

BLOOD PRESSURE ESTIMATION FROM
PHONOCARDIOGRAM BASED ON DEEP HYBRID LEARNING



A Thesis Submitted in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy in Telecommunication
and Computer Engineering
Suranaree University of Technology
Academic Year 2023

การประมาณค่าความดันโลหิตจากเสียงหัวใจบนพื้นฐานการเรียนรู้แบบ
ผสมผสานเชิงลึก



นายกษิตศ กกขุนทด

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาปรัชญาดุษฎีบัณฑิต
สาขาวิชาวิศวกรรมโทรคมนาคมและคอมพิวเตอร์
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2566

BLOOD PRESSURE ESTIMATION FROM
PHONOCARDIOGRAM BASED ON DEEP HYBRID LEARNING

Suranaree University of Technology has approved this thesis submitted in partial fulfillment of the requirements for The Degree of Doctor of Philosophy.

Thesis Examining Committee


.....

(Assoc. Prof. Dr. Chanchai Thaijiam)

Chairperson


.....

(Assoc. Prof. Dr. Peerapong Uthansakul)

Member (Thesis Advisor)


.....

(Assoc. Prof. Dr. Khomdet Phapatanaburi)

Member (Thesis Co-Advisor)


.....

(Assoc. Prof. Dr. Piyaporn Mesawad)

Member


.....

(Assoc. Prof. Dr. Monthippa Uthansakul)

Member


.....


(Dr. Dheerasak Anantakul)

Member


.....

(Assoc. Prof. Dr. Pornsiri Jongkol)

Dean of Institute of Engineering


.....
(Assoc. Prof. Dr. Yupaporn Ruksakulpiwat)
Vice Rector for Academic Affairs
and Quality Assurance

กษิติศ กกขุนทด : การประมาณค่าความดันโลหิตจากเสียงหัวใจบนพื้นฐานการเรียนรู้แบบผสมผสานเชิงลึก (BLOOD PRESSURE ESTIMATION FROM PHONOCARDIOGRAM BASED ON DEEP HYBRID LEARNING)

อาจารย์ที่ปรึกษา : รองศาสตราจารย์ ดร.พีระพงษ์ อุซหารสกุล, 170 หน้า

คำสำคัญ: ความดันโลหิต/โฟโนคาร์ดิโอแกรม/การเรียนรู้แบบผสมผสานเชิงลึก

การติดตามความดันโลหิตเป็นสิ่งสำคัญในการปกป้องสุขภาพของแต่ละบุคคล เนื่องจากการตรวจพบระดับความดันโลหิตที่ผิดปกติตั้งแต่เนิ่น ๆ ช่วยให้การรักษาทางการแพทย์เป็นไปอย่างทันท่วงที สิ่งนี้สามารถลดอัตราการเสียชีวิตที่เกี่ยวข้องกับโรคหลอดเลือดหัวใจได้อย่างมาก ดังนั้นการพัฒนาาระบบตรวจวัดความดันโลหิตอย่างต่อเนื่องจึงมีความสำคัญอย่างยิ่ง

ในวิทยานิพนธ์นี้ มีการนำเสนอวิธีการใหม่ โดยมีโมเดลการถดถอยการเรียนรู้เชิงลึกที่เป็นนวัตกรรมใหม่ ซึ่งควบคุมข้อมูลเสียงการเต้นของหัวใจหรือโฟโนคาร์ดิโอแกรม (PCG) เพื่อการประมาณความดันโลหิต วิธีการนี้รวมเอาโมเดลการถดถอยตามการเรียนรู้แบบไฮบริดเชิงลึกซึ่งไม่เพียงแต่เพิ่มความสามารถในการปรับตัวต่อการเปลี่ยนแปลงเชิงพื้นที่ แต่ยังช่วยให้สามารถจับรูปแบบที่ซับซ้อนภายในสัญญาณ PCG ได้อีกด้วย ความก้าวหน้าเหล่านี้มีส่วนต่อการพัฒนาการประมาณความดันโลหิต

เพื่อยืนยันประสิทธิผลของวิธีการที่เรานำเสนอ จึงได้รวบรวมข้อมูลสัญญาณ PCG จากอาสาสมัคร 78 คน ตามแนวทางจริยธรรมของมหาวิทยาลัยเทคโนโลยีสุรนารี (จริยธรรมการวิจัยมนุษย์หมายเลข EC-65-78) ต่อมา เราใช้ขั้นตอนการตรวจสอบข้าม K-fold สำหรับการแบ่งพาร์ติชันและการจัดแนวข้อมูล ด้วยการรวมชุดข้อมูลผลลัพธ์เข้ากับการเรียนรู้แบบผสมผสานเชิงลึกเพื่อการประมาณค่าความดันโลหิต ทำให้ผลการทดลองมีแนวโน้มที่ดี ได้ระบุค่า Mean Absolute Error (MAE) และค่าเบี่ยงเบนมาตรฐาน (STD) ที่ 10.20 ± 6.67 mmHg สำหรับความดันซิสโตลิก และ 7.00 ± 5.30 mmHg สำหรับความดันไดแอสโตลิก

สาขาวิชา วิศวกรรมโทรคมนาคม

ปีการศึกษา 2566

ลายมือชื่อนักศึกษา กษิติศ

ลายมือชื่ออาจารย์ที่ปรึกษา พีระพงษ์

ลายมือชื่ออาจารย์ที่ปรึกษาร่วม พีระพงษ์

KASIDIT KOKKHUNTHOD : BLOOD PRESSURE ESTIMATION FROM PHONOCARDIOGRAM BASED ON DEEP HYBRID LEARNING. THESIS ADVISOR: ASSOC. PROF. PEERAPONG UTHANSAKUL, 170 PP.

Keyword: BLOOD PRESSURE/ PHONOCARDIOGRAM / DEEP HYBRID LRARNINN

Monitoring blood pressure is crucial for safeguarding individual health because early detection of abnormal blood pressure levels facilitates timely medical intervention, significantly reducing mortality rates associated with coronary heart disease. Consequently, the development of continuous blood pressure monitoring systems is of paramount importance.

In this thesis, a new method is presented, featuring an innovative deep learning regression model that harnesses cardiac sound data or phonocardiogram (PCG) for blood pressure estimation. The method incorporates a deep hybrid learning-based regression model that not only increases adaptability to spatial changes but also enables the capture of complex patterns within the PCG signal. These advances have contributed significantly to the development of blood pressure estimation techniques.

To confirm the effectiveness of our proposed method, PCG signal data were collected from 78 volunteers following the ethical guidelines of Suranaree University of Technology (Human Research Ethics No. EC-65-78). Subsequently, a K-fold cross-validation procedure was employed for data partitioning and alignment. By combining the resulting dataset with deep hybrid learning for estimating blood pressure, the experimental results were very promising. They indicated Mean Absolute Error (MAE) and standard deviation (STD) values of approximately 10.20 ± 6.67 mmHg for systolic pressure and 7.00 ± 5.30 mmHg for diastolic pressure.

School of Telecommunication Engineering

Academic Year 2023

Student's Signature กษิต

Advisor's Signature พีระพงษ์

Co-Advisor's Signature.....

ACKNOWLEDGEMENT

The author wishes to acknowledge the funding support from Suranaree University of Technology (SUT).

I would like to express my sincere thanks to my thesis advisor, Assoc. Prof. Dr. Peerapong Uthansakul, and Assoc. Prof. Dr. Khomdet Phapatanaburi for their consistent supervision, thoughtful comments on several drafts, and valuable advice towards the completion of this study.

My thanks go to Assoc. Prof. Dr. Piyaporn Mesawad, Assoc. Prof. Dr. Chanchai Thajiam, Assoc. Prof. Dr. Monthippa Uthansakul, and Dr. Dheerasak Anantakul for their valuable suggestions and guidance as examination committee members.

The author is also grateful to all faculty and staff members of the School of Telecommunication Engineering and colleagues for their help and assistance throughout the period of this work.

I would like to extend my sincere gratitude to Suranaree University of Technology Hospital for their invaluable support and guidance during the course of the new data collection. Special thanks to the doctors in the cardiology unit of Suranaree University of Technology Hospital, whose expertise and insights were instrumental in this cardiology research. Our appreciation extends to the team members, colleagues, and peers whose contributions were invaluable.

Finally, I would also like to express my deep sense of gratitude to the scholarship program, my parents, and my advisor for their support and encouragement throughout my study at Suranaree University of Technology.

Kasidit Kokkhunthod

TABLE OF CONTENTS

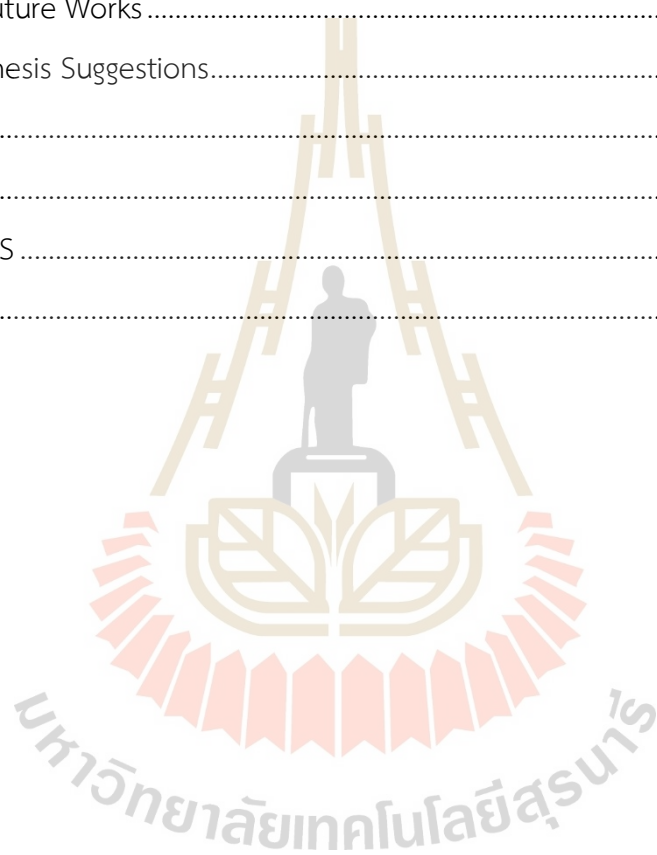
	Page
ABSTRACT (THAI)	I
ABSTRACT (ENGLISH)	III
ACKNOWLEDGEMENT	III
TABLE OF CONTENTS	IV
LIST OF TABLES	VII
LIST OF FIGURES.....	VIII
LIST OF ABBREVIATIONS	X
CHAPTER	
I INTRODUCTION	1
1.1 Background and problem statement.....	1
1.2 Thesis objectives	2
1.3 Scope and limitation of the thesis	3
1.4 Contributions.....	3
1.5 Organization of the thesis.....	4
II BACKGROUND THEORY.....	6
2.1 Introduction.....	6
2.2 Properties of Blood Pressure	7
2.3 Neural Networks	11
2.4 Feedforward Neural Network.....	14
2.5 Deep Learning	18
2.6 Convolutional Neural Network.....	20
2.7 Tuning a Neural Network	30

TABLE OF CONTENTS (Continued)

	Page
2.8 Learning System	33
2.9 K-Fold Cross-Validation	43
2.10 Mean Absolute Error (MAE)	45
2.11 Standard Deviation (STD)	47
2.12 Techniques for estimation of blood pressure values.....	48
2.13 Literature review related	55
2.14 Summary	67
III METHODOLOGY	68
3.1 Introduction	68
3.2 Data Collection and Data Recording	70
3.3 Data Categorizing	80
3.4 Data Collection Process	82
3.5 Verify The Signal Information of The Phonocardiogram Signal.....	93
3.6 Adding Sample Data	95
3.7 Estimating Blood Pressure From Phonocardiograms.....	97
3.8 Dividing The Dataset For Building Training And Testing Process	107
3.9 Summary	114
IV RESULTS AND DISCUSSION.....	115
4.1 Introduction	115
4.2 Information Collected From Volunteers	116
4.3 Performance Of Estimate With 1D CNNs Base Approach.....	120
4.4 Performance Of Estimate With Deep Hybrid Learning.....	127
4.5 Summary	132
V CONCLUSIONS	137

TABLE OF CONTENTS (Continued)

	Page
5.1 Discussion.....	137
5.2 Conclusions	138
5.3 Future Works.....	139
5.4 Thesis Suggestions.....	140
REFERENCES	143
APPENDIX.....	148
PUBLICATIONS	149
BIOGRAPHY	170



LIST OF TABLES

Table	Page
2.1 Comparison with Works are Use PCG Signal.....	67
3.1 Physical information of volunteer.....	98
3.2 Dividing the 1st fold data according to the number of the data	110
3.3 Dividing the 2nd fold data according to the number of the data	111
3.4 Dividing the 3th fold data according2.1 to the number of the data	112
4.1 Information collected from volunteers.	117
4.2 Performance of PCG signals data with CNNs model	121
4.3 Performance of PCG signals with additive white gaussian noise (AWGN)	122
4.4 Performance of numbers CNNs block layer.....	127
4.5 Performance of number of Linear Layers.....	124
4.6 Performance MAE of Batch Sizes and Learning Rates.....	126
4.7 Performance MAE of Deep Hybrid learning.....	129
4.8 Performance MAE of Deep Hybrid learning with physical information	130
4.9 Comparison Performance MAE with using PCG signal.....	133
4.10 Comparison of the Methods for Estimation.....	134

LIST OF FIGURES

Figure	Page
2.1 Sequence of phonocardiograms while working.	9
2.2 Relationship of phonocardiograms to blood pressure	11
2.3 Characteristics of nerve cells in the brain	12
2.4 Architecture of A Neural Network	13
2.5 Feed-forward back-propagation mechanism.	14
2.6 Venn Diagram of Artificial Intelligence, Machine Learning, and Deep Learning.....	19
2.7 A picture that shows the main calculations.	24
2.8 Effects of different convolution matrices	25
2.9 Stride 1, the filter windows move only one time for each connection	26
2.10 The effect of stride	26
2.11 Zero-padding	27
2.12 One-dimensional convolutional neural network (1D-CNNs) architecture for the timeseries forecasting model	29
2.13 Max pooling	30
2.14 Average pooling	31
2.15 Average pooling and Max pooling (Dhuma, 2019).....	32
2.16 Dropout neural network model. (a) A standard neural network. (b) An example of a thinned net produced by applying dropout.....	32
2.17 Architecture Machine learning	34
2.18 Structure of KELM.....	36
2.19 Support Vector Machine and Support Vector Regression (SVR)	38
2.20 The basic structure of the XGB algorithm	39
2.21 Data set arrangement format using K-Fold Cross Validation method	45
2.22 Shows the training and testing datasets.....	45
3.1 Process of estimate blood pressure with Deep Learning	70

LIST OF FIGURES (Continued)

Figure	Page
3.2 Process of estimate blood pressure with Deep Hybrid Learning.	70
3.3 Sequence of recording PCG signal from volunteer	83
3.4 Automatic blood pressure measuring device and digital medicalstethoscope	84
3.5 Characteristics of the position of using the arm cuff	87
3.6 Standard position for listening to phonocardiograms.....	90
3.7 Volunteers sitting and waiting for blood pressure measurement.....	91
3.8 Recording phonocardiograms and blood pressure values.....	91
3.9 (a) Page of the phonocardiogram recording program. (b) Page while recording phonocardiograms	92
3.10 Positioning the equipment used for recording sound and measuring blood pressure.....	93
3.11 Picture of components of phonocardiogram S1 and S2.....	94
3.12 Example of a volunteer's phonocardiogram.....	95
3.13 Filtered sound in the frequency range 60-500 Hz.....	95
3.14 Size of original file of phonocardiogram recorded.....	96
3.15 Segmentation data from original to 11 samples	97
3.16 A schematic representation of the proposed Conv1D model architecture.....	103
3.17 Block diagram depicting the overall system of estimate BP from PCG signal...	104
3.18 Learning process of deep learning algorithm	104
3.19 Testing process of Deep hybrid learning algorithm.....	107
3.20 Data set arrangement format using K-Fold Cross Validation method.....	108
3.21 D Data organization for this research with 78 cases	109
3.22 Shows the training and testing datasets.....	113
4.1 A schematic representation of the proposed Conv1D model.....	128

LIST OF ABBREVIATIONS

PCG	=	phonocardiogram
MAE	=	Mean Absolute Error
STD	=	standard deviation
WHO	=	World Health Organization
BP	=	Blood Pressure
mm/Hg	=	millimeters of mercury
SYS	=	systolic pressure
DIA	=	diastolic pressure
SA node	=	sinoatrial node
ANN	=	Artificial Neural Network
OPD	=	Out-Patient Department
DBMS	=	Database Management System
NN	=	Neural Networks
DL	=	Deep Learning
ML	=	machine learning
DNNs	=	deep neural networks
NLP	=	Natural Language Processing
SER	=	speech emotion recognition
CNN	=	Convolutional Neural Network
BN	=	Batch Normalization
KELM	=	Kernel Extreme Learning Machine
SVR	=	Support Vector Regression
XGB	=	Extreme Gradient Boosting
PTT	=	transit time. of pulse
CVDs	=	cardiovascular disease
AAMI	=	Association for the Advancement of Medical Instrumentation

LIST OF ABBREVIATIONS (Continued)

AWGN	=	Additive White Gaussian Noise
RNNs	=	recurrent neural networks
WHO	=	World Health Organization
PtP	=	Peak to Peak
W	=	Weight
H	=	Height
BMI	=	body mass index



CHAPTER I

INTRODUCTION

1.1 Background and problem statement

Individuals can be at risk of developing various diseases, particularly when they are sensitive to pollutants and germs that, in most people, may cause respiratory or infectious diseases. However, there is a critical illness that is often overlooked, which anyone can contract at any age, regardless of environmental factors or specific causes. This disease is characterized by underdiagnosis and a lack of awareness. It can occur at any age, and data gathered from inpatient histories or statistical studies often fall below normal levels, adhering to World Health Organization (WHO) principles. Notably, this disease commonly affects individuals aged 30-79 years, with approximately 1.3 million people suffering from it. Often asymptomatic initially, it can lead to diagnosis only when linked as a risk factor to other conditions such as ischemic heart disease, cerebral diseases, and kidney diseases, among others.

In Thailand, the incidence of high blood pressure is continually rising. According to the Thai Population Health Survey of 2019-2020, it was discovered that 13 million individuals aged 18 and older have high blood pressure, with up to 7 million of these individuals unaware of their condition. If left untreated for an extended period, the severity of the condition can escalate. Blood pressure measurement is a critical method for checking, assessing, and evaluating symptoms in patients who may have high blood pressure.

Regular blood pressure measurement is essential as it provides crucial information about the health of the circulatory system. Health care centers can detect potential issues early by measuring blood pressure regularly and take

preemptive actions before serious problems develop. Moreover, tracking blood pressure over time helps health care providers monitor the effectiveness of treatment for high blood pressure and other related symptoms.

Certain clinical scenarios require automated, repetitive measurements of blood pressure over several hours or days. However, automatic repetitive cuff-based blood pressure measurements or continuous intra-arterial monitoring are often not feasible due to patient discomfort. Research studying the relationship between phonocardiograms and blood pressure has shown that human phonocardiograms correlate with blood pressure levels. Consequently, the potential for measuring systolic blood pressure by analyzing phonocardiograms was explored. Continuous blood pressure recording might prove beneficial in cases involving patients at risk of shock, those who are severely injured, and newborns with unclear diagnoses.

In this research, we describe the physiological basis for a continuous, non-invasive blood pressure measurement method that uses the correlation between heart and blood vessel sounds to estimate blood pressure. This method is enhanced by deep machine learning to ensure that blood pressure estimations are as accurate as those obtained with a cuff device. The machine learning process utilizes data from a sample group of Thai individuals with varied physiologies for training, enabling the blood pressure estimation to accommodate a broad range of physiological characteristics. This approach will generate new information that has been recorded alongside phonocardiograms, thereby improving the accuracy and reliability of blood pressure estimation.

1.2 Thesis objectives

1.2.1 Utilize the phonocardiogram Recording Dataset.

The dataset containing phonocardiogram recordings made during blood pressure measurements will be used to analyze the characteristics of

phonocardiogram signals. This analysis aims to estimate blood pressure values from test phonocardiogram.

1.2.2 Evaluate Blood Pressure Estimation Algorithms.

Test the capabilities of algorithms designed to estimate blood pressure values.

1.2.3 Compare Algorithm Performance.

Assess and compare the performance of algorithms capable of estimating blood pressure values against the actual pressure values recorded during measurements.

1.3 Scope and limitation of the thesis

1.3.1 Analyze Components of the phonocardiogram Dataset.

Study the components of the phonocardiogram dataset that were recorded from the sample group. These components will be used in the analysis to estimate blood pressure.

1.3.2 Implement deep hybrid Learning.

Apply deep hybrid learning techniques comprehensively to the study's analytical capabilities.

1.3.3 Evaluate and Compare Efficiency.

Compare the efficiency of traditional methods used in pressure estimation with those involving the application of machine learning.

1.4 Contributions

1.4.1 Comparative Performance Analysis. Provide results from a performance comparison for estimating blood pressure, applying machine learning techniques in deep hybrid learning.

1.4.2 Blood Pressure Estimation Mean Absolute Error (MAE). Mean Absolute Error (MAE) as close to the AAMI standard 5 ± 8 mmHg as possible for estimating blood pressure from phonocardiogram.

1.4.3 Data Collection from a Specific New Sample. Gather data from a new and specific sample group, consisting of Thai individuals.

1.5 Organization of the thesis

The remainder of this thesis is outlined as follows:

Chapter II provides an extensive exploration of the underlying theoretical framework, establishing the fundamental principles associated with this research. It serves as the bedrock by introducing the central concepts and principles of the thesis, such as blood pressure estimation, phonocardiogram (PCG) signal processing, and deep learning techniques. Notably, it delves into the relationship between phonocardiograms and blood pressure, and critically examines relevant literature, including previous studies on blood pressure estimation, deep learning methodologies, and hybrid learning models.

Chapter III concentrates on elucidating the methodology employed in designing deep learning models for blood pressure estimation from PCG signals. The chapter places particular emphasis on the data collection process, signal preprocessing, and feature extraction techniques. It delves into the diverse methodologies and techniques used to train and construct the models, including band-pass filtering, Convolutional Neural Networks (CNNs), and hybrid learning approaches. Detailed explanations of the experimental setup and the integration of physical information into the models are provided, offering insights into their implementation and deployment.

Chapter IV presents the outcomes of the developed models, accompanied by comprehensive discussions. The model results are subdivided into several subsections, enabling an in-depth analysis. The first subsection evaluates the

performance of the models using 1D CNNs-based approaches, examining key metrics such as Mean Absolute Error (MAE) and standard deviation (STD). The second subsection investigates the impact of data augmentation techniques like Additive White Gaussian Noise (AWGN) on model performance. Further subsections compare the models' performance with different configurations of CNNs layers and linear layers, as well as different batch sizes and learning rates. Lastly, the performance of deep hybrid learning models, including the integration of physical information, is compared against baseline systems, highlighting the improvements achieved.

Chapter V serves as the concluding chapter of the thesis, encapsulating the key findings and presenting conclusions drawn from the research. Additionally, this chapter provides valuable suggestions and recommendations for future studies, identifying potential areas for further investigation and improvement within the field of blood pressure estimation from PCG signals. The chapter emphasizes the practical implications of the research findings and discusses possible extensions to enhance the robustness and accuracy of the developed models.

CHAPTER II

BACKGROUND THEORY

2.1 Introduction

Blood pressure is a measurement of the force of blood against the arterial walls as the heart pumps it throughout the body. Regularly measuring blood pressure is important because high blood pressure can increase the risk of serious health problems such as heart disease, stroke, and kidney disease. In many health monitoring situations, it is necessary to measure blood pressure repeatedly at several intervals over several hours or days. However, automated repeated cuff-based blood pressure measurements or continuous monitoring of intra-arterial pressure may be uncomfortable for the patient or may interrupt rest during treatment. Research studies have shown a relationship between phonocardiograms and blood pressure, indicating the possibility of measuring systolic blood pressure by analyzing phonocardiograms. Continuous blood pressure recording appears to be particularly valuable for patients at risk for shock, severely injured patients, and newborns with unclear diagnoses.

Phonocardiograms, also known as auscultation, refer to the process of listening to the internal sounds of the heart using a stethoscope. It is an important tool for measuring blood pressure as well as checking for potential heart problems. Phonocardiograms can reveal crucial information about the heart's function and health, such as heart rate, heart rhythm, and the presence of noise or abnormal sounds. By measuring blood pressure and listening to phonocardiograms, healthcare professionals can obtain a comprehensive view of a person's heart health and make a diagnosis if necessary. Additionally, regular monitoring of phonocardiograms and

blood pressure can help detect any changes in heart function over time, potentially serving as an early warning sign of possible health problems.

Research studies have indicated a relationship between phonocardiograms and blood pressure, leading to the exploration of blood pressure estimation using new methods. This involves estimating blood pressure by listening to phonocardiograms. The sounds produced by the heart within the human body can be used to carry important information, which can be applied in the diagnosis of certain diseases or in estimating blood pressure to monitor symptoms that may be serious for the patient. Currently, the development of machine learning algorithms plays a significant role in analyzing or estimating a wide variety of disease symptoms. By recording signals in various formats, these can be used to train machine learning models, providing information that can assist doctors in their diagnoses. This chapter discusses the theories involved in the development of machine learning algorithm systems, some principles of heart function necessary to capture heart activity sounds, and the relationship between phonocardiograms and blood pressure.

2.2 Properties of Blood Pressure.

In the human body, various organs perform different functions, and one of the most important organs is the heart. The heart pumps blood to various parts of the body through blood vessels, making it possible to measure blood pressure. Blood pressure is a measure of the force of blood against the artery walls as the heart pumps it throughout the body. It is generally measured in millimeters of mercury (mm/Hg) and expressed as two numbers: systolic pressure (the higher value) and diastolic pressure (the lower value).

The relationship between phonocardiograms and blood pressure is complex. Changes in blood pressure can affect the heart's ability to pump blood, which in turn influences the sounds the heart makes.

One way blood pressure affects phonocardiograms is through changes in the volume of blood the heart pumps. When blood pressure is high, the heart must work harder to pump blood through the blood vessels, leading to increased blood flow through the heart's chambers. This increased blood flow can cause the phonocardiograms to be louder due to the higher volume of blood moving through the heart.

On the other hand, when blood pressure is low, the heart may not be able to pump enough blood to meet the body's needs, causing a decrease in blood flow through the heart chambers. This reduced blood flow can result in quieter phonocardiograms.

2.2.1 Blood Pressure Function

Blood pressure is caused by the contraction and relaxation of the heart's ventricles. When the left ventricle of the heart contracts, it pumps oxygen-rich blood into the aorta, which is the main artery that carries blood away from the heart. This creates systolic pressure, which is the higher number in a blood pressure reading. The pressure in the aorta begins to decrease as the left ventricle relaxes, allowing blood to fill the ventricle again. This creates diastolic pressure, which is the lower number in a blood pressure reading.

2.2.2 Heart Function

The heart works like a pump, sending blood to the organs, tissues, and cells within the body. Deoxygenated blood filled with carbon dioxide enters the heart starting in the right atrium, and then is squeezed down to the lower right chamber, the ventricle, to be carried to the lungs for purification and oxygenation. Oxygen-laden blood is then transported to the left atrium and pumped down to the left ventricle to deliver nutrients and oxygen through the major arteries to the capillaries, nourishing cells throughout the body. This cardiac cycle works continuously in order. Each heartbeat is triggered by electrical signals generated by the sinoatrial (SA) node, a group of cells on the wall of the heart's right atrium. These

signals travel through nerves and stimulate the heart muscle to contract, resulting in blood movement. The contraction of the heart's chambers and the movement of blood produce the heartbeat sound, as well as the pulse that can be felt at the wrist and neck.

2.2.3 The Sound of The Heart Working

The heart is an organ surrounded by specialized muscles that can contract at any time. The working cycle of the heart consists of three periods: the period of contraction called systole, the period of relaxation called diastole, and a period of rest (pause). This complete cycle is known as the cardiac cycle. During one cardiac cycle, various mechanical changes occur, such as changes in pressure, volume, and blood flow, which produce phonocardiograms. Phonocardiograms are caused by several factors, including the contraction of the heart's chamber walls, turbulent blood flow, and the opening and closing of heart valves. Phonocardiograms are divided into two main types: phonocardiograms and heart murmurs. Phonocardiograms are low-frequency sounds caused by the vibration of the heart valves after they close and open. In contrast, heart murmurs have a complex structure and a signal similar to noise, resulting from turbulent blood flow. Heartbeats can be heard by using the ear or with the aid of a stethoscope, which allows for clearer hearing. An example of a cardiac stethoscope is shown in Figure 2.1.

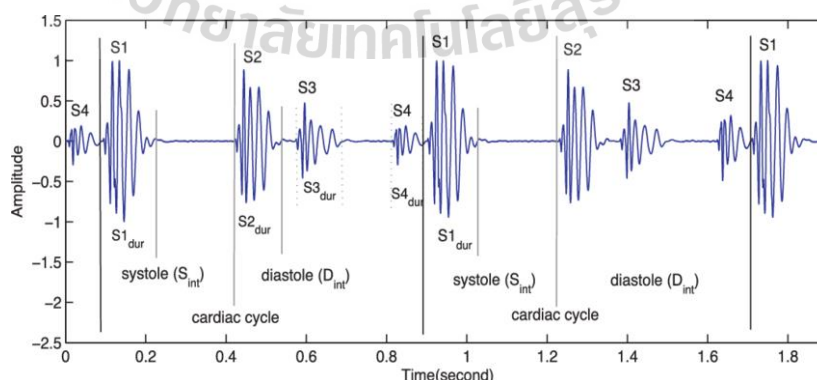


Figure 2.1 Sequence of phonocardiograms while working.

From Figure 2.1, it can be seen that phonocardiograms consist of four distinct sounds, namely sounds 1 to 4. However, in auscultation, only sounds 1 and 2 can be heard. To detect all four sounds, a phonocardiograph device is required. By placing a microphone on the chest, it receives sound signals and converts them into electrical signals. These signals are then sent to an amplifier and displayed as sound to be heard or recorded as sound waves on a graph. The sound waves recorded on this graph are called a "Phonocardiogram (PCG)." The symbols are defined as follows:

S1 is the First phonocardiogram.

S2 is the Second phonocardiogram.

S3 is the Third phonocardiogram.

S4 is the Fourth phonocardiogram.

The sounds S1 and S2 are normal heartbeat sounds and are typically heard clearly. Normally, the S1 and S2 sounds should be almost the same loudness. The sounds that indicate disease are S3, S4, Click, OS, and heart murmur. The murmur sound can occur during the contraction phase (between S1 and S2) or the relaxation phase (between S2 and S1).

2.2.4 Relationship of Blood Pressure and Phonocardiograms

The relationship between blood pressure and phonocardiograms can be observed through a process called auscultation, which involves listening to the heart using a medical stethoscope. For example, a doctor or nurse typically measures a patient's blood pressure and then listens to the phonocardiograms while the patient is in various postures (sitting, standing, etc.) and under different conditions (resting, exercising, etc.).

Research studies have shown that blood pressure values are related to phonocardiograms S1 and S2. These sounds are associated with the periods of compression (or contraction), known as systole, and relaxation, known as diastole, that occur during the cardiac cycle, as shown in Figure 2.2.

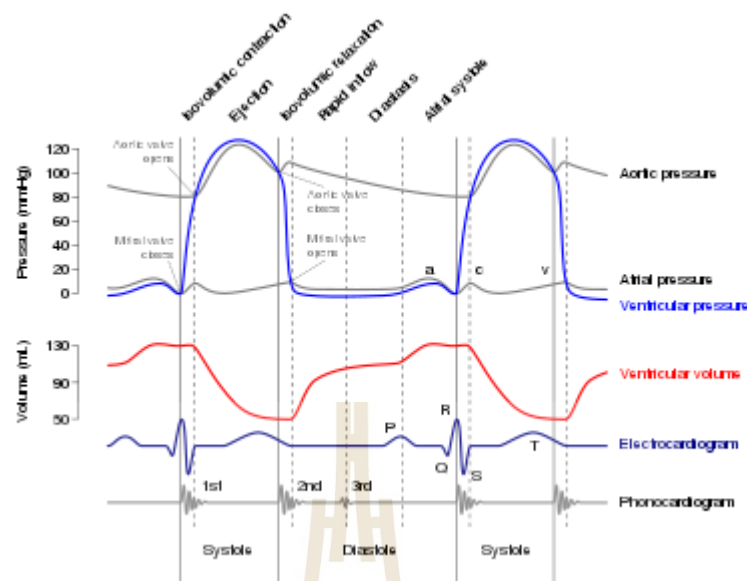


Figure 2.2 Relationship of phonocardiograms to blood pressure

2.3 Neural Networks

Artificial neural networks are a technique designed to emulate how the human brain functions. This concept is inspired by the study of the brain's bioelectric network, which comprises nerve cells, or "neurons," and their connections, known as "synapses." Each neuron features an end called a "dendrite" that receives input signals, and a terminal called an "axon," which functions as the output of the cell. These cells operate through electrochemical reactions. When activated by external stimuli or signals from other cells, the nerve impulse travels through the dendrites to the nucleus. Here, it is determined whether the impulse is strong enough to warrant propagation to other cells. If the impulse is sufficient, the nucleus sends it further through its axons. The characteristics of nerve cells in the brain are illustrated in Figure 2.3.

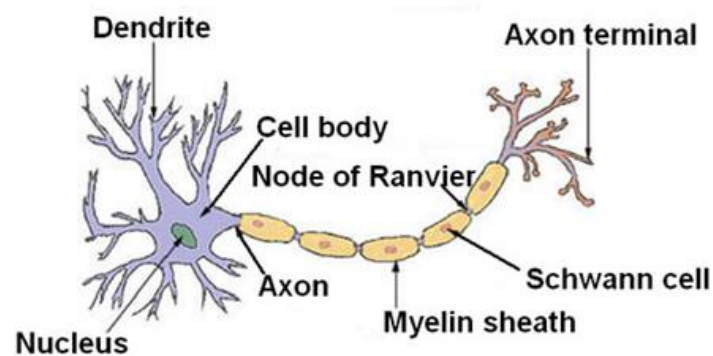


Figure 2.3 Characteristics of nerve cells in the brain

In computing, neural networks consist of many interconnected neurons that accumulate knowledge through prior training. The neural network learns to recognize various fonts of each letter, enabling it to handle diverse types of text images in real-world applications. The training data provided to the neural network does not need to be in the form of images exactly as we see them. Typically, the input undergoes a key feature extraction step and other preliminary processing procedures first.

Neural Network Structure.

The architecture of a neural network (NN) is a key determinant of its performance. Typically, an NN consists of three types of layers: an input layer, one or more hidden layers, and an output layer. Each layer is composed of units or nodes that are interconnected by weights. These weights are adjusted as learning progresses. The structure of these networks can vary significantly, depending on the specific task they are designed to perform, as shown in Figure 2.4.

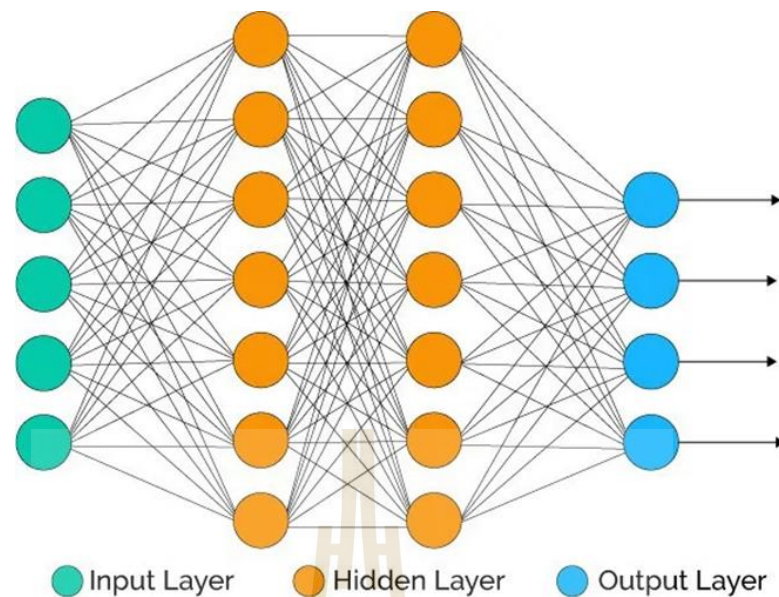


Figure 2.4 Architecture of A Neural Network

The fundamental building block of a neural network (NN) is the neuron, a unit that receives inputs, processes them, and generates an output. Each neuron in the network performs a dot product of the inputs and the weights, adds a bias, and then passes the result through a nonlinear function, often referred to as an activation function. This process enables neural networks to learn complex patterns in data.

Input Layer

The input layer is the initial phase of processing in a NN. This layer's primary function is to receive the inputs and pass them on to the next layers. The number of neurons in the input layer corresponds to the attributes or features of the input data.

Hidden Layer

The hidden layers of a NN are where most of the computations take place. These layers use weights to propagate signals from the input layer to the output layer, and they typically contain a nonlinear activation function to help the network learn complex patterns. The number of hidden layers and the number of neurons in each layer are important parameters that can affect the network's capacity and the complexity of the functions it can learn.

Output Layer

The output layer is the final layer in a NN and is responsible for outputting a vector of values that are understandable in the context of the application. For classification tasks, these outputs are usually the probabilities of the input belonging to a certain class. For regression tasks, they are continuous values.

2.4 Feedforward Neural Network

In the context of feedforward neural networks (FNNs), understanding the forward and backward processes is crucial for both the functioning and training of the network. These processes define how data is propagated through the network and how the model learns from data, as shown in Figure 2.5. Here's a detailed description:

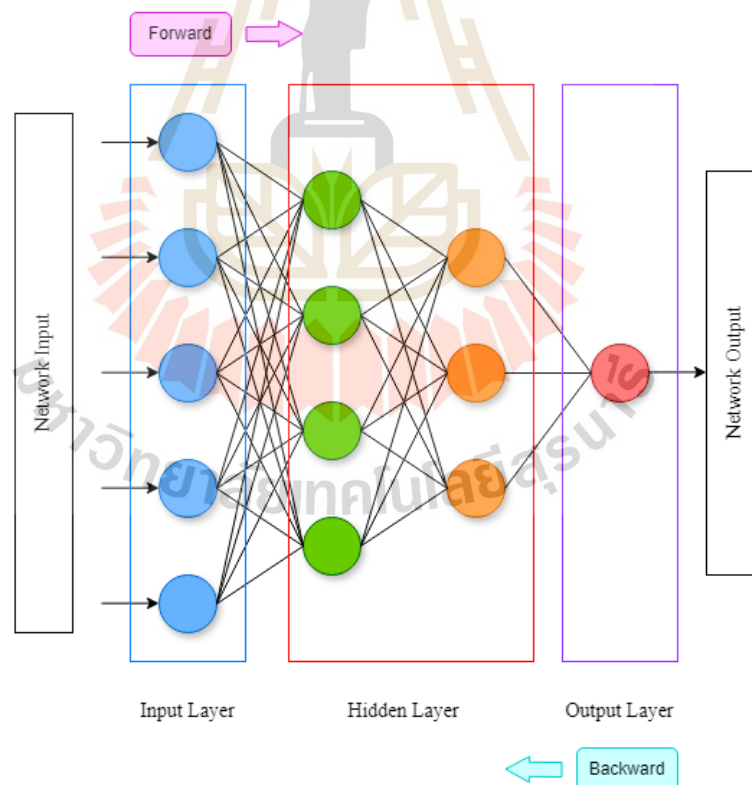


Figure 2.5 Feed-forward back-propagation mechanism.

The Figure 2.5 shown illustrates the structure and data flow within a neural network. It comprises an input layer, hidden layers, and an output layer. During the forward propagation phase, data enters the network through the input layer, passes through the hidden layers where it is processed, and finally reaches the output layer to produce the network's output. In the backpropagation phase, the network adjusts the weights of the connections based on the error between the predicted output and the actual output, propagating this error backward from the output layer to the input layer. This iterative process helps the network learn and improve its predictions.

Network Input: The data fed into the neural network.

Network Output: The result produced by the neural network after processing the input data.

Forward Propagation: The process of passing input data through the network layers to generate an output.

Backpropagation: The method used during training to update the weights of the network based on the error of the output compared to the expected result.

This neural network structure is fundamental in machine learning, allowing the model to learn from data by adjusting the weights during training to make accurate predictions.

2.4.1 Forward Process

The forward process in a feedforward neural network (FNN) involves the propagation of input data through the network to produce an output. This is also referred to as the forward pass. The main steps involved are:

2.4.1.1 Input Layer

The input layer receives the initial data. Each feature of the input data corresponds to a neuron in this layer.

For example, if the input data has n features, the input layer will have n neurons.

2.4.1.2 Hidden Layers

Each hidden layer consists of multiple neurons. Each neuron takes the weighted sum of inputs from the previous layer, adds a bias term, and applies an activation function to produce an output.

The operation performed by each neuron can be described as:

$$z_j^{(l)} = \sum_{i=1}^n w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)} \quad (2.1)$$

$$a_j^{(l)} = f(z_j^{(l)}) \quad (2.2)$$

where:

$z_j^{(l)}$ is the weighted sum of inputs for neuron j in layer l .

$w_{ij}^{(l)}$ is the weight between neuron i in layer $l-1$ and neuron j in layer l .

$a_i^{(l-1)}$ is the activation of neuron i in layer $l-1$.

$b_j^{(l)}$ is the bias term for neuron j in layer l .

f is the activation function.

$a_j^{(l)}$ is the output (activation) of neuron j in layer l .

2.4.1.3 Output Layer

The final hidden layer outputs are fed into the output layer.

The number of neurons in the output layer corresponds to the number of classes in a classification problem or the number of output dimensions in a regression problem.

The output layer applies a suitable activation function, such as the softmax function for classification or a linear activation for regression.

2.4.1.4 Activation Functions

Activation functions introduce non-linearity into the network, enabling it to learn complex patterns.

Common activation functions include:

$$\text{Linear : } f(z) = z$$

$$\text{Sigmoid} : f(z) = \frac{1}{1 + e^{-z}}$$

$$\text{Tanh} : f(z) = \tanh(z)$$

$$\text{ReLU (Rectified Linear Unit)} : f(z) = \max(0, z)$$

2.4.1.5 Output

The final output of the network is used to make predictions. In a classification problem, this could be the predicted class probabilities, and in a regression problem, it could be the predicted continuous values.

2.4.2 Backward Process

The backward process, also known as backpropagation, involves calculating the gradients of the loss function with respect to the network's weights and biases and then updating these parameters to minimize the loss. The main steps involved are:

2.4.2.1 Loss Calculation

The loss function quantifies the difference between the network's predictions and the actual target values.

Common loss functions include:

Mean Squared Error (MSE) for regression:

$$MES = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2.3)$$

Cross-Entropy Loss for classification:

$$CEL = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{i,c} \log(\hat{y}_{i,c}) \quad (2.4)$$

where y_i is the true label, \hat{y}_i is the predicted label, N is the number of samples, and C is the number of classes.

2.4.2.2 Gradient Calculation (Backpropagation)

Backpropagation calculates the gradients of the loss function with respect to each weight and bias in the network.

The chain rule is used to propagate the gradients backward through the network.

For each weight $w_{ij}^{(l)}$ and bias $b_j^{(l)}$:

$$\frac{\partial L}{\partial w_{ij}^{(l)}} = \frac{\partial L}{\partial a_j^{(l)}} \cdot \frac{\partial a_j^{(l)}}{\partial z_j^{(l)}} \cdot \frac{\partial z_j^{(l)}}{\partial w_{ij}^{(l)}} \quad (2.5)$$

$$\frac{\partial L}{\partial b_j^{(l)}} = \frac{\partial L}{\partial a_j^{(l)}} \cdot \frac{\partial a_j^{(l)}}{\partial z_j^{(l)}} \cdot \frac{\partial z_j^{(l)}}{\partial b_j^{(l)}} \quad (2.6)$$

2.4.2.3 Weight and Bias Updates

The network's weights and biases are updated using an optimization algorithm, such as Stochastic Gradient Descent (SGD) or Adam.

The update rule for each weight $w_{ij}^{(l)}$ and bias $b_j^{(l)}$ is:

$$w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} - \eta \frac{\partial L}{\partial w_{ij}^{(l)}} \quad (2.7)$$

$$b_j^{(l)} \leftarrow b_j^{(l)} - \eta \frac{\partial z_j^{(l)}}{\partial b_j^{(l)}} \quad (2.8)$$

where η is the learning rate.

2.4.2.4 Iterative Training

The forward and backward processes are repeated for multiple iterations (epochs) over the training dataset.

In each epoch, the network's parameters are adjusted to minimize the loss function, leading to improved performance.

2.5 Deep Learning

Artificial intelligence encompasses a broad field of intelligence demonstrated by machines, as opposed to the natural intelligence exhibited by humans and animals. Machine learning is considered a subset of artificial intelligence, characterized by a more precise set of definitions. In recent years, machine learning (ML) has emerged as a new paradigm, shifting away from rule-based systems. In this new paradigm, computers are tasked with identifying patterns and autonomously

creating rules. This shift is highlighted by a widely recognized and commonly used definition of machine learning algorithms.

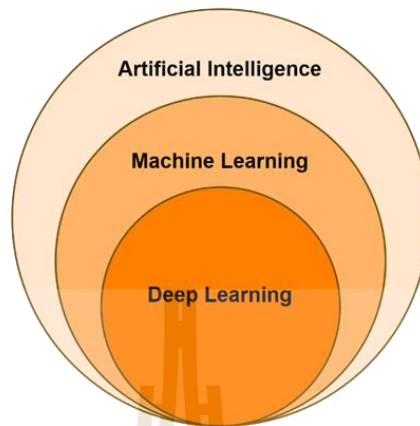


Figure 2.6 Venn Diagram of Artificial Intelligence, Machine Learning, and Deep Learning.

A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T , as measured by P , improves with experience E .

Machine learning algorithms learn from features within a dataset; these features are consistent across all instances in the dataset and can be continuous, categorical, or binary. Machine learning is typically divided into supervised and unsupervised learning, utilizing methods such as Support Vector Machines, Kernel Extreme Learning Machine, and eXtreme Gradient Boosting. A critical enabler for machine learning is data. The surge in big data has significantly propelled the growth of ML, particularly gaining real traction in 2011, driven by promotional efforts from leading technology companies like IBM, which have helped establish a niche market in analytics.

Deep learning represents the latest paradigm in which neural networks with multiple layers—hence the term "deep"—are used. These deep neural networks (DNNs) incorporate a vast number of parameters and typically require significant computing power and large datasets for training. Deep learning first gained

prominence in image processing and has since been widely adopted in fields such as Natural Language Processing (NLP), recommendation systems, and speech emotion recognition (SER). A key factor contributing to the success of deep learning has been the availability of the ImageNet dataset, which, as of 2020, contains approximately 14.20 million hand-labeled images.

2.6 Convolutional Neural Network

2.6.1 Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) are artificial intelligence systems that use multiple layers of neural networks to identify, recognize, and classify objects. They can also detect and separate objects in images. CNNs, also known as ConvNets, are well-known discriminative deep learning architectures that can learn from input data without requiring humans to extract features (Koushik, 2016; Bezdán & Džakula, 2019; Bouchard, 2022). These networks are often used for visual identification, medical image analysis, image segmentation, Natural Language Processing (NLP), and more. Designed to work with a wide range of 2D shapes, CNNs perform better than standard networks because they can automatically identify essential parts of the input without human intervention.

In CNNs, the layers consist of filters or kernels that process data. Initially given random weights, these kernels learn to extract features from the input data through training, transforming random values into meaningful weights. CNNs understand multi-channelled images, such as RGB images with three color channels or single-channelled grayscale images, whereas traditional neural networks work with vector data. By applying these kernels to the input image and displaying the results as N-dimensional matrices, the feature map is created. Within a high-dimensional, implicit feature space, the kernel finds the inner products of all data pairs without knowing the coordinates of the data in that space. This "kernel trick" turns a linear model into a non-linear model, enabling CNNs to better learn and extract complex features.

A CNNs requires a convolutional layer consisting of different filters or kernels that transform the input data. Initially, these kernels are assigned random weights, which gradually change based on training data to extract features from the input, represented as N-dimensional matrices. During training, each kernel, composed of a set of integers, learns to identify important features. This allows the CNNs to operate in a high-dimensional feature space without directly computing data coordinates, instead determining the inner products in the feature space. The kernel trick can be applied to a linear model to convert it into a non-linear model. Unlike traditional neural networks that use vector format, CNN input format is set before convolution, enabling it to handle images with multiple channels.

Examples of the convolution process include RGB images with three channels or grayscale images with one channel. To find patterns in a 4x4 grayscale image, a 2x2 kernel is set up with random weights. As the kernel moves across the image horizontally, the dot product of the input image and the kernel is calculated. This involves multiplying corresponding coordinate values and summing them to obtain a single scalar value. This process is repeated after each iteration until the kernel can no longer move across the image.

Transitioning from this practical example, it becomes evident that the underlying parameters greatly determine the effectiveness of convolution operations. These parameters are crucial in shaping the outcome of these processes, as they dictate how the convolution is executed. They influence everything from the precision of pattern detection to the speed and efficiency of the operation. Understanding these parameters is key to optimizing convolution processes for different applications and requirements.

Building further on the importance of these parameters, in the realm of CNNs, they take on an even more significant role. The convolutional layer serves as a fundamental building block, primarily utilized for feature extraction and spatial analysis in multidimensional data. The efficacy and efficiency of these layers are

largely governed by a set of parameters that define their operational behavior. These parameters not only influence the transformation of data within the layer but also have far-reaching implications for the overall network architecture, computational complexity, and the nature of features extracted. Understanding these parameters is pivotal for designing effective CNNs models. The primary parameters that define the operations of a convolutional layer are (PyTorch, 2023):

- (1) Input dimensions $(N, C_{in}, H_{in}, W_{in})$: The input to a convolutional layer is typically a four-dimensional tensor, characterized by the batch size (N), the number of input channels (C_{in}), and the spatial dimensions - height (H_{in}) and width (W_{in}). These dimensions represent the size and complexity of the input data.
- (2) Kernel size: The kernel, or filter, is a small matrix used to apply the convolution operation. The size of the kernel (height and width) determines the extent of the local area in the input to which the convolution is applied. Larger kernels encompass more input units, capturing broader features, while smaller kernels focus on finer, localized features.
- (3) Padding: Padding involves adding extra pixels around the edge of the input tensor. This technique is used to control the spatial dimensions of the output tensor, allowing for adjustments in feature map size, and is critical for handling edge cases where the kernel overlaps the bounds of the input.
- (4) Stride: Stride defines the step size with which the kernel moves across the input tensor. A larger stride results in broader spatial sampling, leading to smaller output dimensions, whereas a smaller stride offers finer sampling, preserving more spatial information in the output.

- (5) Dilation: Dilation refers to the spacing between the elements within the kernel. This parameter allows the kernel to expand and cover a larger receptive field without increasing the number of parameters. Dilation introduces an additional level of flexibility in manipulating the field of view of the convolutional filters.

The output dimensions $(N, C_{out}, H_{out}, W_{out})$ of a convolutional layer are influenced by its configuration. In this case, (C_{out}) is determined by the number of filters. The height (H_{out}) and width (W_{out}) of the output are calculated as follows:

$$H_{out} = \left\lceil \frac{H_{in} + 2 \times padding[0] - dilation[0] \times (kernelsize[0] - 1) - 1}{stride[0]} + 1 \right\rceil \quad (2.9)$$

$$W_{out} = \left\lceil \frac{W_{in} + 2 \times padding[1] - dilation[1] \times (kernelsize[1] - 1) - 1}{stride[1]} + 1 \right\rceil \quad (2.10)$$

Figure 2.7 shows the main calculations that were done at each stage. The kernel is shown in the smaller square (2x2), and the input picture is shown in the larger square (4x4). Then, a product is shown as a number multiplied by both. This sum is used as an input value for the output feature map (Koushik, 2016; Bezdán & Džakula, 2019; Bouchard, 2022).

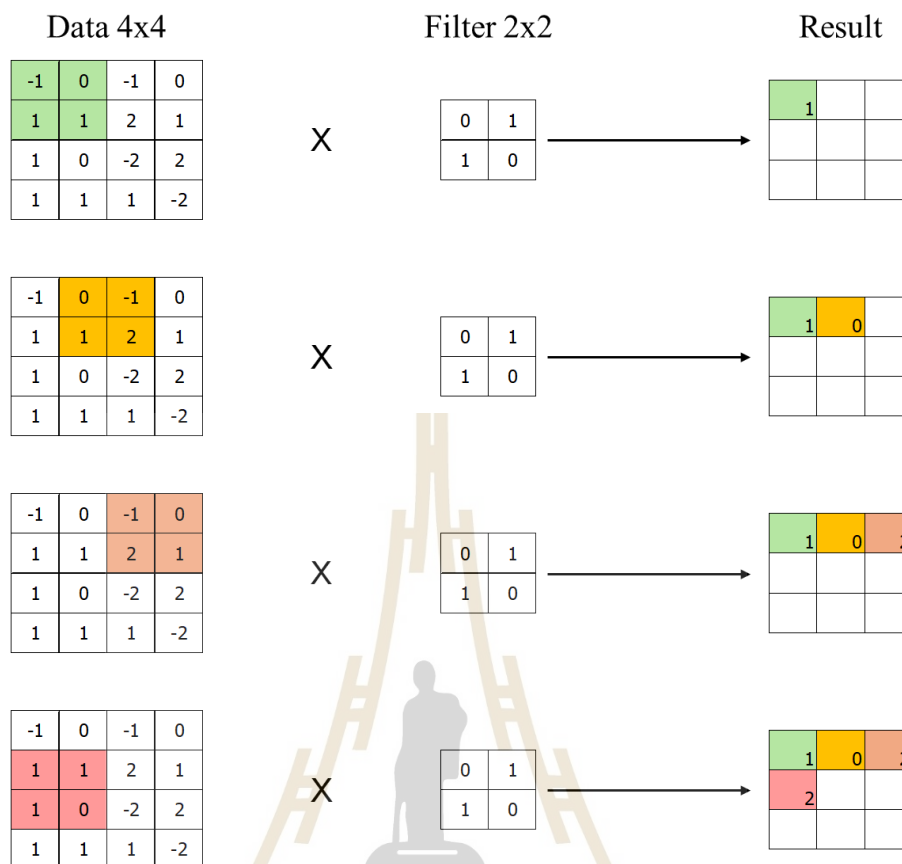


Figure 2.7 A picture that shows the main calculations.

In the example mentioned before, the kernel works with a stride of 1, which sets the step size across the input image in both the vertical and horizontal directions. It is important to note that this example does not add padding to the input image. Remember that this method is flexible; a different stride value can be chosen based on specific needs. One of the best things about choosing a higher stride value is that it makes the feature map less multidimensional. This change can be significant for managing the feature map's size and complexity that the convolutional process creates.

However, padding dramatically affects the size of the picture's borders. It differs from the border side, which changes significantly over time. When the picture gets bigger, the feature map gets bigger, too. Each filter could stand for a feature. If the filter moves over an image and does not find a match, it does not work. CNNs

uses these steps to find the best object-description filters. Figure 2.8 shows how the matrix can be set up to see the edges of a picture. These matrices are also called filters because they work like the filters usually used in image processing.

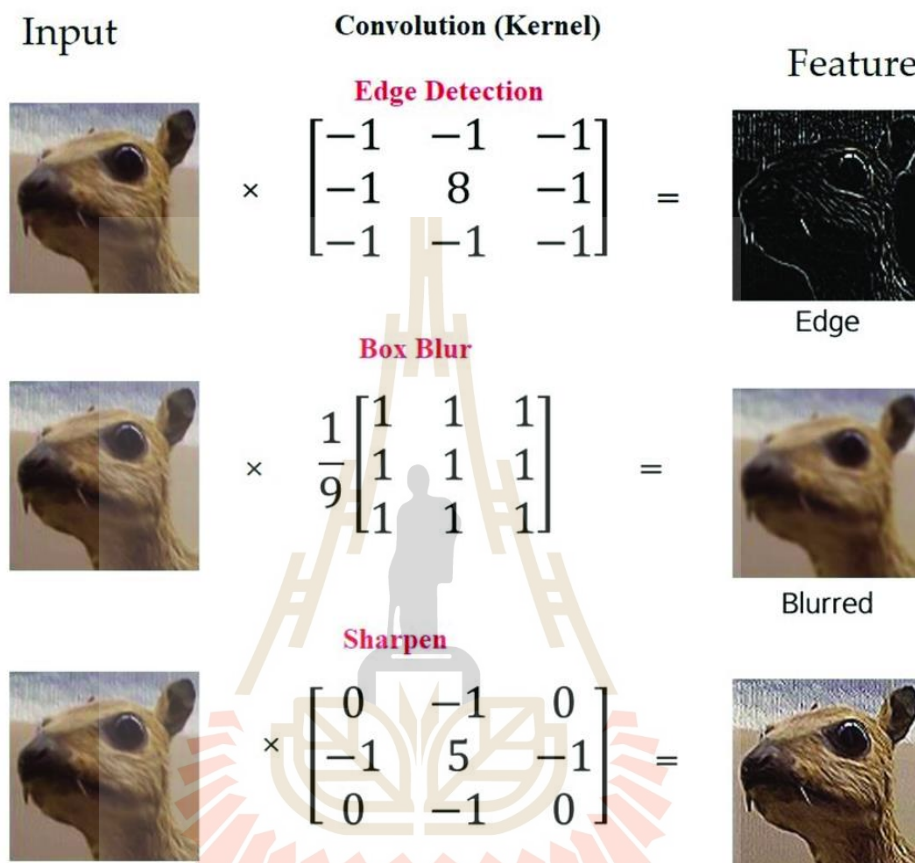


Figure 2.8 Effects of different convolution matrices

But at CNNs, these filters are used before the form filters that are better for the job at hand and are used during training.

Stride: In fact, CNNs offer more options that allow for further fine-tuning of settings while also mitigating some negative effects. One of these options is the stride. In the scenario described above, the next-layer node overlaps with its neighbors when examining the areas. The stride can adjust this overlap. Figure 2.9 shows a unique 6x6 image. The maximum output size it can achieve is 4x4 because the filter can only move by one node at a time. The outputs of the three left matrices, along with those of the three middle matrices and the three right matrices,

are shown in Figure 2.9. If the stride is set to two, the result will be a 3x3 output. This means that the total size of the output and the amount of overlap will decrease (Koushik, 2016; Dhillon & Verma, 2019; Edureka, 2022).

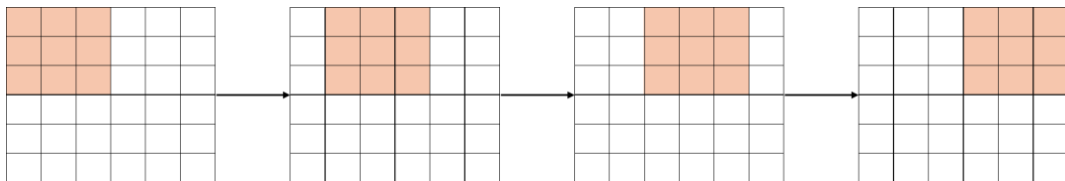


Figure 2.9 Stride 1, the filter windows move only one time for each connection.

Equation 2.11 shows a simple that leads to the output size, as seen in Figure 2.9. The size of the image ($D \times D$) and the size of the filter ($K \times K$) are used to figure this out.

$$output = 1 + \frac{D - K}{S} \quad (2.11)$$

where D is the input size, K is the filter size, and S is the stride size.

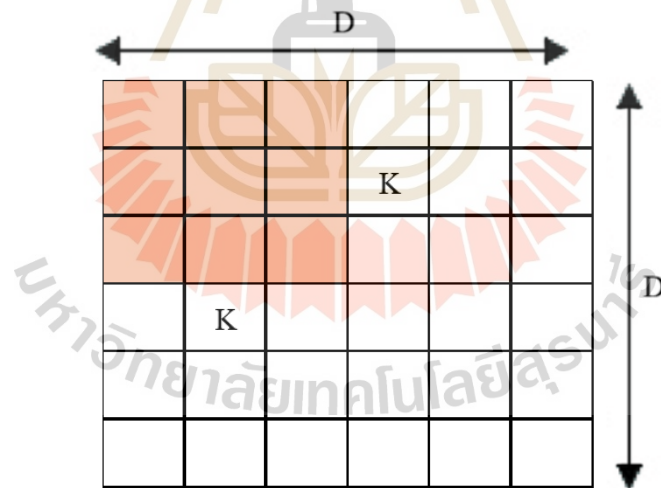


Figure 2.10 The effect of stride

Padding: One of the problems with the convolution step is the loss of detail at the edges of the image. These details may not be captured because the filter only picks them up when it moves over the edges. A simple and helpful solution to this problem is zero padding. By using zero padding, the edges of the image are padded

with zeros, allowing the filter to capture more information. Additionally, zero padding can be used to adjust the output size.

For example, in Figure 2.10, the output will be 4×4 , less than the input of 6×6 . This is because $D = 6$, $K = 3$ and stride 1 were used. However, incorporating a single layer of zero-padding results in a 6×6 output identical to the original input. In this scenario, the actual dimension D becomes 9. This adjustment is reflected in the modified formula, which accounts for the scenario without padding, as detailed in Equation 2.12.

$$\text{output} = 1 + \frac{D + 2P - K}{S} \quad (2.12)$$

Where P is the number of layers of zero-padding (for example, $P = 1$ in Figure 2.11), this padding idea keeps the network output size from getting smaller as the depth goes up.

0	0	0	0	0	0	0	0	0
0								0
0								0
0								0
0								0
0								0
0								0
0								0
0	0	0	0	0	0	0	0	0

Figure 2.11 Zero-padding

2.6.2 One Dimension Convolutional Neural Network (1D-CNNs)

A One -Dimensional Convolutional Neural Network (1D-CNNs) is a type of deep learning model specifically designed for sequence data, such as time series,

audio signals, and other types of data that can be represented as a one-dimensional array. Here is a detailed breakdown of the background theory and components of 1D-CNNs.

1D Convolution: The primary operation in a 1D-CNNs is the 1-dimensional convolution, where a kernel (or filter) slides over the input sequence. The convolution operation involves element-wise multiplication and summation of the input values with the kernel values to produce a feature map.

$$(x * w)(t) = \sum_{k=0}^{K-1} x(t+k) \cdot w(k) \quad (2.13)$$

Where:

x is the input sequence.

w is the kernel.

K is the kernel size.

Convolutional neural networks (CNNs) are a type of deep neural network (DNN) used to process data in multiple arrays, such as images. The key feature of CNNs is the use of convolutional layers, which apply a set of filters to the input data, extracting relevant features and patterns. This allows image-specific features to be incorporated into the network design, improving performance for image-based tasks and reducing configuration parameters. CNNs are based on four key concepts that exploit the properties of natural signals: local connections, shared weights, pooling, and multiple layers. CNNs are renowned for their significant computational demands, necessitating specialized hardware, particularly during the learning process. As a result, real-time applications on mobile devices and devices with limited power and memory resources should refrain from employing 2D CNNs. Furthermore, achieving a satisfactory level of generalization in deep CNNs requires a sufficiently large dataset for training. However, in many real-world applications involving 1D signals, acquiring labeled data can be challenging, making this approach less feasible. A modified version of CNNs, called 1D CNNs, has recently been developed. Studies have shown

that for some applications, 1D CNNs are advantageous and therefore preferable to their 2D counterparts for 1D signal processing. 1D CNNs use three main types of layers: convolutional layers, pooling layers, and fully connected layers. Figure 2.12 shows a streamlined 1D CNNs architecture.

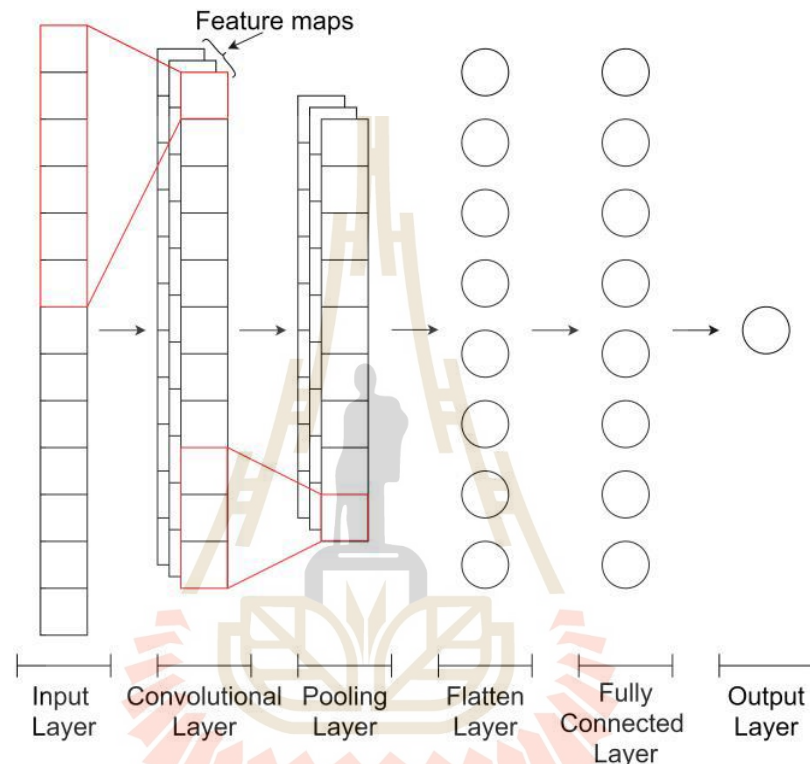


Figure 2.12 One-dimensional convolutional neural network (1D-CNNs) architecture for the timeseries forecasting model.

1D filter kernels start with a sequence of convolutions that pass through the activation function as a sum. This is followed by a pooling operation that reduces the output of every convolutional layer. This distinguishes 1D-CNNs from 2D-CNNs, which employ 2D matrices for feature maps and kernels. In the next phase, the fully connected (FC) layers help the CNNs layers "learn to extract" the characteristics needed for classification from the unprocessed 1D data. They create connections between every neuron and the neurons in the layer above, allowing for complex interactions and the fusion of data from different aspects. The final output is

produced by the FC layers by first performing a weighted sum of the inputs and then applying an activation function. Consequently, feature extraction and classification are merged into a single backpropagation process, making it possible to optimize and improve classification accuracy. This integration is a major benefit of 1D-CNNs. Additionally, because the primary operation of 1D CNNs is a series of 1D convolutions, which are inherently lower-dimensional, 1D-CNNs can reduce computational complexity more effectively than 2D-CNNs. A 1D-CNNs uses less memory and requires simpler filter operations with fewer parameters.

2.7 Tuning a Neural Network

2.7.1 Pooling

Typically, there is a subsampling or pooling layer next to each convolutional layer. This combination lowers the resolution of the feature map, making the output less likely to shift or become distorted (LeCun & Bengio, 1995). The following are the most commonly used pooling algorithms:

Max Pooling: Max pooling selects the brightest pixels in the image, focusing on the lighter pixels while ignoring the darker background. As shown in Figure 2.13, the pooling operation identifies the highest value for each window range, sends that value to the neuron, and then moves to the next layer.

Average Pooling: The average pooling method smooths out the image, which may result in the loss of sharp features. This pooling method is illustrated in Figure 2.14.

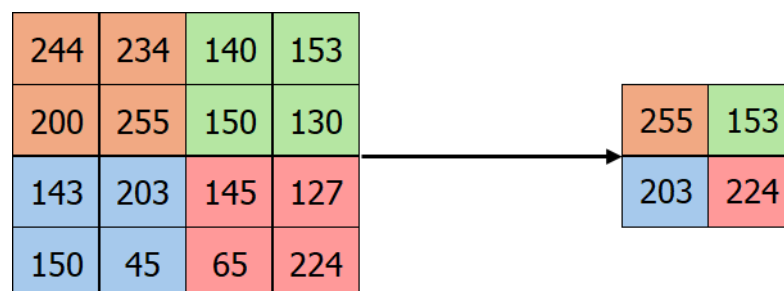


Figure 2.13 Max pooling

244	234	140	153	→	233	143
200	255	150	130			
143	203	145	127			
150	45	65	224			
					135	140

Figure 2.14 Average pooling

For example, Figure 2.15 shows a visual comparison between an original image of a purple flower and two processed versions of the same image using different pooling methods. These techniques are used in image processing to reduce the size of the image while retaining important features. The original image at the top is a detailed and clear photograph of the flower. To its bottom left, the image labeled “Average pooling” appears as a diminished rendition, with a noticeable decrease in resolution and a softened detail profile. This is indicative of the average pooling technique, where pixel values within a designated area are combined, assigning the mean value to the entire region. This results in a more uniform but less detailed image. Conversely, in the bottom right, the “Max pooling” image, despite its lower resolution, conserves more of the flower's intricate details and contours. This effect arises from the max pooling method, which selects the peak value from a specified pixel cluster to represent the whole area, thus maintaining more textural and edge clarity.

In conclusion, this set of images demonstrates the effects of two common pooling operations used in image processing and machine learning, particularly in CNNs. Average pooling results in smoother but less detailed images, whereas max pooling preserves more detail at the cost of losing some smoothness. These techniques are critical in reducing computational load and extracting robust features for tasks such as image classification and pattern recognition.



Figure 2.15 Average pooling and Max pooling (Dhuma, 2019)

2.7.2 Dropout

In 2014, Srivastava et al. introduced a technique called Dropout to prevent overfitting by randomly setting input units and dropping their related connections in the network during training. This means that the neurons will adapt to optimal weights that are less dependent on the weights and performance of other neurons. Dropout, as shown in Figure 2.16, helps improve the robustness and generalization of the network.

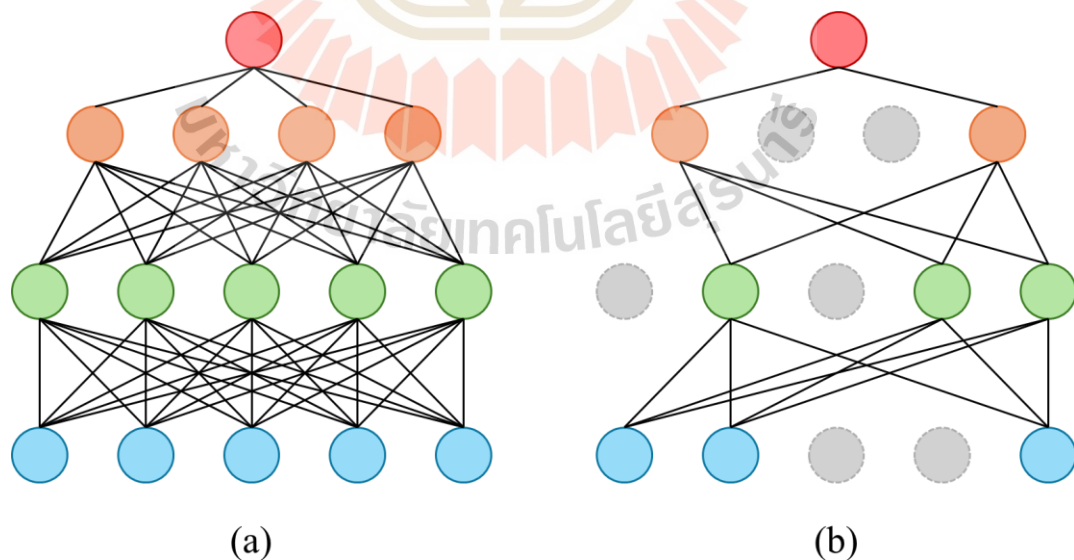


Figure 2.16 Dropout neural network model. (a) A standard neural network. (b) An example of a thinned net produced by applying dropout

2.7.3 Batch Normalization

In 2015, Batch Normalization (BN) was proposed by Ioffe and Szegedy (Ioffe & Szegedy, 2015). BN is used to make the training of neural networks faster and more stable, and to mitigate the problem of internal covariate shift.

It is called "batch" normalization because, during training, it uses the mean and standard deviation (or variance) of the values in the current batch to normalize the values coming in from each layer. During training, a batch normalization layer performs the following steps:

$$\text{Batch mean}(\mu) = \frac{1}{m} \sum_{i=1}^m O_i \quad (2.14)$$

$$\text{Batch variance}(\sigma) = \sqrt{\delta + \frac{1}{m} \sum_{i=1}^m (O_i - \sigma_i)^2} \quad (2.15)$$

Where:

O_i is the previous layer, m is the number of samples in the given batch.

σ is the batch standard deviation.

δ is a small number to make sure.

$\sigma > 0$ i.e., to make sure.

O_i from becoming undefined when divide it by zero.

BN speeds up learning by allowing higher learning rates and making the initial learning rates less critical. BN normalizes a feature along with other features in each batch, meaning that each normalized feature is not just a deterministic value. This effect reduces overfitting and sometimes eliminates the need for other regularization methods, such as Dropout (Srivastava et al., 2014; Ioffe & Szegedy, 2015).

2.8 Learning System

2.8.1 Machine Learning

Machine Learning (ML) is a system that can learn from examples by itself without the input of programmers. This breakthrough came with the idea that computers could learn from data to produce accurate results.

Machine learning can be combined with data and statistical tools to predict outcomes. The results obtained can be used for further insights within the organization. Machine learning is closely related to data mining and Bayesian predictive models. Computers take in data and use algorithms to find answers.

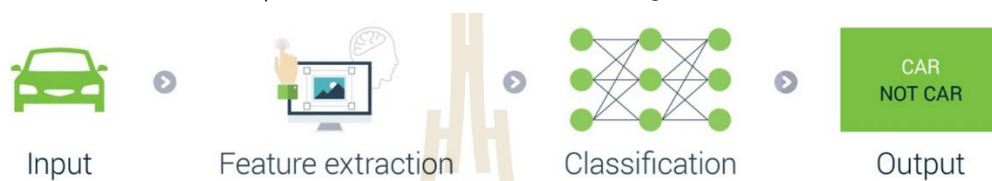


Figure 2.17 Architecture Machine learning.

Machine learning is a branch of artificial intelligence that focuses on developing algorithms and models that can learn from data and make predictions or decisions. The algorithms used for training are typically categorized as follows:

Supervised learning is a type of learning where data is provided with labeled examples and results. In this approach, the computer is trained to learn from input data and corresponding desired results provided by a teacher or supervisor. The computer then generates a model to make predictions based on this data. In supervised learning, the algorithm learns from the labeled examples to generalize patterns and relationships between the input data and their corresponding outputs. The goal is to create a purposeful model that can accurately predict results for new input data based on its learned knowledge. This type of learning is commonly used in tasks such as classification, where the model learns to assign input data into predefined categories, and regression, where the model learns to predict numerical values based on input features. Supervised learning is widely applied in machine learning due to its ability to make accurate predictions with labeled training data (Nasteski, 2017).

Unsupervised learning is a type of learning where the computer receives input data without corresponding desired outputs or labeled examples. It aims to mimic the functioning of the human brain more closely. The learning process uses statistical principles to analyze and group the data into different levels. This approach doesn't rely on labeled examples but instead looks for patterns or information within the data itself. Unsupervised learning is especially helpful when there is no labeled data available. The method is used in various domains, including exploratory data analysis, recommendation systems, and anomaly detection (Celebi & Aydin, 2016).

Reinforcement learning is a learning approach that involves trial and error to learn and determine the most effective course of action. It is often used when an agent learns how to interact with an environment to maximize a reward signal. Examples include learning to play games, optimizing product recommendations, and predicting customer behavior. In reinforcement learning, the agent learns through interactions with the environment. The model receives feedback through rewards or penalties based on its actions, enabling the model to learn and improve to seek the maximum overall reward and achieve an optimal solution. This learning allows the agent to learn and improve without human intervention, even when the data is complex and unstructured (Vidyasagar, 2023).

While supervised learning, as discussed in the previous section, encompasses a range of techniques and models, one of the most effective and widely used in this category is the Support Vector Machine (SVM) (Kadiri & Alku, 2019). This method stands out for its unique approach in solving classification problems, leveraging the principles of supervised learning to achieve high accuracy and efficiency.

2.8.1.1 Kernel Extreme Learning Machine (KELM)

Kernel Extreme Learning Machine (KELM) is a type of Extreme Learning Machine (ELM), which is a feed-forward neural network with a single hidden

layer. KELM differs from traditional ELM in that it uses kernel tricks to map input data into a higher-dimensional feature space before applying the weights of the hidden layer.

In KELM, the weights of the hidden layers are randomly generated, and the resulting layer weights are determined by solving the normalized least squares problem. This allows KELM to be trained much faster than traditional neural networks.

The kernel trick is a technique that allows modeling non-linear decision boundaries using linear combinations of input data in a higher-dimensional feature space. This makes KELM particularly suitable for processing non-linear data, such as images and videos.

KELM is used in a wide variety of applications, including image classification, time series prediction, and control systems. It has been proven to compete with other machine learning methods such as support vector machines and traditional neural networks.

KELM training process is faster than that of traditional neural networks, and it is especially suitable for non-linear data processing. It is used in various applications, including image classification and time series prediction.

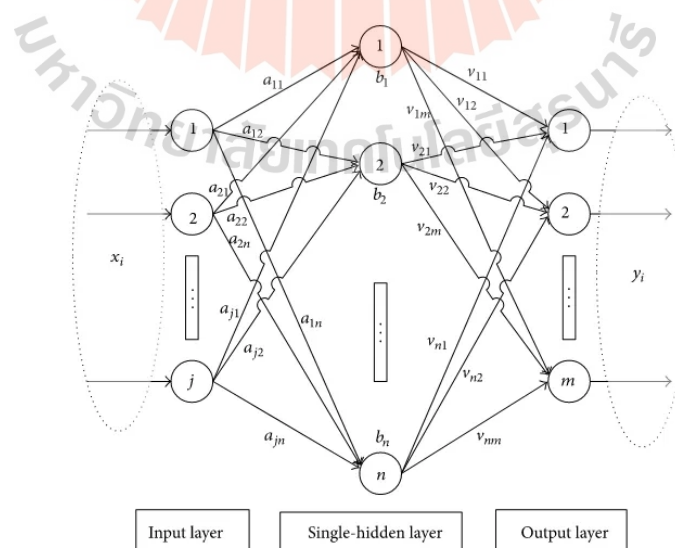


Figure 2.18 Structure of KELM

In KELM, kernel functions are directly utilized for feature capture. The kernel matrix is expressed using the following equation:

$$\Omega_{KELM} = HH^T \quad (2.16)$$

where H is the output matrix of the hidden layer. Ω_{KELM} is a kernel function.

$$\Omega_{KELM} = h(d_r).h(d_s) = K(d_r, d_s) \quad (2.17)$$

Since the general Moore-Penrose inverse is used to calculate the output weights, the output function of the KELM-based classifier can be described as follows:

$$f(d) = \begin{bmatrix} K(d, d_1) \\ K(d, d_2) \\ \vdots \\ K(d, d_N) \end{bmatrix} \left(\frac{1}{C} + \Omega_{KELM} \right)^{-1} T \quad (2.18)$$

In this context, 'T' denotes the target matrix (label), similar to SVM. 'I' represents the identity matrix, and 'C' denotes the regularization coefficient. The procedure for testing the decision between the normality class and the seizure class in EEG signals depends on the differences between the two classes as follows:

$$\Lambda_{KELM} (Y) = P(t_{circle} | f(Y)) - P(t_{line} | f(Y)) \quad (2.19)$$

2.8.1.2 Support Vector Regression (SVR)

Support Vector Regression (SVR) is a type of Support Vector Machine (SVM) algorithm commonly used for regression analysis. SVMs are powerful supervised learning algorithms primarily used for classification problems.

VMs aim to find the optimal hyperplane that best separates the two classes in the input data. A hyperplane is a flat subspace of dimension p-1 in a p-dimensional space, where p is the number of input features. The optimal hyperplane is the one that maximizes the margin, which is the distance between the hyperplane and the closest data points from each class, known as support vectors.

Similar to SVMs, SVR uses the concept of a hyperplane and margin but there are differences in their definitions. In SVR, the margin is defined as the error tolerance of the model, which is also called as the ϵ -insensitive tube. This tube allows some deviation of the data points from the hyperplane without being counted as errors. The hyperplane is the best fit possible to the data that fall within the ϵ -insensitive tube. The difference of SVM and SVR is summarized in the Figure 2.19.

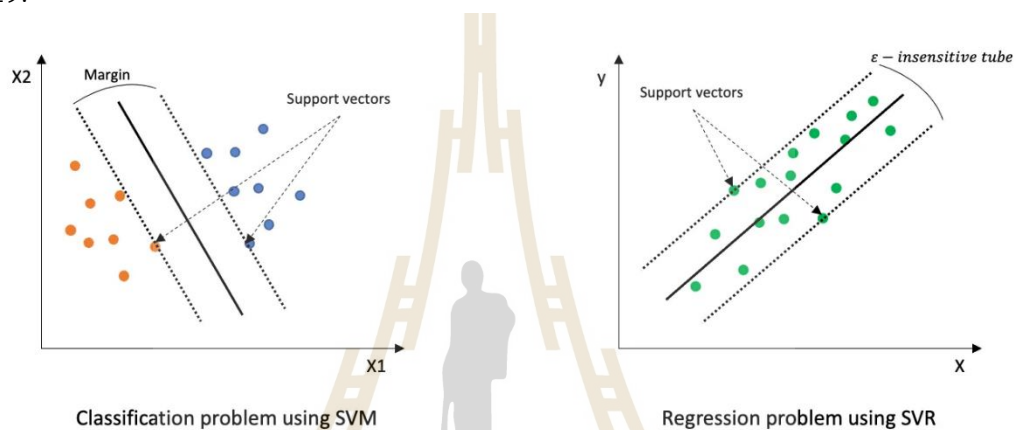


Figure 2.19 Support Vector Machine and Support Vector Regression

2.8.1.3 Extreme Gradient Boosting (XGB)

Extreme Gradient Boosting is a model that sequentially trains multiple Decision Trees, with each Decision Tree learning from the mistakes of the previous tree. This process does not involve using predictions continuously but rather focuses on iteratively improving the model. The training continues until a sufficient number of trees have been built and the model has effectively minimized the error patterns from previous trees.

XGBoost builds an ensemble of decision trees sequentially, where each new tree aims to correct the errors made by the previously trained trees. This process helps to improve the overall predictive accuracy of the model.

The XGB model is an implementation of gradient boosting to reduce the training loss function. The XGB consists of a set of decision trees. The model expression can be represented as follows:

$$\tilde{y}_i = \sum_{k=1}^K f_k(x_i) \quad (2.20)$$

where \tilde{y}_i represents the predicted value for input x_i , $f_k(x_i)$ is the k th tree.

Finally, the expression for OF is as follows:

$$OF = \sum_{i=1}^n L(\tilde{y}_i, y_i) + \sum_{k=1}^K \Omega(f_k) \quad (2.21)$$

where OF includes the regularization item Ω and the loss function L which denotes the gap between the predicted result \tilde{y}_i and the actual result y_i . The basic structure of XGB is shown in Figure 2.20 and provides a more comprehensive description and characterization of XGB.

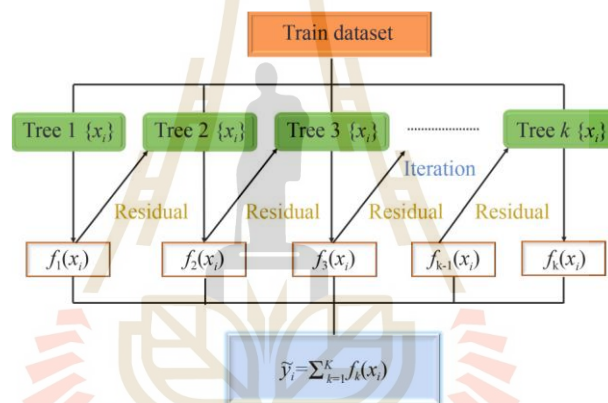


Figure 2.20 The basic structure of the XGB algorithm.

2.8.2 Deep hybrid learning

Deep hybrid learning is an advanced area of study in the field of artificial intelligence that merges various neural network architectures and machine learning techniques to create sophisticated, multi-functional systems. This approach seeks to harness the strengths of different models to improve performance, adaptability, and efficiency across a wide range of applications from image and speech recognition to complex decision-making tasks in healthcare and finance.

Traditional deep learning models, while powerful, often specialize in specific types of data or tasks. For instance, convolutional neural networks (CNNs) excel at processing spatial information such as images, whereas recurrent neural

networks (RNNs) are better suited for sequential data such as text or time series. However, many real-world problems involve data that is both complex and heterogeneous, encompassing various types and structures. Deep hybrid learning models address this challenge by combining these specialized networks in a cohesive framework that can process and analyze diverse data types effectively.

The rationale behind deep hybrid learning is not just to combine technologies but to create systems where the integration leads to emergent properties that surpass the capabilities of individual models. For example, a hybrid model combining CNNs and RNNs can extract features from video frames with the CNNs component and interpret sequences of these frames over time with the RNN component, leading to superior performance in video classification tasks.

Moreover, deep hybrid learning extends beyond the mere fusion of neural networks. It also involves integrating machine learning models with algorithms from other domains such as fuzzy logic, genetic algorithms, and reinforcement learning, creating a versatile toolkit for tackling the most challenging AI problems. These hybrid models can optimize their own architecture and adapt to new, unseen data more effectively than traditional models.

As businesses and industries generate increasingly complex datasets and demand more sophisticated decision-making tools, deep hybrid learning stands out as a pivotal innovation. It not only promises enhanced accuracy and adaptability but also opens up new possibilities for automation and intelligent system design in every sector of the economy. The goal of this introduction is to explore the principles, methodologies, and applications of deep hybrid learning, providing a foundation for understanding its role and potential in advancing artificial intelligence.

2.8.3 Architecture of Deep Hybrid Learning systems

1. Feature extraction using CNNs: The first stage involves a CNNs architecture that processes raw voice data, represented as time-series frames. This

CNNs, often with multiple convolutional layers, is adept at automatically extracting a rich set of features from the raw input without the need for manual feature engineering.

2. Extracted features from the CNNs are then fed into an SVM classifier. SVMs are known for their effectiveness in high-dimensional spaces and their ability to find the optimal boundary between classes with a maximum margin, which is crucial for medical diagnosis where the distinction between healthy and pathological samples is often subtle.

2.8.4 Details of a Deep Hybrid Learning model.

1. Audio samples are first segmented into frames, which are then transformed into a suitable form, such as spectrograms, for CNNs processing.

2. CNNs feature learning: The CNNs layers apply various convolution and pooling operations to the input, progressively abstracting and enhancing salient features. Each convolutional operation is defined by kernels of specific sizes, followed by pooling operations that reduce dimensionality and focus on the most relevant features.

3. Integration with machine learning: The high-level features extracted by the CNNs undergo dimensionality reduction, typically through a global average pooling operation. These condensed features, which retain the essential information, are then presented to the machine learning classifier.

4. Classification and decision making: The machine learning classifier processes the features to classify the voice as healthy or pathological. The decision is based on the learned hyperplane that best separates the feature space into distinct classes.

2.8.4 Implementation and training

1. Model training: The training process begins with an initial phase where the CNNs layers are trained using a labeled dataset. This dataset consists of a

variety of voice recordings, including those from individuals with various voice disorders.

2. Integration and fine-tuning: After the CNNs training, the extracted features are used to train the machine learning classifier. This phase may include fine-tuning the CNNs in conjunction with machine learning training to better align the feature extraction with the classification goals.

3. Testing and validation: The fully trained hybrid model is then validated and tested using separate datasets to ensure its diagnostic accuracy and generalizability to unseen data.

2.8.5 Implementation Physical Information

1. Incorporating Physical Data for Enhanced Blood Pressure Prediction

Several studies have emphasized the significance of incorporating physical data such as age, gender, height, weight, and heart rate (HR) to improve the accuracy of blood pressure estimation models. The integration of such data helps personalize the estimation process, capturing individual characteristics that directly influence blood pressure values.

2. Data Augmentation

Physical data augmentation involves increasing the volume and variety of information used in training models. Features like age, gender, height, weight, and HR provide additional context to the machine learning models, enabling them to identify patterns and correlations more effectively. This augmentation is crucial for developing robust models that can generalize well across different populations.

3. Personalization and Individual Characteristics

Incorporating physical data helps capture the unique characteristics of individuals, which significantly impact blood pressure values. For example:

1 Age: Blood pressure tends to change as people age due to the natural aging process, which includes the stiffening of arteries.

2. Gender: Men and women may have different average blood pressure levels due to hormonal differences and physiological factors.

3 Height: Height is believed to indicate the size of blood vessels and the volume of blood circulating in the body, influencing blood pressure values.

4. Weight: Weight affects blood pressure through the distribution of body fat. Overweight individuals may place additional strain on the heart and blood vessels, affecting blood pressure.

5. Heart Rate (HR): Heart rate is a vital indicator of the heart and vascular system's functioning, directly influencing blood pressure values. HR is directly related to the cardiovascular system's functioning, and its inclusion helps in understanding the dynamics of blood pressure variations.

4. Improving Model Accuracy with Physical Data

Physical data helps capture the relationship between various health parameters and blood pressure. Studies have shown that models incorporating such data are better at estimating blood pressure because they can consider the physiological context of each individual. This leads to more personalized and accurate predictions.

The integration of physical information in blood pressure estimation models represents a significant advancement in personalized healthcare. By leveraging comprehensive data, these models can achieve higher accuracy and reliability, ultimately leading to better management and treatment of hypertension and related cardiovascular conditions.

2.9 K-Fold Cross-Validation

In the pursuit of creating robust and generalizable machine learning models, the method of evaluating and validating these models plays a crucial role. K-fold cross-validation stands as a cornerstone technique in the field of machine learning

for assessing how the results of a statistical analysis will generalize to an independent data set. This introduction lays the foundation for a detailed exploration of the K-fold cross-validation process, its importance in model training, the challenges associated with its implementation, and best practices that enhance its effectiveness.

K-fold cross-validation addresses one of the central problems in machine learning: model overfitting. Overfitting occurs when a model is too closely fit to a limited set of data points and fails to generalize well to new data. The essence of K-fold cross-validation is to test the model's ability to predict new data based on various subsets of the available data, thereby minimizing errors like overfitting.

The fundamental concept behind K-fold cross-validation involves dividing the entire data set into ' K ' smaller sets or folds. The procedure follows a simple but powerful iterative process: for each unique group, the given fold of data serves as the test set (or the validation set), while the remaining $K-1$ folds make up the training set. The model is then trained on the $K-1$ folds and validated on the one remaining fold. This process is repeated K times, with each of the K folds used exactly once as the validation data. The K results can then be averaged (or otherwise combined) to produce a single estimation. The advantage of this method is that all observations are used for both training and validation, and each observation is used for validation exactly once.

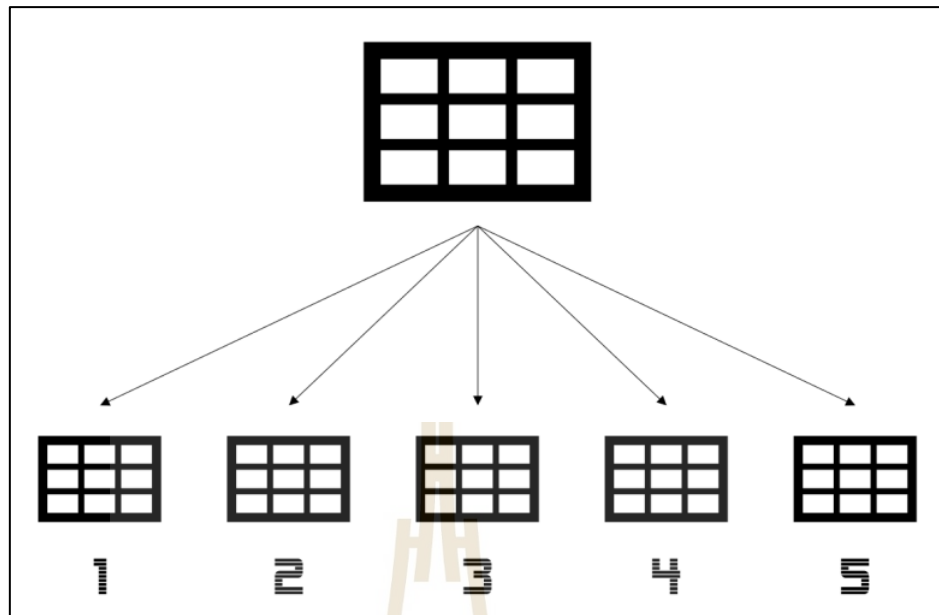


Figure 2.21 Data set arrangement format using K-Fold Cross Validation method.

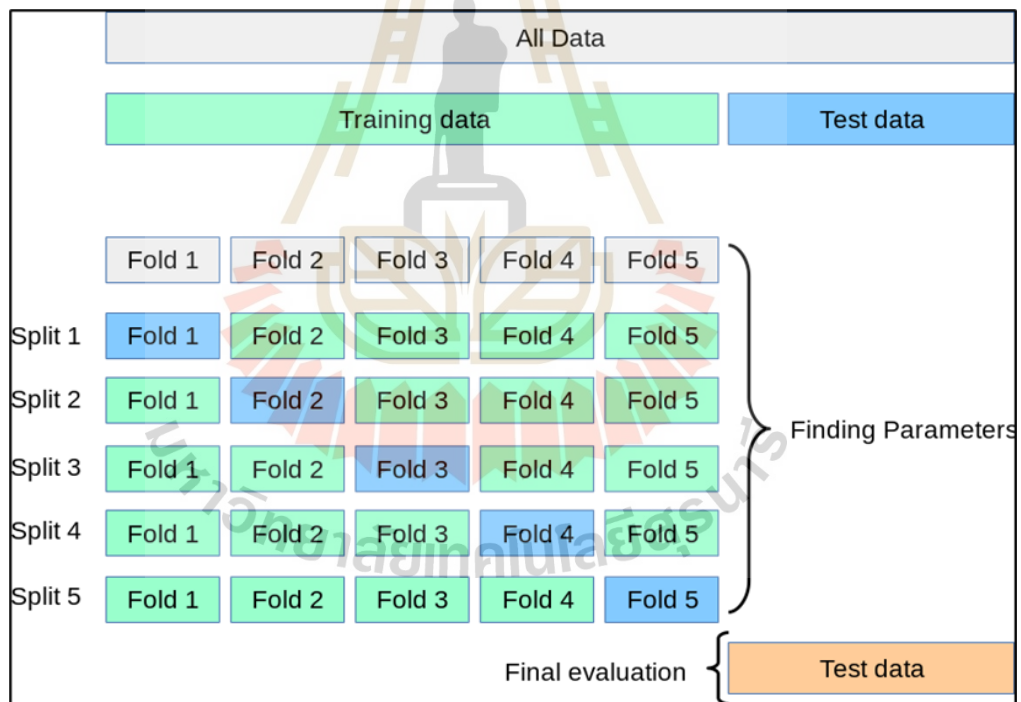


Figure 2.22 Shows the training and testing datasets.

2.10 Mean Absolute Error (MAE)

In the complex realm of predictive modeling and machine learning, evaluating a model's performance is pivotal to ensuring its reliability and accuracy in

real-world applications. Among the various metrics available for this purpose, Mean Absolute Error (MAE) serves as a critical tool. This introduction offers a comprehensive overview of MAE, discussing its mathematical formulation, significance in model assessment, and practical implications in various predictive modeling contexts.

Mean Absolute Error (MAE) is a measure used to quantify the accuracy of a model in regression tasks. It calculates the average magnitude of the errors in a set of predictions without considering their direction. It aggregates the absolute differences between predictions and actual observations, providing a clear numerical measure that represents the average error magnitude. MAE is especially valuable because it is easy to understand and interpret, making it a practical choice for many predictive modeling scenarios.

The formula for MAE is straightforward:

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.22)$$

where n is the number of observations, y_i is the actual value of the observation, and \hat{y}_i is the predicted value. The absolute values of the differences $|y_i - \hat{y}_i|$ ensure that all errors are treated equally, making MAE a robust measure against outliers and skewed data.

This metric is particularly favored in real-world applications where large errors are to be avoided. It helps in understanding the 'typical' error magnitude, which is critical for businesses and industries where decisions are based on predictive insights. Furthermore, the scale of MAE is the same as the data being predicted, which aids stakeholders in forming concrete interpretations of the model's performance.

The introductory chapter will explore how MAE contrasts with other metrics like Mean Squared Error (MSE) and why choosing the right error metric is crucial, depending on the specific characteristics of the data and the impact of prediction

errors on decision-making processes. Subsequent sections will delve into the applications of MAE in various industries, its limitations, and how these can be mitigated.

Through this examination, the thesis aims to provide a deep understanding of Mean Absolute Error, showcasing its importance and utility in guiding the development of robust predictive models. This will include discussions on when MAE should be preferred over other metrics and how it can be effectively implemented in practical modeling scenarios to ensure that the models perform optimally according to the expectations and requirements of different stakeholders.

2.11 Standard Deviation (STD)

A standard deviation (STD) is a measure of how dispersed the data is in relation to the mean. A low or small standard deviation indicates that data points are clustered tightly around the mean, while a high or large standard deviation indicates that data points are more spread out. A standard deviation close to zero indicates that data points are very close to the mean, whereas a larger standard deviation indicates that data points are spread further away from the mean.

Standard deviation is a crucial statistical measure that quantifies the amount of variation or dispersion in a set of values. STD can be used to assess the variability of predictions compared to actual measurements, providing insight into the precision and reliability of a model.

Definition and Formula

The standard deviation of a dataset is defined as the square root of the variance. The variance itself is the average of the squared differences between each data point and the mean of the dataset. Mathematically, the standard deviation (σ) is represented as:

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (X_i - \mu)^2}{N}} \quad (2.23)$$

where

X_i is each individual value in the dataset.

μ is the mean of the dataset.

N is the number of values in the dataset.

Calculation Steps

1. Compute the Mean (μ)

$$\mu = \frac{1}{N} \sum_{i=1}^N X_i \quad (2.24)$$

Calculate the average of all the data points.

2. Compute the Variance

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (X_i - \mu)^2 \quad (2.25)$$

3. Compute the Standard Deviation.

$$\sigma = \sqrt{\sigma^2} = \sqrt{\frac{\sum_{i=1}^N (X_i - \mu)^2}{N}} \quad (2.26)$$

Take the square root of the variance to get the standard deviation.

2.12 Techniques for estimation of blood pressure values

In general, heart recordings include many data points. Redundant data can slow down the classification process and often result in inaccurate outcomes. To enhance performance and reduce dimensionality, feature extraction techniques are often employed. This step involves extracting salient features of the data signal and involves a process to format and group the data appropriately. This adjusted data is then used to estimate blood pressure values.

2.12.1 Bandpass filter

The analysis of phonocardiograms, also known as phonocardiography, is a critical component in diagnosing cardiovascular diseases. A vital step in this analysis involves filtering the phonocardiogram signals to isolate relevant frequency components. A bandpass filter is commonly used in this process to remove unwanted noise and emphasize the frequency range that contains significant information about the heart's activity.

Bandpass Filter

A bandpass filter is a critical component in signal processing used to isolate a specific range of frequencies from a broader spectrum of signals. Here's a detailed look at the background theory of bandpass filters:

Definition:

A bandpass filter allows frequencies within a certain range (the passband) to pass through while attenuating frequencies outside this range (the stopbands).

Frequency Response:

The frequency response of a bandpass filter shows how the amplitude of the output signal varies with frequency. It typically has three main regions:

1. Passband: The range of frequencies that the filter allows to pass with minimal attenuation.
2. Stopbands: The ranges of frequencies that the filter attenuates significantly.
3. Transition bands: The regions between the passband and stopbands where the filter's response transitions from pass to stop.

Design Parameters:

Center Frequency (f_0): The midpoint frequency of the passband.

Bandwidth (BW): The width of the passband, defined as the difference between the upper and lower cutoff frequencies.

Quality Factor (Q): A measure of how selective the filter is, defined as the center frequency divided by the bandwidth ($Q = \frac{f_0}{BW}$).

Mathematical Representation

The transfer function:

$H(s)$ of a bandpass filter in the Laplace domain (for analog filters) can be represented as:

$$H(s) = \frac{s \cdot w_0}{s^2 + \frac{w_0}{Q} s + w_0^2} \quad (2.27)$$

where

s is the complex frequency variable.

w_0 is the angular center frequency.

Q the quality factor.

In the z-domain (for digital filters), the transfer function $H(z)$ can be represented as:

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^M a_k z^{-k}} \quad (2.28)$$

where

b_k and a_k are the filter coefficients.

A bandpass filter allows frequencies within a specific range to pass through while attenuating frequencies outside this range. In the context of phonocardiogram analysis, the bandpass filter is designed to focus on the frequency range where most of the phonocardiogram information is concentrated, typically between 60 Hz and 500 Hz.

Importance in Phonocardiogram Analysis

Noise Reduction: Phonocardiogram recordings often contain various types of noise, including environmental noise and artifacts from body movements. A

bandpass filter helps to reduce this noise, making the phonocardiograms clearer for analysis.

Frequency Isolation: Phonocardiograms, such as the first phonocardiogram (S1) and second phonocardiogram (S2), have distinct frequency components. By applying a bandpass filter, these components can be isolated, making it easier to analyze the timing, intensity, and quality of the phonocardiograms.

Signal Enhancement: Enhancing the relevant frequency range improves the accuracy of automated phonocardiogram analysis systems. This enhancement is crucial for the reliable detection of murmurs and other abnormal phonocardiograms that may indicate underlying cardiovascular issues.

Applications

Clinical Diagnosis: Bandpass filtering is used in digital stethoscopes and other diagnostic tools to provide clearer phonocardiogram recordings for clinicians.

Research: Researchers use bandpass filters to study phonocardiogram characteristics and develop algorithms for automated diagnosis systems.

Telemedicine: In remote monitoring systems, bandpass filters help ensure that transmitted phonocardiogram data is of high quality, facilitating accurate remote diagnosis.

Detection of Heart Murmurs: Studies have shown that bandpass filtering phonocardiogram signals significantly improves the detection of heart murmurs, which are typically found in the frequency range of 200 Hz to 600 Hz. Automated Classification, Research on automated classification of phonocardiograms has demonstrated that preprocessing with a bandpass filter enhances the performance of machine learning models in distinguishing between normal and abnormal phonocardiograms.

2.12.2 A Convolutional Neural Network (CNNs)

A Convolutional Neural Network (CNNs) is a type of artificial neural network designed to process data with a grid-like topology, such as images. A CNNs is

typically composed of multiple layers. Including the input layer One layer or several layers Convolutional One or more layers, combining layers, and one or more fully connected layers.

A CNNs is a type of neural network designed to process grid-like data, such as images. A CNNs consists of several layers, including convolutional, pooling, and fully connected layers. It is used to extract features from the input data and perform prediction or classification. It is widely used for image and video processing tasks and has been successful in many applications.

2.12.3 Machine Learning

In the realm of estimating blood pressure values using machine learning, several advanced techniques have been developed. One such technique is the Kernel Extreme Learning Machine (KELM), a powerful variant of the Extreme Learning Machine (ELM). Additionally, Support Vector Machines (SVM) have been effectively utilized for this purpose due to their robustness and accuracy in handling non-linear data. Furthermore, Extreme Gradient Boosting (XGB) has proven to be highly effective in providing accurate predictions by leveraging the boosting technique to sequentially improve model performance.

Kernel Extreme Learning Machine (KELM)

Kernel Extreme Learning Machine (KELM) is a type of Extreme Learning Machine (ELM), which is a feed-forward neural network with a single hidden layer. Unlike traditional ELM, which uses random weights and biases, KELM leverages kernel tricks to map input data into a higher-dimensional feature space. This mapping allows KELM to handle complex, non-linear relationships in the data more effectively.

Key Features

1. Random Weights in Hidden Layers:

In KELM, the weights of the hidden layers are randomly generated, which simplifies the training process significantly.

2. Kernel Trick:

The kernel trick is a mathematical technique that enables the transformation of input data into a higher-dimensional feature space without explicitly computing the coordinates in that space. This allows KELM to model non-linear decision boundaries using linear combinations of input data in this transformed feature space. Common kernels used include Radial Basis Function (RBF) and polynomial kernels.

3. Training Speed:

KELM can be trained much faster than traditional neural networks because it only requires solving a simple linear system. The resulting output weights are determined by solving a normalized least squares problem, which is computationally efficient.

By incorporating KELM into the methodology for estimating blood pressure values from PCG signals, this approach leverages its strengths in handling non-linear relationships and rapid training, providing a robust and efficient solution to this challenging problem. This technique ensures that the model can capture the complex patterns in PCG signals that are indicative of blood pressure levels, leading to accurate and reliable predictions.

Support Vector Machine (SVM)

Support Vector Machine (SVM) is a powerful supervised learning model used for classification and regression tasks. In the context of blood pressure estimation from PCG signals, SVM can be employed to create a robust model that accurately predicts blood pressure values.

Key Features:

1. Hyperplane and Margins:

SVM works by finding the optimal hyperplane that best separates the data points of different classes. In the case of regression, it finds the hyperplane that best fits the data while maintaining a margin of tolerance for error.

2. Kernel Trick:

Similar to KELM, SVM uses kernel functions to transform the input data into a higher-dimensional space, enabling the model to handle non-linear relationships. Common kernels include Radial Basis Function (RBF), polynomial, and linear kernels.

3. Support Vectors:

Support vectors are the data points closest to the hyperplane, which influence its position and orientation. These points are crucial in defining the decision boundary.

By integrating SVM into the methodology for estimating blood pressure values from PCG signals, the approach benefits from the model's robustness and ability to handle non-linear data. This technique ensures that the model can accurately capture the intricate patterns in PCG signals that correlate with blood pressure levels, leading to reliable predictions.

Extreme Gradient Boosting (XGB)

1. Basic Concept

XGBoost builds an ensemble of decision trees sequentially, where each new tree aims to correct the errors made by the previously trained trees. This iterative process helps to improve the overall predictive accuracy of the model.

2. Objective Function

The objective function in XGBoost consists of two main components: the loss function and the regularization term.

Loss Function: Measures the difference between the predicted values and the actual values, commonly using mean squared error for regression tasks.

Regularization Term: Prevents overfitting by penalizing model complexity, such as the number of leaves and the L2 norm of leaf weights.

3. Gradient Descent

XGBoost minimizes the objective function using gradient descent. In each iteration, a new tree is fitted to the negative gradient of the loss function, also known as the residual error.

The methodology of Extreme Gradient Boosting involves building an ensemble of decision trees sequentially, optimizing the objective function using gradient descent, and incorporating regularization techniques to prevent overfitting. Its capabilities for handling missing data and using parallel and distributed computing make XGBoost a powerful tool for various machine learning tasks.

2.12.4 Tolerance calculation

1. Absolute error is the amount of difference between the actual value and the value obtained from the measurement. Can be found from the equation.

2. Mean absolute error (MAE). The average of the absolute error. According to the number of tests Can be found from the equation.

3. Standard Deviation (STD) is a measure of how dispersed the data is in relation to the mean. Low, or small, standard deviation indicates data are clustered tightly around the mean, and high, or large, standard deviation indicates data are more spread out.

2.13 Literature Review

To gain an understanding of the considerations for estimating blood pressure values. Using information obtained from a sample group that recorded values from locations where phonocardiograms could be received from the sample group To study the sound wave signals of the heart and blood vessels or Cardiovascular. and study techniques for estimating blood pressure values that occur Method for estimating blood pressure from the sound of the heart. Study the direction of literary reviews. and related research To understand the structure and working principles of

phonocardiogram generation related to blood pressure. and estimation techniques For knowledge and use as a guideline for designing the following research. From the study there is a review of past literature and various related research works. The researcher therefore divided it into 1) research that uses other health data to estimate blood pressure values. 2) research that uses blood pressure estimation using machine learning techniques.

2.13.1 Research that uses other health data to estimate blood pressure.

Dastjerdi, Amirhossein Esmaili, Mohammad Kachuee, and Mahdi Shabany. conducted a study titled Non-invasive Blood Pressure Estimation Using Phonocardiogram . This paper introduces a new method for estimating blood pressure (BP) based on transit time. of pulse (PTT) to achieve this goal. Hardware devices are designed for precise sampling of signals. A Phonocardiogram (PCG) and Photoplethysmogram (PPG) are used to obtain the PTT value. A force sensing resistor (FSR) is also placed under the cuff of the BP reference device to record the exact moment of the measurement. The PTT-BP model is derived stepwise. Calibration episodes involve supervised exercise for each individual. The proposed method was tested on 24 subjects, and the results showed that BP can be reliably estimated using PCG for PTT measurement and the proposed model. The use of low-cost PCG-based hardware makes the proposed method a promising solution for ubiquitous BP estimation in portable healthcare devices.

PAUL A. OBRIST, KATHLEEN C. LIGHT, JAMES A. McCUBBIN, J. STANFORD HUTCHESON, and J. LEE HOFFER Study of the relationship between pulse transit time (PTT) and both systolic blood pressure. A (SBP) and diastolic blood pressure (DBP) were studied in 114 humans in both resting and stressed states. The results showed a consistent relationship between PTT and SBP, but not with DBP, when measuring blood pressure noninvasively and invasively. In addition, pharmacological blockade of cardiac sympathetic nerves attenuated the relationship between PTT and SBP

Yuichirou Yano studied whether combining home blood pressure (HBP) and ambulatory blood pressure monitoring (ABPM) with office blood pressure (OBP) can predict changes in cardiovascular biomarkers during hypertension treatment. The study involved 252 hypertensive patients who had their OBP, HBP, ABPM, and cardiovascular biomarkers (urinary albumin excretion (UAE) and brain natriuretic peptide (BNP)) measured before and after six months of treatment with candesartan and thiazide-diuretics. The results showed significant reductions in all blood pressure measurements (OBP, HBP, daytime, and nighttime BP) and UAE levels after treatment. BNP levels decreased only in patients taking diuretics. Changes in home systolic BP and nighttime SBP were independent and significant predictors of changes in UAE. Nighttime SBP changes were important for predicting changes in BNP. Patients who reduced all three SBP values showed significant reductions in UAE, and those who lowered their SBP both at work and at night had a more significant decrease in BNP. The study suggests that HBP and ABPM, particularly nighttime SBP, provide additional information for predicting treatment-induced changes in cardiovascular biomarkers when used alongside office SBP measurements.

Tozawa, Masahiko conducted a study aimed at comparing the pulse pressure (PP) values in patients receiving chronic hemodialysis with those in healthy individuals. A large group of hemodialysis patients and controls matched for age, gender, diabetes, and body mass index was studied. The results showed that at any level of mean arterial pressure (MAP), hemodialysis patients had higher systolic blood pressure (SBP) than diastolic blood pressure. (DBP) was lower and PP was higher compared to the control group. The mean difference in PP was 19.2 mm Hg. Studies suggest that increased PP in hemodialysis patients may be an independent risk factor for cardiovascular events. And further research is needed to determine whether increasing PP reduction improves the prognosis for these patients.

Rastegar, Solmaz, Hamid GholamHosseini, and Andrew Lowe studied that blood pressure (BP) monitoring is important in detecting and controlling the risk of

cardiovascular disease (CVDs) in patients. Blood pressure monitoring (ABP) of the patient is important in the outpatient environment. And this can be done using the latest mobile health technology. In the past few years There is a push to find alternatives to traditional BP monitoring methods, which are often cumbersome and inconvenient. They use equipment that is easy to use, fast and cost-effective. However, many of these techniques are still in prototype stages and face various challenges. in compliance with clinical standards This review considers continuous and cuffless blood pressure monitoring methods. These include pulse transit time (PTT), ultrasound, pulse arrival time (PAT) and machine learning, the accuracy, portability and ease of use of these technologies. The potential integration into wearable healthcare systems is also analyzed. Finally, recommendations are made for the future direction of the field.

Peng, Rong-Chao described a new approach for early detection of cardiovascular disease through continuous blood pressure monitoring using phonocardiogram signals received from a smartphone microphone. The experiment was conducted on 32 healthy individuals using a cold pressor test. And the experimental results show that the support vector machine regression model is effective in predicting systolic, diastolic, and mean blood pressure values with high accuracy. The average correlation coefficients were 0.707, 0.712, and 0.748, respectively, and the mean error was less than 5 mmHg with a standard deviation less than 8 mmHg. These findings suggest that this technique has potential in pressure measurement. Continuous blood flow and no cuff required and can be used in home health services.

2.13.2 Research on pressure estimation using machine learning techniques.

Thambiraj Geerthy described the objective to develop a continuous and accurate blood pressure (BP) monitoring system. This is due to the complexity of controlling and controlling high blood pressure. Existing continuous non-invasive

techniques face challenges such as proper sensor placement and frequent calibration according to the individual. The authors propose a new algorithm that uses time-domain features from ECG and PPG signals to estimate BP. The method is evaluated using performance metrics such as MAE, RMSE, and bias, and the best features are determined using a deterministic algorithm. Genetics with random forest models The results show that the model built with ECG and PPG features outperforms the model built with PPG features alone. And the importance of ECG features for BP has been confirmed. This information can help reduce computational costs and errors in BP estimation.

Tahar Omari described that phonocardiogram (PCGs), which are recordings of phonocardiograms, have several benefits over traditional auscultation methods. One of the main advantages is that spectral and frequency data can be replayed and analyzed. Despite its potential benefits, PCG is not widely used in diagnostics due to vocal damage. Sounds from various sources such as lung sounds and breathing, environmental noise and the sound of blood flow. The murmurs (called murmurs) can interfere with the PCG signal. The murmurs contain important information about the heart's hemodynamics and can be used to detect heart valve disease. An automated system for processing heart murmurs could be a useful tool in diagnosing heart disease using a simple electronic stethoscope. However, before any automation can occur, the PCG signal must be segmented to separate the murmurs. Effective segmentation algorithms require effective denoising techniques. And Wavelet Transform (WT) has been found to give good results in such cases. The main challenge of WT is selecting the level of decomposition and parent wavelet. This paper presents a new method for automatically selecting the best mother wavelet and its level of decomposition for use in phonocardiogram extraction. Results from simulated and real PCG signals show that the proposed approach can select the best decomposition level and the best mother wavelet for extracting the main components of PCG noise (S1 and S2) from different types of drones.

Irene, D. Shiny presented a hybrid method to improve heart disease diagnosis through automatic prediction. The method combines fuzzy k-medoids clustering attribute-based weighting (FKMAW) and deep networks and advanced machine learning (DBNKELM) to improve the medical diagnosis process. The input attributes are first weighted using FKMAW, and then the medical data classification performance is improved by transforming the non-linearly separated data into a linearly separable data set. The weighted attributes are used in a heart disease prediction project based on a regression model combining DBNKELM. The results show that the FKMAW + DBNKELM method has good performance in dealing with the challenges of classifying medical data for both datasets. 6 sets

Hsiao, Chun-Chieh developed a continuous blood pressure measurement device that uses both phonocardiograms. (phonocardiogram, PCG) and blood flow signals (photoplethysmography, PPG). PCG records the first and second phonocardiograms (S1 and S2) during systole and diastole, while PPG measures changes in blood volume in the arteries. This affects the amount of light that is reflected and received to detect the heart's pulse wave. The device filters noise from the signal using Finite Impulse Response (FIR) and Discrete Wavelet Transform (DWT) and calculates three properties (peaks S1, S2, and PPG) including vascular transit time (VTT), ejection time. (ET) and heart rate (HR) blood pressure were estimated using a regression analysis model that included cardiac parameters and physiological factors such as gender, age, height, and weight. The results of this study indicate that blood pressure The estimated systolic blood pressure had an error of 6.67 ± 8.47 mmHg, which is close to the AAMI standard of 5 ± 8 mmHg, indicating the feasibility of estimating blood pressure using PCG and PPG signals.

Esmaelpoor, Jamal, Mohammad Hassan Moradi, and Abdolrahim Kadkhodamohammadi present an invasive blood pressure (BP) measurement technique. This study proposes a multi-step model using deep neural networks to estimate blood pressure. systolic and diastolic signals using Photoplethysmogram

(PPG) Method: The proposed model consists of two main components. using two consecutive steps The first step consists of two convolutional neural networks (CNNs) to extract morphological features from each PPG segment and then to estimate systolic and diastolic BPs separately. The second step uses long-term short-term memory (LSTM) to capture temporal dependencies. Additionally, the method incorporates dynamic relationships between systolic and diastolic BPs to improve accuracy. Results: The proposed multistep model was evaluated on 200 subjects using British standards. Hypertension Society (BHS) and the Association for the Advancement of Medical Instrumentation (AAMI). The results show that the performance of our model meets the requirements of the AAMI standard. In addition, according to the BHS standard, it received grade A in both blood pressure estimation. Systolic and diastolic The mean and standard deviation of errors for estimating systolic and diastolic blood pressure were 1.91 ± 5.55 mmHg and 0.67 ± 2.84 mmHg respectively.

Martinez-Ríos, Erick in this article presents a review of recent studies in the medical field that used machine learning techniques to detect high blood pressure. The authors aim to identify best practices, challenges, and opportunities in developing these machine learning models by reviewing the techniques used. Publicly available datasets and the predictors used in previous studies. They also reviewed the feature selection techniques used to simplify the model. The authors found a lack of studies that combined demographic or clinical data with physiological signals. Although there is a relationship between these two factors and blood pressure levels. As a result, there is an opportunity to improve the accuracy of these models by using both types of data for hypertension detection and blood pressure measurement.

Fan, Xiaomao presented a new method for estimating blood pressure using ECG signal and photoplethysmography signal together. Most wearable monitoring devices receive only one type of physiological signal due to limitations such as

power costs and device size. The proposed method uses a 2-layer bidirectional long short-term memory network with three fully connected networks to estimate blood pressure. In addition, a weight learning model that adapts to the trend of detection loss is introduced. Results from experiments on the Physionet Multiparameter Intelligent Monitoring in Intensive Care database show that the proposed method can meet the standards set by the British Hypertension Society and the US Association of Advancement of Medical Instrumentation. This method can be used in the healthcare system to provide continuous blood pressure monitoring and help reduce the incidence of complications related to high blood pressure.

Jiang, Hengbing presented a new technique for continuous blood pressure estimation using a neural network with multi-task learning (MST-net) that extracts multi-scale features. The process begins with signal pre-processing. (ECG and photoplethysmography) and blood pressure data. By removing false peak values, MST-net is then designed to identify and learn the relationship between multiscale features and blood pressure values, which are evaluated together. The performance of this method is tested using a publicly available multi-parameter smart monitoring waveform database. It shows good results with low mean absolute error and high correlation coefficient. It meets the standards set by the Association for the Advancement of Medical Instrumentation and reaches Level A of the British Hypertension Society standards. The proposed technique is a simple and effective method for continuous blood pressure measurement. This makes it a possible solution for mobile health devices.

Lee, Soojeong researched and presented a new method for more reliable and accurate cuffless blood pressure (BP) measurement and confidence interval (CI) estimation. The method is based on a hybrid feature selection and decision algorithm using a Gaussian process and involves using F-test and robust neighbor component analysis to select a set of heavily weighted features. Akaike's information criterion algorithm is used to determine the best subset of features. The proposed

method is evaluated and compared with conventional algorithms. This was shown to provide the most accurate BP and CI estimates. The authors believe that this is the only method currently available for reliable estimation of BP and CI.

Renna, Francesco, Jorge Oliveira, and Miguel T. Coimbra investigated the use of deep convolutional neural networks. To separate phonocardiograms into their main components The proposed technique uses a deep neural network architecture influenced by previous image segmentation methods. The output of the neural network depends on different temporal modeling techniques. To ensure that the state sequence in the output corresponds to the normal state sequence within the phonocardiogram signal (S1, systole, S2, diastole), a neural network with a hidden Markov model and a semi-Markov model was used. Hidden is used to estimate the distribution of emissions. The performance of the proposed method was tested on phonocardiogram signals from the publicly available PhysioNet dataset. and is shown to be superior to existing phonocardiogram segmentation methods. with an average sensitivity of 93.9% and an average positive predictive value of 94% in detecting S1 and S2 sounds.

Rastegar, Solmaz, Hamid Gholam Hosseini, and Andrew Lowe have used continuous blood pressure (BP) monitoring as essential for patients at high risk of cardiovascular disease. Blood pressure can be influenced by factors such as heart rate. blood vessel elasticity, blood volume, peripheral resistance, respiration, and emotional behavior. However, traditional BP measurement methods are difficult and ineffective in continuously monitoring BP. This paper proposes a new hybrid model that combines a convolutional neural network (CNNs) and support vector regression (SVR). CNNs serves as a feature extractor and SVR as a regression model to estimate blood pressure. systolic (SBP) and diastolic blood pressure (DBP) from electrocardiogram (ECG) and photoplethysmography (PPG) signals. The model was tested on 120 patients from MIMIC III database and achieved a mean absolute error

(MAE) of 1.23 ± 2.45 mmHg for SBP and 3.08 ± 5.67 for DBP, which meets the accuracy standards set by AAMI SP10.

Ali, Noor Faris, and Mohamed Atef focused on developing machine learning models for continuous blood pressure (BP) estimation. and selecting optimal features with interpretable models for medical professionals. To meet this challenge The study proposed a set of six physiologically motivated features based on the physiological background of blood pressure factors. Another set of commonly used properties is included for comparison purposes. A multi-step transfer learning method based on long short-term memory (LSTM) network is used for BP prediction, which consists of three steps: BP classification, mean arterial pressure (MAP) regression. and final BP estimation. The model performance was compared with a traditional one-step artificial neural network (ANN) model and an LSTM-based model using data from 40 subjects from the MIMIC II database. When using LSTM, there was a Mean Absolute Error (MAE) \pm Standard Deviation (SD) of 2.03 ± 3.12 for systolic blood pressure (SBP) and 1.18 ± 1.70 mmHg for diastolic blood pressure (DBP). Results The research found that the proposed feature set significantly reduces errors by up to 86.21% compared to models trained on commonly used features. It demonstrates the effectiveness of the proposed robust feature set and highly efficient multi-step topology in increasing the accuracy of BP estimation using photoplethysmography (PPG) signals.

Yang, Sen used traditional blood pressure (BP) measurement methods, which have limitations such as drawing blood and using cuffs. Or does it require someone to operate? There is great interest in developing non-invasive and continuous blood pressure measurements based on physiological signals, however, extracting features from these signals is challenging due to noise or distortion. This study investigates using raw signals as direct input for a deep learning model. A hybrid model that uses both raw signals and physical characteristics such as age, height, weight, and gender has been developed to compare with traditional machine learning models that use

features from photoplethysmograms and waveforms. electrical heart The hybrid model provided the best results for both diastolic BP and systolic BP, with mean absolute errors of 3.23 ± 4.75 mmHg and 4.43 ± 6.09 mmHg, respectively. These results meet the performance requirements of the British Hypertension Society.

Panwar, Madhuri introduced a new deep learning model named 'PP-Net' which can estimate three physiological parameters: diastolic blood pressure (DBP), systolic blood pressure (FvSBP) and heart rate (HR) from single-channel photoplethysmogram (PPG) signals. The model is designed using a long-range Recurrent Convolutional Network (LRCN) framework, allowing to extract features without the need to select and extract features. This makes it suitable for deployment on resource-constrained platforms such as mobile devices. The performance of PP-Net was tested on the large, publicly available MIMIC-II database and yielded average NMAE values of 0.09 mmHg for DBP, 0.04 mmHg for SBP, and 0.046 bpm for HR estimates in a population of 1557 critically ill patients. Accurate estimation of HR and BP in such a large population It demonstrates the effectiveness of the proposed deep learning framework and its robustness in cardiovascular rehabilitation monitoring.

Li, Yung-Hui proposed blood pressure measurement as an important part of monitoring people's health. And early detection of abnormal blood pressure levels can lead to timely treatment. and reduce death rates from heart disease and stroke to achieve real-time blood pressure measurement This paper presents a deep learning regression model used. electrocardiogram (ECG) and photoplethysmogram (PPG) signals to estimate systolic blood pressure (SBP) and diastolic blood pressure (DBP). The model uses a bidirectional layer of short-term memory. long-term (LSTM) and has residual connections in each segment based on the LSTM layer. The performance of the proposed model is compared with traditional machine learning methods and existing deep learning models using a set of MIMIC II data which provides ECG, PPG and blood pressure signals The results show that the proposed

model has better performance. Existing methods and accurate estimates are suitable for clinical use.

Şentürk, Ümit, İbrahim Yücedağ, and Kemal Polat studied and presented a hybrid model that combines signals. Electrocardiography (ECG) and Photoplethysmographic (PPG) to continuously assess blood pressure. The model consists of two steps: 1) calculating 22 time-domain attributes from ECG and PPG signals to determine systolic and diastolic blood pressure values, and 2) using these attributes as input to The structure of the recurrent neural network (RNN) spans two directions - the Short-Term Memory (BLSTM) layer, the LSTM layer and the Rectified-Linear unit (ReLU) layer. The BLSTM layer helps explain the non-linear physiological dynamics of Blood pressure, LSTM layer improves the learning ability of the model, and ReLU layer improves the processing speed of the network. The results showed that the mean square errors between estimated and measured systolic and diastolic blood pressure values were 3.63 and 1.48, respectively. The proposed model is systematic. Calibration-free blood pressure measurement using only PPG and ECG signals.

Lee, Soojeong, Gangseong Lee, and Gwanggil Jeon analyzed normal blood pressure values. oscillometric (BP) using statistical methods such as kurtosis, skewness, Kolmogorov-Smirnov and correlation testing To improve accuracy and reduce uncertainty This study used a deep learning regression model to determine confidence limits of blood pressure measurements based on normal values. The proposed deep learning model is shown to reduce the standard deviation of error and mean absolute error of BP measurements, resulting in greater accuracy and reduced uncertainty. The study also demonstrated the normality of the distribution of the BP estimates and the independence of the pseudo-systolic and diastolic blood pressure estimates using the rank test. The study concludes that the proposed method using deep learning provides a more accurate BP estimation. and reduce the uncertainty associated with confidence limits and standard deviations of error.

2.14 Summary

The second chapter provides a survey of the literature reviewed and explains the meaning and properties of phonocardiograms produced by the human body. It also aims to understand accurate blood pressure measurement. This chapter lays the groundwork for comprehending the heart's functions, the relationship between blood pressure values and cardiac activity, and the techniques for analyzing sound wave signals by extracting their distinctive features based on time and frequency. The chapter covers pressure estimation using machine learning techniques, combining various deep learning methods to recognize patterns and structures and identify relationships. Additionally, it includes a review of relevant literature and past research on feature extraction. The researcher proposes a new approach to developing feature extraction techniques that yield better weight values and reduce data size through machine learning-based feature extraction. This approach aims to minimize data size while retaining its essential characteristics and to enable the use of data from new samples, which will be explained in the next chapter.

Table 2.1 Comparison with Works are Use PCG Signal.

Work	Number of subjects	Signals	Index	SBP mmHg		SBP mmHg	
				MAE	STD	MAE	STD
[1]	37	PCG	S21	6.48	4.48	3.91	2.58
[2]	85	PCG-PPG	VTT	6.67	8.47	-	-
[3]	24	PCG-PPG	PTT	7.47	11.08	3.56	4.53
[4]	10	PCG-PPG	VTT	4.07	3.07	5.61	4.09
[5]	32	PCG-PPG	PTT	6.22	9.44	3.97	5.15

CHAPTER III

METHODOLOGY

3.1 Introduction

This chapter discusses the design and working principles of blood pressure estimation through an experiment using a data set obtained from a sample group. The steps involved are as follows: 1) classify data sets for easy understanding, 2) design a process for PCG signal features, and 3) design the estimation process. The estimation process uses the following methods in order: bandpass filtering, CNNs deep learning, deep hybrid learning, and deep hybrid learning with physical information. These methods will be used to create a blood pressure estimation model that consists of upper (systolic) and lower (diastolic) pressure values. The obtained estimation model will then be tested on the data set to evaluate its accuracy.

3.1.1 Classify Data Sets

Classifying datasets is an essential step to ensure that the data is organized and easily understandable. This involves categorizing the data based on relevant attributes such as demographic information, physiological measurements, and signal characteristics.

3.1.2 Design a Process for PCG Signal Features

The next step involves designing a process to extract relevant features from PCG signals. These features are crucial for accurately estimating blood pressure.

3.1.3 Design the Estimation Process

The estimation process involves using various machine learning techniques to develop a model that can predict blood pressure values from the extracted features. The following methods are employed in sequence:

3.1.3.1 Bandpass Filtering

Bandpass filtering is used to isolate the frequency range of interest in the PCG signals, removing noise and irrelevant frequency components.

3.1.3.2 Convolutional Neural Network (CNNs) Deep Learning

CNNs are used to automatically learn hierarchical features from the PCG signals, which are then used for blood pressure estimation.

3.1.3.3 Deep Hybrid Learning

Deep hybrid learning combines CNNs with other machine learning models to improve the accuracy and robustness of blood pressure estimation.

1. Feature Extraction: Use CNNs to extract features from the PCG signals.

2. Hybrid Model: Integrate the CNNs features with additional models such as Extreme Gradient Boosting (XGB), Support Vector Machines (SVM) and Kernel Extreme Learning Machines (KELM).

3. Training: Train the hybrid model on the labeled dataset to learn the mapping between features and blood pressure values.

Hybrid models leverage the strengths of multiple techniques to achieve better performance than individual models.

3.1.3.4 Deep Hybrid Learning with Physical Information

Incorporating physical information, such as demographic and physiological data (e.g., age, weight, height), into the deep hybrid learning model to enhance prediction accuracy. Integrating deep hybrid learning with physical information for blood pressure estimation from phonocardiogram (PCG) signals enhances predictive accuracy by capturing complex relationships between physiological parameters and blood pressure. This approach leverages diverse data sources, such as age, weight, and height, to improve feature representation and model robustness, reducing the risk of overfitting and data imbalance. Additionally, incorporating physical attributes increases the clinical relevance and interpretability of the model, aligning predictions with medical knowledge and making them more

useful for healthcare practitioners. By combining the strengths of convolutional neural networks (CNNs) for PCG feature extraction with traditional machine learning models like Extreme Gradient Boosting (XGB), support vector regression (SVR) and kernel extreme learning machines (KELM), the hybrid model achieves better performance and generalization, providing a comprehensive, end-to-end learning solution that is both accurate and reliable for clinical applications.

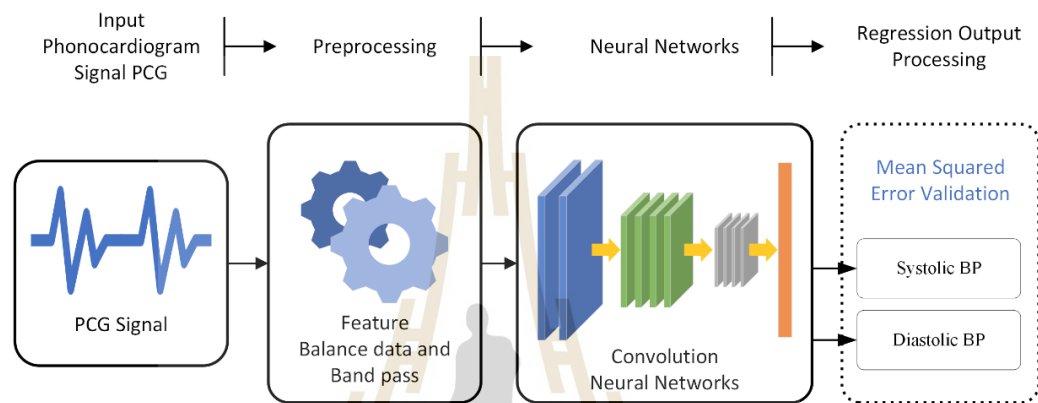


Figure 3.1 Process of estimate blood pressure with Deep Learning

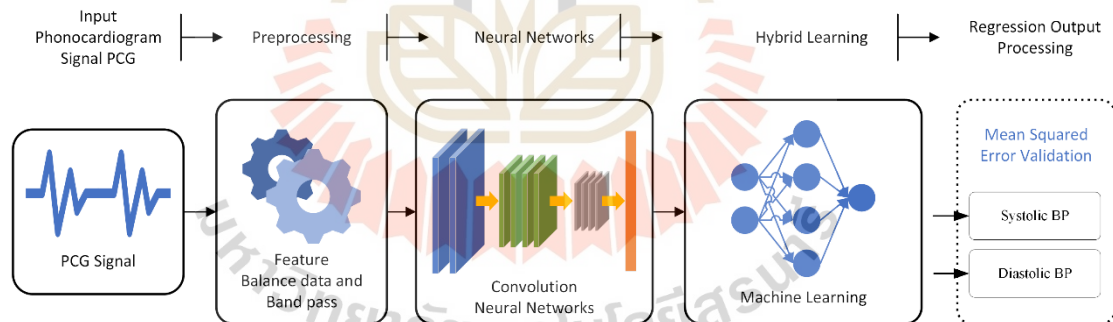


Figure 3.2 Process of estimate blood pressure with Deep Hybrid Learning.

3.2 Data Collection and Data Recording

Hospitals at all levels, from sub-district health promotion hospitals to community hospitals, general hospitals, and regional hospitals, have a primary mission to provide care and treatment for diseases as well as to promote healthy living. Ensuring good health for the people in their area of responsibility is a priority. Carrying out this mission naturally involves managing a large amount of data,

including information related to medical treatment activities and other related information. Good quality data management begins with the collection and recording of accurate data. Operations must adhere to established standards, starting with data collection and data recording standards as follows:

3.2.1 Data collection

Data collection involves requesting necessary information from various sources. Within healthcare facilities, important sources of information include patients and their relatives, who must provide accurate details to doctors and hospital staff. This information is crucial for diagnosing diseases and planning treatment or health promotion to achieve optimal outcomes. Collecting information from patients and relatives occurs frequently and across many departments within a hospital. For instance, upon first entering the hospital, the reception or patient registration department will request the patient's first and last names to search for previous medical history if the patient is new. They will also need to collect other details such as the patient's address, identification number, marital status, and parents' names for registration purposes.

During a visit to the doctor's examination room, the nurse will take a medical history. The doctor will then gather additional details about the illness and perform a physical examination. Upon receiving a prescription and visiting the dispensary, the pharmacist will inquire about the patient's history of drug allergies and their understanding of medication usage.

It is evident that data collection occurs at multiple points within the hospital, with individuals in various roles responsible for gathering this information. Effective data collection methods are required to ensure that the information gathered is complete and accurate.

There are five different methods of data collection: 1) Registration: This involves formally recording patient information upon their admission or during their visit. 2) Answers to Forms or Questionnaires: Patients or their relatives fill out documents that gather necessary health information. 3) Questions or Interviews: Data

collectors ask direct questions to patients or their relatives to obtain specific information. 4) Direct Observation by Data Collectors: Observers gather data based on their assessment and observation of patients. 5) Viewing Reports from Another Source: This method includes reviewing information provided by external sources or previous medical records.

3.2.1.1 Registration

Registration involves requesting information from patients and recording it in a registry or database. This process ensures that a list of patients with important details is maintained. For instance, when a doctor schedules surgery for a patient, the attendant and surgical appointment registrar will request the patient's information and record the patient's name along with other important details in the surgical appointment registry. When it is time to queue for the operating room, they will refer to the names in the registry and proceed with the queuing process, facilitating a smooth transition into surgery. Examples of registries used in healthcare facilities include:

1. New patient registration
2. Inpatient registration
3. Outpatient department examination appointment registration
4. Surgery appointment registration
5. Diabetic patient registration
6. Register for follow-up appointments after treatment

3.2.1.2 Giving responses to forms or questionnaires.

Obtaining information from patients can be facilitated by having them complete a form or questionnaire. This form should include important topics, and the patient is asked to fill in the necessary information. Utilizing this method can reduce the workload of staff responsible for data collection. For example, when a new patient visits a healthcare facility for the first time, they are given registration forms to complete. These forms require details such as the

patient's first name, last name, identification number, date of birth, address, parents' names, marital status, occupation, nationality, and religion. After filling out the forms, patients submit them to the staff for further processing. However, this method may have limitations in cases where patients are physically or intellectually unable to write. Examples of forms or questionnaires used in healthcare facilities include:

1. New patient registration form
2. Form for requesting changes to information in the patient register
3. Service satisfaction questionnaire

3.2.1.3 Questioning or interview

Questioning or interviewing involves engaging with patients to gather information by asking questions about predetermined important topics. This method is advantageous because it is convenient for patients and quick. However, it has limitations in cases where verbal communication is not possible, such as with foreign patients or those who are unable to speak due to physical or intellectual disabilities. Additionally, the individuals conducting the interviews must be trained in effective questioning techniques to ensure that they collect complete and crucial information within a limited time frame. Examples of questioning or interviews used in medical facilities include:

1. Medical records staff ask for patient information to make cards.
2. The nurse asks about illness symptoms to determine which department of doctor to send for examination.
3. The doctor asks about the history of the illness in order to use the history to make a diagnosis.

3.2.1.4 Direct observation by the data collector

Direct observation is a method where data collectors observe a patient's appearance, posture, or conduct physical examinations. This involves measuring various signals from the patient using predefined techniques. It is

commonly employed in healthcare settings. For example, during the physical examination of patients by doctors, direct observation is a clear instance. When a doctor examines a patient, they utilize techniques such as looking, feeling, tapping, and listening to different parts of the body. These actions help identify any abnormal symptoms, which are then recorded as results of the physical examination to aid in further diagnosis. Examples of direct observation used in hospitals include:

Measurement of blood pressure, and pulse of the patient.

3.2.1.5 Viewing reports from other sources

Viewing reports from other sources refers to examining documents that patients bring or that data collectors request to see. For instance, when a doctor takes a patient's history and learns that the patient has been taking 2-3 medications, the doctor may ask to see the medicines. Reviewing the name of the drug or the drug label is considered viewing reports from other sources. Often, patients may also bring results from x-rays or laboratory tests that were conducted at other hospitals. Common examples of reports from other sources frequently encountered in hospitals include:

Referral letter from a doctor in hospital Report.

Data collection through these various methods allows for the integration of information from multiple sources into medical patient records. Typically, information in various registers originates from registries. General patient information is usually obtained from filling out forms and conducting additional inquiries. Information about a patient's illness history is derived from interview questions, while details of physical examinations come from direct observations. It is important to be cautious with information from other sources; such data should be recorded separately to clearly indicate that it is information the patient has brought in, such as laboratory test results from another hospital. This approach ensures that it is evident the information originates from external sources.

3.2.2 Data Recording

Recording data involves converting information into a format that is presentable, distributable, and replicable. Typically, data is recorded on various storage media, including paper and different forms of electronic media. This process also includes recording and storing data in computer databases. Recording information provides reliable evidence that can be easily referenced and used to generate new information and knowledge.

Data recording is considered a primary responsibility of all personnel in a hospital and is regarded as a fundamental ethical duty. All medical and public health professionals must treat it as their duty to adhere to professional standards, particularly in activities related to patient care. It is essential that all records are complete, accurate, and detailed, and that they are completed within the specified timeframe.

Current data recording practices in healthcare facilities typically involve three common formats: 1) Recording data on paper: This traditional method involves manually entering information into physical documents. It is often used for initial patient intake forms, notes during consultations, and other situations where immediate electronic access is not available. 2) Recording data into a database system: This method involves entering data directly into a digital database. This system facilitates easier data retrieval and management, enabling healthcare providers to access and update patient records efficiently. 3) Recording data in electronic files: This format involves storing data in electronic files rather than in a structured database. It allows for the storage of various types of data, including scans of physical documents, digital images, and other multimedia formats.

3.2.2.1 Recording Information on Paper

Recording information on paper has been a long-established practice since the inception of modern medical facilities in Thailand. This process begins with a prescription, where the doctor must meticulously record the name of the medicine, the dosage, and the instructions for use to ensure the pharmacist can accurately prepare the medication for the patient. Additionally, every hospital

maintains a logbook for patient care activities, where doctors, nurses, and other staff record various care procedures. When a logbook is filled, it is replaced with a new one, and the old notebooks are stored systematically. This organization allows for easy retrieval, enabling staff to quickly access information whenever needed.

As medical record systems evolved, including the Out-Patient Department (OPD) Card or In-patient File, doctors, nurses, and other staff began documenting the information gathered from patient interactions directly into each patient's medical record. Most of the information is recorded using letters and numbers. Occasionally, doctors may include diagrams to illustrate the location of abnormal symptoms or to depict surgical procedures as a visual supplement to the textual description. With the advent of typewriters in hospitals, some staff members started typing information directly into the medical records system, replacing handwritten entries. This transition has made the information in medical records easier to read and more accessible than traditional handwriting.

Recording information on paper offers several advantages: it is easy to execute, requiring only a pen and paper, with no need for electronic tools. There are established standard practices for recording information on paper in medical facilities, which include the following:

1. Use only a pen or printer for recording. Do not use a pencil.
2. Ensure handwriting is neat and at least semi-detailed to facilitate easy reading.
3. Always include the date and time on every recorded entry.
4. Provide clear and complete details in each entry. Avoid vague or incomplete records, such as noting only that 'the same medicine was dispensed' or using 'RM' without additional context.
5. Do not use abbreviations or symbols that are not approved by the hospital's standards. Healthcare facilities should have a list of allowed abbreviations and symbols for recording information. Any not on the list are considered non-standard and should not be used.

6. Record events immediately or soon after they occur. For instance, in an emergency, document the treatment as soon as it is safe to do so.

7. Clearly sign each entry to identify the recorder. If a signature is used, it should be accompanied by a legible indication of the signer's identity. Signatures should also be registered in a signature register for verification.

8. To correct a recorded entry, draw a single straight line through the incorrect text, write the new message beside it, and then sign the correction. Do not scrape off the original text, use correction fluid, or delete text in any other way.

Disadvantages of recording information on paper include, There is a chance of getting lost or the message is easily blurred. Some people's handwriting is unreadable. Making paper copies is more expensive than other formats.

3.2.2.2 Recording Information into the Database System.

Recording information into the database system began in hospitals in Thailand around 1987, following the introduction of computers and hospital system programs. While storing information in the database has advantages, it also has several disadvantages: Technology Dependency: Recording requires a computer. Skill Requirement: Users must possess proficient keyboard skills to ensure data can be recorded conveniently and quickly. Access Limitations: Data browsing is typically restricted to a computer monitor, except when data is printed out. Only printed records can be physically viewed in the medical record file.

The advantages of recording data into a database system include making information simultaneously available at multiple points. For instance, when several doctors collaborate to care for the same patient, they can view the patient's information from different computers at the same time. This system reduces issues associated with unreadable handwriting. Additionally, using computers enables the use of copy and paste functions, allowing text to be duplicated in the same format. This facilitates the easy and quick processing of long messages.

Currently, most medical facilities in Thailand encourage doctors to record patient information into a database system, and an increasing number of doctors are cooperating. This shift has led some hospital administrators to consider discontinuing the use of paper records. However, not all crucial patient information can be effectively recorded using computers. For instance, physical examination results that require illustrative drawings by doctors cannot yet be effectively captured using computers. Additionally, some diagnostic information cannot be accurately entered into the database system; there is a common misconception that entering an ICD code equates to recording a disease diagnosis. Therefore, healthcare facilities should not yet abandon the use of paper for recording patient information. It is essential to ensure that the data recording program meets the necessary standards before phasing out paper records, as per the standard protocol for evaluating data recording systems in healthcare settings.

Before deciding to discontinue paper recording in hospitals, management must implement a process to evaluate whether using digital systems can match the detail and volume of information recorded on paper. If it is found that digital systems allow doctors to record less information than they can on paper, the transition should not proceed. For instance, recording the results of a physical exam might be more convenient and thorough on paper than using a digital system. If using the program results in doctors not recording essential details of physical examinations, critical information could be lost. Therefore, such decisions must be carefully considered to ensure there is no compromise in the quality and completeness of patient records.

Standards for checking data recording programs in healthcare facilities before canceling paper recording.

1. Check that the program has all the fields to record data according to the standard data list that must be recorded. of the Ministry of Public Health or not? For example, there must be a space to record the name of the disease, not using an ICD code to record instead of the name of the disease.

2. The program must be dated and recorded. Always label recorded messages.

3. The program must not have automated systems that cause ambiguous or non-descript messages to be recorded, such as recording only the original prescription or RM without any other details.

4. The program must have a system for checking the use of abbreviations or symbols that are not standardized. of the hospital (Health care facilities should specify abbreviations and symbols Characteristics that are allowed for use in recording information. If any abbreviations or symbols Not on the list Treat, it as an abbreviation or symbol. that are not standard).

5. The program must have a system for checking and recording who recorded the program. When was it recorded?

6. The program must not allow users to delete saved entries from the database if they want to edit saved messages. Add the message as a new item. then mark cancel and edit old entries with reference numbers What number will the new item be edited to replace the old item with? Record the name of the editor and the date and time of the edit.

7. The database system must maintain the highest level of stability and security. Data copies are made. There is a log system that records every entry to edit the database or database copy.

3.2.2.3 Recording Information Into Electronic Files

Electronic file recording began with the introduction of computers into hospitals, paralleling the implementation of database systems. Recording information in electronic files shares similar disadvantages with database systems. Firstly, access to a computer is necessary for recording data. Additionally, users must possess proficient keyboarding skills to enter data efficiently and quickly. Furthermore, browsing data is restricted to computer screens unless the information is printed out, which then allows the printed data to be viewed physically in the medical record file.

Recording information into electronic files typically involves storing data separately from medical records and database information, often across different departments. For example, radiology examination results may be stored in a Microsoft Word document on only the radiologist's computer. If this information is not properly managed, it can lead to significant issues, including the inability to integrate the information into the central medical record system or the healthcare facility's central database, or to share the information across multiple departments. Proper management is essential to prevent such negative impacts and ensure effective use of data throughout the healthcare facility.

Standards for managing data stored in electronic files separated into different departments.

1. Carry out registration to know what information is stored in electronic files separated into different departments. What information is stored in which departments?
2. If the data stored separately is information related to patient care, such as reports of radiology examination results. Print out a report and keep it in the patient's medical record. Each auxiliary person, every time, prepares a report.
3. Ensure that the registered data files are backed up periodically. Verify backed up files that it is always practical.
4. In the medium term, a mechanism should be created to link separately stored data files with the hospital's central database system.
5. In the long term, a data warehouse should be created that uses data stored in separate files to be used in data services and Analyze data to lead to further development of the hospital.

3.3 Data Categorizing

3.3.1 Define Data Categories

Identify the main attributes of the data that are relevant to study or project. This could include variables such as age, gender, location, time of recording, type of data.

Determine the subcategories within those main attributes if applicable, such as different age groups, various locations, or subtypes of data.

3.3.2 Structure Data Storage

Create a hierarchical folder system on data storage platform where each main category has its own folder.

Within each main category folder, create subfolders for each subcategory.

Name the folders clearly and consistently to avoid confusion.

3.3.3 Standardize Data Entry

Develop a data entry protocol that all team members will use. This might include standardized forms or templates that prompt for specific information, ensuring that all necessary data categories are captured.

Implement quality control checks to ensure data is categorized correctly at the time of entry.

3.3.4 Utilize Metadata

For every data record, include metadata that describes the data. Metadata can include the date of entry, source of the data, the person who recorded it, and any relevant category information.

Metadata should be stored in a consistent format that is easy to access and understand.

3.3.5 Data Tagging and Labeling

Apply tags or labels to data files and records based on their categories. This can be done manually or automated with software that recognizes patterns or keywords.

Tags help in quick retrieval of data for analysis or review.

3.3.6 Implement a Database Management System (DBMS)

Use a DBMS to store, retrieve, manage, and analyze data. A DBMS can help categorize data more efficiently and enable complex queries that involve multiple categories.

Ensure that the DBMS supports the data formats to using and allows for flexible categorization.

3.3.7 Review and Adjust Categories Regularly

Periodically review the categories and the data categorization process to ensure they still meet the needs of project or research.

Be prepared to adjust categories as the scope of the project expands or shifts.

3.3.8 Documentation and Training

Document the categorization process and any updates to it. This documentation should be accessible to all team members.

Train new team members on the proper categorization procedures to ensure consistency across the project.

3.4 Data Collection Process

3.4.1 Step Data Collection Process

The data collection was carried out under strictly controlled conditions to minimize the impact of external variables on the PCG signals and the corresponding BP readings. Participants were comfortably seated in a calm, temperature-regulated environment to ensure consistent and accurate PCG signal recordings. The dataset captured signals within the frequency range of 20-2000 Hz. Sensors were positioned for optimal PCG signal reception on participants. Efforts were made to minimize distractions and noise. We meticulously collected PCG signal data from 78 volunteers, which was considered and approved by the Human Research Ethics Committee Suranaree University of Technology. The research was conducted in accordance with the ethical principles for human research in accordance with the Declaration of Helsinki. Received research ethics certificate

number Suranaree University of Technology. The procedure for collecting PCG signal data, along with blood pressure values, is outlined in Figure 3.1:

1. Participants sat comfortably in a calm, temperature-controlled environment for approximately 5 minutes to ensure consistent and accurate recording of the PCG signal.
2. Interview basic important information from volunteers, including age, weight, height, and gender.
3. Prepare the automatic blood pressure monitor. Ready to use a pressure measuring strap (Cuff) to wrap around the upper arm approximately 1-2 inches above the elbow and tie it tightly.
4. Prepare the digital stethoscope equipment to record signals. and set the signal file name to match the information of each volunteer.
5. Place the digital stethoscope in the position specified by the attending physician. After that, start the automatic blood pressure monitor and record the PCG signal.

Equipment for data collection:

Blood pressure monitor: In this research, an automatic blood pressure recording device named Blood Pressure Monitor NISSEI DSK-1031 was used.

Digital stethoscope: The ThinkLabs One Digital Stethoscope was employed, featuring a frequency range of 20-2000 Hz.

Figure 3.3 shows the sequence of recording PCG in the new dataset being tested in a hospital. The physician is responsible for overseeing the collection of data from each volunteer. The test was conducted under human research ethics number EC-65-78.

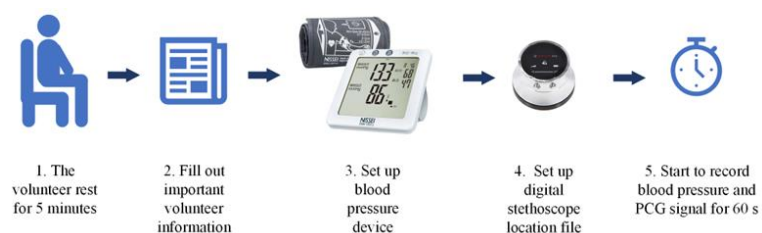


Figure 3.3 Sequence of recording PCG signal from volunteer

3.4.2 Data Collection

The data used as input is of the echocardiographic signal type. This database was compiled from a sample of volunteers and consists of echocardiogram recordings from 78 individuals who are all in good health. The upper blood pressure ranged from 100-145 mmHg, and the lower blood pressure ranged from 50-100 mmHg. The recordings were divided into 78 cases, each recorded sequentially along with basic information including weight, height, age, gender, and heart rate measured at the time the blood pressure was recorded using a blood pressure monitor.

3.4.2.1 Set volunteer selection criteria

1. Have consent to participate in the experiment as determined by the researcher.
2. Not a person with abnormal symptoms or have a disease related to the heart or blood pressure
3. Be a person between the ages of 16-80 years.

3.4.2.1 Device for recording the sound of the heart.



Figure 3.4 Automatic blood pressure measuring device (a) and digital medical stethoscope (b)

In this research, an automatic blood pressure recording device named Blood Pressure Monitor (NISSEI DSK-1031) was used, as shown in Figure 3.4(a). Additionally, the device used for recording phonocardiograms was a Digital Electronic Stethoscope (ThinkLabs One), as shown in Figure 3.4(b).

1. Blood pressure monitor (NISSEI DSK-1 0 3 1) There is technical information as follows.

1. Operating Principle : Oscillometric method
2. Indicator : 15 digits liquid crystal display
3. Pressure Indicating Range : 3 to 300 mmHg (cuff pressure)
4. Measuring Range : 50 to 250 mmHg (systolic), 40 to 180 mmHg (diastolic), 40 to 160 bpm (pulse rate)
- 5.- Accuracy : ± 3 mmHg (cuff pressure), $\pm 5\%$ of reading (pulse rate)
6. Inflation : Automatic inflation (FUZZY)
7. Deflation : Automatic (electric control valve)
8. Exhaust : Automatic exhaust valve
9. Power Supply : Four 1.5 volt LR6 (AA alkaline) batteries or AC adaptor
10. ADP-W5 series
11. Power Consumption : 4W (max.)
12. Electric Rating : DC6V/4W, with four LR6 batteries
13. Adaptor; AC100-240V, 50-60Hz, 0.12A,
14. Unit; DC6V, 500mA, with designated AC adaptor
15. Memory : 2 banks, each store 60 readings
16. Operating Condition : $+10^{\circ}\text{C}$ to $+40^{\circ}\text{C}$, 15% to 85% RH (noncondensing)
17. Transportation/Storage Condition : -20°C to $+60^{\circ}\text{C}$, 10% to 95% RH (noncondensing)
18. Cuff : Model; DSC-EP10, Coverage arm circumference; 22 to 42 cm, applied part; BF
19. Main Unit : Weight; Approx. 250 g (without batteries)
20. Size; Approx. 115 x 115 x 65.9 mm (W x D x H)

2. Digital electronic stethoscope (thinklabsone) There is technical information as follows.

1. Amplification : More than 100x (>40dB)
- 2.- Audio Filters : 5 bandpass filters
3. Display : Volume, Filter, Battery LED scale
4. Output Signal Level : Low impedance headphone driver, 3V p-p
5. Power Input : 5V DC (USB charger compatible)
6. Power Source : Internal Lithium-Ion cell
7. Battery Capacity : > 120 patient exams per charge (2 minutes per patient)
8. Connector : 4-conductor 3.5mm jack
9. Transducer : Thinklabs-patented Electromagnetic Diaphragm™
10. Dimensions : 46mm x 28mm
11. Weight : 50g
12. Headphones : Thinklabs high performance earbud headphones included, compatible with any high-quality audio headphones
13. Hearing Aid Compatibility : Connects to most hearing aid streaming devices. Recommended for use with closed-fit hearing aids that provide low frequency amplification (e.g. Oticon, Phonak, Resound, Sivantos, Starkey, Widex, Cochlear)
14. Accessories : Thinklabs high performance earbud headphones, carrying case, 100-240V USB-style charger, charger cable, Thinklink Interface kit (PC/Mac/Mobile etc.)
15. Telemedicine Platform Compatibility : Connects to audio inputs of most audio equipment including videoconferencing codecs (e.g. Cisco, Polycom), PC/Mobile (e.g. Vidyo, Zoom, WebRTC)

16. Apps/Software : Compatible with PC/Mac/iOS/Android audio recording/audio transmission apps, Thinklabs Stethoscope App (iOS), forthcoming Thinklabs apps (iOS, Android)

3.4.3 Location for placing data recording devices.

3.4.3.1 Blood pressure measurement location

To position the arm cuff, place it on the left arm, ensuring that the rubber air hose is positioned in front of the arm. The lower edge of the cuff should be 2-3 cm above the inside of the elbow. The cuff must fit snugly but not be wrapped too tightly or too loosely, as improper fitting can lead to inaccurate blood pressure readings. Additionally, adjust the position of the cuff so that the rubber tube is aligned over the inner arm on the brachial artery, as shown in Figure 3.5.

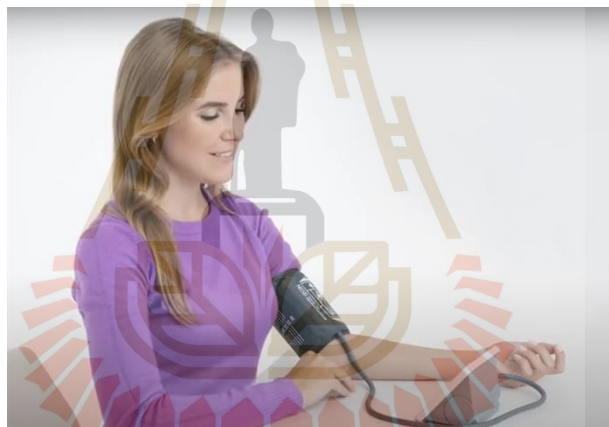


Figure 3.5 Characteristics of the position of using the arm cuff.

3.4.3.2 Location of phonocardiogram recording

Phonocardiograms are produced by specific cardiac events, such as the closing of valves or the contraction of chordae tendineae. These sounds are generated by the heart beating and blood flowing, specifically from the turbulent resonance that occurs when the heart valves are completely closed. To listen to the heart.

In healthy adults, there are two normal phonocardiograms, often referred to as "Lub" and "Dub," which occur sequentially with each heartbeat. These are the first phonocardiogram (S1) and the second phonocardiogram (S2), caused by the closing of the atrioventricular and semilunar valves, respectively. In

addition to these normal sounds, there may be other sounds present, including heart murmurs, unusual sounds, and galloping rhythms such as S3 and S4.

The first phonocardiogram, or S1, which comprises the "lub" of the "lub-dub," includes components M1 (mitral valve closure) and T1 (tricuspid valve closure). Normally, M1 slightly precedes T1. It is caused by the closure of the atrioventricular valves, such as the tricuspid and mitral (bicuspid), at the start of ventricular contraction or systole. As the ventricles begin to contract, the papillary muscles within each ventricle also contract. These muscles are connected to the cusps, or leaflets, of the tricuspid and mitral valves via the chordae tendineae (heart strings). When the papillary muscles contract, they tighten the chordae tendineae, which prevents blood from flowing back into the low-pressure environment of the atria. The chordae tendineae act like the ropes of a parachute, allowing the valve leaflets to rise slightly into the atria but not enough to break the seal at the leaflet tips, which would cause blood to flow backward. It is the pressure created by ventricular contraction that closes the valves, not the action of the papillary muscles alone. Contraction of the ventricles begins before the AV valves close and before the semilunar valves open. The sudden contraction of the ventricles and the compression within the ventricles with the semilunar valves closed send blood rushing back toward the atria, and the parachute-like valves catch the blood in the leaflets, causing them to close. The S1 sound results from the resonance within the blood associated with the sudden obstruction of backflow by the valve. A greater than normal delay in T1 causes splitting of the S1 sound, which is heard in conditions like right bundle branch block.

The second phonocardiogram, or S2, constitutes the "dub" of the heart's "lub-dub" sound and is composed of the components A2 (closure of the aortic valve) and P2 (closure of the pulmonary valve). Typically, A2 precedes P2, especially during inspiration when the splitting sound of S2 can be heard, caused by the closure of the semilunar valves (aortic and pulmonary valves) located at the junction between the heart's ventricles and the major arteries. When the left

ventricle empties, the pressure within it drops below the pressure in the aorta, causing blood to flow back rapidly towards the left ventricle, which catches the pocket-like parts of the aortic valve and leads to its closure. Similarly, when the pressure in the right ventricle falls below that in the pulmonary artery, the pulmonary valves close. The S2 sound results from resonance within the blood associated with the sudden blockage of the return flow. S2 splitting, also known as physiological splitting, typically occurs during inspiration due to a decrease in intrathoracic pressure, which delays the time it takes for the pulmonary pressure to exceed right ventricular pressure.

Widely separated S2 sounds can indicate various cardiovascular conditions, with the separation sometimes being wide and variable, while at other times it remains wide and constant. Wide and variable splits occur in conditions such as right bundle branch block, pulmonary stenosis, pulmonary hypertension, and ventricular septal defects. A wide and constant separation of S2 occurs in atrial septal defects. In conditions like pulmonary hypertension and embolism, the pulmonary S2 (P2) is accentuated, a phenomenon known as P2 accentuation. Conversely, in pulmonary stenosis, the S2 sound diminishes as the pulmonary artery narrows, as illustrated in Figure 3.6.

This is where S1 and S2 can be heard. The four standard auscultation points are:

Aortic – Located on the right side of the sternum.

Pulmonary – On the left side of the sternum

Tricuspid – in the fourth intercostal space along the lower left edge of the sternum

Mitral – in the fifth intercostal space along the midline of the collarbone

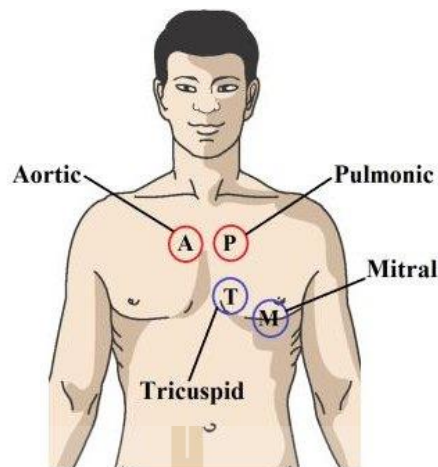


Figure 3.6 Standard position for listening to phonocardiograms.

3.4.4 Determining the Format of Phonocardiogram and Blood Pressure Recordings

phonocardiogram recording

In the recording process, the listening position for capturing phonocardiograms was designated as position P. Subsequently, recordings were obtained from groups comprising both male and female volunteers. At certain listening positions, discomfort may have been experienced due to the nature of the sounds. To maintain consistency in recording standards, we opted to utilize only one listening position throughout the data collection process.

Recording blood pressure values

Blood pressure values will be recorded using an automatic blood pressure monitor. The measurements will be taken with an arm cuff wrapped around the front of the left arm, positioned approximately 2-3 cm above the elbow. To ensure accuracy, measurements will be conducted while the individual is in a seated position, ensuring that the level of the arm cuff aligns with the heart. The recorded values will then be read from the displayed screen of the monitor.

Step 1: Begin by measuring the blood pressure. Wrap the armband around the left arm, ensuring the volunteer is seated and resting to align the armband's level with the heart. The volunteer should rest for at least 5 minutes

before the blood pressure measurement is taken. The seated position is illustrated in Figure 3.7.



Figure 3.7 Volunteers sitting and waiting for blood pressure measurement.

Step 2: To commence blood pressure measurement, employ a digital stethoscope to capture the phonocardiogram at position P. Listen to and record the sound for at least 60 seconds, adjusting for longer recordings based on volunteer variability. Simultaneously conduct blood pressure measurement and phonocardiogram recording to synchronize blood pressure values with heart operation sounds. The concurrent recording process is depicted in Figure 3.8.

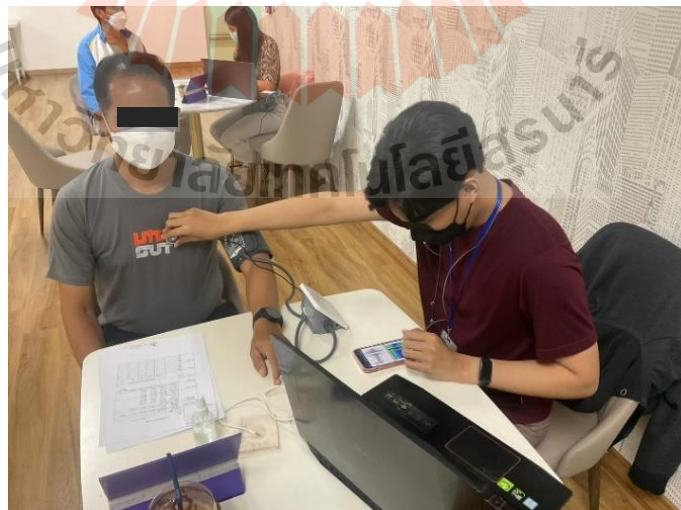


Figure 3.8 Recording phonocardiograms and blood pressure values.

3.4.5 Arrange the collection of phonocardiogram data.

To capture phonocardiograms using a digital stethoscope, ensure access to the device's dedicated software, enabling adjustments or file name editing post-recording. Figure 3.9 illustrates the operational interface of the stethoscope, detailing its functionalities.

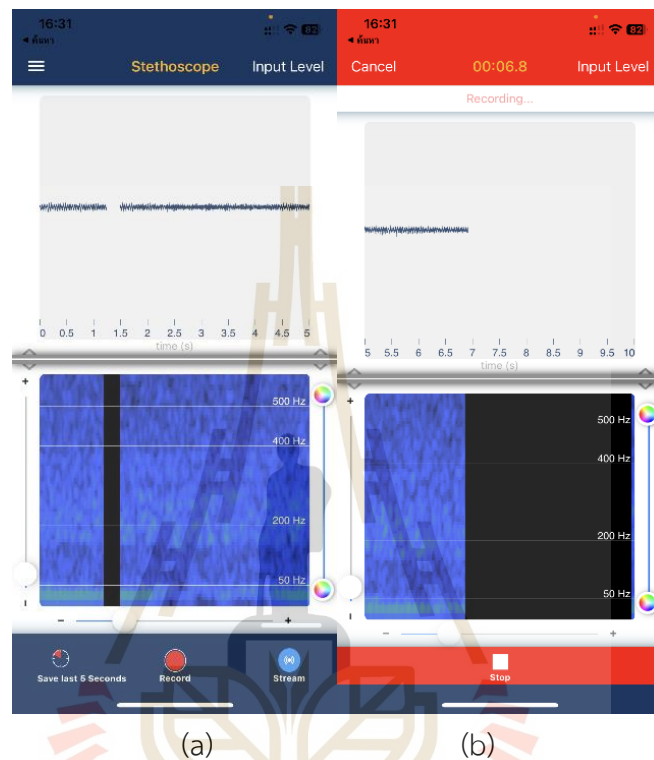


Figure 3.9 (a) Page of the phonocardiogram recording program. (b) Page while recording phonocardiograms.

3.4.6 Data collection

Collection of phonocardiogram data along with blood pressure values during measurement will be done using the following steps:

1. Have the volunteers sit and rest for 5 minutes to ensure their bodies are in a normal state and not too tired.
2. Use the arm cuff of a blood pressure monitor to wrap around the volunteer's left arm. Position it approximately 2 cm above the elbow and ensure that the air tube is not bent or folded, as shown in Figure 3.10.

3. Prepare to record phonocardiograms using a digital stethoscope connected to a recording program on mobile phone.

4. Record the physical data of the volunteers, including weight, height, age, and gender, before recording blood pressure and phonocardiograms.

5. Start recording by placing the digital stethoscope at position P, as shown in Figure 3.3. Listen for the phonocardiograms S1 and S2. Once to hear the sounds, start measuring the blood pressure, which the machine will automatically measure. The duration of blood pressure measurement may vary depending on the physiology of the volunteer. As for recording the heartbeats, it will take no less than 1 minute.



Figure 3.10 Positioning the equipment used for recording sound and measuring blood pressure.

3.5 Verify the Signal Information of the Phonocardiogram Signal.

After recording the blood pressure value along with the phonocardiogram, the subsequent step entails examining the recorded audio signal. This process involves confirming whether the pattern of phonocardiograms captured can be identified in the format of the audio signal, and assessing if it corresponds to the audible functions of the heart.

When analyzing the sound signal, it is imperative to confirm its inclusion of phonocardiogram elements from the designated recording location. As depicted in Figure 3.9, the provided example illustrates an audio signal acquired from a recording session. Predominantly, the recorded sound signal encompasses the phonocardiogram components S1 and S2. The sound captured by volunteers using a digital stethoscope spans a frequency range of 20-2000 Hz, as demonstrated in Figure 3.9. Within this frequency spectrum, the digital stethoscope has the capability to capture various physiological activities, such as lung expansion sounds during inhalation and exhalation, gastric digestion noises, or the movement of other bodily organs. Notably, studies on phonocardiograms have revealed their distinct audibility within the range of 60-500 Hz, prompting initial signal analysis procedures like filtering the sound signal within the 60-500 Hz range, as depicted in Figure 3.10. Subsequent scrutiny of the sound signal identified the presence of undesired noises. Consequently, the commencement and conclusion segments of the audio signal were trimmed to reduce its temporal duration to a mere 10 seconds, significantly shorter than the original recording duration of 60 seconds or more.

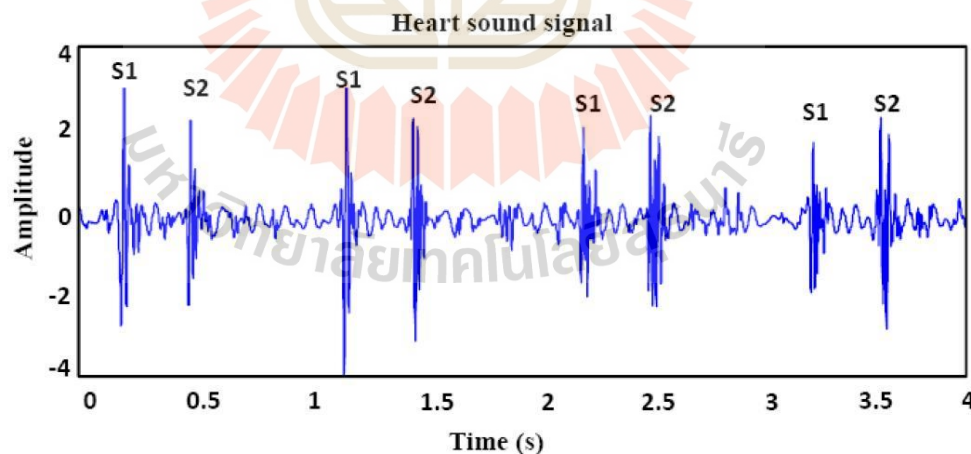


Figure 3.11 Picture of components of phonocardiogram S1 and S2

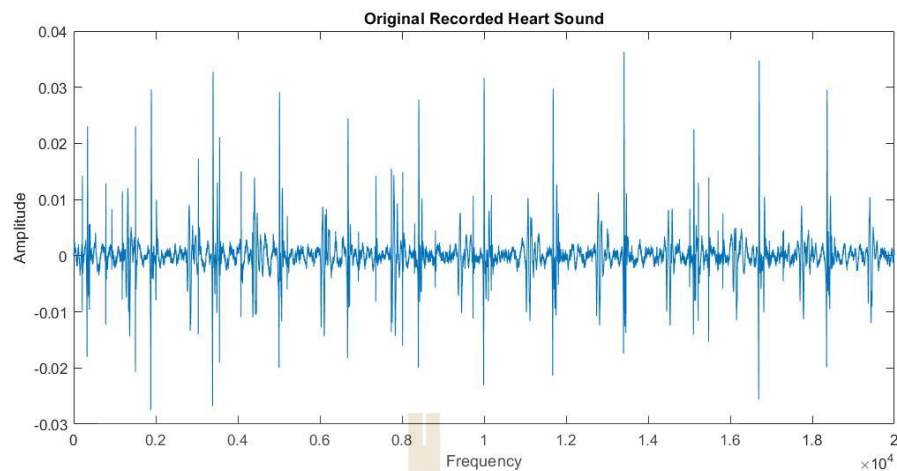


Figure 3.12 Example of a volunteer's phonocardiogram.

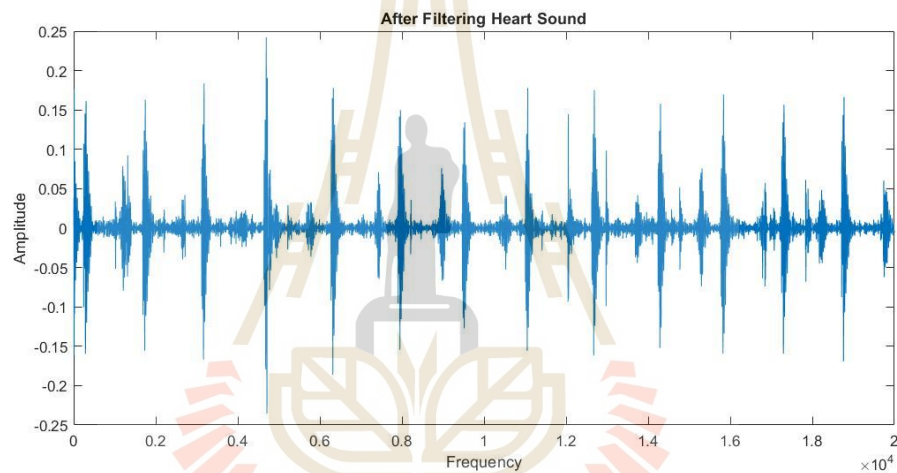


Figure 3.13 Filtered sound in the frequency range 60-500 Hz

Filtering the audio signal within a specific frequency range, as depicted in Figure 3.13, facilitates the observation of sound variations when confined within certain frequency parameters. Sounds falling within the 60-500 Hz range render the distinctive characteristics of phonocardiograms more discernible, while preserving their fundamental features.

3.6 Adding Sample Data

In the dynamic realm of data analysis and machine learning, proficient data management and preprocessing play vital roles in deriving valuable insights. Among the array of preprocessing techniques, data segmentation emerges as a fundamental method for organizing and streamlining intricate datasets. Also referred to as data

partitioning or slicing, data segmentation entails partitioning a larger dataset into smaller, more digestible segments or chunks according to predefined criteria. This procedure holds significant importance across diverse domains such as signal processing, time-series analysis, and pattern recognition, enabling improved analysis and streamlined processing workflows.

Data segmentation serves as a key strategy to render large datasets more manageable for analysis and to optimize the performance of machine learning models. In time-series analysis, for example, segmentation aids in isolating events, trends, or temporal changes, facilitating targeted analyses on specific intervals devoid of irrelevant data noise. Likewise, in image processing, segmentation techniques segment digital images to pinpoint and discern objects or boundaries more efficiently.

Data segmentation can be applied through several methodologies, depending on the nature of the data and the specific requirements of the analysis or application. Common techniques include:

1. Fixed-Interval Segmentation: Data is sliced into segments of a predefined fixed length, useful in scenarios where the temporal or spatial consistency of data points is crucial or regular sampling of sensor data.

2. Event-Based Segmentation: Segments are defined by specific events or changes detected within the dataset, such as a sudden spike in a signal or a change in market conditions. This method is particularly useful in anomaly detection and event prediction.

3. Sliding Window Techniques: Involves creating overlapping segments where each new segment shares some portion of data with the previous one. This technique is widely used in streaming data analysis and real-time monitoring systems.

Original file of heart sound recorded

60 second

Figure 3.14 Size of original file of phonocardiogram recorded.

Separate file Original to 11 files for match label

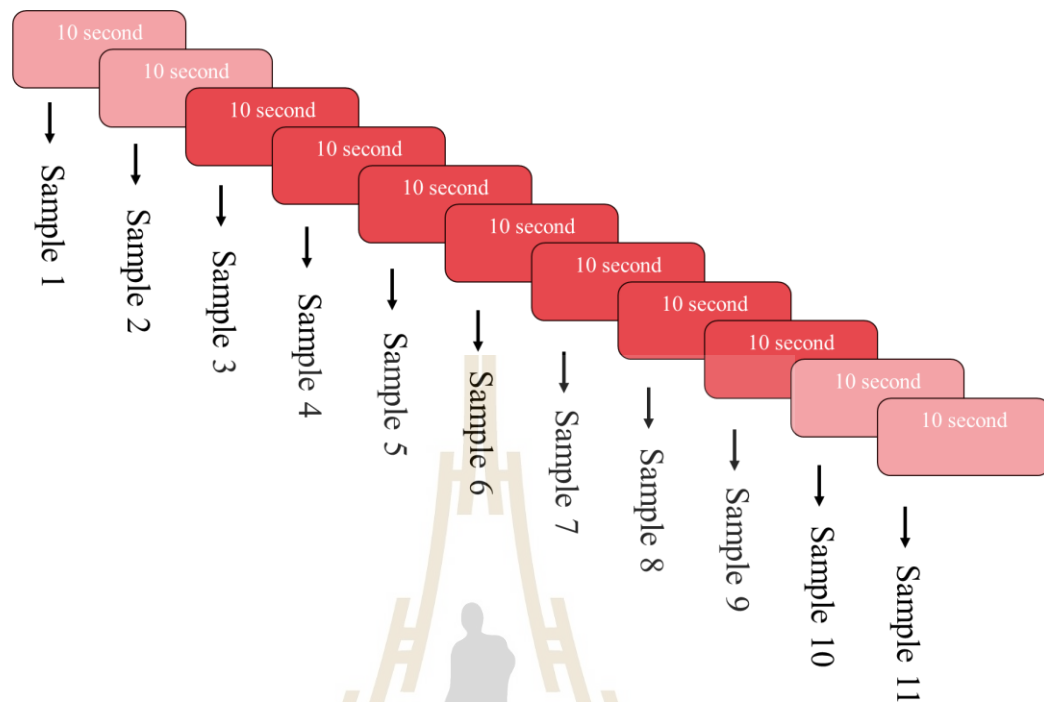


Figure 3.15 Segmentation data from original to 11 samples.

The data segmentation technique employed in this study partitions the data into phases, each comprising a 10-second window overlapping the preceding one by 5 seconds. This method is pivotal in modern data analysis and machine learning as it offers a systematic approach to handling extensive and intricate datasets. By breaking down the information into manageable segments, analysts and models can concentrate on specific attributes or trends, fostering deeper insights and more precise predictions. As data volumes and complexities expand, the significance of proficient data segmentation techniques escalates in extracting actionable insights from expansive datasets.

3.7 Estimating Blood Pressure from Phonocardiograms.

In this research, the estimation of blood pressure from phonocardiograms will be performed through the creation of an algorithm using deep machine learning methods. This form of machine learning enables the adjustment of internal values or

components to achieve an accurate pressure estimate. The algorithm development process is outlined as follows:

3.7.1 Preparation of data on phonocardiograms and blood pressure values

To record blood pressure and phonocardiogram data for one cycle, each recorded file captured 60 seconds of audio data from 78 volunteers, resulting in a total of 78 files. The systolic pressure ranged from 100-145 mmHg, and the diastolic pressure ranged from 50-100 mmHg. The recordings were divided into 78 cases. Furthermore, the subjects' physical data were documented in Excel for convenient access, as illustrated in Table 3.1.

Table 3.1 physical information of volunteer.

Number	Weight	Hight	Age	gender
1	65	165	33	Male
2	58	166	25	Female
3	59	161	30	Female
4	57	160	29	Female
5	68	165	33	Male
6	64	164	28	Female
7	56	161	28	Female
8	65	170	26	Male
9	54	155	28	Female
10	43	155	35	Female
11	56	164	34	Female
12	79	177	35	Male
13	86	161	29	Female
14	67	150	31	Female
15	84	171	34	Male
16	55	165	27	Male
17	64	164	34	Female
18	80	183	33	Male

Table 3.1 physical information of volunteer. (Continued)

Number	Weight	Hight	Age	gender
19	69	171	27	Male
20	66	163	58	Female
21	45	150	45	Female
22	72	175	36	Male
23	68	174	41	Male
24	60	161	31	Female
25	52	165	30	Male
26	72	155	47	Female
27	62	156	27	Female
28	58	162	31	Female
29	68	165	28	Female
30	56	157	41	Female
31	91	181	61	Male
32	51	167	22	Female
33	60	173	21	Male
34	71	174	51	Male
35	62	163	34	Female
36	60	156	51	Female
37	55	163	29	Male
38	62	162	30	Female
39	55	158	33	Female
40	65	178	29	Male
41	60	165	29	Male
42	85	163	36	Female
43	87	157	22	Female
44	58	155	57	Female
45	70	171	39	Male
46	95	173	40	Male

Table 3.1 physical information of volunteer. (Continued)

Number	Weight	Hight	Age	gender
47	89	163	35	Female
48	77	160	52	Female
49	52	162	33	Male
50	82	170	30	Male
51	57	166	21	Female
52	50	153	28	Female
53	58	150	35	Female
54	71	155	51	Female
55	68	170	24	Male
56	50	149	21	Female
57	55	145	56	Female
58	56	160	38	Female
59	82	176	34	Male
60	53	160	66	Female
61	87	158	22	Female
62	75	160	40	Female
63	64	156	52	Female
64	104	170	28	Male
65	60	172	40	Male
66	47	163	23	Female
67	57	165	28	Male
68	84	178	39	Male
69	60	165	61	Male
70	74	158	59	Female
71	75	175	34	Male
72	79	175	51	Male
73	82	170	33	Female
74	50	159	29	Female

Table 3.1 physical information of volunteer. (Continued)

Number	Weight	Hight	Age	gender
75	70	168	29	Male
76	60	163	28	Female
77	80	180	35	Male
78	43	153	29	Female

Incorporating physical data such as age, gender, height, weight, and heart rate (HR) significantly enhances the accuracy of blood pressure prediction models. This integration allows for the personalization of the estimation process, capturing individual characteristics that directly influence blood pressure values.

Physical data augmentation involves increasing the volume and variety of information used in training models. Features like age, gender, height, weight, and HR provide additional context, enabling machine learning models to identify patterns and correlations more effectively. This approach is crucial for developing robust models that generalize well across different populations.

Incorporating physical data helps capture the unique characteristics of individuals, significantly impacting blood pressure values:

1. Age: Blood pressure tends to change with age due to the natural aging process, including arterial stiffening.
2. Gender: Men and women may have different average blood pressure levels due to hormonal and physiological differences.
- 3 Height: Height indicates the size of blood vessels and the blood volume circulating in the body, influencing blood pressure values.
4. Weight: Weight affects blood pressure through body fat distribution, with overweight individuals placing additional strain on the heart and blood vessels.
5. Heart Rate (HR): HR is directly related to the cardiovascular system's functioning, aiding in understanding blood pressure dynamics.

Physical data enhances the relationship between various health parameters and blood pressure. Studies show that models incorporating such data are better at

estimating blood pressure, considering each individual's physiological context. This leads to more personalized and accurate predictions.

In this research, a total of 78 samples of phonocardiogram data were recorded, divided into two parts: 1) the training set used to train the machine learning model, comprising 52 samples, which accounts for 66 percent of the total data; and 2) the testing set used to evaluate model performance, consisting of 26 samples, which accounts for 33 percent of the total data.

3.7.2 Creating a deep learning algorithm to estimate blood pressure.

The information from the PCG signals is input into a convolutional neural network (CNNs) using the traditional CNNs-based method to estimate blood pressure. CNNs are adapted to leverage their ability to learn hierarchical patterns from sequences observed over time, making them suitable for time-dependent applications. This involves applying 1D convolutional neural networks over an input signal composed of multiple input planes. These modifications enable CNNs to handle various time series tasks, as illustrated in equations (3.1, 3.2). Let X denote the feature matrix extracted from the PCG signals. The CNNs utilizes the weights θ within the network to transform these features into an estimated blood pressure:

$$y[t] = \sum_{i=0}^{k-1} x[t+i] \cdot w[i] + b \quad (3.1)$$

Where

$y[t]$ is the value of the output at time step t .

$x[t+i]$ represents the input values at time steps $t, t+1, t+2, \dots, t+k-1$.

$w[i]$ represents the convolutional kernel weights at index i .

b is the bias term.

$$L_{out} = \left[\frac{L_{in} + 2(p) - 1(k-1) - 1}{s} + 1 \right] \quad (3.2)$$

L_{out} is the length of the output sequence.

L_{in} is the length of the input sequence.
 P is the amount of padding added to each side of the input sequence.
 k is size of the convolving kernel.
 s is stride of the convolution.

The network comprises several convolutional layers, pooling layers, fully connected layers, and an output layer for regression. Convolutional layers apply a set of learnable filters to the input features. Figure 3.17 illustrates the intended structure of the Conv1D model, derived from the input to the solution equation (3.2). The method employs a 5-layer CNNs, showcasing the constituent elements of each layer.

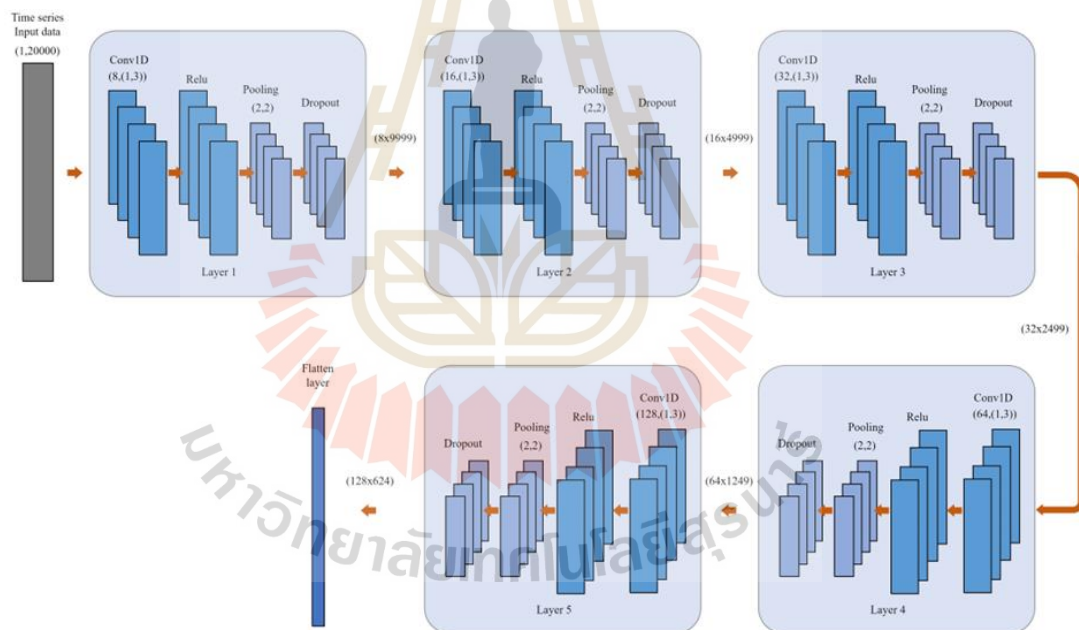


Figure 3.16 A schematic representation of the proposed Conv1D model architecture.

Overall method using 5-layer CNNs and shown the component of CNNs layer.

The architecture of CNNs-based models comprises multiple layers, such as convolutional layers, pooling layers, and fully connected layers. Specifically, convolutional layers within the CNNs-based model dynamically adjust the sampling locations within the feature map illustrated in Figure 3.16. This adaptive adjustment

enhances the model's capability to capture intricate patterns present in the PCG signals.

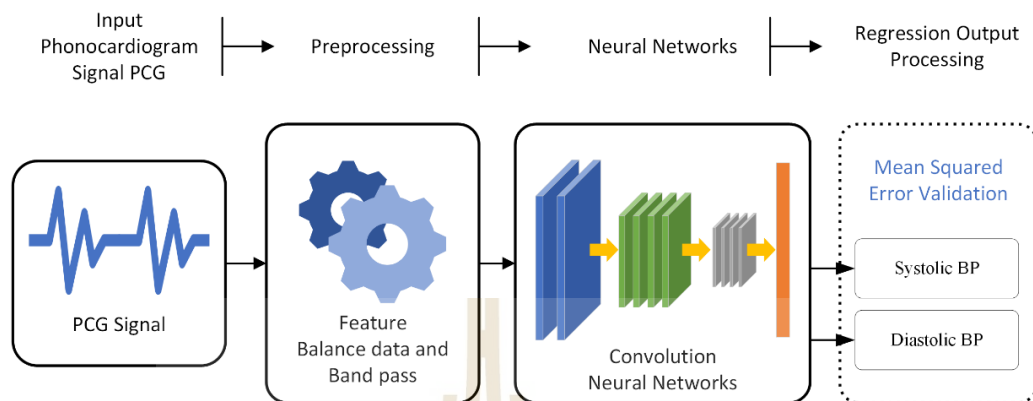


Figure 3.17 A block diagram depicting the overall system of estimate BP from the PCG signal.

3.7.3 Creating a deep hybrid learning algorithm to estimate blood pressure.

Developing an algorithm for estimating blood pressure values entails employing deep hybrid machine learning techniques. This involves utilizing labeled training datasets to train the algorithm, enabling it to learn, retain information, and construct a predictive model. This process is depicted in Figure 3.18.

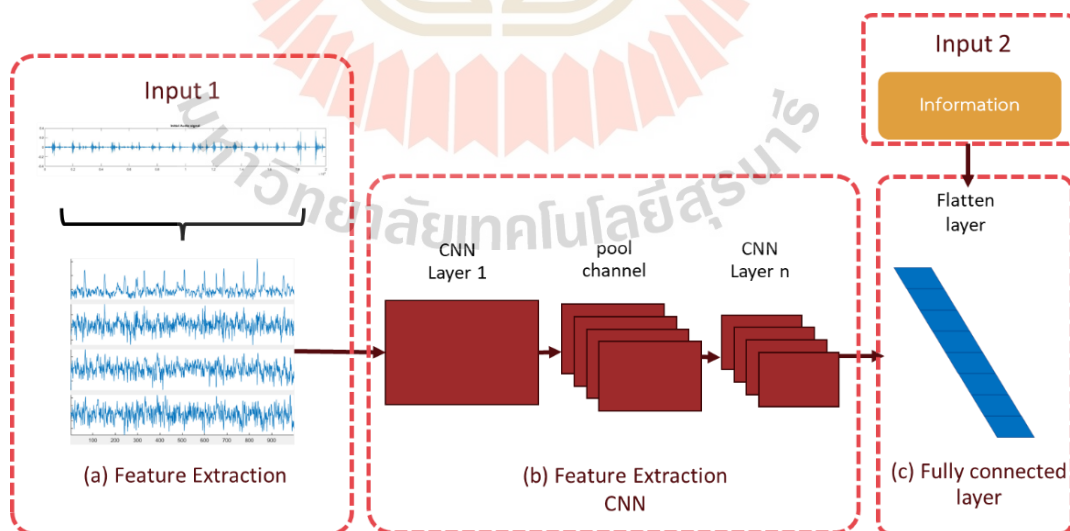


Figure 3.18 Learning process of deep learning algorithm.

3.7.3.1 Algorithm performance testing

Creating a data model for use in the classification process is a key component of this research. A machine learning method will be employed for data modeling. Based on the model from the research "A comprehensive comparison of different regression techniques and nature-inspired optimization algorithms to predict carbonation depth of recycled aggregate concrete," the model was used as follows.

Support Vector Regression (SVR)

SVR uses the principles of Support Vector Machines (SVM) but adapts them for continuous regression prediction problems instead of classification. SVR finds the hyperplane with the widest margin between the predicted value and the actual value, taking into account data points close to the hyperplane, called support vectors.

Advantages

1. Flexibility: SVR can handle non-linear relationships by using various kernel functions (e.g., polynomial, radial basis function).
2. Robustness: It is less affected by outliers because it focuses on data points near the hyperplane (support vectors).
3. Efficiency in Small Datasets: Performs well with small to medium-sized datasets, making it suitable for scenarios with limited data.

Disadvantages

4. Computational Complexity: Training can be computationally intensive, especially with large datasets.
5. Parameter Sensitivity: Requires careful tuning of parameters such as the choice of kernel and regularization parameter, which can be complex.
6. Scaling: Performance can degrade with large datasets due to its quadratic complexity in terms of the number of data points.

Kernel Extreme Learning Machine (KELM)

KELM is an extension of the Extreme Learning Machine (ELM) that uses a kernel trick to increase its flexibility and ability to handle high-dimensional data.

KELM randomizes hidden layer parameters and uses a linear resolution equation to calculate the parameters of the output layer.

Advantages

1. Fast Learning: KELM has a fast-training speed due to the randomization of hidden layer parameters.
2. High-Dimensional Data Handling: The use of kernel tricks allows it to effectively manage high-dimensional data.
3. Simplicity: It has a simple structure and fewer parameters to tune compared to traditional neural networks.

Disadvantages

1. Randomization Dependency: The random initialization of hidden layer parameters can lead to variability in results.
2. Generalization: May not generalize as well as other methods for complex datasets.
3. Single Hidden Layer: Limited to a single hidden layer, which can restrict its ability to capture complex patterns compared to deep learning models.

Extreme Gradient Boosting (XGB)

XGB is an enhancement of ensemble learning using a boosting technique that builds multiple Decision Tree models sequentially. Each model attempts to correct the errors of the previous model, resulting in highly accurate final results.

Advantages

1. High Accuracy: XGB often provides highly accurate predictions due to its sequential error-correction approach.
2. Flexibility: Can handle various types of data (e.g., numerical, categorical) and missing values effectively.
3. Feature Importance: Provides insights into feature importance, helping in feature selection and understanding the model.

Disadvantages

1. Computationally Intensive: Training can be slow and resource-intensive, especially with large datasets.
2. Overfitting: Prone to overfitting if not properly tuned, requiring careful cross-validation and parameter tuning.
3. Complexity: The model and its implementation can be complex, requiring a good understanding of boosting techniques and hyperparameter optimization.

The next stage is to evaluate the algorithm's performance using unlabeled testing datasets—datasets the algorithm has never encountered before—after determining the appropriate parameter values. Figure 3.19 illustrates this testing procedure. The selection between SVR, KELM, and XGB for blood pressure prediction depends on the characteristics of the data and the advantages and disadvantages of each method. SVR requires flexibility in the use of kernel functions and is suitable for learning from small datasets. Using the kernel method, KELM is ideal for managing complex data and enabling quick learning. Meanwhile, XGB excels in handling complex, large datasets and making highly accurate predictions.

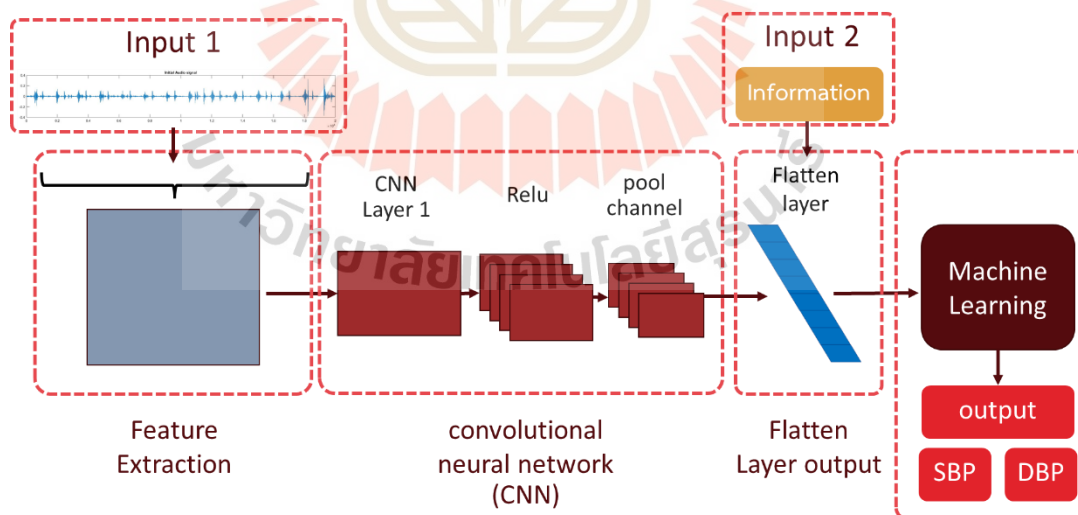


Figure 3.19 Testing process of Deep hybrid learning algorithm.

3.8 Dividing The Dataset For Building Training And Testing Process

Dividing The Dataset For Building Training And Testing

K-Fold Cross Validation involves dividing data into K equal parts to build and test a model. This method calculates the average accuracy or error before using the model to predict a dataset. Dividing the data into K folds means splitting the dataset into sets for training and testing by specifying the number K as the number of sets to be divided. Two values commonly used in practice are K=5 or K=10. In this study, the researcher has divided the data into K=3 folds as in Figure 3.21.

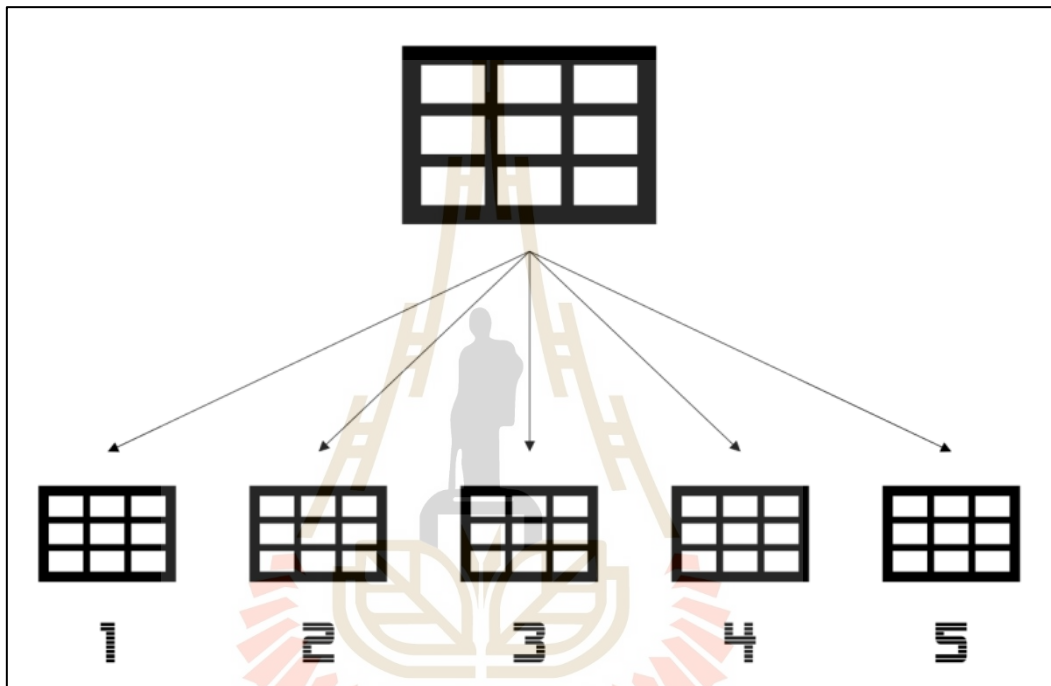


Figure 3.20 Data set arrangement format using K-Fold Cross Validation method.

Once the data has been partitioned into K=3, the model is built and tested until every fold of data has been used. With K=3, the model must be trained three times using the training folds, and similarly, it must be tested three times using the test sets (validation folds).

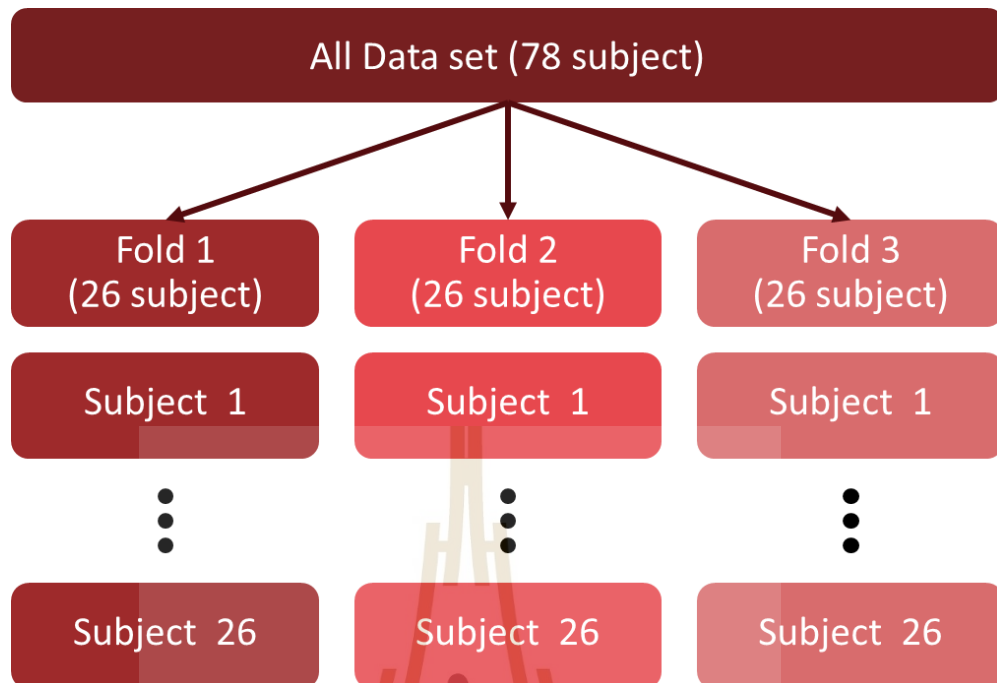


Figure 3.21 Data organization for this research with 78 cases.

The researcher defines the data set for each fold as follows.

- 1) Fold 1 is set to store data sets of cases 1, 4, 6, 14, 15, 16, 22, 23, 26, 32, 34, 36, 40, 48, 52, 54, 55, 58, 59, 63, 65, 66, 70, 71, 75 and 76
- 2) Fold 2 is set to store data sets of cases 3, 5, 7, 8, 11, 12, 18, 19, 20, 24, 28, 33, 37, 38, 41, 42, 43, 44, 51, 56, 61, 62, 64, 72, 77 and 78.
- 3) Fold 3 is set to store data sets of cases 2, 9, 10, 13, 17, 21, 25, 27, 29, 30, 31, 35, 39, 45, 46, 47, 49, 50, 53, 57, 60, 67, 68, 69, 73 and 74.

The Fold Cross Validation process is used to group data before utilizing these grouped datasets to train models across three different learning types.

Table 3.2 Dividing the 1st fold data according to the number of the data.

Number	Level	SYS	DIA
4	1	102	66
6	1	101	57
36	1	105	72
52	1	102	57
76	1	104	75
32	1	110	66
66	1	107	72
1	1	117	70
15	1	117	70
23	1	114	79
40	1	113	69
58	1	113	62
71	1	116	64
14	2	114	84
48	2	116	81
22	2	125	74
59	2	126	84
75	2	124	65
34	2	129	80
63	3	142	86
65	3	126	90
16	3	130	76
55	3	131	85
54	3	135	87
70	3	137	74
26	3	139	81

Table 3.3 Dividing the 2nd fold data according to the number of the data.

Number	Level	SYS	DIA
7	1	101	65
11	1	102	55
51	1	103	61
56	1	103	62
18	1	106	72
33	1	108	69
78	1	111	71
3	1	119	75
19	1	116	67
24	1	113	78
41	1	114	64
61	1	116	74
77	1	117	76
28	2	122	84
62	2	113	82
38	2	126	79
43	2	124	77
37	2	128	74
64	2	127	80
12	3	119	86
72	3	129	87
20	3	132	65
42	3	133	77
8	3	136	73
5	3	138	79
44	3	141	83

Table 3.4 Dividing the 3th fold data according to the number of the data.

Number	Level	SYS	DIA
25	1	100	65
27	1	105	74
30	1	102	69
74	1	100	68
29	1	106	76
60	1	112	71
69	1	109	65
13	1	118	78
21	1	113	57
35	1	115	73
57	1	118	72
67	1	117	69
10	1	106	80
46	2	117	82
2	2	125	82
47	2	126	82
73	2	124	78
45	2	128	78
68	2	127	78
49	3	125	90
31	3	130	87
39	3	132	78
53	3	135	85
17	3	136	94
9	3	139	78
50	3	142	79

3.8.2 Testing Process

From the Fold Cross Validation process used to group the data sets for training the model. This process will also be used for testing the dataset as follows.



Figure 3.22 Shows the training and testing datasets.

From Figure 3.22 it tells how to take a group of data sets for testing.

The first iteration test is to use the Fold 2 and Fold 3 data groups as training datasets. To test the Fold 1 data group

The second iteration test is to use Fold 1 and Fold 3 data groups as training datasets. To test the Fold 2 data group

The third iteration test is to use Fold 1 and Fold 2 data groups as training datasets. To test the Fold 3 data group

Absolute Error is the amount of difference between the actual value and the value obtained from the estimation, which can be calculated as in this equation below:

$$Absolute\ Error = |BP_{measure} - BP_{predict}| \quad (3.3)$$

Mean Absolute Error

$$Mean\ Absolute\ Error = \frac{1}{n} \sum_{i=1}^n |BP_{measure} - BP_{predict}| \quad (3.4)$$

Standard Deviation

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (X_i - \mu)^2}{N}} \quad (3.5)$$

3.9 Summary

This chapter outlines methodologies for gathering phonocardiogram data and blood pressure values using conventional data storage devices. It delineates the recording locations and establishes standards to ensure consistent recording formats and postures among experimenters. The acquired information undergoes analysis, with phonocardiogram signals segmented using specialized software. Subsequently, the dataset is split into a learning dataset, for modeling blood pressure estimation from phonocardiograms, and a testing dataset, for model performance evaluation. Furthermore, the chapter details the creation of an automatic algorithm adjustment system, termed the Calibration system, alongside notable feature extraction techniques. These techniques encompass data band decomposition and deep data scaling, with bandpass filtering employed for frequency band decomposition and CNNs convolutional neural networks for data size adjustment (Deep Learning). Additionally, the Extreme Gradient Boosting (XGB), support vector regression (SVR) and kernel extreme learning machines (KELM) estimation method, a machine learning approach, is utilized to discern estimation pattern disparities by learning from extracted outstanding features adjusted to the data size. Finally, the chapter evaluates the performance of these methodologies in blood pressure estimation.

CHAPTER IV

RESULTS AND DISCUSSION

4.1 Introduction

This chapter introduces a dataset comprising phonocardiograms recorded simultaneously with blood pressure values using an automatic blood pressure monitor. These datasets are examined and prepared for deep testing with hybrid machine learning algorithms. The process involves capturing phonocardiograms and filtering these signals to isolate specific sounds. Signal feature extraction is performed using a band-pass filter. This dataset is then fed into a Convolutional Neural Network (CNNs) to generate a new dataset suitable for machine learning and estimation testing. The pressure values from the test dataset are used to record the estimation results.

From the development and testing of the estimation, experiments were conducted, and the results were collected in four formats as follows:

4.1.1 The phonocardiograms dataset from 78 volunteers is band-pass filtered for pre-processing and then processed using different CNNs modalities.

4.1.2 Phonocardiograms are processed using Conv1D and then through different CNNs formats.

4.1.3 The phonocardiograms dataset from 78 volunteers is band-pass filtered and processed using various types of CNNs hidden layers; the values obtained from the flatten layer are then input into machine learning models.

4.1.4 Phonocardiograms are processed through CNNs; the values obtained from the flatten layer are combined with the physical information of the volunteers and then entered machine learning models.

4.2 Information Collected Blood Pressure From Volunteers.

This table provides a comprehensive dataset encapsulating vital statistics collected from volunteers who participated in a health-related study. Each column in the table is meticulously designed to capture specific aspects of the volunteers' physiological and demographic characteristics, which are essential for the study's objectives. The recorded data spans various parameters, each playing a critical role in analyzing and understanding health trends among the sampled population.

Table Columns Explained:

1. Number: This column assigns a unique identifier to each volunteer, ensuring that individual data can be accurately tracked and referenced throughout the study without compromising participant privacy.

2. SYS (Systolic Blood Pressure): This measure indicates the maximum pressure in the arteries when the heart beats and fills them with blood.

3. DIA (Diastolic Blood Pressure): This measure records the pressure in the arteries when the heart rests between beats.

4. Pulse: The pulse rate is recorded to assess the heart rate and rhythm, which are indicative of the volunteers' cardiac function.

5. PtP (Peak to Peak): This measurement, typically used in signal processing, might refer to variations in a physiological signal, providing insights into the variability of certain health parameters over time.

6. W (Weight): The weight of each volunteer is tracked, offering data crucial for evaluating general health and risks related to weight-related conditions.

7. H (Height): Height is recorded to assist in calculating body mass index (BMI) and other health-related metrics that require stature as a parameter.

8. Age: Age data is fundamental for assessing risks and conditions that may vary with different life stages.

9. Gender: Gender information helps in analyzing trends and differences in health markers across male and female participants, which can be vital for personalized healthcare insights.

Table 4.1 Information collected from volunteers.

Number	SYS	DIA	Pulse	PtP	W	H	Age	Gender
1	117	70	83	47	65	165	33	Male
2	125	82	73	43	58	166	25	Female
3	119	75	103	44	59	161	30	Female
4	102	66	83	36	57	160	29	Female
5	138	79	91	59	68	165	33	Male
6	101	57	72	44	64	164	28	Female
7	101	65	73	36	56	161	28	Female
8	136	73	80	63	65	170	26	Male
9	139	78	72	61	54	155	28	Female
10	106	80	83	26	43	155	35	Female
11	102	55	81	47	56	164	34	Female
12	119	86	96	33	79	177	35	Male
13	118	78	96	40	86	161	29	Female
14	114	84	73	30	67	150	31	Female
15	117	70	64	47	84	171	34	Male
16	130	76	75	54	55	165	27	Male
17	136	94	93	42	64	164	34	Female
18	106	72	98	34	80	183	33	Male
19	116	67	58	49	69	171	27	Male
20	132	65	85	67	66	163	58	Female
21	113	57	53	56	45	150	45	Female
22	125	74	80	51	72	175	36	Male
23	114	79	96	35	68	174	41	Male
24	113	78	83	35	60	161	31	Female
25	100	65	64	35	52	165	30	Male
26	139	81	69	58	72	155	47	Female
27	105	74	91	31	62	156	27	Female

Table 4.1 Information collected from volunteers. (Continued)

Number	SYS	DIA	Pulse	PtP	W	H	Age	Gender
28	122	84	68	38	58	162	31	Female
29	106	76	65	30	68	165	28	Female
30	102	69	83	33	56	157	41	Female
31	130	87	72	43	91	181	61	Male
32	110	66	76	44	51	167	22	Female
33	108	69	76	39	60	173	21	Male
34	129	80	75	49	71	174	51	Male
35	115	73	60	42	62	163	34	Female
36	105	72	73	33	60	156	51	Female
37	128	74	83	54	55	163	29	Male
38	126	79	93	47	62	162	30	Female
39	132	78	80	54	55	158	33	Female
40	113	69	61	44	65	178	29	Male
41	114	64	85	50	60	165	29	Male
42	133	77	80	56	85	163	36	Female
43	124	77	71	47	87	157	22	Female
44	141	83	85	58	58	155	57	Female
45	128	78	80	50	70	171	39	Male
46	117	82	87	37	95	173	40	Male
47	126	82	98	44	89	163	35	Female
48	116	81	69	35	77	160	52	Female
49	125	90	81	35	52	162	33	Male
50	142	79	73	63	82	170	30	Male
51	103	61	72	42	57	166	21	Female
52	102	57	93	45	50	153	28	Female
53	135	85	76	50	58	150	35	Female
54	135	87	64	48	71	155	51	Female

Table 4.1 Information collected from volunteers. (Continued)

Number	SYS	DIA	Pulse	PtP	W	H	Age	Gender
55	131	85	93	46	68	170	24	Male
56	103	62	89	41	50	149	21	Female
57	118	72	67	46	55	145	56	Female
58	113	62	64	51	56	160	38	Female
59	126	84	96	42	82	176	34	Male
60	112	71	89	41	53	160	66	Female
61	116	74	64	42	87	158	22	Female
62	113	82	72	31	75	160	40	Female
63	142	86	60	56	64	156	52	Female
64	127	80	87	47	104	170	28	Male
65	126	90	66	36	60	172	40	Male
66	107	72	103	35	47	163	23	Female
67	117	69	91	48	57	165	28	Male
68	127	78	85	49	84	178	39	Male
69	109	65	75	44	60	165	61	Male
70	137	74	80	63	74	158	59	Female
71	116	64	54	52	75	175	34	Male
72	129	87	72	42	79	175	51	Male
73	124	78	87	46	82	170	33	Female
74	100	68	81	32	50	159	29	Female
75	124	65	57	59	70	168	29	Male
76	104	75	73	29	60	163	28	Female
77	117	76	81	41	80	180	35	Male
78	111	71	98	40	43	153	29	Female

Table 4.1 is designed to offer a comprehensive dataset, systematically organized to facilitate the analysis of health variables influenced by physiological measurements and demographic factors. This structured information will empower

researchers to derive meaningful insights regarding health conditions, the factors affecting them, and associated outcomes. Such insights can inform the development of targeted health strategies and interventions customized for specific populations. Beyond serving as a cornerstone for statistical analysis, this table enriches the understanding of diverse health dynamics prevalent within communities.

4.3 Performance of Estimate With 1D CNNs Base Approach.

The Mean Absolute Error (MAE) for systolic and diastolic pressure estimation by the proposed model is presented, along with a comparison to the acceptable margin of error defined by the Association for the Advancement of Medical Instrumentation (AAMI) standard. This comparison underscores the clinical relevance and applicability of the model's performance.

Detailed Statistical Analysis:

1. Report confidence intervals for MAE values to provide a measure of statistical certainty.
2. Conduct hypothesis testing to validate the significance of the observed improvements.

4.3.1 Performance of PCG signals data with CNNs model

In this subsection, we conduct a comparative analysis of our proposed method against several established systems. Due to variances in experimental models used in database-based approaches, some systems may not be included. Here, we focus on aligning experimental conditions with our own and present a comparison of results accordingly. Selected known systems are included to facilitate continuous testing and comparison with our tested methods. Table 1 showcases the comprehensive performance metrics, encompassing mean absolute error (MAE) and standard deviation (STD) for both training and testing phases of each network. Notably, the CNNs architecture featuring three CNNs layers and three hidden layers, along with a linear layer for testing the new dataset, is highlighted. This network leverages PCG signals recorded from 78 volunteers in our dataset.

Mean blood pressure (MBP) is usually calculated with a standard formula as follows: $MBP = \text{diastolic blood pressure (DBP)} + \frac{1}{3} [\text{systolic blood pressure (SBP)} - \text{DBP}]$.

Table 4.2 Performance of PCG signals data with CNNs model

MAE SBP	MAE DBP	STD SBP	STD DBP	MBP
mmHg	mmHg	mmHg	mmHg	mmHg
32.46	19.81	24.97	15.39	24.02

The MAE SBP value of 32.46 mmHg represents the average systolic blood pressure in the STD dataset, while an SBP STD of 24.97 mmHg indicates the degree of variability or dispersion of SBP values. A higher standard deviation signifies a wider measurement range. Similarly, a broader SBP MAE DBP value of 19.81 mmHg represents the average diastolic blood pressure in the STD dataset, with a DBP STD of 15.39 mmHg indicating variability in DBP measurements. The MBP of 26.13 mmHg represents the average of both SBP and DBP, offering insight into overall blood pressure levels. In summary, Table 4.2 presents a test of this dataset when imported into the estimation process using the CNNs technique, assessing whether the newly recorded data can yield a preliminary estimation. However, as indicated in the table, while this new dataset can be utilized in the basic CNNs system, the tolerances of the test results are deemed unacceptable. Consequently, the next step involves further improving the test by rebalancing the dataset to ensure a balance in the values that can be estimated, thereby enhancing the machine learning process's ability to learn from an expanded dataset.

4.3.2 Performance of PCG signals with additive white gaussian noise

In our analysis of the newly acquired PCG signal data, we observed significant fragmentation, particularly with an underrepresentation of higher blood pressure readings. To address this imbalance and enhance the dataset's effectiveness for model training, we deliberately introduced Additive White Gaussian Noise (AWGN). This method was chosen to augment the dataset with a broader spectrum of blood pressure readings. By enriching the dataset in this manner, we aimed to incorporate

additional data points spanning a wider range of blood pressures, thereby fostering more robust model training.

The effect of this data augmentation is visually illustrated in Figure 8. Kernel density estimates offer a clear comparison between the distribution of the original dataset and the one enriched with AWGN. These density plots vividly show how the application of AWGN effectively broadens the representation of higher blood pressure values, thereby alleviating the initial disproportionality and promoting a more balanced dataset for our model development. Table 4.3 presents the MAE results obtained with the AWGN method.

Table 4.3 Performance of PCG signals with additive white gaussian noise (AWGN).

Model	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
1D CNNs w/o AWGN	32.46	19.81	24.97	15.39	24.02
1D CNNs with AWGN	15.46	9.53	11.32	7.76	11.50

The introduction of AWGN notably impacts blood pressure estimation, as evident from the comparison between the two rows in Table 2. The data reveal a considerably lower MAE when the dataset is balanced across diverse blood pressure values. This underscores the notion that augmenting the distribution of information during training enhances the efficacy of learning, as it provides a more comprehensive set of datasets with known values for training. However, testing the results indicates that incorporating a more balanced number of training sessions into the dataset leads to weaker estimation outcomes.

4.3.3 Performance of Different Numbers of CNNs Block Layers

To assess the efficacy of the CNNs model design, we conducted tests and compared the outcomes, focusing on the variability in extracting PCG signal features across varying numbers of CNNs layers. The results of these investigations are detailed in Table 4.4.

Table 4.4 Performance of numbers CNNs block layer.

Number of CNNs block	MAE SBP	MAE DBP	STD SBP	STD DBP	MBP
	mmHg	mmHg	mmHg	mmHg	mmHg
1	147.66	74.81	102.99	49.47	99.10
2	15.90	10.53	13.66	8.29	12.32
3	15.46	9.53	11.32	7.76	11.50
4	14.21	10.03	10.35	6.59	11.42
5	12.68	8.31	8.27	5.86	9.07
6	14.27	9.92	10.88	7.66	11.37
7	13.84	9.63	10.31	7.43	11.03

Experiments were conducted to evaluate how altering the number of CNNs layers affected model accuracy. The findings indicated that increasing the number of CNNs blocks initially enhanced model accuracy, reaching peak performance with five layers. However, beyond this threshold, additional layers yielded no further improvements, suggesting an optimal number of layers for this particular application.

The experimental table presents various modifications of the CNNs layers and compares their performance. Notably, one experiment yielded significantly low error values, with a pressure error of 12.68 mmHg for the upper pressure and 8.31 mmHg for the lower pressure, as depicted in Table 3. The results in Table 3 demonstrate a consistent trend in the CNNs model's performance across different numbers of convolutional blocks, as measured by MAE for SBP and DBP blood pressure, their STD, and MBP. There is a discernible decrease in MAE for both SBP and DBP as the number of CNNs blocks increases from 1 to 5, indicating enhanced model accuracy. However, this improvement levels off beyond five layers, suggesting an optimal number of layers for this specific application. The standard deviations also exhibit a similar trend, decreasing with an increasing number of layers, indicating more consistent model predictions. This analysis underscores the efficacy of deep learning models in capturing intricate data patterns, striking a balance between model complexity and performance. Notably, the experimental results in Table 3 include

models with up to 7 layers of CNNs. Subsequent tests involving additional layers (8, 9, 10) resulted in underfitting of estimations, a trend that persisted with further layer additions, prompting exploration of alternative parameterizations.

4.3.4 Performance of Different Numbers of Linear Layers.

The subsequent phase entails scrutinizing Table 4.5 to evaluate the influence of different numbers of linear layers, following the model with 5 CNNs layers. This exploration seeks to gauge how diverse configurations of linear layers affect the model's accuracy and consistency in blood pressure estimation. Such an examination is pivotal for identifying the most efficient number and arrangement of linear layers to enhance the model's efficacy for the particular application under consideration.

Table 4.5 Performance of number of Linear Layers.

Number of classifier layer	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
1	19.05	74.73	13.08	8.58	56.17
2	18.77	10.18	12.40	7.22	13.05
3	12.68	8.31	8.27	5.86	9.77
4	12.50	8.80	9.25	6.20	10.04
5	14.66	10.27	10.37	7.57	11.73
6	11.68	7.88	7.44	6.11	9.15
7	12.18	8.81	8.15	5.68	9.93
8	15.05	9.94	10.63	7.09	11.64

In our analysis of Table 4.5, the focus was on the MAE performance of models with varying numbers of linear layers for blood pressure estimation. Several noteworthy trends have emerged, with the primary focus being on the number of linear layers within the model architecture.

Firstly, there is a clear relationship between the number of linear layers and MAE values. As the number of linear layers increases from 1 to 8, a consistent impact is observed on both systolic blood pressure (SBP) and diastolic blood pressure (DBP)

MAE. For SBP MAE, the highest value of 19.05 mmHg is associated with a model featuring 1 linear layer. However, as the number of linear layers progressively increases, SBP MAE consistently decreases, with the most accurate SBP estimation, an MAE of 11.68 mmHg, achieved when employing models with 6 linear layers. A similar trend is observed for DBP MAE, where the highest value of 74.73 mmHg is linked to a model with 1 linear layer, while the lowest DBP MAE of 7.88 mmHg is attained with 6 linear layers.

Secondly, the analysis extends to the standard deviation (STD) of SBP and DBP MAE. These STD values exhibit a consistent pattern mirroring the behavior of MAE. As the number of linear layers increases, STD values decrease, reflecting a reduction in variability. Models with 1 linear layer produce the highest STD values, signifying greater variability in MAE. Conversely, models with 6 linear layers yield the lowest STD values, indicating reduced variability in MAE.

Lastly, the Mean Blood Pressure (MBP) values align with the trends observed for SBP and DBP MAE. The highest MBP MAE is associated with 1 linear layer, while the lowest MBP MAE is achieved when employing models with 6 linear layers.

Further experiments evaluated the influence of the number of linear layers on model performance. Models with six linear layers provided the most accurate blood pressure estimates, with lower MAE and standard deviation values, indicating improved precision and consistency.

In summary, the analysis underscores the crucial role of selecting the appropriate number of linear layers within the model architecture for accurate blood pressure estimation. The model with 6 linear layers outperformed the other models, consistently demonstrating lower MAE and lower STD in MAE. These findings highlight the importance of optimizing the number of linear layers to effectively enhance the accuracy of blood pressure estimation. From the experimental results in Table 4.5, up to 8 linear layers are presented. This is because adding a 9th layer resulted in underfitting of the estimation, a trend that persisted with the addition of more layers.

Consequently, this led us to explore other types of parameterizations for further testing.

4.3.5 Performance MAE of Batch Sizes and Learning Rates.

The next step involves testing by adjusting the values in specific learning areas. This includes scaling the data to be fed into the learning section and compiling the learning rate values, as shown in Table 4.6.

Table 4.6 Performance MAE of Batch Sizes and Learning Rates.

Batch Size /Learning rates	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
256/0.01	11.68	7.88	7.44	6.11	9.15
256/0.001	11.43	7.97	7.79	6.37	9.13
256/0.0001	11.75	9.04	8.30	5.47	9.94
128/0.01	11.73	8.11	7.70	5.80	9.31
128/0.001	11.11	7.91	7.96	4.78	8.98
128/0.0001	12.22	8.87	8.11	5.39	9.99
64/0.01	10.26	7.81	7.95	4.99	8.63
64/0.001	12.43	9.03	9.22	6.02	10.16
64/0.0001	12.69	8.84	8.67	6.28	10.12
32/0.01	12.56	8.22	8.84	5.59	9.67
32/0.001	11.77	8.75	7.85	5.53	9.76
32/0.0001	14.65	10.55	10.47	7.19	11.92
16/0.01	10.69	6.89	7.23	5.22	8.16
16/0.001	14.18	9.62	9.89	6.53	11.14
16/0.0001	16.12	11.50	11.11	7.85	13.04

The analysis of the results presented in Table 5, which investigates the MAE performance of models across different batch sizes and learning rates, offers valuable insights into the influence of these hyperparameters on the accuracy of blood pressure estimation.

Considering the impact of learning rate on MAE, the results demonstrate that learning rate selection plays a critical role in accurate blood pressure estimation. A higher learning rate (0.01) tends to result in lower MAE values, indicating improved accuracy in estimating blood pressure. Conversely, the lowest learning rate (0.0001) often leads to higher MAE values, underscoring the importance of selecting an appropriate learning rate to achieve model convergence and accuracy. Table 5, which examines the MAE performance across different batch sizes and learning rates, reveals several notable trends. Smaller batch sizes, particularly 16, often demonstrate lower MAE values, suggesting better model performance. This improvement can be attributed to smaller batch sizes offering more frequent updates and potentially better generalization. The interaction between batch size and learning rate is also significant; for instance, a batch size of 16 with a learning rate of 0.01 shows notably lower MAE values across most entries in the table, indicating an optimal combination for this specific model and dataset. These findings suggest that both batch size and learning rate play crucial roles in model accuracy and should be carefully tuned to achieve optimal performance.

In Table 4.6, a batch size of 16 and a learning rate of 0.01 are observed to yield the lowest Mean Absolute Error (MAE) values across the board: SBP (10.69 mmHg), DBP (6.89 mmHg), STD SBP (7.23 mmHg), STD DBP (5.22 mmHg), and MBP (8.79 mmHg). This indicates that for this model and dataset, a smaller batch size combined with a moderately high learning rate optimizes performance, resulting in more accurate blood pressure predictions with lower variability.

4.4 Performance of Estimate with Deep Hybrid Learning.

Deep hybrid learning represents a cutting-edge approach in the field of artificial intelligence, integrating different machine learning models, including deep learning and traditional algorithms, to leverage their unique strengths. This synergy enhances model performance, particularly in tasks involving complex data interpretations and predictions. The goal of deep hybrid learning is to combine the

abstract feature extraction capabilities of deep learning with the precision and efficiency of classical machine learning techniques.

4.4.1 Performance MAE of Deep Hybrid learning.

Deep hybrid learning, which integrates diverse machine learning methodologies to harness their unique strengths, is increasingly pivotal in tackling complex predictive tasks across various domains. A prominent example of this approach is the combination of Convolutional Neural Networks (CNNs) to Support Vector Regression (SVR), Extreme Gradient Boosting (XGB) and Kernel Extreme Learning Machine (KELM). This hybrid model leverages the hierarchical feature extraction capabilities of CNNs alongside the robust regression techniques of SVR, aiming to deliver precise and reliable predictive performance in scenarios where high-dimensional data and complex regression tasks converge.

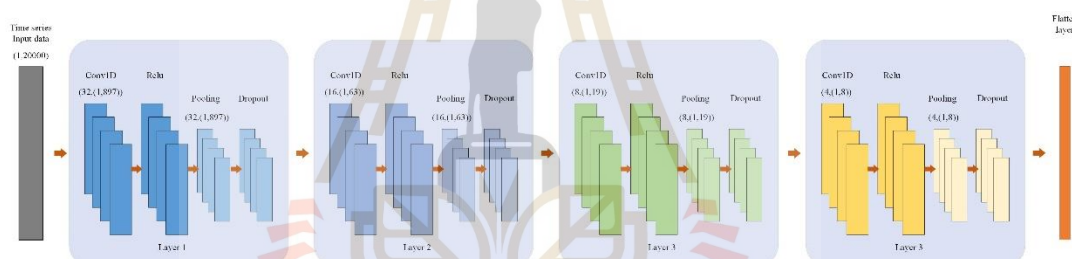


Figure 4.1 A schematic representation of the proposed Conv1D model.

Table 4.7 presents the performance of three different hybrid learning techniques for estimating systolic blood pressure (SBP) and diastolic blood pressure (DBP) using a combination of convolutional neural networks (CNNs) and various machine learning algorithms: Extreme Gradient Boosting (XGB), Support Vector Regression (SVR), and Kernel Extreme Learning Machine (KELM). The performance metrics considered are the Mean Absolute Error (MAE), Standard Deviation (STD) for both SBP and DBP, and Mean Blood Pressure (MBP).

1. CNNs to SVR

The combination of CNNs and SVR provides the lowest MAE and STD for both SBP and DBP, indicating that this method achieves the most accurate and

consistent blood pressure estimates among the three methods. SVR's ability to handle small datasets and its flexibility in using kernel functions likely contribute to its superior performance.

2. CNNs to KELM

While KELM provides fast learning capability and effectively handles complex data using the kernel trick, it shows a slightly higher STD for SBP compared to SVR, indicating less consistency. However, the MAE values are comparable to those of SVR, making KELM a competitive alternative.

3. CNNs to XGB

XGB shows higher MAE and MBP values, indicating that it is less accurate than SVR and KELM for this specific task. XGB builds multiple decision trees sequentially, with each tree attempting to correct the errors of the previous tree. This approach, while generally effective for high-accuracy predictions, may introduce inconsistencies in this context due to the complex nature of blood pressure data.

Table 4.7 Performance MAE of Deep Hybrid learning.

Hybrid Learning	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
CNNs + XGB	11.38	8.06	7.92	5.13	9.17
CNNs + SVR	10.49	7.08	6.83	5.20	8.22
CNNs + KELM	10.47	7.15	8.41	5.50	8.26

Deep hybrid learning, which integrates CNNs with other machine learning models like Extreme Gradient Boosting (XGB), Support Vector Regression (SVR) and Kernel Extreme Learning Machine (KELM), was evaluated. This hybrid approach leverages the strengths of both deep learning and traditional algorithms, resulting in improved predictive performance.

Integration with Physical Information:

Incorporating physical information (e.g., age, weight, height) into the deep hybrid models further enhanced predictive accuracy. This integration provided additional context, significantly improving the models' performance.

4.4.2 Performance MAE of Deep Hybrid learning with physical information

Incorporating physical information into deep hybrid models involves using data that captures the fundamental physical properties or behaviors of the system under study. This can include measurements from sensors, environmental data, or other tangible data that directly reflect the physical characteristics of the subject matter. The integration of such data is particularly valuable in fields like healthcare, environmental science, and engineering, where physical parameters are critical for enhancing interpretability and relevance in real-world scenarios.

Table 4.8 Performance MAE of Deep Hybrid learning with physical information.

Hybrid Learning	MAE SBP	MAE DBP	STD SBP	STD DBP	MBP
	mmHg	mmHg	mmHg	mmHg	mmHg
CNNs + XGB + information	11.08	7.38	8.00	5.39	8.61
CNNs + SVR + information	10.20	7.00	6.67	5.30	8.07
CNNs + KELM + information	11.87	7.29	7.94	5.84	8.82

The results presented in Table 4.7 provide a comparative analysis of three deep hybrid learning models—XGB (Extreme Gradient Boosting), SVR (Support Vector Regression), and KELM (Kernel Extreme Learning Machine) each enhanced with additional physical information. The analysis focuses on key performance metrics: Mean Absolute Error (MAE) and standard deviation (STD) for both systolic (SBP) and diastolic (DBP) blood pressures, as well as the Mean Blood Pressure (MBP).

Performance Metrics

Mean Absolute Error (MAE)

SVR + Information: This model consistently shows the lowest MAE for both SBP and DBP, indicating it provides the most accurate predictions among the three models. The low MAE values suggest that the SVR model, when augmented with physical information, effectively captures the underlying patterns in the data related to blood pressure.

XGB + Information: This model performs well but slightly underperforms compared to SVR. The MAE values are marginally higher, indicating a slight trade-off in prediction accuracy.

KELM + Information: This model has the highest MAE, indicating it is less accurate in predicting blood pressure compared to the SVR and XGB models. However, it still performs reasonably well, showing that KELM can be a viable option depending on the application context.

Standard Deviation (STD)

SVR + Information: The lowest STD values for both SBP and DBP indicate that the SVR model provides the most consistent predictions. This is crucial in clinical settings where variability in predictions can lead to significant differences in treatment outcomes.

XGB + Information: This model also shows low STD values, suggesting good consistency but slightly higher variability than the SVR model.

KELM + Information: The highest STD values for both SBP and DBP indicate that the KELM model has greater prediction variability, which could impact its reliability in clinical applications.

Using physical data such as age, gender, height, weight, and heart rate (HR) together with deep learning and machine learning has had a significant impact on blood pressure estimation as follows:

1. Increasing the efficiency of learning

Integrating physical data allows models to capture relationships between a wider range of data, such as the relationship between weight and blood pressure or heart rate and blood pressure.

Physical data improves the model's accuracy in estimating blood pressure because these data influence the blood pressure values of each person.

2. Personalization

Physical data helps models better capture individual characteristics, such as whether people are overweight or underweight or have different heart rates.

Combining this data allows the model to more accurately predict blood pressure values for each individual.

3. Managing small data

Using physical data allows models to learn better from smaller datasets. This is especially beneficial with limited blood pressure and PCG signal data.

Physical data increases the amount and variety of data used to train models, enabling the model to better capture various characteristics and patterns.

From Tables 4.7-4.8, which show the effectiveness of using physical data with deep learning and machine learning:

The model using CNNs + SVR + information produced the lowest MAE and STD values, demonstrating better blood pressure estimation performance than other techniques.

Models using CNNs + KELM + information and CNNs + XGB + information, although having higher MAE and STD values, still show better results than using these techniques without physical information.

4.5 Summary

4.5.1 Performance with difference machine learning

In our results, we conducted a detailed comparative analysis of three prominent deep hybrid learning models—XGB (Extreme Gradient Boosting), SVR (Support Vector Regression), and KELM (Kernel Extreme Learning Machine). Our evaluation focused on two key performance indicators: Mean Absolute Error (MAE) and the standard deviation of predictions for both systolic and diastolic blood pressure readings. The results indicate that the SVR model, when enhanced with additional physical information, not only surpasses its counterparts by achieving the lowest MAE values (10.20 mmHg for SBP and 6.67 mmHg for DBP) but also demonstrates remarkable consistency in its performance, evidenced by its competitive standard deviation metrics (7.00 mmHg for SBP and 5.30 mmHg for DBP).

The findings of this research underscore the critical impact of integrating physical information into traditional machine learning frameworks. By incorporating such data, the SVR model showed a significant improvement in predictive accuracy for both systolic and diastolic blood pressures. This approach holds substantial promise for enhancing clinical decision-making tools, providing a methodologically sound basis for the development of more reliable and accurate health monitoring systems. These advancements are especially valuable in clinical settings where precise blood pressure measurements are essential for the diagnosis and management of hypertensive patients.

Table 4.9 Comparison Performance MAE with using PCG signal.

work	Number of subjects	Signals	Feature	SBP mmHg		DBP mmHg	
				MAE	STD	MAE	STD
This work	78	PCG	CNNs	10.20	7.00	6.67	5.30
[1]	37	PCG	S21	6.48	4.48	3.91	2.58
[2]	85	PCG-PPG	VTT	6.67	8.47	-	-
[3]	24	PCG-PPG	PTT	7.47	11.08	3.56	4.53
[4]	10	PCG-PPG	VTT	4.07	3.07	5.61	4.09
[5]	32	PCG-PPG	PTT	6.22	9.44	3.97	5.15

4.5.2 Comparison estimation techniques with related work

From the research results, it was found that using technical data in testing allows for detailed and level comparisons of each test method to measure the effectiveness of the methods, as shown in Table 4.10.

Table 4.10 Comparison of the Methods for Estimation.

work	Number of subjects	Signals	Feature	Estimation Techniques
This work	78	PCG	CNNs	Deep Hybrid Learning
[1]	37	PCG	S21	Machine Learning
[2]	85	PCG-PPG	VTT	Multiple Regression
[3]	24	PCG-PPG	PTT	Calculated from PTT
[4]	10	PCG-PPG	VTT	Calculated from VTT and ET
[5]	32	PCG-PPG	PTT	Calculated from PTT -

In Table 4.10 compares the blood pressure estimation method used in the present study with previous research. The estimation techniques are considered. The comparison in the table shows that the current research used 78 volunteers and employed a Deep Hybrid Learning technique using a Convolutional Neural Network (CNNs) to create features from Phonocardiogram (PCG) signals. This approach combines deep learning and machine learning to increase accuracy in estimating blood pressure values. In contrast, other studies use a combination of PCG and Photoplethysmogram (PPG) signals and different features such as S21, Vascular Transit Time (VTT), Pulse Transit Time (PTT), and Ejection Time (ET), employing techniques such as Machine Learning, Multiple Regression, and direct feature calculations. The comparative method demonstrates that using Deep Hybrid Learning techniques can streamline the process to merely identifying signal characteristics and feeding them into machine learning, which effectively estimates blood pressure.

Research results using the Deep Hybrid Learning technique show advantages over other methods used in previous research as follows:

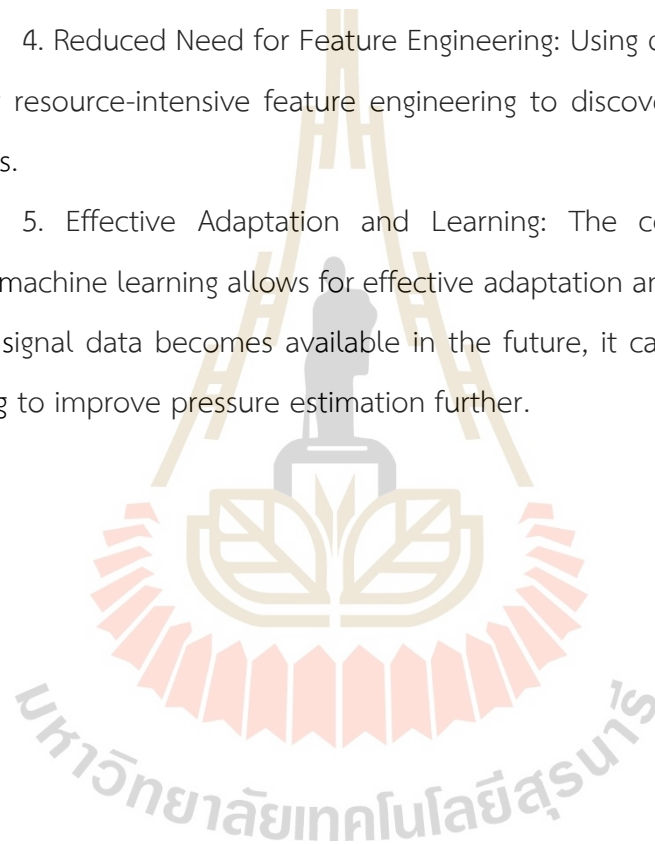
1. Accurate Feature Creation: Using Convolutional Neural Networks (CNNs) helps create complex, high-quality features from Phonocardiogram (PCG) signals, allowing for accurate estimation of blood pressure.

2. Process Simplification: While other research requires the use of multiple signals such as PCG and PPG, as well as multiple features, the hybrid learning techniques we use can reduce these complexities by relying on PCG signals alone.

3. Effective Small Data Management: Machine learning techniques in hybrid learning help deal with small amounts of data well, making it possible to work even with limited data.

4. Reduced Need for Feature Engineering: Using deep learning reduces the need for resource-intensive feature engineering to discover and process signal characteristics.

5. Effective Adaptation and Learning: The combination of deep learning and machine learning allows for effective adaptation and learning from data. If more PCG signal data becomes available in the future, it can be adapted to use deep learning to improve pressure estimation further.



CHAPTER V

CONCLUSIONS

5.1 Discussion

5.1.1 Machine Learning in Deep Hybrid Learning

Analyzing the experimental results using three different machine learning-based methods, including regression techniques, the results in Table 4.8 show that SVR is the most appropriate for blood pressure estimation in this case. This is because it has the lowest MAE and STD values, while XGB and KELM have slightly worse results. The SVR technique finds the hyperplane with the widest margin between the predicted value and the actual value, taking into account the data points used for learning. This allows for the estimation of values that are consistent with the available data sets. In contrast, the KELM technique uses random parameters in only one hidden layer, causing the calculation to quickly approach the test value but not as accurately as it should. The XGB technique, based on many decision trees sequentially correcting errors, may result in individual adjustments not being as consistent with the test data as possible, leading to a higher mean error compared to the SVR technique.

5.1.2 Adding Physical Information to Improve Blood Pressure Estimation

1. Data Augmentation

Incorporating physical data such as age, gender, height, weight, and heart rate (HR) enhances the diversity and quantity of information used in training the model. This provides the model with more data to identify patterns and make predictions more effectively.

2. Capturing Individual Characteristics (Personalization)

Physical data aids in capturing the unique characteristics of an individual, which can influence blood pressure values. For instance, individuals with higher

weight or varying heart rates may exhibit different blood pressure levels. By integrating this data, the model can more accurately predict blood pressure values tailored to each person.

3. Enhanced Correlation Between Data

Combining physical data allows the model to recognize relationships among various data aspects, such as the correlation between weight and blood pressure or heart rate and blood pressure. This improves the accuracy of the predictions.

4. Assumptions of the Relationship of the Data

Age, gender, height, weight, and heart rate (HR) are crucial factors influencing blood pressure values. Age impacts blood pressure through vascular deterioration and other age-related physical changes. Gender differences, driven by hormonal and physiological factors, lead to variations in average blood pressure levels between men and women. Height correlates with the size of blood vessels and blood volume, thereby affecting blood pressure. Weight influences blood pressure through fat distribution, with overweight individuals experiencing additional strain on the heart and vessels. Lastly, heart rate, as an indicator of cardiovascular function, directly affects blood pressure levels.

The physical data obtained from this collection was found to have several limitations. One of the main issues is that the collected data does not cover the full range of specified blood pressure values. Additionally, the physical information is limited. This limitation is primarily due to the data being collected under uniform conditions, as screened by medical experts. Consequently, in some cases, there is a lack of comprehensive blood pressure data along with the corresponding PCG signals.

For future development, efforts should focus on collecting a more diverse dataset. This includes expanding the range of blood pressure values to encompass a broader spectrum of levels. Such improvements will enhance the model's learning efficiency, enabling it to learn and estimate blood pressure values in various formats and levels with greater accuracy. This approach aims to create a more robust and reliable model capable of handling a wide range of blood pressure scenarios.

5.2 Conclusions

Blood pressure estimation is a rapidly evolving field with significant potential to positively impact healthcare. This thesis introduces deep hybrid learning as a method for estimating blood pressure from phonocardiogram (PCG) signals. The new design incorporates a convolutional neural network. The experiment collected phonocardiogram data sets along with blood pressure values while recording phonocardiograms. Data were collected through the human research process at Suranaree University of Technology.

By developing a hybrid model for use with phonocardiogram signals, this hybrid model combines the trainable feature representation of deep learning with the robust classification capabilities of Support Vector Regression (SVR). Testing these proposed models on the appropriate database demonstrated promising performance. The experimental results yielded Mean Absolute Error (MAE) and standard deviation (STD) values of approximately 10.49 ± 6.83 mmHg for systolic pressure and 7.08 ± 5.20 mmHg for diastolic pressure. Further development with deep hybrid learning improved the MAE and STD to approximately 10.20 ± 6.67 mmHg for systolic pressure and 7.00 ± 5.30 mmHg for diastolic pressure.

This thesis presents a new framework for estimating blood pressure from PCG and emphasizes the importance of hybrid models in optimizing performance. These findings demonstrate the potential of integrating neural networks with traditional machine learning models, paving the way for innovative blood pressure estimation methods. The potential applications of these models in real-world healthcare settings, such as clinics and telemonitoring systems, mark a promising shift in the early detection and management of high blood pressure and related conditions.

The choice of SVR, KELM, and XGB to predict blood pressure depends on the nature of the data and the strengths and weaknesses of each technique. SVR is suitable for small datasets and requires flexibility with kernel functions. KELM is ideal for fast learning and handling complex data using the kernel trick. XGB excels in high-accuracy predictions and managing large, complex datasets.

Analyzing the results in Table 4.8, SVR is the most appropriate for blood pressure estimation, having the lowest MAE and STD values. The SVR technique finds the hyperplane with the widest margin between the predicted and actual values, ensuring consistency with the available data. In contrast, KELM uses random parameters for a single hidden layer, and XGB's sequential correction in decision trees may not align perfectly with the test data, resulting in higher errors.

5.3 Future Works

Deep hybrid learning optimized by feature selection establishes a new benchmark for estimating blood pressure from PCG signals. Although these models demonstrate significant accuracy and reliability, there is always room for improving deep learning models. Future research should focus on optimizing neural network layers, experimenting with various activation functions, and improving the overall reliability of the models. Increasing the number of datasets of PCG signals with blood pressure values will also enhance model performance.

Expanding the research to include datasets beyond the initial database will help evaluate the model's performance across different blood pressure levels and population groups. The increasing demand for real-time blood pressure monitoring highlights the importance of implementing these models in practical settings. Integrating these models with advanced diagnostic tools such as medical computing could lead to a more comprehensive diagnostic platform for the detection and management of high blood pressure.

Model interpretability remains crucial, especially in healthcare, where decisions have significant impacts. Incorporating a variety of blood pressure-related conditions in future research will improve the applicability of these studies. Additionally, efforts should focus on making these models compatible with clinical devices and accessible in resource-limited settings. Prioritizing models that perform well on low-power devices is essential for real-world applications.

In the future, development may focus on collecting more diverse data, including expanding the range of blood pressure values to cover more levels. This approach aims to increase the learning efficiency of the model, enabling it to learn and estimate blood pressure values in various formats and levels more accurately.

5.4 Thesis Suggestions

From the test of estimating blood pressure with phonocardiogram data while measuring blood pressure, different signal characteristics were observed for each volunteer. Research studies and testing revealed that extracting salient features from phonocardiogram data and processing them through deep learning retains the distinctive characteristics of the signals. The resulting weight values are then utilized in machine learning for estimation purposes. To enhance the accuracy of this algorithm, phonocardiogram data can be further developed and expanded to facilitate learning from multiple datasets.

The researcher has organized a development plan to more efficiently estimate blood pressure from phonocardiograms as follows.

5.4.1 Increase the feature extraction process to obtain values that are consistent with blood pressure, retaining as many distinctive characteristics as possible while maintaining high efficiency.

5.4.2 Use high-performance medical stethoscopes for recording phonocardiograms. These recordings will be integrated into machine learning models derived from the extracted outstanding features.

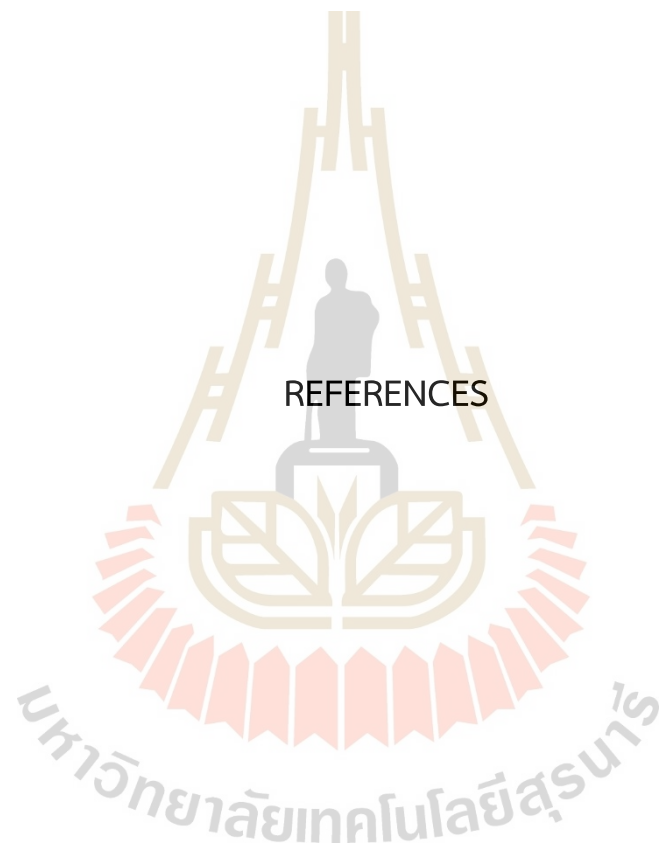
5.4.3 Develop deep integrated machine learning models for estimation that provide acceptable values based on testing.

5.4.4 Study the use of phonocardiogram datasets and recorded blood pressure values that were outside the test range to assess their ability to test other blood pressure levels.

This research involves developing a feature extraction algorithm and applying deep hybrid machine learning to estimate blood pressure from phonocardiograms

with an acceptable error range. Phonocardiograms were recorded from 140 volunteers, but in this test, only the phonocardiograms of volunteers with specific systolic and diastolic blood pressure values were used.





REFERENCES

- Sukonthasarn, A. et al. Thai Guidelines on The Treatment of Hypertension, Thailand, 2019. Accessed: Feb 14,2022. [Online]. Available: https://www.thaihypertension.org/files/GL%20HT%202015.pdf
- Lin, Y. H., Harfiya, L. N., Purwandari, K., & Lin, Y. D. "Real-time cuffless continuous blood pressure estimation using deep learning model," *Sensors*, vol. 20, no.19, pp. 5606, Sep. 2020.
- Omari, T. & Bereksi-Reguig, F. "A new approach for blood pressure estimation based on phonocardiogram," *Biomedical engineering letters*, vol. 9, pp. 395-406, Jun. 2019.
- Lee, S., Joshi, G. P., Shrestha, A. P., Son, C. H., & Lee, G. "Cuffless Blood Pressure Estimation with Confidence Intervals using Hybrid Feature Selection and Decision Based on Gaussian Process," *Applied Sciences*, vol. 13, no. 2, pp. 1221, Jan. 2023.
- Ümit, Ş., İbrahim, Y., & Kemal, P. "Repetitive neural network (RNN) based blood pressure estimation using PPG and ECG signals," in *2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) IEEE*, Ankara, Turkey, pp. 1-4, Oct. 19-21, 2018.
- Parry, F., Guy, D., Craig, R., Chris, M., & Mark, A. "Continuous noninvasive blood pressure measurement by pulse transit time," in *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society IEEE*, San Francisco, CA, USA, pp. 738-741, Sep. 01-05, 2004.
- Zhang, X. Y., & Zhang, Y. T. "Model-based analysis of effects of systolic blood pressure on frequency characteristics of the second phonocardiogram," in

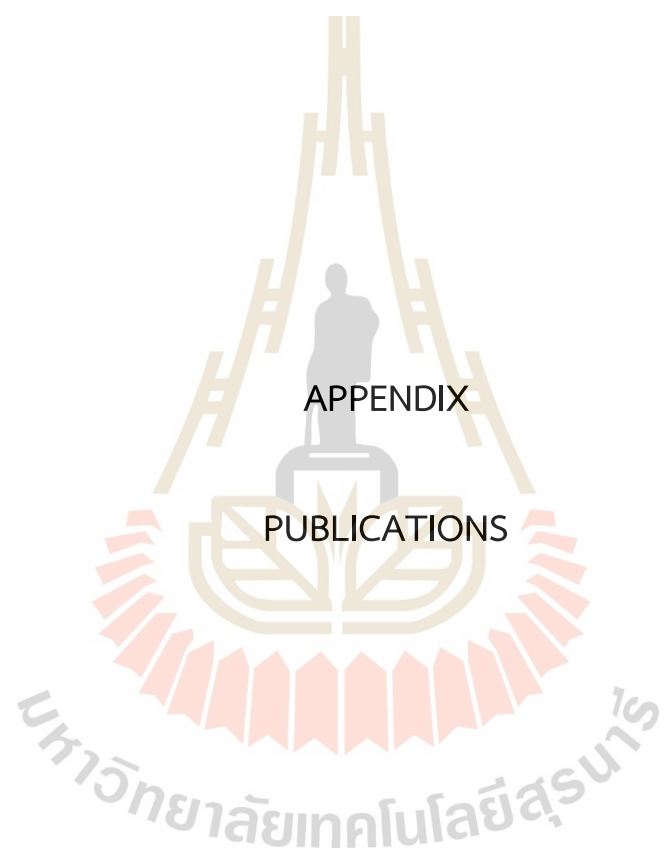
- 2006 International Conference of the IEEE Engineering in Medicine and Biology Society IEEE, New York, NY, USA, Aug. 30- Sep. 3, 2006, pp. 2888-2891.
- 8Marzorati, D., Bovio, D., Salito, C., Mainardi, L., & Cerveri, P. "Chest wearable apparatus for cuffless continuous blood pressure measurements based on PPG and PCG signals," *IEEE Access*, vol. 8, pp. 55424-55437, Mar. 2020.
- Nigam, V. & Priemer, R. "Simplicity based gating of phonocardiograms," in 48th Midwest Symposium on Circuits and Systems IEEE, Covington, KY, USA, Aug. 07-10, 2005, pp. 1298-1301.
- Israr, M., Zia, M., Ur Rehman, N., Ullah, I. & Khan, K. "Classification of normal and abnormal heart by classifying PCG signal using MFCC coefficients and CGP-ANN classifier," *Mehran University Research Journal Of Engineering and Technology*, vol. 42, no. 3, pp. 160-166, Jul. 2023.
- Yano, Y., Hoshide, S., Shimizu, M., Eguchi, K., Ishikawa, J. et al., "Association of home and ambulatory blood pressure changes with changes in cardiovascular biomarkers during antihypertensive treatment," *American journal of hypertension*, vol. 25, no. 3, pp. 306-312, Mar. 2012.
- Hsiao, C. C., Horng, J., Lee, R. G., & Lin, R. "Design and implementation of auscultation blood pressure measurement using vascular transit time and physiological parameters," in 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Banff, AB, Canada, Oct. 5-8, 2017, pp. 2996-3001.
- Martinez-Rios, E., Montesinos, L., Alfaro-Ponce, M., & Pecchia, L. "A review of machine learning in hypertension detection and blood pressure estimation based on clinical and physiological data," *Biomedical Signal Processing and Control*, vol. 68, pp. 102813, Jul. 2021.
- Demir, F., Abdullah, D. A., & Sengur, A. "A new deep CNNs model for environmental sound classification," *IEEE Access*, vol. 8, pp. 66529-66537, Apr. 2020.

- Lee, S., Lee, G., & Jeon, G. "Statistical approaches based on deep learning regression for verification of normality of blood pressure estimates," *Sensors*, vol. 19, no. 9, pp. 2137, May. 2019.
- Aizenberg, I., & Aizenberg, I. "Multilayer feedforward neural network based on multi-valued neurons (MLMVN)", *Complex-Valued Neural Networks with Multi-Valued Neurons*, vol. 353, pp. 133-172, 2011.
- Dastjerdi, A. E., Kachuee, M., & Shabany, M. "Non-invasive blood pressure estimation using phonocardiogram," in 2017 IEEE international symposium on circuits and systems (ISCAS), Baltimore, MD, USA, May. 28-31, 2017, pp. 1-4.
- Haque, C. A., Kwon, T. H., & Kim, K. D. "Cuffless blood pressure estimation based on Monte Carlo simulation using photoplethysmography signals," *Sensors*, vol. 22, no. 3, pp. 1175, Jan. 2022.
- Rastegar, S., Gholam Hosseini, H., & Lowe, A. "Hybrid CNNs-SVR Blood Pressure Estimation Model Using ECG and PPG Signals," *Sensors*, vol. 23, no. 3, pp. 1259, Jan. 2023.
- Omari, T., & Bereksi-Reguig, F. "An automatic wavelet denoising scheme for phonocardiograms. *International Journal of Wavelets*", *Multiresolution and Information Processing*, vol. 13, no.03, pp. 1550016, Apr. 2015.
- Obrist, P. A., Light, K. C., McCubbin, J. A., Hutcherson, J. S., & Hoffer, J. L. "Pulse transit time: Relationship to blood pressure," *Behavior Research Methods and Instrumentation*, vol. 10, no.5, pp. 623-626, Sep. 1978.
- Rastegar, S., GholamHosseini, H., & Lowe, A. "Non-invasive continuous blood pressure monitoring systems: current and proposed technology issues and challenges," *Physical and Engineering Sciences in Medicine*, vol. 43, pp. 11-28, Nov. 2022.
- Renna, F., Oliveira, J., & Coimbra, M. T. "Deep convolutional neural networks for phonocardiogram segmentation," *IEEE journal of biomedical and health informatics*, vol. 23, no.6, pp. 2435-2445, Jan. 2019.

- Kiranyaz, S. et al. "Real-time phonocardiogram anomaly detection by adaptive 1D convolutional neural networks," *Neurocomputing*, vol. 411, pp. 291-301, Oct. 2020.
- Chen, J., Sun, S., Zhang, L. B., Yang, B., & Wang, W. "Compressed sensing framework for phonocardiogram acquisition in internet of medical things," *IEEE transactions on industrial informatics*, vol. 18, no.3, 2000-2009, Jun. 2021.
- Esmaelpoor, J., Moradi, M. H., & Kadkhodamohammadi, A. "A multistage deep neural network model for blood pressure estimation using photoplethysmogram signals," *Computers in Biology and Medicine*, vol. 120, pp. 103719, May. 2020.
- Ali, N. F., & Atef, M. "LSTM Multi-Stage Transfer Learning for Blood Pressure Estimation Using Photoplethysmography," *Electronics*, vol. 11, no.22, pp.3749, Nov. 2022.
- Wongsathon, P. et al. "RS-MSConvNet: A Novel End-to-End Pathological Voice Detection Model," *IEEE Access*, vol. 10, pp. 120450-120461, Nov. 2022.
- Panwar, M., Gautam, A., Biswas, D., & Acharyya, A. "PP-Net: A deep learning framework for PPG-based blood pressure and heart rate estimation," *IEEE Sensors Journal*, vol. 20, no.17, pp. 10000-10011, Apr. 2020.
- Jiang, H., Zou, L., Huang, D., & Feng, Q. "Continuous Blood Pressure Estimation Based on Multi-Scale Feature Extraction by the Neural Network With Multi-Task Learning," *Frontiers in Neuroscience*, vol. 16, pp. 883693, May. 2022.
- Le, T., Ellington, F., Lee, T. Y., Vo, K., Khine, M., Krishnan, S. K., Nikil, D., & Cao, H. "Continuous non-invasive blood pressure monitoring: a methodological review on measurement techniques," *IEEE Access*, vol. 8, pp. 212478-212498, Nov. 2020.
- Hoang, T., Montanari, A., & Fahim, K. "Non-invasive blood pressure monitoring with multi-modal in-ear sensing." in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, Singapore, Singapore, May. 23-27, 2022, pp. 6-10.

Esmaili, A., Kachuee, M. & Shabany, M. "Nonlinear Cuffless Blood Pressure Estimation of Healthy Subjects Using Pulse Transit Time and Arrival Time," IEEE Transactions on Instrumentation and Measurement, vol. 66,





List of Publications

International Journal Paper

Kokkhunthod, K., Phapatanaburi, K., Pathonsuwan, W., Jumphoo, T., Anchuen, P. et al. (2024). Blood pressure estimation with phonocardiogram on cnn-based approach. *Computers, Materials & Continua*, 79(2), 1775-1794.





ARTICLE

Blood Pressure Estimation with Phonocardiogram on CNN-Based Approach

Kasidit Kokkhunthod¹, Khomdet Phapatanaburi², Wongsathon Pathonsuwan¹, Talit Jumphoo¹, Patikorn Anchuen³, Pornnip Nimbuntod⁴, Monthippa Uthansakul¹ and Peerapong Uthansakul^{1,*}

¹School of Telecommunication Engineering, Suranaree University of Technology, Nakhon Ratchasima, 30000, Thailand

²Telecommunications Engineering, Rajamangala University of Technology Isan, Nakhon Ratchasima, 30000, Thailand

³Navaminda Kasatriyadhiraj Royal Air Force Academy, Saraburi, 18180, Thailand

⁴Institute of Medicine, Suranaree University of Technology Hospital, Nakhon Ratchasima, 30000, Thailand

*Corresponding Author: Peerapong Uthansakul. Email: uthansakul@sut.ac.th

Received: 02 January 2024 Accepted: 21 March 2024

ABSTRACT

Monitoring blood pressure is a critical aspect of safeguarding an individual's health, as early detection of abnormal blood pressure levels facilitates timely medical intervention, ultimately leading to a reduction in mortality rates associated with cardiovascular diseases. Consequently, the development of a robust and continuous blood pressure monitoring system holds paramount significance. In the context of this research paper, we introduce an innovative deep learning regression model that harnesses phonocardiogram (PCG) data to achieve precise blood pressure estimation. Our novel approach incorporates a convolutional neural network (CNN)-based regression model, which not only enhances its adaptability to spatial variations but also empowers it to capture intricate patterns within the PCG signals. These advancements contribute significantly to the overall accuracy of blood pressure estimation. To substantiate the effectiveness of our proposed method, we meticulously gathered PCG signal data from 78 volunteers, adhering to the ethical guidelines of Suranaree University of Technology (Human Research Ethics number EC-65-78). Subsequently, we rigorously preprocessed the dataset to ensure its integrity. We further employed a K-fold cross-validation procedure for data division and alignment, combining the resulting datasets with a CNN for blood pressure estimation. The experimental results are highly promising, yielding a Mean Absolute Error (MAE) and standard deviation (STD) of approximately 10.69 ± 7.23 mmHg for systolic pressure and 6.89 ± 5.22 mmHg for diastolic pressure. Our study underscores the potential for precise blood pressure estimation, particularly using PCG signals, paving the way for a practical, non-invasive method with broad applicability in the healthcare domain. Early detection of abnormal blood pressure levels can facilitate timely medical interventions, ultimately reducing cardiovascular disease-related mortality rates.

KEYWORDS

Blood pressure; phonocardiogram; CNN-based; deep learning

1 Introduction

Blood pressure (BP) is a crucial vital sign in the human body, serving as a significant risk indicator for serious health conditions, notably cardiovascular diseases (CVD) and hypertension



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.



ARTICLE

Blood Pressure Estimation with Phonocardiogram on CNN-Based Approach

Kasidit Kokkhunthod¹, Khomdet Phapatanaburi², Wongsathon Pathonsuwan¹, Talit Jumphoo¹, Patikorn Anchuen³, Pornnip Nimmkuntod⁴, Monthippa Uthansakul¹ and Peerapong Uthansakul^{1,*}

¹School of Telecommunication Engineering, Suranaree University of Technology, Nakhon Ratchasima, 30000, Thailand

²Telecommunications Engineering, Rajamangala University of Technology Isan, Nakhon Ratchasima, 30000, Thailand

³Navaminda Kasatriyadhiraj Royal Air Force Academy, Saraburi, 18180, Thailand

⁴Institute of Medicine, Suranaree University of Technology Hospital, Nakhon Ratchasima, 30000, Thailand

*Corresponding Author: Peerapong Uthansakul. Email: uthansakul@sut.ac.th

Received: 02 January 2024 Accepted: 21 March 2024

ABSTRACT

Monitoring blood pressure is a critical aspect of safeguarding an individual's health, as early detection of abnormal blood pressure levels facilitates timely medical intervention, ultimately leading to a reduction in mortality rates associated with cardiovascular diseases. Consequently, the development of a robust and continuous blood pressure monitoring system holds paramount significance. In the context of this research paper, we introduce an innovative deep learning regression model that harnesses phonocardiogram (PCG) data to achieve precise blood pressure estimation. Our novel approach incorporates a convolutional neural network (CNN)-based regression model, which not only enhances its adaptability to spatial variations but also empowers it to capture intricate patterns within the PCG signals. These advancements contribute significantly to the overall accuracy of blood pressure estimation. To substantiate the effectiveness of our proposed method, we meticulously gathered PCG signal data from 78 volunteers, adhering to the ethical guidelines of Suranaree University of Technology (Human Research Ethics number EC-65-78). Subsequently, we rigorously preprocessed the dataset to ensure its integrity. We further employed a K-fold cross-validation procedure for data division and alignment, combining the resulting datasets with a CNN for blood pressure estimation. The experimental results are highly promising, yielding a Mean Absolute Error (MAE) and standard deviation (STD) of approximately 10.69 ± 7.23 mmHg for systolic pressure and 6.89 ± 5.22 mmHg for diastolic pressure. Our study underscores the potential for precise blood pressure estimation, particularly using PCG signals, paving the way for a practical, non-invasive method with broad applicability in the healthcare domain. Early detection of abnormal blood pressure levels can facilitate timely medical interventions, ultimately reducing cardiovascular disease-related mortality rates.

KEYWORDS

Blood pressure; phonocardiogram; CNN-based; deep learning

1 Introduction

Blood pressure (BP) is a crucial vital sign in the human body, serving as a significant risk indicator for serious health conditions, notably cardiovascular diseases (CVD) and hypertension



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

in clinical environments, heralding significant progress in the precision and dependability of non-invasive blood pressure monitoring methods. The introduction of this method stands to revolutionize the management and preliminary detection of coronary artery disease and hypertension.

2 Materials and Methods

2.1 Dataset

For this study, a specialized database was diligently compiled, focusing exclusively on phonocardiogram signals to ensure the robustness and reliability of the proposed blood pressure estimation models [11,25]. The database encompasses PCG signal data from a diverse cohort of participants, spanning various ages, genders, and health conditions. This extensive diversity is pivotal for affirming the effectiveness and reliability of the proposed models across a wide array of individuals, ensuring the models' broad applicability and generalizability in real-world settings. Fig. 1 is the sample of the PCG signal.

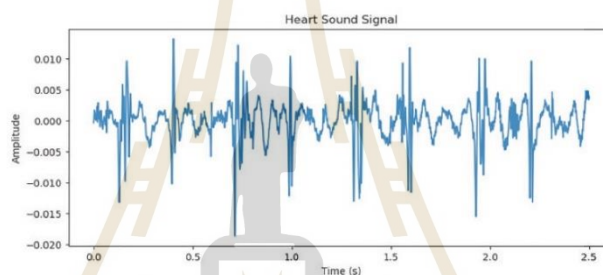


Figure 1: Simultaneous presentation physiological signals PCG of the arterial pressure

Data collection:

The data collection was carried out under strictly controlled conditions to minimize the impact of external variables on the PCG signals and the corresponding BP readings [11]. Participants were comfortably seated in a calm, temperature-regulated environment to ensure consistent and accurate PCG signal recordings. The dataset captured signals within the frequency range of 20–2000 Hz. Sensors were positioned for optimal PCG signal reception on participants. Efforts were made to minimize distractions and noise. We meticulously collected PCG signal data from 78 volunteers, which was considered and approved by the Human Research Ethics Committee Suranaree University of Technology. The research was conducted in accordance with the ethical principles for human research in accordance with the Declaration of Helsinki. Received research ethics certificate number Suranaree University of Technology. The procedure for collecting PCG signal data, along with blood pressure values, is outlined in Fig. 2.

1. Participants sat comfortably in a calm, temperature-controlled environment for approximately 5 min to ensure consistent and accurate recording of the PCG signal.
2. Interview basic important information from volunteers, including age, weight, height, and gender.
3. Prepare the automatic blood pressure monitor. Ready to use a pressure measuring strap (Cuff) to wrap around the upper arm approximately 1–2 inches above the elbow and tie it tightly.

4. Prepare the digital stethoscope equipment to record signals, and set the signal file name to match the information of each volunteer.
5. Place the digital stethoscope in the position specified by the attending physician. After that, start the automatic blood pressure monitor and record the PCG signal.

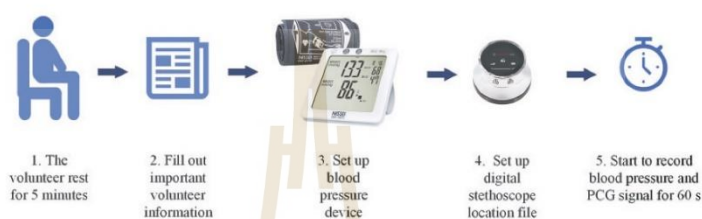


Figure 2: Sequence of recording PCG signal from volunteer

Equipment for data collection:

Blood pressure monitor: In this research, an automatic blood pressure recording device named Blood Pressure Monitor NISSEI DSK-1031 was used.

Digital stethoscope: The ThinkLabs One Digital Stethoscope was employed, featuring a frequency range of 20–2000 Hz.

Fig. 3 shows the new dataset being tested in a hospital. The physician is responsible for overseeing the collection of data from each volunteer. The test was conducted under human research ethics number EC-65-78.



Figure 3: Simultaneous recording of PCG signals and measurement of blood pressure take place concurrently

2.2 PCG Signal Processing

In the preparation of the database for model development and evaluation, a comprehensive data pre-processing and PCG signal processing phase was meticulously conducted. This multifaceted process entailed the meticulous removal of noise and artifacts from the PCG signals, ensuring the presence of pristine and high fidelity signals ideal for feature extraction and subsequent model training [3,24]. Simultaneously, the associated blood pressure readings underwent rigorous verification and validation procedures to validate their accuracy and reliability. Within the PCG signal processing framework, a rigorous pre-processing routine was applied to enhance signal quality, encompassing noise and artifact removal, amplitude normalization for standardization, and segmentation to partition the continuous PCG signals into manageable segments, facilitating further in-depth analysis [5,10].

2.3 Blood Pressure Estimation Methods

2.3.1 1D CNN-Based Method

In the Traditional CNN-based Method, the features from the PCG signals are used as the input to a Convolutional Neural Network for blood pressure estimation [24]. To adapt CNNs for time-dependent applications, modifications are made to exploit their ability to learn hierarchical patterns, now from sequences observed over time. By applies a 1D convolutional neural networks over an input signal composed of several input planes. Such adaptations make CNNs suitable for a wide range of time series, which following in the Eqs. (1) and (2). Let X be the matrix of PCG signals features, the CNN maps these features to an estimated blood pressure using a function by the weights θ of the network:

$$y[t] = \sum_{i=0}^{k-1} x[t+i] \cdot w[i] + b \quad (1)$$

where

$y[t]$ is the value of the output at time step t .

$x[t+i]$ represents the input values at time steps $t, t+1, t+2, \dots, t+k-1$.

$w[i]$ represents the convolutional kernel weights at index i .

b is the bias term.

$$L_{out} = \left[\frac{L_m + 2(p) - 1(k-1) - 1}{s} + 1 \right] \quad (2)$$

L_{out} is the length of the output sequence.

L_m is the length of the input sequence.

p is the amount of padding added to each side of the input sequence.

k is size of the convolving kernel.

s is stride of the convolution.

The network comprises multiple convolutional layers followed by pooling layers, fully connected layers, and a regression output layer. The convolutional layers apply a set of learnable filters to the input features. Fig. 4 shows a diagram illustrating the envisioned Conv1D model structure, from input into the solving Eq. (2). The overall approach employs a 5-layer CNN and depicts the constituent elements of each CNN layer [30].

6

CMC, 2024

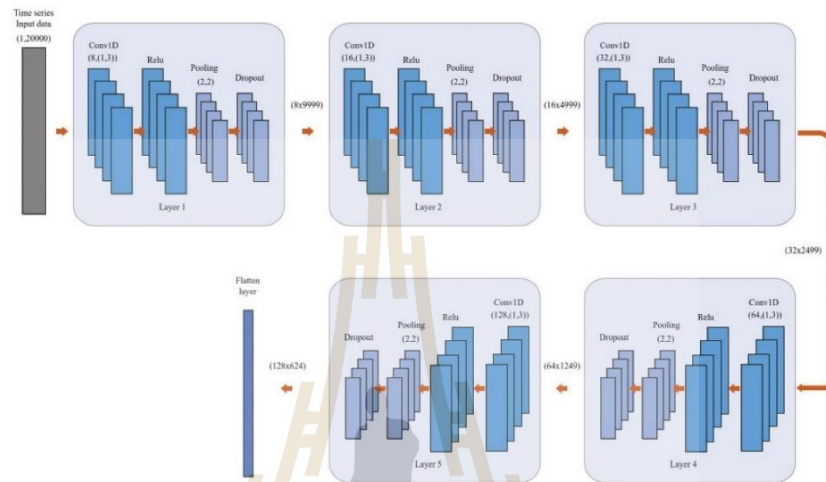


Figure 4: A schematic representation of the proposed Conv1D model architecture. Overall method using 5-layer CNN and shown the component of CNN layer

3 Experimental Setup

3.1 Design Model Training

3.1.1 Dataset Split

In this study, a 3-fold cross-validation approach is employed to ensure the robust evaluation of the models. The entire dataset, encompassing PCG signals and corresponding blood pressure readings, is evenly divided into three folds. In each iteration of the validation process, two folds (approximately 67% of the data) are used for training the model, and the remaining one-fold (approximately 33% of the data) is used for validation [31]. This process is repeated three times, with each fold used for validation exactly once. After the completion of the 3-fold cross-validation, the average performance metrics from each fold are computed to obtain a comprehensive and reliable assessment of the models' performance. This approach ensures that every data point is used for both training and validation, enhancing the reliability of the model evaluation and ensuring the models' effectiveness and generalizability across diverse data samples. For final testing and evaluation, a separate test set, which was not involved in the 3-fold cross-validation, is used. This test set in each iteration comprises 1 fold of the total 3 fold shown in Fig. 5 and 2 fold using for training. K-fold cross-validation is utilized to assess the models' performance on unseen data, providing insights into their real-world applicability and effectiveness in blood pressure estimation from PCG signals. In each iteration, the test performance of each fold after iteration. Result will be taken to the next average performance.



Figure 5: Illustration of the K-fold cross-validation

3.1.2 Data Augmentation

To enhance the balance of the models and mitigate overfitting, data augmentation techniques are employed. These techniques include adding white Gaussian noise, time shifting, and time stretching to the PCG signals. This augmentation enriches the diversity of the training data, enabling the models to learn more generalized and robust features [28].

Additive white Gaussian noise (AWGN): In this technique, white Gaussian noise is introduced into the PCG signals. This noise, following a Gaussian distribution, simulates real-world variations and adds a level of randomness to the signals. By doing so, the models become more resilient to noise in actual data, leading to improved generalization and reduced overfitting.

We have added the adding white Gaussian noise as one of the data augmentation techniques to the text. This technique helps to further diversify the training data and improve the model's robustness.

3.1.3 Model Architecture

Architecture the CNN-based models are constructed with multiple layers, including convolutional layers, pooling layers, and fully connected layers. The CNN-based model incorporates convolutional layers that adaptively adjust the sampling locations in the feature map in Fig. 6, enhancing the model's ability to capture complex patterns in the PCG signals.

3.1.4 Loss Function

The Mean Squared Error (MSE) loss function is used to train the models. It calculates the average squared differences between the estimated and actual blood pressure values, guiding the models to minimize this error during the training process [28,31].

$$MSE = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2 \quad (3)$$

where (x_i) and (\hat{x}_i) are the actual and estimated blood pressure values, respectively, and n is the number of observations. The Mean Squared Error (MSE) loss function is used to measure the average squared difference between predicted values and actual values in a regression task.

The MSE loss measures the discrepancy between predicted values (\hat{x}_i) and actual values (x_i) for each data point in the dataset. The goal is to minimize this loss during the training process to make predicted values as close as possible to the ground truth values.

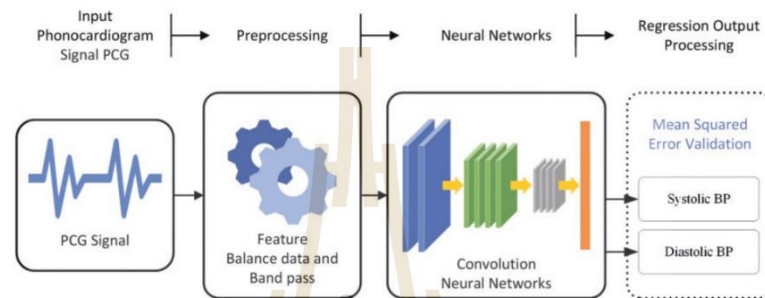


Figure 6: A block diagram depicting the overall system of estimate BP from the PCG signal

3.1.5 Optimizer

The Adam optimizer is employed for training the models. It adaptively adjusts the learning rate during training, providing efficient and effective optimization.

3.1.6 Training Process

In the training process, machine learning or deep learning models are iteratively optimized to achieve optimal performance on a given task. This section outlines the key aspects of the training process.

1. Epoch-Based Training:

The training process is organized into epochs, where each epoch represents a complete pass through the entire training dataset. This approach allows the model to iteratively learn and improve from the data. The training process will repeat 2000 times, each time adjusting the model's parameters to reduce the error in predictions. In Fig. 7, through experimentation, we observed that our model's loss function started stabilized around 1750 epochs, with marginal improvements thereafter. Extending the training to 2000 epochs allowed us to confirm that the model reached in learning, and for training with the newly PCG dataset, thereby ensuring that we had fully captured the learning potential of the newly data without prematurely halting the process.

2. Optimize Model Performance:

The Adam optimizer is used to adjust and improve model parameters based on the calculated gradients. Adam is chosen for its effectiveness in handling varying data scales and its adaptive approach to adjusting the learning rate, which is essential for converging to optimal solutions efficiently.

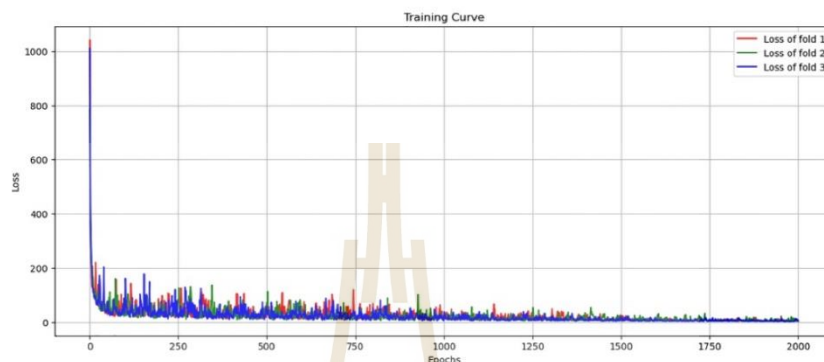


Figure 7: Training loss curve across 3-fold cross-validation

3. Loss Function:

The Mean Squared Error Loss (MSE) function is used to measure the difference between the model's predictions and the actual values. MSE is a common choice for regression problems as it penalizes larger errors more severely, encouraging the model to focus on reducing these larger discrepancies.

3.1.7 Hyperparameter Tuning

Various hyperparameters including learning rate and batch size. This process involved employing a systematic method to select the most effective conditions based on their performance outcomes in initial experiments. These conditions were then iteratively applied to refine subsequent experiments. To explore the conditions of various parameters systematically and identify the configuration that provides the best model performance from this new recording dataset.

1. CNN layer and pooling: The design of our CNN model's architecture, specifically the configuration of its convolutional and pooling layers, is informed by the guidelines established in reference [28]. Initially, the model is structured to include three convolutional layers, a foundational setup aimed at capturing a broad range of features from the input data. Following each convolutional layer, a max pooling layer is integrated, serving as a critical component for reducing dimensionality, enhancing computational efficiency, and ensuring the model's focus on the most salient features. This initial configuration, chosen as our test's starting condition.
2. Learning rate: The learning rate is a critical hyperparameter in the training of neural networks. In this section, the learning rates to be tuned are 0.01, 0.001, and 0.0001. The learning rate influences how much the model's weights are adjusted during training. A higher rate can lead to faster convergence but might overshoot the minimum, while a lower rate ensures more precise updates but can slow down the training process. The goal is to find an optimal learning rate that balances convergence speed and accuracy.
3. Batch size: Batch size is another key hyperparameter, and the sizes being experimented with are 256, 128, 64, 32, and 16 [30]. The batch size determines the number of training examples used in one iteration of model updates. Larger batch sizes offer more stable gradient estimates but

require more memory and computational power. Conversely, smaller batch sizes can lead to faster convergence and can also introduce beneficial noise, aiding generalization. The ideal batch size would efficiently utilize computational resources while maintaining good model performance.

3.1.8 Model Evaluation

Upon completion of training, the models are evaluated on the test set to assess their performance using the metrics outlined in the Performance Criteria section. The results from this evaluation provide insights into the models' effectiveness in blood pressure estimation from PCG signals and inform potential refinements and enhancements for further model development.

3.2 Performance Criteria

In evaluating the performance of the proposed CNN-based learning regression model for blood pressure estimation, several key metrics are employed to ensure a comprehensive assessment. These metrics provide a quantitative measure of the model's accuracy, reliability, and overall effectiveness in estimating blood pressure from PCG signals. Below are the detailed explanations and equations for each performance criterion used in this study:

- Mean Absolute Error (MAE): This metric calculates the average absolute errors between the estimated and the actual blood pressure values. It provides a straightforward measure of the model's estimation error. A lower MAE value indicates better performance of the model.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (4)$$

where (y_i) and (\hat{y}_i) are the actual and estimated blood pressure values, respectively, and n is the number of observations.

- Standard Deviation (STD): The standard deviation of the errors measures the dispersion of the errors from the mean error. A lower STD indicates that the errors are closely clustered around the mean, suggesting consistent model performance.

$$\text{STD} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (5)$$

Each of these performance criteria plays a crucial role in thoroughly evaluating the effectiveness of the proposed model, ensuring its reliability and accuracy in real-world blood pressure estimation tasks. The comprehensive assessment allows for the identification and addressing of any potential limitations or areas of improvement, contributing to the continual enhancement of the model's performance in blood pressure estimation from PCG signals.

4 Results and Discussion

4.1 Results Based on Our Proposed Method

The model proposed a Mean Absolute Error (MAE) of systolic and diastolic pressure estimation. These results are expected to approach the acceptable margin of error within the Association for the

Advancement of Medical Instrumentation (AAMI) standard. This emphasizes the clinical applicability of the mode.

4.1.1 Comparison with Method

In this subsection, we compare the performance of our proposed method with that of several well-known systems. As mentioned in the introduction section, some systems may not be included due to differences in the experimental models employed in database-based approaches. Here, we present a comparison of experimental results to align with the conditions of our experiment. As a result, we include the results of select known systems for the purpose of adjusting continuous testing and comparing them with the methods we have tested. [Table 1](#) presents the overall performance, including the Mean Absolute Error (MAE) and Standard Deviation (STD), for both the training and testing phases of each network. It can be observed that the CNN with three CNN layers and three hidden layers has a linear layer for test the new dataset. This network utilizes the PCG signals recorded from 78 volunteers in the dataset.

Table 1: Performance of PCG signals data with CNN model

MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
32.46	19.81	24.97	15.39	26.13

The MAE SBP value of 32.46 mmHg represents the average systolic blood pressure in the STD dataset. An SBP of 24.97 mmHg indicates a degree of variability or dispersion of SBP values. A higher standard deviation indicates a measurement range. A broader SBP MAE DBP value of 19.81 mmHg represents the average diastolic blood pressure in the STD dataset. DBP of 15.39 mmHg represents the variability in DBP measurements. MBP of 26.13 mmHg represents the average of both SBP and DBP. And provide insight into overall blood pressure. In summary [Table 1](#), it is a test of this dataset when imported into the estimation process with the CNN technique to see if the newly recorded data can produce a preliminary estimation or not. As shown in the table, Using this new dataset can be used in the basic CNN system, but the tolerances of the test results are not acceptable. The next step was to further improve the test by rebalancing the dataset to balance the values that could be estimated, and increase the chances of the machine learning to learn from an increased dataset

In our examination of the newly acquired PCG signal data, we identified significant fragmentation with an underrepresentation of higher blood pressure readings. To rectify this imbalance and improve the dataset's efficacy for model training, we implemented Additive White Gaussian Noise (AWGN) as a deliberate method to enrich the dataset with a wider spectrum of blood pressure readings. This enrichment strategy was aimed at bolstering the dataset with additional data points across a broader blood pressure range, thereby facilitating a more comprehensive model training.

The impact of this data augmentation is graphically depicted in [Fig. 8](#). Here, kernel density estimates provide a stark comparison between the distribution of the unmodified dataset and that which has been enhanced with AWGN. The density plots demonstrate how the application of AWGN successfully expands the representation of higher blood pressure values, mitigating the initial disproportionality and fostering a more balanced dataset for the development of our model. [Table 2](#) shows the MAE with AWGN method.

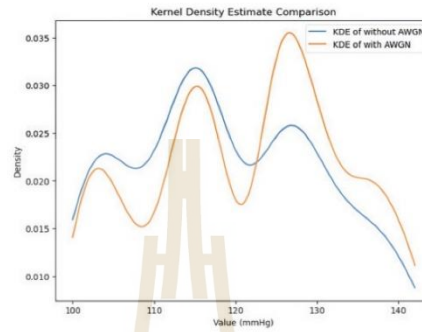


Figure 8: Kernel density estimates for the distribution of blood pressure data comparing without AWGN to those with AWGN

Table 2: Performance of PCG signals with additive white gaussian noise (AWGN)

Model	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
1D CNN w/o AWGN	32.46	19.81	24.97	15.39	26.13
1D CNN with AWGN	15.46	9.53	11.32	7.76	12.49

The presence of AWGN clearly has a significant effect on blood pressure estimation when comparing the two rows. Table 2 shows that when testing the same model, The finding that the MAE was much lower when the data were balanced across different blood pressure values. This is due to the assumption that if there is an increase in the distribution of information in training, It will make learning work better because there are a number of datasets that are known along with the values used in training. When testing the results, it was found that adding a more balanced number of trainings sessions to the dataset resulted in weaker estimation results. To evaluate the CNN model design, we performed tests and compared the results. Specifically, investigating the variation in extracting PCG signal features under different numbers of CNN layers, the results are shown in Table 3.

Table 3: Performance of numbers CNN block layer

Number of CNN block	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
1	147.66	74.81	102.99	49.47	111.24
2	15.90	10.53	13.66	8.29	13.22
3	15.46	9.53	11.32	7.76	12.49
4	14.21	10.03	10.35	6.59	12.12

(Continued)

Table 3 (continued)

Number of CNN block	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
5	12.68	8.31	8.27	5.86	10.50
6	14.27	9.92	10.88	7.66	12.09
7	13.84	9.63	10.31	7.43	11.73

The results of the experimental table where different modifications of the CNN layers are presented and compared. Among these modifications, the experiment yielded significantly low error values. It had a pressure error of 12.68 mmHg for the upper pressure and 8.31 mmHg for the lower pressure. As shown in Table 3, the results in Table 3 reveal a trend in the performance of a CNN model with varying numbers of convolutional blocks, as measured by MAE for SBP, DBP blood pressure, and their STD, along with MBP. A notable observation is the decrease in MAE for both SBP and DBP as the number of CNN blocks increases from 1 to 5, indicating improved model accuracy. However, this improvement plateaus beyond five layers, suggesting an optimal number of layers for this specific application. The standard deviations also follow a similar trend, decreasing with an increasing number of layers, indicating more consistent model predictions. This analysis reflects the efficiency of deep learning models in capturing complex patterns in data, with a balance between model complexity and performance. From the experimental results in the Table 3, there are 7 layers of CNN presented, because in the test adding layers at 8, 9, 10, the results obtained from the estimation were under-fitting of estimation. And this continued to happen as more layers of the CNN were added, allowing us to continue testing other types of parameterizations.

The next phase involves examining Table 4, which is to analyze the impact of different numbers of linear layers, following the model with 5 CNN layers. This investigation aims to assess how various linear layer configurations influence the model's precision and reliability in blood pressure measurement. Such an analysis is crucial for determining the most effective number and configuration of linear layers to optimize the model's performance for the specific application at hand.

Table 4: Performance of number of linear layers

Number of classifier layer	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
1	19.05	74.73	13.08	8.58	46.89
2	18.77	10.18	12.40	7.22	14.48
3	12.68	8.31	8.27	5.86	10.50
4	12.50	8.80	9.25	6.20	10.65
5	14.66	10.27	10.37	7.57	12.46
6	11.68	7.88	7.44	6.11	9.78
7	12.18	8.81	8.15	5.68	10.49
8	15.05	9.94	10.63	7.09	12.49

In our analysis of [Table 4](#), we focused on the MAE performance of models with different numbers of linear layers for estimating blood pressure. Several noteworthy trends have emerged. The key factor under investigation is the number of linear layers within the model architecture.

Firstly, we observe a clear relationship between the number of linear layers and MAE values. As the number of linear layers increases from 1 to 8, a consistent impact is observed on both systolic blood pressure (SBP) and diastolic blood pressure (DBP) MAE. For SBP MAE, the highest value of 19.05 mmHg is associated with a model featuring 1 linear layer. However, as the number of linear layers progressively increases, SBP MAE consistently decreases. The most accurate SBP estimation, with an MAE of 11.68 mmHg, is achieved when employing models with 6 linear layers. A parallel trend is observed for DBP MAE, with the highest MAE of 74.73 mmHg linked to 1 linear layer, while the lowest DBP MAE of 7.88 mmHg is attained with 6 linear layers.

Secondly, the analysis extends to the STD of SBP and DBP MAE. These STD values exhibit a consistent pattern mirroring the behavior of MAE. As the number of linear layers increases, STD values decrease, reflecting a reduction in variability. Models with 1 linear layer produce the highest STD values, signifying greater variability in MAE. Conversely, models with 6 linear layers yield the lowest STD values, indicating reduced variability in MAE.

Lastly, the Mean Blood Pressure (MBP) values align with the trends observed for SBP and DBP MAE. The highest MBP MAE is associated with 1 linear layer, while the lowest MBP MAE is achieved when employing models with 6 linear layers.

In summary, the analysis highlights the important role of selecting the appropriate number of linear layers within the model architecture. For accurate blood pressure estimation, the model with 6 linear layers outperformed the other models. Consistently, showing lower MAE and lower STD in MAE, these findings highlight the importance of optimizing the number of linear layers to effectively increase the accuracy of blood pressure estimation. From the experimental results in [Table 4](#), there are 8 linear layers presented because in the test plus the 9th layer, the results obtained from the estimation caused underfitting to the estimation. And this continues to happen as more linear layers are added. This leads us to continue testing other types of parameterizations.

The next step involves testing by adjusting the values in specific learning areas. Scale the data to be fed into the learning section and compile the learning rate values in [Table 5](#).

Table 5: Performance MAE of batch sizes and learning rates

Batch size/Learning rates	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
256/0.01	11.68	7.88	7.44	6.11	9.78
256/0.001	11.43	7.97	7.79	6.37	9.70
256/0.0001	11.75	9.04	8.30	5.47	10.40
128/0.01	11.73	8.11	7.70	5.80	9.92
128/0.001	11.11	7.91	7.96	4.78	9.51
128/0.0001	12.22	8.87	8.11	5.39	10.55
64/0.01	10.26	7.81	7.95	4.99	9.04
64/0.001	12.43	9.03	9.22	6.02	10.73

(Continued)

Table 5 (continued)

Batch size/Learning rates	MAE SBP mmHg	MAE DBP mmHg	STD SBP mmHg	STD DBP mmHg	MBP mmHg
64/0.0001	12.69	8.84	8.67	6.28	10.77
32/0.01	12.56	8.22	8.84	5.59	10.39
32/0.001	11.77	8.75	7.85	5.53	10.26
32/0.0001	14.65	10.55	10.47	7.19	12.60
16/0.01	10.69	6.89	7.23	5.22	8.79
16/0.001	14.18	9.62	9.89	6.53	11.90
16/0.0001	16.12	11.50	11.11	7.85	13.81

The analysis of the results presented in Table 5, which investigates the MAE performance of models across different batch sizes and learning rates, offers valuable insights into the influence of these hyperparameters on the accuracy of blood pressure estimation.

Considering the impact of learning rate on MAE, the results show that learning rate selection plays an important role in accurate blood pressure estimation. A higher learning rate (0.01) tends to result in lower MAE values, indicating increased accuracy in estimating blood pressure. On the contrary, the lowest learning rate (0.0001) often leads to higher MAE values, highlighting the importance of an appropriate learning rate in achieving model convergence and accuracy. Table 5, which examines the MAE performance across different batch sizes and learning rates, reveals several trends. Smaller batch sizes, particularly 16, demonstrate lower MAE values in many cases, suggesting better model performance. This can be attributed to smaller batch sizes offering more frequent updates and potentially better generalization. The interaction between batch size and learning rate also appears significant; for instance, a batch size of 16 with a learning rate of 0.01 shows notably lower MAE values across most table, indicating an optimal combination for this specific model and dataset. This suggests that both the batch size and learning rate play crucial roles in model accuracy and should be carefully tuned to achieve optimal performance.

Table 5 is observed with a batch size of 16 and a learning rate of 0.01. This combination yields the lowest Mean Absolute Error (MAE) values across the board: SBP (10.69 mmHg), DBP (6.89 mmHg), STD SBP (7.23 mmHg), STD DBP (5.22 mmHg), and MBP (8.79 mmHg). This indicates that for this model and dataset, a smaller batch size combined with a moderately high learning rate optimizes performance, leading to more accurate blood pressure predictions with lower variability.

Fig. 9 shows is a scatter plot illustrating the correlation between blood pressure values predicted by our model and the actual blood pressure measurements. The plot showcases how closely our model's estimates align with the true values. This visual representation demonstrates the effectiveness of our proposed in estimating blood pressure.

4.1.2 Systolic Blood Pressure Estimation

To estimate systolic blood pressure model performed. It has an MAE 10.69 mmHg, Fig. 10a, and a standard deviation is 7.23 mmHg, indicating that the model estimates can provide an approximation to the real data. This indicates linear relationship between the estimated and actual systolic blood pressure values. These findings reinforce the model's feasibility for estimating systolic blood pressure.

This points to the potential usefulness of the PCG signal feature in this context, and it is clear from the picture that the estimated vs. actual values tend to be similar.

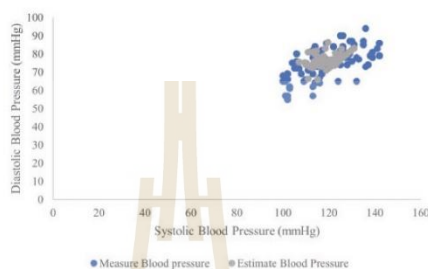


Figure 9: Scatter plot both actual and estimated blood pressure values by the proposed model

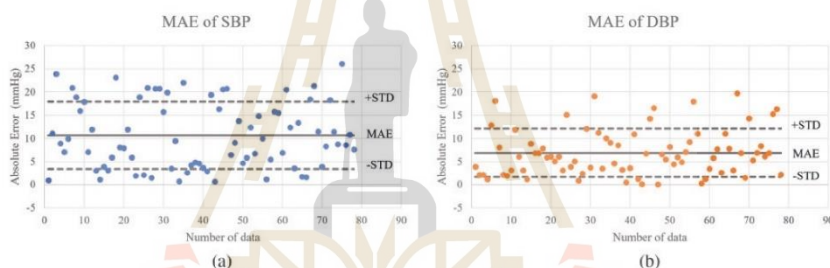


Figure 10: Scatter plot of the algorithm for (a) systolic BP estimation and (b) diastolic BP estimation

4.1.3 Diastolic Blood Pressure Estimation

In terms of diastolic blood pressure estimation, it comes to estimating diastolic blood pressure model performed. With a mean absolute error of 6.89 mmHg. This indicates a lower error rate compared to systolic blood pressure values. The standard deviation of the error was recorded as 5.22 mmHg. Fig. 10b visually shows that the model's estimates correspond closely to the real data. These findings also confirm the model's consistent performance in estimating systolic and diastolic blood pressure.

4.2 Comparison of the Results with the Related Works

Table 6 presents a comprehensive comparative analysis of the performance of PCG-based blood pressure estimation using the proposed CNN approach in this study, along with recent related works from the literature. The evaluation is based on key metrics, including Mean Absolute Error and Standard Deviation [3,32], for both systolic blood pressure and diastolic blood pressure. Table 6 presents a comprehensive comparison of CNN-based PCG methods for blood pressure estimation. This work is detailed alongside other contemporary studies such as the use of PCG signals with

traditional machine learning techniques and artificial neural networks. Work using dual signals, namely PCG and PPG for blood pressure estimation. This demonstrates the diversity of approaches within the field. This table includes the standard deviation and absolute mean error. Table 6 shows that the estimated SBP using the recommended technique is close to the measured value, as the MAE \pm STD is 10.69 ± 7.23 mmHg reported. The proposed approach yields similar results to the method. Given in [3] when it comes to estimating DBP, this result is of course higher but due to the new method used, namely CNN, and the simplification of data readout. Along with testing with new data collected under the supervision of a doctor.

Table 6: Comparison with other works

Work	Number of subjects	Signals	Index	SBP mmHg		DBP mmHg	
				MAE	STD	MAE	STD
This work	78	PCG	CNN	10.69	7.23	6.89	5.22
[3]	37	PCG	S21	6.48	4.48	3.91	2.58
[12]	85	PCG-PPG	VTT	6.67	8.47	–	–
[15]	24	PCG-PPG	PTT	7.47	11.08	3.56	4.53
[32]	10	PCG-PPG	VTT	4.07	3.07	5.61	4.09
[33]	32	PCG-PPG	PTT	6.22	9.44	3.97	5.15

Indeed, the MAE \pm STD was 10.69 ± 7.23 mmHg for the former and 3.91 ± 2.58 for the latter, respectively. Although the results obtained by the proposed method are globally very comparable to the results described in reference [3], the proposed method is much simpler because only one signal (PCG signal) is required to measure blood pressure.

5 Conclusion and Future Work

In this study, we introduced a novel method for storing phonocardiogram (PCG) signal data, collected from 78 volunteers at the Suranaree University of Technology Hospital. This method leverages a unique device and is subject to the Human Research Ethics approval number EC-65-78. Utilizing cutting-edge techniques such as convolutional neural networks (CNN), a form of deep learning, we demonstrated the potential of using a single PCG signal for blood pressure estimation.

Our approach centers on optimizing dataset adjustments and fine-tuning the parameters within the CNN's layers to capture the characteristics of the PCG signal. Analysis revealed a distinct correlation between the features of the PCG signal and blood pressure readings, underscoring the viability of this method for non-invasive blood pressure monitoring.

Despite these promising results, further research is necessary to expand the scope of our study, including the exploration of additional physical parameters and other measurement units that are compatible with our outpatient blood pressure measurement techniques. There is significant potential to enhance healthcare delivery through this methodology, as it enables direct blood pressure estimation from a single PCG signal without the need for expensive equipment. Moreover, the compatibility of PCG recordings with smartphone technology paves the way for integrating mobile devices to refine PCG signal analysis accuracy. Embracing this single-signal approach also presents new challenges in analysis, making it a compelling direction for future research.

Acknowledgement: We would like to express our sincere gratitude to Suranaree University of Technology Hospital for their invaluable support and guidance throughout the course of this new data collection. Special thanks to doctor in Cardiology Unit of Suranaree University of Technology Hospital, whose expertise and insights were instrumental in cardiology. We are also grateful to Suranaree University of Technology for providing the necessary resources and funding that greatly facilitated our work. Our appreciation extends to the team members, colleagues, and peers whose contributions. Finally, we thank our families and friends for their understanding and encouragement during the demanding phases of this endeavor.

Funding Statement: This work was supported by Suranaree University of Technology, Thailand Science Research and Innovation (TSRI), and National Science, Research, and Innovation Fund (NSRF) (NRIIS Number 179292).

Author Contributions: Study conception and design: K. Kokkhunthod, K. Phapatanaburi, W. Pathonsuwan, T. Jumphoo, P. Anchuen, P. Nimkuntod, M. Uthansakul, P. Uthansakul; data collection: K. Kokkhunthod, K. Phapatanaburi, W. Pathonsuwan, T. Jumphoo; analysis and interpretation of results: K. Kokkhunthod, K. Phapatanaburi, W. Pathonsuwan, T. Jumphoo, P. Anchuen, P. Nimkuntod, M. Uthansakul, P. Uthansakul; draft manuscript preparation: K. Kokkhunthod, K. Phapatanaburi, T. Jumphoo, P. Nimkuntod, M. Uthansakul, P. Uthansakul. Author reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data are available from the corresponding author upon reasonable request.

Ethics Approval: This research was conducted with human volunteers and received approval from the Human Research Ethics Committee Suranaree University of Technology. Adhering to the ethical principles outlined in the Declaration of Helsinki for human research, our study was granted the research ethics certificate number EC-65-78. All participating human volunteers provided informed consent.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] A. Sukonthasarn *et al.*, *Thai Guidelines on the Treatment of Hypertension*. Thailand, 2019. Accessed: Feb. 14, 2022. [Online]. Available: <https://www.thaihypertension.org/files/GL%20HT%202015.pdf>
- [2] Y. H. Lin, L. N. Harfiya, K. Purwandari, and Y. D. Lin, "Real-time cuffless continuous blood pressure estimation using deep learning model," *Sens.*, vol. 20, no. 19, pp. 5606, Sep. 2020. doi: 10.3390/s20195606.
- [3] T. Omari and F. Bereksi-Reguig, "A new approach for blood pressure estimation based on phonocardiogram," *Biomed. Eng. Lett.*, vol. 9, no. 3, pp. 395–406, Jun. 2019. doi: 10.1007/s13534-019-00113-z.
- [4] S. Lee, G. P. Joshi, A. P. Shrestha, C. H. Son, and G. Lee, "Cuffless blood pressure estimation with confidence intervals using hybrid feature selection and decision based on gaussian process," *Appl. Sci.*, vol. 13, no. 2, pp. 1221, Jan. 2023. doi: 10.3390/app13021221.
- [5] S. Ümit, Y. Ibrahim, and P. Kemal, "Repetitive neural network (RNN) based blood pressure estimation using PPG and ECG signals," in *2018 2nd Int. Symp. Multidis. Stud. Innov. Technol. (ISMSIT) IEEE*, Ankara, Turkey, Oct. 19–21, 2018, pp. 1–4.

- [6] F. Parry, D. Guy, R. Craig, M. Chris, and A. Mark, "Continuous noninvasive blood pressure measurement by pulse transit time," in *26th Annual Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE*, San Francisco, CA, USA, Sep. 1–5, 2004, pp. 738–741.
- [7] X. Y. Zhang and Y. T. Zhang, "Model-based analysis of effects of systolic blood pressure on frequency characteristics of the second heart sound," in *2006 Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE*, New York, NY, USA, Aug. 30–Sep. 3, 2006, pp. 2888–2891.
- [8] D. Marzorati, D. Bovio, C. Salito, L. Mainardi, and P. Cerveri, "Chest wearable apparatus for cuffless continuous blood pressure measurements based on PPG and PCG signals," *IEEE Access*, vol. 8, pp. 55424–55437, Mar. 2020. doi: [10.1109/ACCESS.2020.2981300](https://doi.org/10.1109/ACCESS.2020.2981300).
- [9] V. Nigam and R. Priemer, "Simplicity based gating of heart sounds," in *48th Midwest Symp. Circuit. Syst. IEEE*, Covington, KY, USA, Aug. 7–10, 2005, pp. 1298–1301.
- [10] M. Israr, M. Zia, N. Ur Rehman, I. Ullah, and K. Khan, "Classification of normal and abnormal heart by classifying PCG signal using MFCC coefficients and CGP-ANN classifier," *Mehran Univ. Res. J. Eng. Technol.*, vol. 42, no. 3, pp. 160–166, Jul. 2023. doi: [10.22581/muet1982.2303.16](https://doi.org/10.22581/muet1982.2303.16).
- [11] Y. Yano *et al.*, "Association of home and ambulatory blood pressure changes with changes in cardiovascular biomarkers during antihypertensive treatment," *American J. Hypertens.*, vol. 25, no. 3, pp. 306–312, Mar. 2012. doi: [10.1038/ajh.2011.229](https://doi.org/10.1038/ajh.2011.229).
- [12] C. C. Hsiao, J. Horng, R. G. Lee, and R. Lin, "Design and implementation of auscultation blood pressure measurement using vascular transit time and physiological parameters," in *2017 IEEE Int. Conf. Syst., Man, Cybernet. (SMC)*, Banff, AB, Canada, Oct. 5–8, 2017, pp. 2996–3001.
- [13] E. Martinez-Rios, L. Montesinos, M. Alfaro-Ponce, and L. Pecchia, "A review of machine learning in hypertension detection and blood pressure estimation based on clinical and physiological data," *Biomed. Signal Process.*, vol. 68, no. 2, pp. 102813, Jul. 2021. doi: [10.1016/j.bspc.2021.102813](https://doi.org/10.1016/j.bspc.2021.102813).
- [14] F. Demir, D. A. Abdullah, and A. Sengur, "A new deep CNN model for environmental sound classification," *IEEE Access*, vol. 8, pp. 66529–66537, Apr. 2020. doi: [10.1109/ACCESS.2020.2984903](https://doi.org/10.1109/ACCESS.2020.2984903).
- [15] S. Lee, G. Lee, and G. Jeon, "Statistical approaches based on deep learning regression for verification of normality of blood pressure estimates," *Sens.*, vol. 19, no. 9, pp. 2137, May 2019. doi: [10.3390/s19092137](https://doi.org/10.3390/s19092137).
- [16] I. Aizenberg and I. Aizenberg, "Multilayer feedforward neural network based on multi-valued neurons (MLMVN)," *Complex-Valued Neural Netw. Multi-Valued Neur.*, vol. 353, pp. 133–172, 2011. doi: [10.1007/978-3-642-20353-4_4](https://doi.org/10.1007/978-3-642-20353-4_4).
- [17] A. E. Dastjerdi, M. Kachuce, and M. Shabany, "Non-invasive blood pressure estimation using phonocardiogram," in *2017 IEEE Int. Symp. Circuit. Syst. (ISCAS)*, Baltimore, MD, USA, May 28–31, 2017, pp. 1–4.
- [18] C. A. Haque, T. H. Kwon, and K. D. Kim, "Cuffless blood pressure estimation based on Monte Carlo simulation using photoplethysmography signals," *Sens.*, vol. 22, no. 3, pp. 1175, Jan. 2022. doi: [10.3390/s22031175](https://doi.org/10.3390/s22031175).
- [19] S. Rastegar, H. Gholam Hosseini, and A. Lowe, "Hybrid CNN-SVR blood pressure estimation model using ECG and PPG signals," *Sens.*, vol. 23, no. 3, pp. 1259, Jan. 2023. doi: [10.3390/s23031259](https://doi.org/10.3390/s23031259).
- [20] T. Omari and F. Bereksi-Reguig, "An automatic wavelet denoising scheme for heart sounds," *Int. J. Wavelets Multiresolut Inf. Process.*, vol. 13, no. 3, pp. 1550016, Apr. 2015. doi: [10.1142/S0219691315500162](https://doi.org/10.1142/S0219691315500162).
- [21] P. A. Obrist, K. C. Light, J. A. McCubbin, J. S. Hutcheson, and J. L. Hoffer, "Pulse transit time: Relationship to blood pressure," *Behav. Res. Methods Inst.*, vol. 10, no. 5, pp. 623–626, Sep. 1978. doi: [10.3758/BF03205360](https://doi.org/10.3758/BF03205360).
- [22] S. Rastegar, H. Gholam Hosseini, and A. Lowe, "Non-invasive continuous blood pressure monitoring systems: Current and proposed technology issues and challenges," *Phys. Eng. Sci. Med.*, vol. 43, no. 1, pp. 11–28, Nov. 2020. doi: [10.1007/s13246-019-00813-x](https://doi.org/10.1007/s13246-019-00813-x).
- [23] F. Renna, J. Oliveira, and M. T. Coimbra, "Deep convolutional neural networks for heart sound segmentation," *IEEE J. Biomed. Health Inform.*, vol. 23, no. 6, pp. 2435–2445, Jan. 2019. doi: [10.1109/JBHI.2019.2894222](https://doi.org/10.1109/JBHI.2019.2894222).

- [24] S. Kiranyaz, M. Zabihi, A. B. Rad, T. Ince, R. Hamila and M. Gabbouj, "Real-time phonocardiogram anomaly detection by adaptive 1D convolutional neural networks," *Neurocomputing*, vol. 411, no. 12, pp. 291–301, Oct. 2020. doi: [10.1016/j.neucom.2020.05.063](https://doi.org/10.1016/j.neucom.2020.05.063).
- [25] J. Chen, S. Sun, L. B. Zhang, B. Yang, and W. Wang, "Compressed sensing framework for heart sound acquisition in internet of medical things," *IEEE Trans. Ind. Inf.*, vol. 18, no. 3, pp. 2000–2009, Jun. 2021. doi: [10.1109/TII.2021.3088465](https://doi.org/10.1109/TII.2021.3088465).
- [26] J. Esmaelpoor, M. H. Moradi, and A. Kadkhodamohammadi, "A multistage deep neural network model for blood pressure estimation using photoplethysmogram signals," *Comput. Biol. Med.*, vol. 120, pp. 103719, May 2020. doi: [10.1016/j.compbiomed.2020.103719](https://doi.org/10.1016/j.compbiomed.2020.103719).
- [27] N. F. Ali and M. Atef, "LSTM multi-stage transfer learning for blood pressure estimation using photoplethysmography," *Electronics*, vol. 11, no. 22, pp. 3749, Nov. 2022. doi: [10.3390/electronics11223749](https://doi.org/10.3390/electronics11223749).
- [28] P. Wongsathon *et al.*, "RS-MSCovNet: A novel end-to-end pathological voice detection model," *IEEE Access*, vol. 10, pp. 120450–120461, Nov. 2022. doi: [10.1109/ACCESS.2022.3219606](https://doi.org/10.1109/ACCESS.2022.3219606).
- [29] M. Panwar, A. Gautam, D. Biswas, and A. Acharyya, "PP-Net: A deep learning framework for PPG-based blood pressure and heart rate estimation," *IEEE Sensors J.*, vol. 20, no. 17, pp. 10000–10011, Apr. 2020. doi: [10.1109/JSEN.2020.2990864](https://doi.org/10.1109/JSEN.2020.2990864).
- [30] H. Jiang, L. Zou, D. Huang, and Q. Feng, "Continuous blood pressure estimation based on multi-scale feature extraction by the neural network with multi-task learning," *Front. Neurosci.*, vol. 16, pp. 883693, May. 2022. doi: [10.3389/fnins.2022.883693](https://doi.org/10.3389/fnins.2022.883693).
- [31] T. Le *et al.*, "Continuous non-invasive blood pressure monitoring: A methodological review on measurement techniques," *IEEE Access*, vol. 8, pp. 212478–212498, Nov. 2020. doi: [10.1109/ACCESS.2020.3040257](https://doi.org/10.1109/ACCESS.2020.3040257).
- [32] T. Hoang, A. Montanari, and K. Fahim, "Non-invasive blood pressure monitoring with multi-modal in-ear sensing," in *ICASSP 2022-2022 IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Singapore, IEEE, May 23–27, 2022, pp. 6–10.
- [33] A. Esmaili, M. Kachuee, and M. Shabany, "Nonlinear cuffless blood pressure estimation of healthy subjects using pulse transit time and arrival time," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 12, pp. 3299–3308, Sep. 2017. doi: [10.1109/TIM.2017.2745081](https://doi.org/10.1109/TIM.2017.2745081).



BIOGRAPHY

Kasidit Kokkhuntod was born on September 27, 1996. He began his studies and graduated with a bachelor's degree. Department of Electronics and Telecommunications Engineering Faculty of Engineering and Architecture Rajamangala University of Technology Isan, Nakhon Ratchasima Province, 2018. He graduated with a master's degree of Engineering in Telecommunication and Computer Engineering in 2021, respectively. He is pursuing his Ph.D. in telecommunication and computer engineering at Suranaree University of Technology, Nakhon Ratchasima, Thailand.

