Specialised Non-Invasive Blood Pressure Measurement Algorithm

Han-Chun (Vivien) Lin

A thesis submitted to
Auckland University of Technology
in partial fulfilment of the requirements for the degree of
Master of Engineering (ME)

2007

School of Engineering

Primary Supervisor: Professor Ahmed Al-Jumaily

ACKNOWLEDGEMENT

ABSTRACT

Blood pressure is one of the fundamental clinical measures. For more than 100 years, clinicians and researchers have used the mercury sphygmomanometer for blood pressure measurement. Environmental concern about mercury contamination has highlighted the need to find a replacement for traditional mercury sphygmomanometers.

A number of currently used non-invasive blood pressure measurement methods have been studied in this research. The most commonly used automatic pressure monitoring method nowadays is the Oscillometric method. Height-based and Slope-based criteria are the two general means used to determine the systolic and diastolic pressures. However, these two criteria have many disputed points, making them debatable as a good standard for blood pressure measurement. For this reason, the auscultatory method continues to be the gold-standard for non-invasive blood pressure measurement.

Current research uses a newly developed cuff with three different lengths of piezo film sensors and a pressure sensor to collect signals from the brachial artery. The objectives of the research are to process the measured signal from the sensors and develop a blood pressure measurement algorithm that will accurately determine the blood pressure non-invasively.

Signal processing and heart beat / heart rate detection software have been developed. The best algorithm has been selected from three developed algorithms for further modification and validation. The final algorithm used two feed-forward Neural Networks to classify the acquired pressure signals into various regions of the pressure signals. The final algorithm has been tested on 258 measurements from 86 subjects. The testing result showed that the algorithm achieved grade A for both systolic and diastolic pressures according to the British Hypertension Society protocol. The mean differences (SD) between the observers and the developed algorithm were 1.44 (5.27) mmHg and 1.77 (6.17) mmHg for systolic and diastolic pressures, respectively, which also fulfilled the Association for the Advancement of Medical Instrumentation protocol. In conclusion, this algorithm was successfully developed and it is recommended for further clinical trial in a wider adult population. Further development of this algorithm

also includes extending to other subgroups such as pregnant women, arrhythmia, diabetics and other subjects with diseases.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

STATEMENT OF ORIGINALITY

'I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a university or other institution of higher learning, except where due acknowledgment is made in the acknowledgments.'

……………………….. (signed)

…………………….. (date)

ABBREVIATIONS

AAMI	Association for the Advancement of Medical Instrumentation
ABP	Arterial Blood Pressure
ANN	Artificial Neural Network
AUTEC	Auckland University of Technology Ethics Committee
BHS	British Hypertension Society
BP	Blood Pressure
BPM	Blood Pressure Measurement
DFT	Discrete Fourier Transform
DP	Diastolic Pressure
FFT	Fast Fourier Transform
HB	Heart Beat
HR	Heart Rate
LPF	Low-Pass Filter
MAP	Mean Arterial Pressure
NIBP	Non-Invasive Blood Pressure
OAs	Oscillation Amplitudes
OpAmp	Operational Amplifier
PCA	Principal Component Analysis
PEG	Performance Error Goal
PSD	Power Spectral Density
RoC	Rate of Change
RWRA	Recursive Weighted Regression Algorithm
SD	Standard Deviation
SFLC	Synthetic Fuzzy Logic Controller
SI	Stroke Index
SP	Systolic Pressure
SSF	Slope Sum Function
STFT	Short-Time Fourier Transform
TPRI	Total Peripheral Resistance Index
VC	Volume-Compensation
VI	Virtual Instrument
VO	Volume-Oscillometric

Chapter 1 INTRODUCTION

1.1 Background

The determination of human blood pressure is very important to medical professionals especially for clinical studies of certain illnesses, blood hypertension classification and monitoring the condition of patients during operations. The general public measures blood pressure to check their cardiovascular health status. Blood pressure measurement (BPM) can be classified into two groups, invasive (direct) and non-invasive (indirect). Invasive techniques of BPM involve inserting a catheter into the vascular system which brings high risks of embolism, arrhythmia, heart attack and a certain percent of mortality [1]. This method is not convenient for everyday application. It will only be used when absolutely necessary. The non-invasive devices are safer, easier to use, and can be utilized in most situations [2, 3]. This research is focussed on the Non-Invasive Blood Pressure (NIBP) measurement.

In 1896 the Italian paediatrician Scipione Riva-Rocci invented the air cuff sphygmomanometer measurement method [4]. Mercury sphygmomanometers soon become the gold-standard of NIBP measurement. Environmental concern about mercury contamination has highlighted the need to find a replacement for traditional mercury sphygmomanometers. Aneroid sphygmomanometers are the other option of manual sphygmomanometers but these have to be checked against a mercury manometer frequently if they are suspected of being out of calibration. The other type of blood pressure measuring devices is called automated or digital sphygmomanometers. These are easily operated and more practical in noisy environments. Although there are many different NIBP measurement devices in the market now, high accuracy clinical use devices are very expensive and in certain sub-groups of populations, such as pregnant women, NIBP devices remain inaccurate. A recent clinical review [5] showed that 23 automated BP measurement devices have been validated according to the British Hypertension Society (BHS) [6] and Association for the Advancement of Medical Instrumentation (AAMI) [7] standard protocols. Five devices have been recommended

for clinical use and only one of the devices, the Omron HEM-772C, when tested on elderly subjects, achieved an A/A grading according to the BHS protocol.

Pulsecor Ltd. is a New Zealand company developing a non-invasive medical measurement device. Pulsecor has developed a new prototype for automated NIBP measurement. Improvements in blood pressure cuffs were researched. Low-frequency, wideband external pulse transducers (sensors) were used and placed in the blood pressure cuffs to measure the pulse pressure waves from the brachial artery. The purpose of this research was to analyse the measured signal from the new developed sensors and develop a BPM algorithm that would accurately determine the blood pressure non-invasively. As the end product of this device will be used in American's market, the AAMI standard protocol was the main standard used in this research. Both of the AAMI and BHS standard protocols will be briefly explained in Chapter 2.

The following sections describe the background of blood pressure, various NIBP measurement techniques and some recent developments.

1.2   Blood Pressure

Blood pressure is the force exerted by the blood on the walls of the blood vessels [8]. Generally speaking "blood pressure" refers to systemic arterial blood pressure. Blood is pumped from the heart and the blood vessels carry blood from the heart to all the tissues. The pressures of the blood in other vessels are much lower than the arterial pressure. Figure 1.2.1 shows the BP changes in the different parts of the circulatory system. The BP in the arteries is determined by 1) the force provided to the blood by the heart during its contraction and ejection of blood into the arterial compartment, 2) the flow rate of blood out of the arterial compartment and into the tissues via the capillaries, 3) the volume of blood within the vascular compartment and 4) the tension generated by the walls of the blood vessel resisting the blood pushed into the arteries by the heart [9].
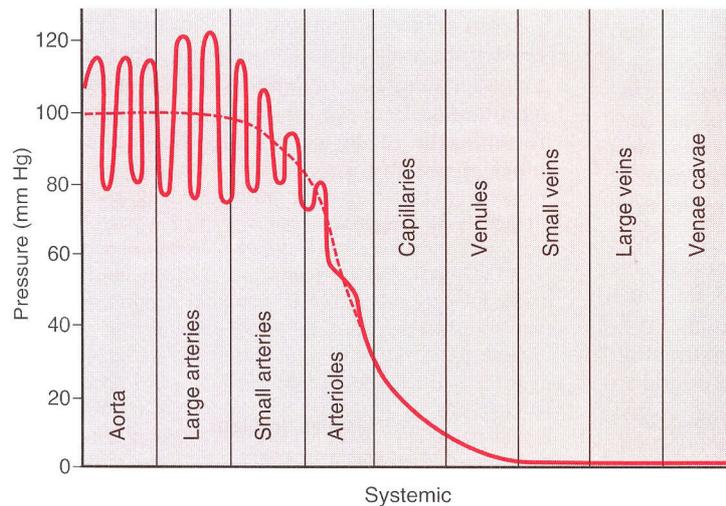
Figure 1.2.1 Normal blood pressures in the different portions of the circulatory system [8].

The heart pumps out blood into the arteries on every heart beat. BP is at its highest value when the heart pumps the blood which is also known as the contraction of the heart. This is called systolic pressure (SP). When the heart refills with blood, between each systole, BP falls to its minimum. This is the diastolic pressure (DP) [10]. Figure 1.2.2 shows a typical arterial blood pressure (ABP) waveform. The difference between the SP and DP is the pulse pressure. Mean blood pressure is often calculated as one third of the pulse pressure because almost 60% of the time the heart is in diastole [8, 11].
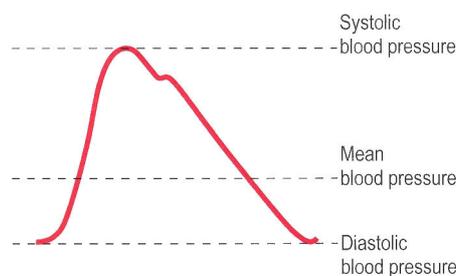
Figure 1.2.2 Illustration of the waveform of a pressure pulse [12].

BP is usually presented as two numbers, the SP and DP. These values are normally measured in millimetres of mercury (mmHg) because the mercury manometer has been used as the the standard reference for measuring pressure for a long time [8]. In a healthy adult, the systolic blood pressure is typically in the range of 120 ± 20 mmHg and the diastolic blood pressure is typically in the range of 75 ± 15 mmHg [13].

Blood pressure values are not constant. It changes from beat to beat and throughout the day. It also changes in response to stress, nutritional factors, drugs, or diseases. BPM is

3

a very important indicator to monitor cardiovascular health and specifically to determine whether a patient has hypertension or not.

1.3 Non-Invasive Blood Pressure Measurement Techniques

There are many different NIBP measurement methods currently used in NIBP monitoring, which include the Auscultatory method, Oscillometric method, Electronic Palpation method, Unloading Plethysmographic method, Volume-Oscillometric (VO) method, Volume-Compensation (VC) method, Arterial Tonometry method, etc. These methods may be classified into two categories: intermittent methods and continuous methods [2, 14]. In intermittent methods, SP, DP, and mean arterial pressure (MAP) values are calculated over a period of time that encompasses more than one heartbeat, such as the Auscultatory method, Oscillometric method, etc. Continuous methods calculate pressure values in every heartbeat, e.g. Arterial Tonometry method, etc.

The Auscultatory and the Oscillometric methods are the most common techniques used in commercial blood-pressure monitoring nowadays [2, 3, 15, 16]. Many studies are based on these two methods to develop new algorithm in order to improve accuracy and stability.

1.3.1 Auscultatory (Riva-Rocci) Method

The auscultatory method is often called the Riva-Rocci/Korotkoff method [14]. This method uses an occluding cuff and a stethoscope to listen to the presence and absence of acoustic pulses generated by an artery, called Korotkoff sounds [3, 17, 18]. The presence or absence of sound is used to identify the pressures at which blood initially begins to flow through the brachial artery (SP) and at which normal blood flow returns (DP) [17].

This method has various sources of potential error, including the inconstant deflating rate, inappropriate cuff size with respect to arm diameter, different standard of identifying Korotkoff phases between observers and so on. Also high noise situations, such as ambulatory environments or patient movement, will cause great measurement errors [17].

1.3.2 Oscillometric Method

The most widely used automatic method nowadays is the oscillometric method. It is based on the principle that the pulsatile blood flowing through an artery creates oscillations of the arterial wall. As the occluding cuff pressure is gradually reduced from above SP to below DP, the actual values of the SP, the DP and the MAP can be determined [2, 16]. A typical example of oscillometric method is shown in Figure 1.3.1. The oscillation in the air pressure of the arm cuff was measured and plotted as an oscillation waveform. The waveform was analysed to identify the occurrence of the systolic and DP. The actual values for SP and DP were determined by the corresponding pressure in the cuff pressure curve.
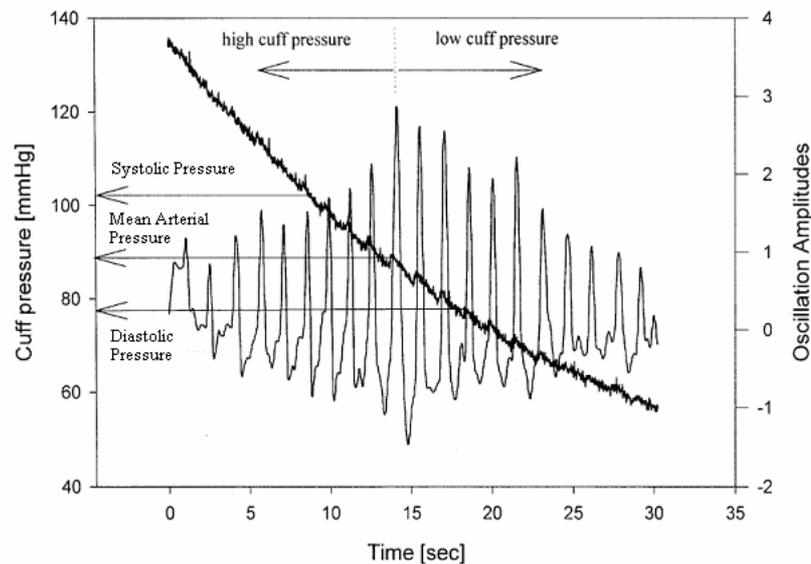
Figure 1.3.1 Cuff pressure signal (dense decreasing line) and oscillation waveform (thin line) [15].

The maximum amplitude of the cuff pressure has been accepted as the MAP [2, 15, 19-21]. There are two general algorithms to determine the SP and DP values; they are height-based and slope-based. In the height-based approach, a ratio is obtained by dividing the amplitude over the maximum value. Those ratios before the maximum amplitude would be used to compare a certain ratio to determine the SP. Whereas those ratios after the maximum amplitude would be compared to another ratio to determine the DP [2, 15, 21]. The slope-based criterion applies the derivative of the oscillation amplitude curve with respect to cuff pressure. The maximum and minimum slope of the curve has been defined as the SP and DP respectively [2, 22].

For the height-based algorithm, there are several selection criteria for the ratio used by different investigators and manufacturers. Some researchers have proposed to take fractions of 40% and 60% [2] out of the maximum amplitude. Nippon Colin Ltd. takes a fraction of 55% for both systolic and diastolic values [23]. Geddes [24] observed that the best correlated values with the auscultatory method are 50% and 80% for the SP and DP respectively. Cuff Link takes fractions of 50% and 67% and BP Pump takes fractions of 54% and 59% [25].

One of the most difficult problems with the oscillometric technique is the motion artefacts [15, 21, 26]. Those motion artefacts include respiration, speaking, involuntary or voluntary movement, etc. Patient motion produces pulses that appear similar to the arterial pulse and cause wide variations in pulse amplitudes. Another shortcoming is that a large number of cardiovascular diseases, such as arrhythmia, will lead to irregular oscillation amplitude.

1.3.3 Electronic Palpation Method

The electronic palpation method uses an arm cuff to occlude the brachial artery and then detect pressure pulses from the radial artery on the wrist. The radial artery sensor is based on a four-channel pulse sensor built in a wrist wrap. The measurement can be made either during the inflation or the deflation of the cuff.

Nissilä et al [3] used a simple algorithm to determine the SP and DP points with great accuracy. Each pressure pulse was detected and its amplitude was determined. A typical pressure pulse signal and cuff pressure in the inflation mode is shown in Figure 1.3.2(a), and Figure 1.3.2(b) shows the deflation mode. In the inflation mode, the diastolic value is the point where the pressure pulse amplitude starts to attenuate; the SP value is the point where the pulse amplitude drops under the noise level. Whereas in the deflation mode, the SP value is the point where the pressure pulse from the radial artery starts to appear, and the diastolic value is at the point where the pressure pulse amplitude levels reach a flat terrain.
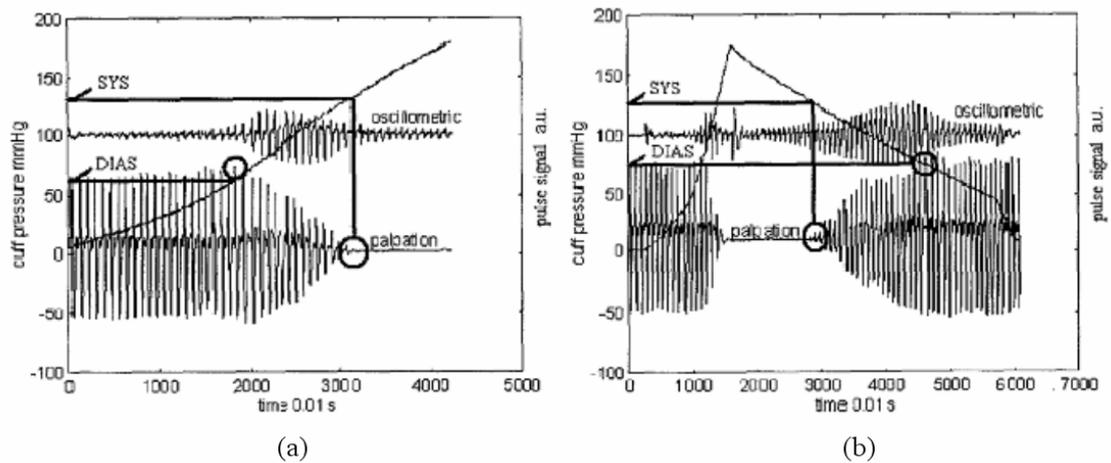
Figure 1.3.2 Typical pressure pulse signal and cuff pressure in a) the inflation mode and b) the deflation mode [3].

1.3.4 Unloading Plethysmographic Method

An indirect unloading continuous plethysmographic technique can be used in the real time BP measurement [1]. This method is based on the idea of the relationship between the cuff pressure, arterial pressure and the arterial walls. When the cuff pressure is equal to the arterial pressure, the arterial walls will be unloaded and the arteries will not change in size. Therefore, the blood volume will be constant.

Holejšovská et al [1] used the photoelectric technique to detect blood flow in the finger. The plethysmographic signal was analysed by the digital signal processor and then found out the character of the signal corresponding to the detected blood flow value. Improper setting of the reference value and the control system produced great measurement errors. Changes in the tension of the arterial wall and the movement of the body also caused a lot of measurement errors. This research is still in progress and the principle of this method has not been extended in the clinical application because of its insufficient accuracy and stability.

1.3.5 Volume-Oscillometric and Volume-Compensation Method

VO and VC methods for NIBP measurement were developed by Yamakoshi et al [27]. Both methods are based on the vascular unloading principle and the characteristics of the pressure-volume relationship in the artery. Both methods employ photoelectric plethysmography to detect the volume changes in the artery. The VO method is similar to the oscillometric method except that it is based on arterial volume oscillations instead

of cuff pressure oscillations. It can measure SP and MAP, and can be used for long-term ambulatory monitoring. The criterion of determining DP from this method is apparently not well established. The VC method allows the continuous measurements of SP and DP and the recording of the pressure waveform non-invasively.

Tanaka et al [28, 29] developed a new system for non-invasive measure of instantaneous blood pressure in the radial artery. The system was based on the VC method. The radial artery was chosen as a measuring site to avoid venous congestion during long term monitoring.

Tanaka et al [28, 29] used a disk-type cuff for local pressurisation and a nozzle-flapper type electro-pneumatic converter for the cuff-pressure control was designed. The cuff-pressure was gradually increased by a commercial air pump, and the unloaded vascular volume was determined from the mean level of the photo-plethysmographic signal. The results indicated that the radial artery could be completely compressed, and the nozzle-flapper type electro-pneumatic converter had sufficient frequency response for BP measurement in humans. The prototype system was capable of measuring instantaneous blood pressure non-invasively in both rest and stressful conditions.

1.3.6   Arterial Tonometry method

The tonometry is a transcutaneous method for continuously monitoring arterial blood pressure. The principle of arterial tonometry is illustrated in Figure 1.3.3. An array of pressure transducers are embedded in a tonometric sensor to increase the chances that at least one transducer will be positioned over the artery. The radial artery is pressed against the radius bone by the tonometric sensor with hold-down pressure in an air chamber. The optimum hold-down pressure is given automatically to flatten a portion of the arterial wall and maximize the pulse pressure measured by the sensor elements which are positioned over the artery. Intra-arterial pressure (P) in Figure 1.3.3 is measured by the pressure transducer positioned over the flattened portion of the arterial wall. The circumferential tension (T) in the flattened arterial wall to the transducer is neglected [30, 31].
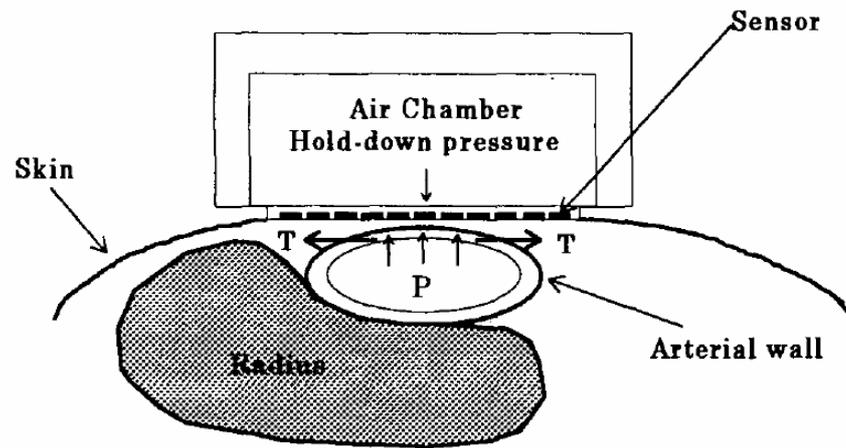
Figure 1.3.3 Illustrations of the principles of arterial tonometry [31].

1.4   Literature Review

It is not the intent of this research to provide an evaluation of the measurement methods or the technologies associated with the implementation of the methods. The measured signals from the new developed BP cuff are similar to an oscillometric waveform. The literature review section is therefore focusing only on the Oscillometric measurement algorithm developments.

1.4.1   Prediction and Smoothing Algorithm

Thomas J. Dorsett [26] developed a prediction and smoothing algorithm to predict the next oscillometric pulse amplitude and cuff pressure. The algorithm used Kalman filter equation to predict the next pulse amplitude, and polynomial curve fitting to produce a smooth curve to determine MAP, SP and DP. The result shows that the Kalman filter has the ability to reject the artefact motion and also meet the requirement of AAMI protocol.

1.4.2   Fuzzy Logic

Wang et al [16, 32] designed a NIBP method to measure blood pressure by detecting arterial volume pulsation. A Kalman filter has been designed to reduce the physiologic and measurement disturbance of the vessel volume oscillation amplitude. A linear predictor has been designed to estimate the changing tendency of the MAP in each HB, and the estimated result were feedback to the synthetic fuzzy logic controller (SFLC). The result showed that the MAP with changing rates of ±10, ±20 or ±30 mmHg/minute

could control the chamber pressure in real time with a mean square error of 1.9, 2.2 or 2.8 mmHg, respectively [32].

Lin et al [15] developed a fuzzy-logic-based recursive weighted regression algorithm (RWRA) to reduce the interference in the oscillation amplitudes (OAs). A fuzzy logic discriminator has been designed to remove the interference in the OAs. The Lorentzian function has been used to fit the patterns of OAs of the cuff pressure. Clinical results demonstrated that the proposed RWRA was more robust than the traditional curve fitting algorithm and improved the accuracy of the oscillometric BP measurement.

1.4.3 Fuzzy Pulse Qualifier

Colak et al [33] developed a pulse qualifier based on fuzzy set theory. Quality assessment is based on the pulse features, pressure, height, width and derivative. This development evaluates the quality of each pulse before entering the analysis procedure. Then the BP estimation can be more accurate.

1.4.4 Neural Network Method

A feedforward neural network has been developed to estimate blood pressure [34]. A set of recorded database is low-pass filtered to eliminate the noise. Based on the research [34], a Lorentzian function has been used to fit the oscillation envelope. The results were compared with the traditional maximum amplitude pressure algorithm. The neural network yields improvements over the maximum amplitude algorithm. One major advantage with the neural network is that the waveform does not need to be above and below the SP and DP points respectively but it requires a huge database.

1.4.5 Pattern Recognition

A cuff signal pattern recognition method has been developed to measure blood pressure [35]. Peak positive (dP/dt_{Max}), peak negative (dP/dt_{Min}), and the time interval between peak positive and peak negative (t_{pp}) were recorded from both invasive and non-invasive waveforms. A comparison has been carried out to validate the accuracy of the non-invasive measurement. Linear regression analysis has been used to compare the invasive and non-invasive methods. The correlation coefficients of the systolic, diastolic and mean blood pressure for these two methods were 0.94, 0.91 and 0.95 respectively. The results obtained a strong correlation between these two methods.

1.4.6 Mathematical Modelling

Ursino et al [36, 37] built a mathematical lumped parameter model to analyse the accuracy of NIBP measurement with the oscillometric technique. The model is used to examine how alterations in some biomechanical factors may affect the accuracy of pressure measurement. The results show that alterations in wall viscoelastic properties and in arterial pressure pulse amplitude may lead to errors as great as 15-20% in the computation of diastolic and systolic arterial pressures. The paper suggests that a better evaluation of the MAP is the lowest pressure at which the cuff pulse amplitude reaches a plateau.

1.4.7 Blood Pressure Classification

Colak et al [38] developed a fuzzy classification system to extract some parameters such as height and ratios of the pulses at certain pressure levels from the cuff pressure. The Principal Component Analysis (PCA) technique was used to extract feature and fuzzy sets to classify pressure profiles. The result shows that diastolic classification performance is worse than systolic performance. Using additional features extracted from the waveform can increase classification accuracy.

Colak et al [39] developed fuzzy set algorithms to classify systolic, mean and diastolic blood pressure. Algorithms are based on fuzzy sets, whose membership functions are determined by using neural networks. Researchers employed Gram-Schmidt orthogonal transformation [40] to select appropriate features for classification. Orthogonal feature subset selection method performs a good classification on the data. Diastolic classification with a selected feature performs better than the systolic classification. This result shows that satisfactory NIBP classification can be obtained, independent of age and arm circumference size for adults by using artificial neural networks (ANNs).

1.4.8 Other Method / Algorithm

Lee et al [21, 41] developed a digital envelope detector to detect the maximum oscillation criterion in the finger artery BP measurement. The volume oscillometric algorithm has been used. The digital envelope detector used a Hilbert transform to analyse the input signal and select peak values from the average of nine sequential point-moving window. Researchers used characteristic ratios of 0.5 for SP and 0.8 for DP estimation. The auscultatory method has been used as their reference. The result has

been compared with those cases where no envelope detection was applied. The mean difference error and standard deviation (SD) were improved by 30 ~ 40%. A summary of the result is shown in Table 1.4.1.

Table 1.4.1 The algorithm results with and without digital envelope detection [21].

	With digital envelope detector (± mmHg)		Without digital envelope detector (± mmHg)	
	Systole	Diastole	Systole	Diastole
Mean difference error	3.8	5.1	8.3	8.6
Standard Deviation	6.5	5	9.3	9.6

Ball-llovera et al [2] developed an algorithm to calculate blood pressure values by applying mathematical methods to the pulse index waveform. Researchers used height criteria to calculate SP and DP values. A computer program was developed and used to statistically indicate the difference between the actual and the calculated values. The mean differences (standard deviation) between the observers and the developed algorithm were 0.09 (5.35) mmHg and -0.66 (4.06) mmHg for SP and DP, respectively. The results showed that the algorithm used in a bedside monitor (DOCTUS IV) fulfilled the AAMI standard requirement.

Zong et al [42] developed an algorithm to detect the onset of ABP pulses. The algorithm employs a windowed and weighted slope sum function (SSF) to extract ABP waveform features. A decision rule has been established for the detection of each SSF pulse onset. The results indicate that the developed algorithm can detect the onset ABP pulse less than or equal to 20 ms compared to the reference value.

Perfetto et al [43] developed an algorithm to detect systoles and diastoles from continuous blood pressure signals in head up tilt situations. The method follows blood pressure curve searching for maximum associated to systole. When systole is pointed, diastole is obtained following the curve in the inverse direction. The algorithm does not rely on the history of the signal. It can detect correctly through the rapid changes in the signal.

Gratze et al [44] developed a real-time software package for non-invasive beat-to-beat monitoring of stroke index (SI), blood pressure (BP) and total peripheral resistance index (TPRI). A meta-analysis has been used for the evaluation of autonomic function which includes spectral analysis of heat rate, BP, SI and TPRI and the automatic calculation of baroreceptor reflex sensitivity. The developed software package is suitable for online and non-invasive detection on a beat-to-beat basis.

Moraes et al [45, 46] developed a controlled linear deflation technique for BP measurement. The correlation among several quantities, such as reference BP measurements, actual cuff pressure, pulse amplitude, characteristic ratios, age, weight, height, arm circumference size, systolic, mean and diastolic blood pressures, has been analysed. Researchers used fixed percentile rule and characteristic ratio in relation to pressure rule to determine SP and DP. Table 2 shows the mean difference error and SD results from different rules. Parameters such as arm circumference size and maximum envelope amplitude have resulted in a more precise BP measurement with the use of the oscillometric method. Other parameters, such as age, weight, and height, are not so important. The method used in such oscillometric BP measurement systems produced better results, after an experimental comparison study was made.

Table 1.4.2 The result obtained with fixed percentile rule and with different characteristic ratio [45].

	With fixed percentile rule (± mmHg)		With different characteristic ratio (± mmHg)	
	Systole	Diastole	Systole	Diastole
Mean difference error	-0.9421	0.9589	-1.4927	0.6152
Standard Deviation	6.9828	6.5146	5.1057	5.8931

1.5 Objective

Published studies of the oscillometric method introduced various algorithmic approaches for the determination of SP, DP and MAP. There is a general agreement for MAP determination but not SP and DP. Height-based and Slope-based criteria are the two general means used to determine the SP and DP values. However, these two criteria have many disputed points and the accuracy has been questioned. Researchers have

improved the algorithms to solve the problem caused by the fixed percentile algorithm. The accuracy of the SP and DP was also improved but most of the researches were more focused on general subjects. Accurately measuring BP from a special group of people such as those with cardiovascular diseases or pregnant women still remains unsolved. The new NIBP measurement prototype was developed by Pulsecor to accurately measure BP for both general and special subjects. In order to reach this goal, a general algorithm needs to be developed for this new device to accurately measure BP from general subjects.

The purpose of this research was:

- ♦ To build a database for healthy subjects, age 16 and above.
- ♦ To develop BPM algorithms based on various methods and select the best algorithm for further development and modification.
- ♦ To validate the final algorithm selected with two standard protocols (AAMI and BHS).

Chapter 2 Theory and Experimental Setup

2.1 Introduction

This chapter covers the basic principles of some digital signal processing and statistical analysis techniques applied in this research. The definition of a signal and the reason of applying those techniques will be explained in detail. Two standard protocols to validate the accuracy of the new developed device and the minimum requirements to pass the standard will be briefly explained. The details of the experimental apparatus, setups and procedures are also included in this chapter.

2.2 Signal Processing

A signal is a physical quantity that varies as a function of an independent variable [47, 48]. Time is frequently used as the independent variable. A signal can be classified as either continuous or discrete. Continuous signals are signals that have a specific value at every instant of time. Discrete signals only have certain defined values at discrete points in time. This can be thought of as the result of continuous signals that have been sampled periodically. In this project, the signal measurements are considered discrete signals since a digital computer can only record a finite number of signal samples.

The purpose of signal processing is to extract information from the signal, find the relationship between different signals, and/or produce an alternative representation of the signal. Signal processing is like a system analysis, the signal is the input to a "black box" which contains the rules for processing the signal, and the output is the desired information signal [48]. This project applied signal processing techniques in both hardware and software to remove unwanted signal components and extract useful information from the measured signal.

A signal can be presented and processed in different domains. This project focused on the time and frequency domain. A signal in the time domain is represented as $x(t)$,

where *x* refers to the dependent (measured) value and *t* is the time variable, and *X*(*f*) is the Fourier transform represented in the frequency domain.

2.2.1 Fourier Transform

The Fourier analysis is a mathematical technique for transforming a signal from the time domain to the frequency domain. It breaks a signal into constituent sinusoids (sines and cosines) of different frequencies. The Fourier transform is a generalisation of the Fourier series where function is represented by the sum of sines and cosines.

The Fourier transform of input signal *x*(*t*) is defined in equation (2.1)

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \tag{2.1}$$

where *f* is the input frequency in cycles per second. Another way of presenting the equation is by using the angular frequency ω in radians per second where $\omega = 2\pi f$ [49].

If a signal is a sum of sinusoids of different frequencies, then equation (2.1) can produce a series of weighting factors of all the sinusoidal components that add up to the original signal *x*(*t*).

The original signal *x*(*t*) can also be reconstructed by applying the inverse Fourier transform technique from the frequency domain as shown in equation (2.2).

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{2\pi ft}df \tag{2.2}$$

The Fourier transform calculates the frequency, amplitude and phase of each sine wave from the original signal. Significant values can be obtained from this transformation. The spectrum of a continuous-time signal can also be found by computing the Fourier transform of the signal [50].

2.2.1.1 Fast Fourier transform (FFT)

Since a digital computer only works with discrete data, the Discrete Fourier Transform (DFT) technique is used for transforming spectrum samples from the time domain to the frequency domain. The FFT is a mathematical algorithm developed by J.W. Cooley and J.W. Tukey in 1965 [49]. The FFT technique greatly reduces the number of

multiplications required in the calculation of the DFT and produces a result much faster with no mathematical difference between the DFT and the FFT calculations [47].

Spectrum samples are computed with the DFT. The DFT (FFT) equation is shown in the equation (2.3)

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j2\pi nk/N} \qquad \text{for } k = 0, 1, 2, \ldots N-1 \qquad (2.3)$$

where $x[n]$ is the input sequence, $X(k)$ is the DFT, $2\pi k$ is the angular frequency of the input sequence frequency (k) and N is the number of samples in both discrete-time and discrete-frequency domains.

The inverse DFT (IFFT) is shown in the equation (2.4) [51]

$$x[n] = \frac{1}{N} \sum_{k=1}^{N} X[k] e^{j2\pi(n-1)(k-1)/N} \qquad \text{for } n = 1, 2, 3, \ldots N \qquad (2.4)$$

The signal in the time domain can be converted into the frequency domain by applying the FFT function and the reverse conversion can be done by applying the IFFT function. Unwanted signals in different frequencies can be eliminated in the frequency domain before applying the IFFT function. Therefore, only the desired signals stay in the time domain. An example is shown in Figure 2.2.1. Figure 2.2.1(a) displayed a pure sine wave signal at 60 Hz with a data sampling frequency of 1000 Hz. Figure 2.2.1(b) is the signal corrupted with some random noise. The frequency spectrum resulting from FFT is demonstrated in Figure 2.2.1(c). It can be seen that the most significant signal occurred around the frequency between 59 Hz to 61 Hz. Figure 2.2.1(d) reconstructed the signal only with the frequency between 59 Hz to 61 Hz which is very similar to the original signal as shown in Figure 2.2.1(a).
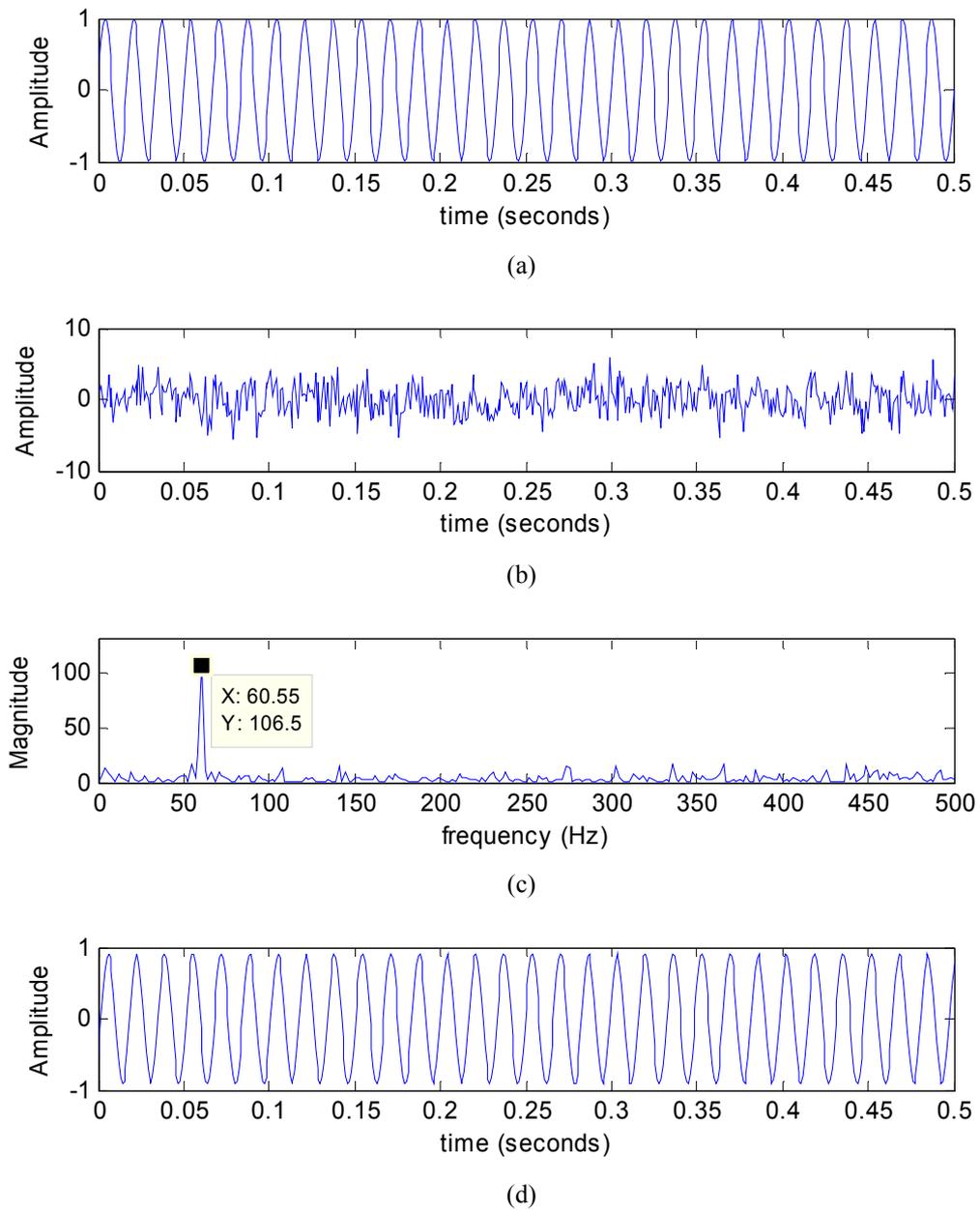
Figure 2.2.1 (a) A pure sine wave signal at 60 Hz in the time domain. (b) Signal corrupted with random noise. (c) Fourier transform of signal (b). (d) Inverse Fourier transform of signal between frequency 59 Hz to 61 Hz.

2.2.1.2 Short-Time Fourier Transform (STFT)

As mentioned before, the Fourier analysis transforms a signal from the time domain to the frequency domain. However, STFT divides the original signal into small segments by means of the FFT and then displays the result spectra as a 'spectrogram' [52]. Each segment may be overlapped by the adjacent segment for a certain amount of time. The STFT represents a two-dimensional function of time and frequency. It can be represented in either a two-dimensional or three-dimensional view. Examples are shown in Figure 2.2.2. The STFT provides the magnitude of each segment signal at a different time and frequency. It shows that during the time between 0 ~ 0.25 seconds the largest magnitude (dark red) occurred at frequency between 60 ~ 70 Hz.

(a)   (b)
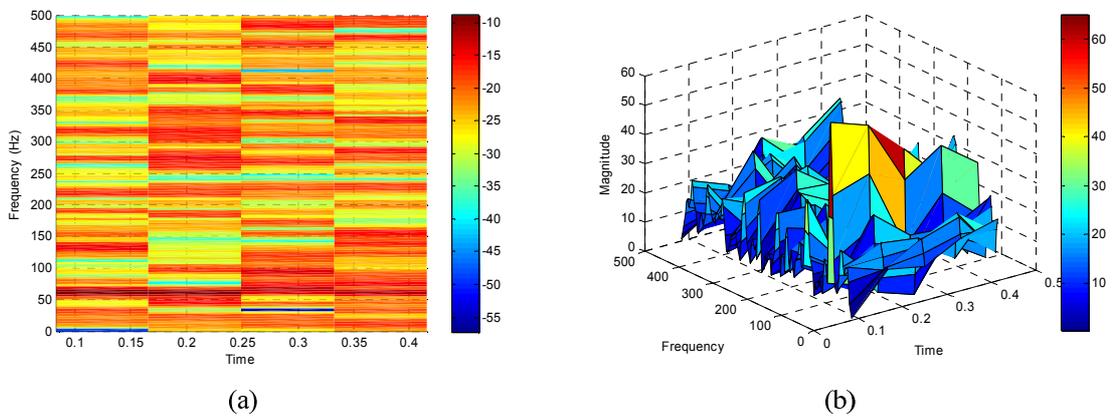
Figure 2.2.2 Time-Frequency diagram. (a) Two-Dimensional View. (b) Three-Dimensional View.

This technique has certain precision limitations because the precision depends on the size of the window. A narrow window gives a good time resolution because it provides more details, but a poor frequency resolution, since the time is too short to observe the signal characteristics. Increases in the window size gives better frequency resolution but poor time resolution [53]. Table 2.2.1 shows the quality of resolutions by selecting different window sizes. Therefore, it is very important to select the right size of window to obtain the desired value during the process.

Table 2.2.1 Time and Frequency Resolution by Window Width.

	Time Resolution	Frequency Resolution
Narrow Window	Good	Poor
Wide Window	Poor	Good

2.2.2   Power Spectral Density (PSD)

The PSD is basically a method to describe the distribution of power contained in a signal, as a function of frequency [50]. This measures the contribution to the signal made by each of its sine wave components. The power spectrum of a signal can be simply obtained by calculating its DFT and summing the squares of its real and imaginary components at each frequency [49]. The power spectra can be used for detecting the signals hidden in wide-band noise.

2.2.3   Noise Removal

Removing noise from a signal is commonly applied in all situations. The goal is to remove the unwanted signal with the least distortion of the desired signal. Filters are designed for this purpose. Common categories of filters include high pass, low pass, band pass and band stop. Designing a filter can remove large amplitude and other unwanted noise signals from the required bandwidth [48, 52]. Therefore, it is very important to understand the properties of a desired signal.

As mentioned above this project recorded the continuous-time signals using digital sampling at a certain sampling rate, therefore the filtering process can be done either before sampling using an analogue filter or after sampling using a digital filter. Analogue filters are circuits normally built from components such as resistors, capacitors and inductors. Applying an analogue filter before sampling can prevent aliasing during the measurement of the continuous-time signal but its operation is very sensitive to the values of the components used. On the other hand, digital filters are normally executed in software and might achieve the narrowing of the transition bands [48].

2.2.4   Windows

The purpose of using windows in signal processing is to force the signal outside the chosen range to be zero [48]. The output value within the chosen range M is computed by multiplying the signal value $x[nT]$ with the window function $w[nT]$. Other values outside the range M will become zero. The simplest window is the rectangular window. Unfortunately, it produces a filter with poor stop band attenuation. It is not sufficient to remove signals outside the pass band. Using a window with smoother edges can solve this problem [53]. The Hanning, Hamming, Blackman, and Kaiser windows give better

stop band attenuations. Figure 2.2.3 shows some commonly used windows and sketches of their forms.

$$w[nT] = \begin{cases} 1 & |n| \leq M \\ 0 & |n| > M \end{cases}$$

(a) Rectangular Window

$$w[nT] = \begin{cases} 1 - |n|/(M + 0.5) & |n| \leq M \\ 0 & |n| > M \end{cases}$$

(b) Triangular (Bartlett) Window

$$w[nT] = \begin{cases} 0.5(1 + \cos[\pi n/(M + 0.5)]) & |n| \leq M \\ 0 & |n| > M \end{cases}$$

(c) Raised Cosine (Hanning) Window

$$w[nT] = \begin{cases} 0.54 + 0.46 \cos[\pi n/(M + 0.5)] & |n| \leq M \\ 0 & |n| > M \end{cases}$$

(d) Hamming Window
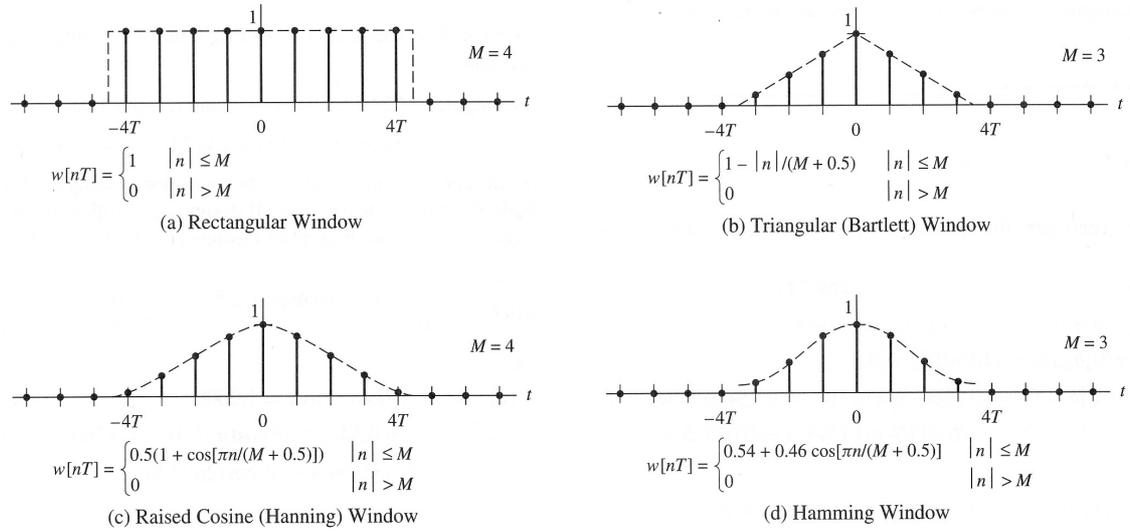
Figure 2.2.3 Window examples and functions [50].

2.2.5   Normalisation

Normalisation is a technique to spread the component range of the signal values uniformly over the whole range of the input [54]. In other words, this technique rescales the input signal to the range which can be interpreted more easily for further analysis. For example, a signal measured in the range between 0.4 to 0.85 volts, by applying normalisation technique can be rescaled into a range within 0 and 1. Therefore, the minimum 0.4 V became 0 V and the maximum 0.85 V became 1 V. This could be used to find the similarity and the changes of a signal in a different signal range.

The measured input variable x can be mapped to a scaled variable y according to equation (2.5).

$$y = \left(\frac{x - x_{\min}}{x_{\max} - x_{\min}} \right) (y_{\max} - y_{\min}) + y_{\min} \qquad (2.5)$$

where $x_{\min}$ is the minimum measured input variable value, $x_{\max}$ is the maximum measured input variable value, $y_{\min}$ is the minimum scaled variable value and $y_{\max}$ is the maximum scaled variable value [54].

2.3 Statistical Analysis and Classification

Statistical analysis and classification can be presented in many different ways. An Artificial Neural Network (ANN) is one of the artificial intelligence techniques which has been selected for use in this research. A basic introduction to the theoretical principles of the ANN will be briefly described in this section. Principle Component Analysis (PCA) is a method of identifying data patterns which can be used to reduce the dimension of the data set. The basic idea and the calculation procedure will be explained but the mathematical background used in the PCA method will not be covered in this thesis. Bland and Altman plot, which is one of the graphical methods for the statistical analysis, will be described as well.

2.3.1 Artificial Neural Network

An ANN is a mathematical model which is trying to model a system similar to the human brain. It consists of a number of simple and highly interconnected processors, called *neurons* which can be compared to the biological neurons in the brain. The ANN gained knowledge through the training process and stored this knowledge in long-term memory, called *weights*. Weights represent the importance of each neuron input. During the training process, weights are adjusted until the network output matches the target. ANNs have been widely used to solve complex functions in various fields including pattern recognition, identification, classification, speech, vision, control systems, and signal processing applications [55].

The basic element of an ANN is a neuron. A neuron receives input data, computes the weighted sum of the input data, calculates the transfer function and then sends the result as the output value [55]. Figure 2.3.1 shows a single-layer neuron model and its transfer function. Where R is the number of elements in input vector, p is the value of individual input, w is the weight of individual input, n is the net weighted input to the neuron, and a is the output of the neuron. The sum of the weighted input n is the argument of the transfer function f. There are four transfer functions commonly used to produce the output a. These functions and its mathematical expression are shown in Figure 2.3.2. The Hard-Limit Transfer Function in Figure 2.3.2(a) is used in decision-making neurons for classification and pattern recognition applications. The Linear Transfer Function in Figure 2.3.2(b) provides an output equal to the neuron weighted input. This function is used for linear approximation. Figure 2.3.2(c) shows the Sigmoid transfer function

which allows any input values and gives output value in the range between 0 and 1. This function is normally used in the non-linear classification [55], e.g. back-propagation networks. Figure 2.3.2(d) shows the Tan-Sigmoid transfer function which is very similar to the Sigmoid transfer function. This function gives output value in the range between -1 and 1.
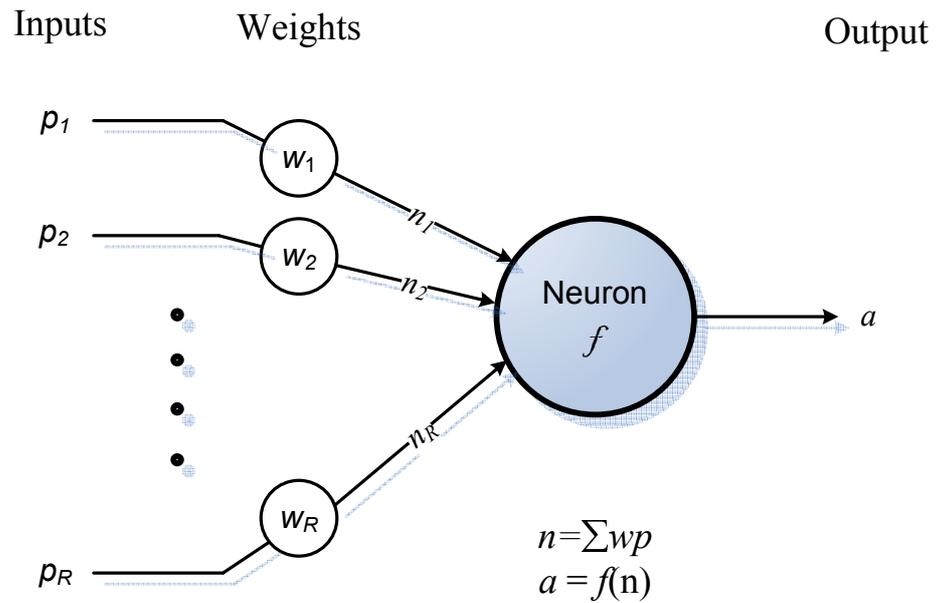
Figure 2.3.1 Diagram of a single layer neuron model and its transfer function.

$$a = \begin{cases} 1, & \text{if } n \geq 0 \\ 0, & \text{is } n < 0 \end{cases}$$

(a) Hard-Limit Transfer Function

$$a = f(n) = n$$

(b) Linear Transfer Function

$$a = \frac{1}{1 + e^{-n}}$$

(c) Sigmoid Transfer Function

$$a = \frac{2}{1 + e^{-n}} - 1$$

(d) Tan-Sigmoid Transfer Function
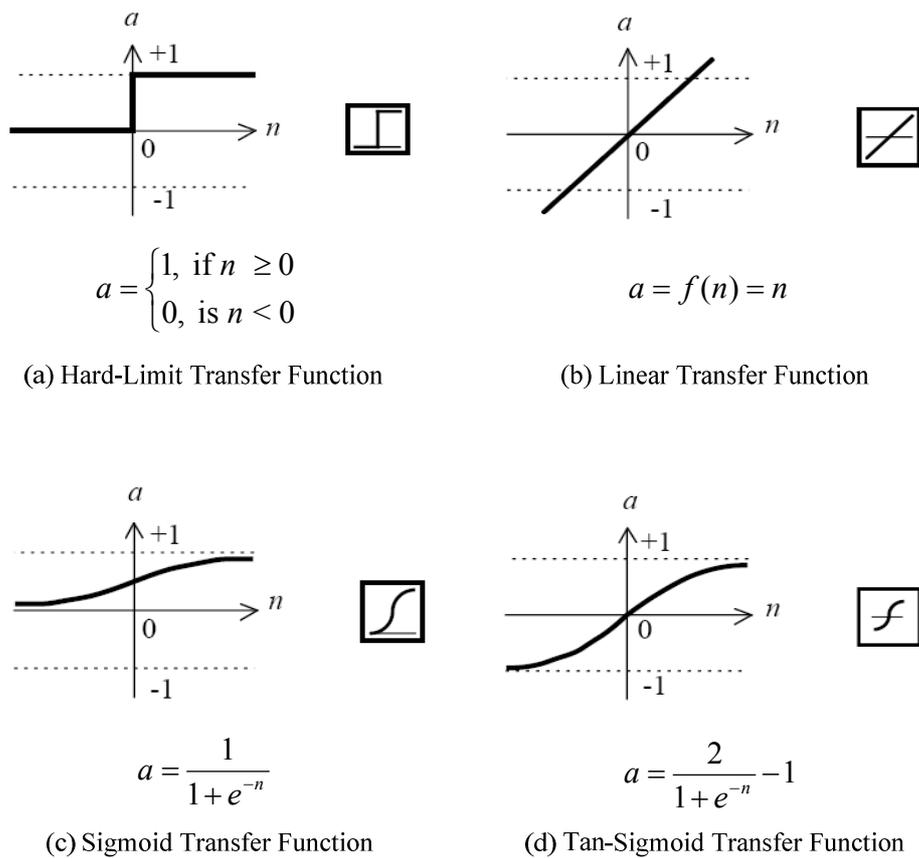
Figure 2.3.2 Transfer Functions of a neuron [56].

A single layer neuron by itself is not a very powerful tool. It can classify only linearly separable patterns. The real capability of ANN comes when neurons are combined into the multilayer structure called the neural networks. The structure of a neural network is based on a single layer neuron model interconnected with each other to form a multilayer network. A multilayer network consists of an input layer of source neurons, at least one or more hidden layer of computational neurons, and an output layer of computational neurons [55]. A multilayer network with two hidden layers is shown in Figure 2.3.3. Each layer in a multilayer neural network has a weight matrix *w*, a bias vector *b*, and an output vector *a*. The bias is much like a weight, except that it has a constant input of 1. It is simply being added to the net weighted input in each neuron [56].

$$a^1 = f^1(IW^{1,1}p + b^1)$$

$$a^2 = f^2(LW^{2,1}a^1 + b^2)$$

$$a^3 = f^3(LW^{3,2}a^2 + b^3)$$

$$a^3 = f^3(LW^{3,2}f^2(LW^{2,1}f^1(IW^{1,1}p + b^1) + b^2) + b^3)$$
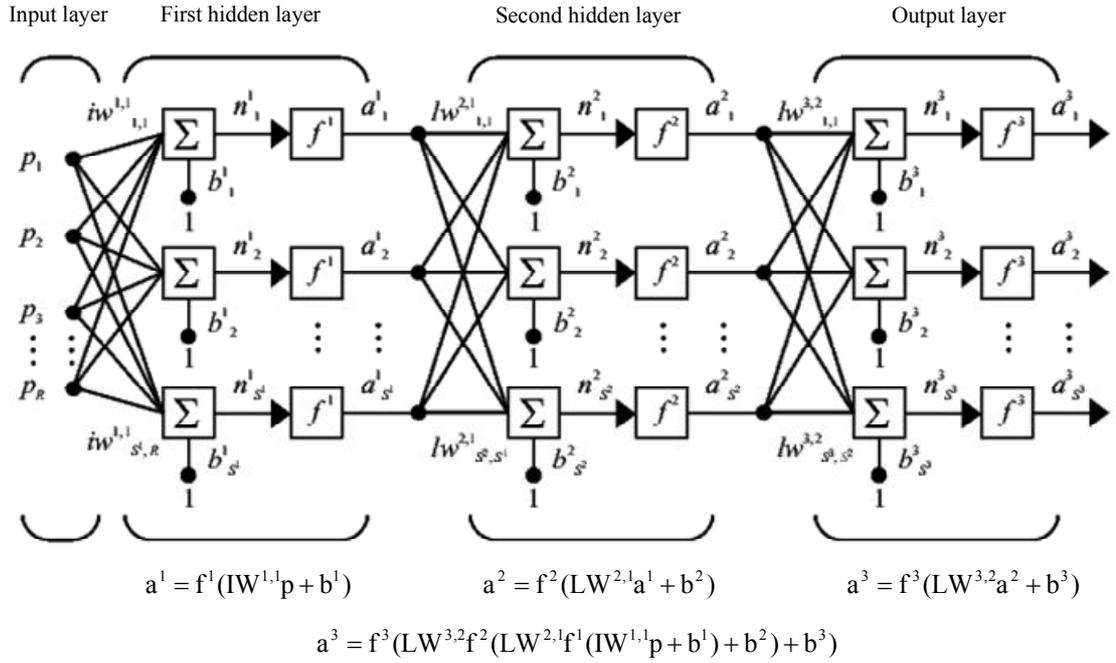
Figure 2.3.3 Multilayer neuron model with two hidden layers [56].

As described earlier in this section, an ANN needs to go through a training process and then adjusts weights until the network output matches the target. There are more than a hundred different training algorithms available but the most popular method is back-propagation [55]. In a back-propagation neural network, the training algorithm has two phases. First, the network propagates the input data from input layer to each hidden layer(s) and generates output data from the output layer. If the output data is different from the target, an error is calculated and then propagated backwards through the network from the output layer back to the input layer. The weights are modified during the backward process. Properly trained back-propagation networks tend to give reasonable outputs when testing with new similar inputs data that the networks have never seen [56]. This is a good method for biomedical signal classification since any new signal data will get similar output if it is similar to the trained signal.

2.3.2   Principal Component Analysis (PCA)

PCA is a classical method of identifying patterns in data. Once these patterns in the data have been found, PCA can be used to reduce, compress or simplify the data set. The number of dimensions in the data can be reduced without losing much information. This method can help to reduce the number of input data in the ANN [57].

For PCA to work properly, each data dimension needs to first subtract its mean value to get a data set whose mean value is zero. The covariance matrix will then be calculated from the zero mean data set. The covariance matrix gives a square matrix which has the same dimension as the number of columns, i.e. for an m-by-n matrix, where m represents the number of rows and n represents the number of columns, the covariance matrix is n-by-n. Since the covariance matrix is a square matrix, the eigenvectors and eigenvalues of this matrix can be calculated. These values provide the information about the pattern in the input data and contribution of each data to the total variation in the data set. The eigenvector with the highest eigenvalue is the principal component of the data set. It is the most significant relationship between the data dimensions. The dimension of the input data set can be reduced by eliminating those principal components that have less significance. A principal component transformation matrix can be constructed by forming a matrix using those eigenvectors with the large eigenvalues in the columns. This transformation matrix can help in transforming original input data to a reduced dimension data set and vice versa [57, 58].

2.3.3   Bland and Altman Plot

Bland and Altman plot was designed for assessing the agreement between two measurement results [59]. It shows the difference between new and old methods based on the same subject. An example is shown in Figure 2.3.4. In this graphical method the differences between the new developed algorithm and the average results from two observers is plotted at y-axis. The average between these two values is plotted at x-axis. Horizontal (solid) lines are drawn at the mean difference, and at the mean difference ±1.96 multiplied by the SD of the differences to show the 95% distribution of the difference values. Horizontal (dotted) lines can also be drawn at zero difference, ±5 mmHg, ±10 mmHg and ±15 mmHg difference for the BHS protocol requirements. The Bland and Altman plot is useful to reveal a relationship between the differences and the averages.
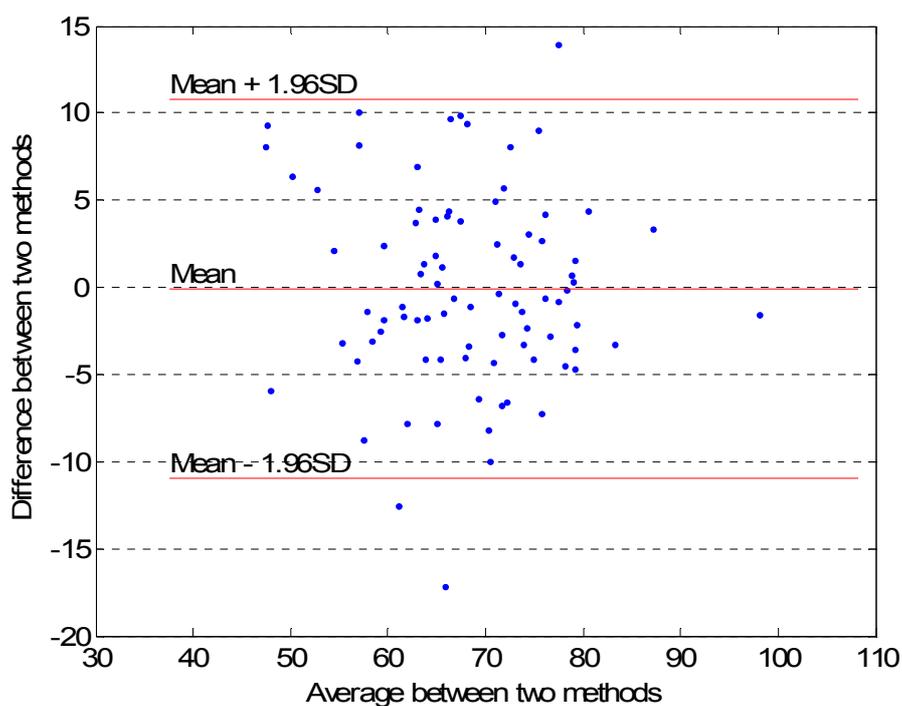
Figure 2.3.4 An example of a Bland and Altman plot.

2.4   Standard Protocols

Accurate measurement of blood pressure is very important from the public health standpoint. Any new developed BP measurement devices should be validated with the standard protocols. Two widely used protocols for testing the accuracy of these devices are set by the AAMI with a pass/fail system, and the protocol from the BHS with an A-D graded system. AAMI standard protocol recommended that for comparison of auscultatory monitors, at least 85 subjects should be studied. 80% of SP and DP should be within the range from 100 to 160 mmHg and 60 to 100 mmHg respectively, 10% above and below the range stated above. 80% of subjects should have an arm size between 25 to 35 cm in circumference and 10% above and below the range. A device would pass the AAMI protocol if its measurement error has a mean value of less than 5 mmHg with a SD of no more than 8 mmHg. The BHS protocol would grant grade A to a device if 60% of its error measurements are within 5 mmHg, 85% of the errors are within 10 mmHg, and 95% errors fall within 15 mmHg. BHS has progressively less stringent criteria for the grades of B and C, and will assign a grade D if a device performs worse than C. Table 2.4.1 shows the summary of the grade system for BHS protocol. A device will only be recommended if it passed the AAMI criteria for both SP and DP and received a grade of A or B under the BHS protocol for both systolic and

diastolic blood pressures.

Table 2.4.1 Grading criteria used by the British Society of Hypertension.

Grade	Absolute difference between reference and test device (%)		
	≤ 5 mmHg	≤ 10 mmHg	≤ 15 mmHg
A	60	85	95
B	50	75	90
C	40	65	85
D	Worse than C		

Both protocols are used in validating the accuracy of the new developed device. The experimental part carried out in this research was based on the AAMI protocol requirement.

2.5   Experimental Setup

2.5.1   Apparatus

The new cuffs with three different sizes were made from Trimline blood pressure cuffs (Trimline Medical Products Corp, NJ, USA). DT1, DT2 and DT4 piezoelectric film sensor elements from Measurement Specialties (VA, USA) were attached on the surface of the cuff (refer to Figure 2.5.1) to collect signals from the brachial artery. The DT1 and DT2 piezoelectric sensors were placed on the inside wall of each cuff in the axial direction. The DT4 piezoelectric sensor was placed on the outside wall of each cuff in the circumferential direction. The inside and outside sensors were attached to the cuff using cyanoacrylate. Signals were measured using the different voltage generated across the positive and negative terminals of each sensor.
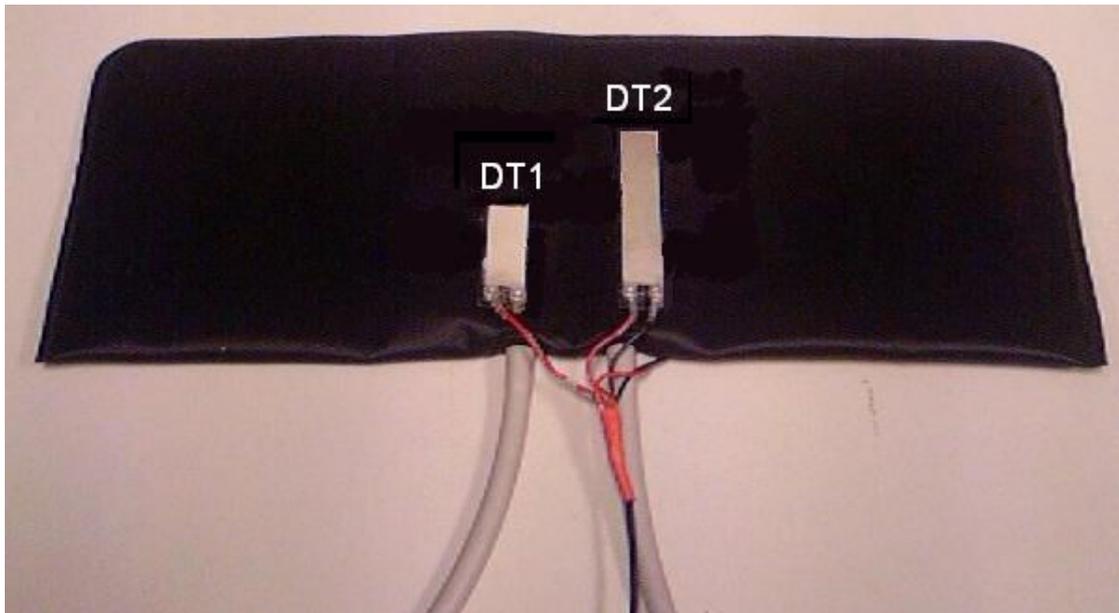
(a) Inside sensors displacement

(b) Outside sensor displacement

Figure 2.5.1 Inside and outside sensors placement of a standard medium cuff.

All negative pins of the sensors were connected together and routed to ground. The positive pin from the sensors was individually connected to an Operational Amplifier (OpAmp). The circuit configuration sheet can be found in Appendix I. The voltage output from each OpAmp was digitized using a data acquisition card, DAQCard-AI-16XE-50 (National Instruments, TX, USA), in differential input mode to maintain a high-input impedance. A Virtual Instrument (VI) has been created for data recording in the text format by using v6.1 LabView software program. The inside, outside and cuff pressure sensor signals were recorded simultaneously for further processing.

The cuff has two air hoses protruding from the bladder. One air hose was connected to a Medisave aneroid sphygmomanometer. The second air hose was connected to Welch Allyn® NIBP module which was used for inflation and deflation of the cuff automatically. Poemtalk software program (Welch Allyn … etc.) was used to control the NIBP module. The NIBP module was also connected to an ADP1 semiconductor pressure sensor (Matsuchita Electrical Works Ltd, Japan) via a tee-junction. The ADP1 was powered using the 5V reference from the DAQCard, and the output voltage corresponding to pressure was measured using a differential analogue input on the DAQCard.

The schematic setup of the apparatus is illustrated in Figure 2.5.2. Although two different software programs were run on the same computer, it is clearer to show them separately in the figure to demonstrate their different functions. Figure 2.5.3 to Figure 2.5.8 show the actual apparatus setup in the data collection.
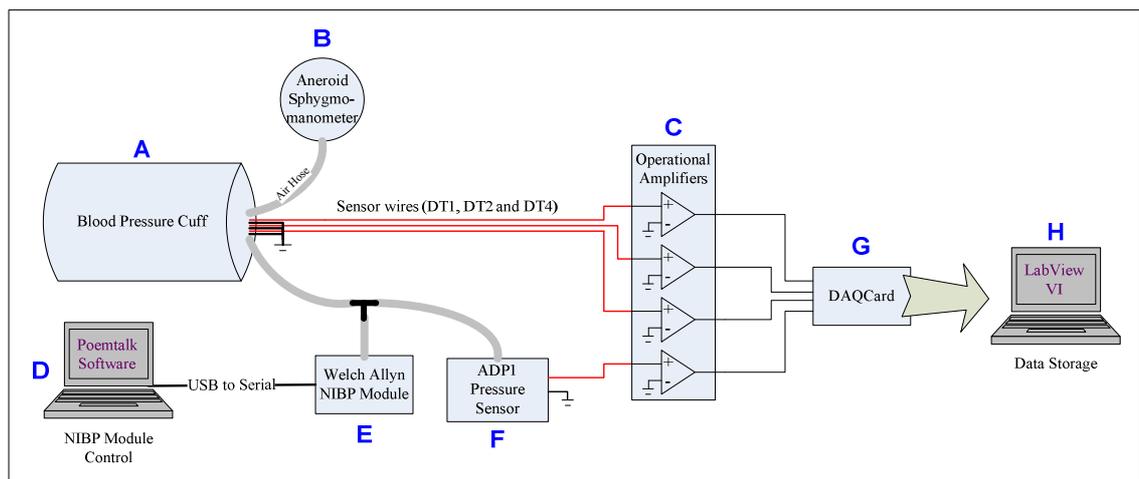
Figure 2.5.2 Data collection apparatus setup schematic diagram.
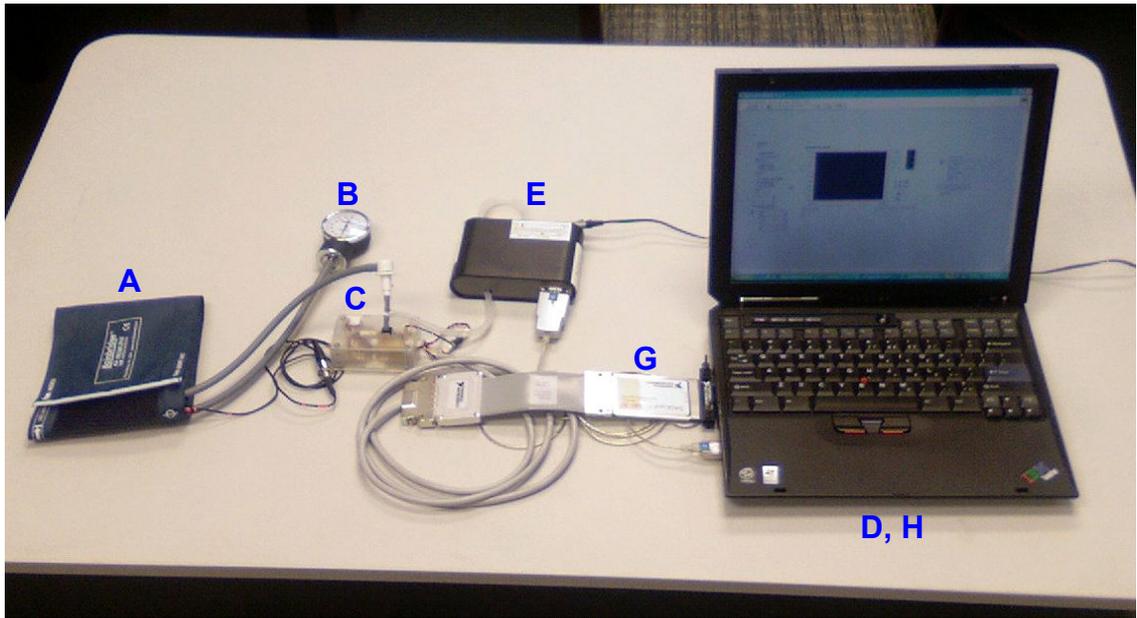
Figure 2.5.3 Actual apparatus used in the data collection.

Figure 2.5.4 Standard medium cuff and an aneroid sphygmomanometer.

Figure 2.5.5 Data acquisition card.
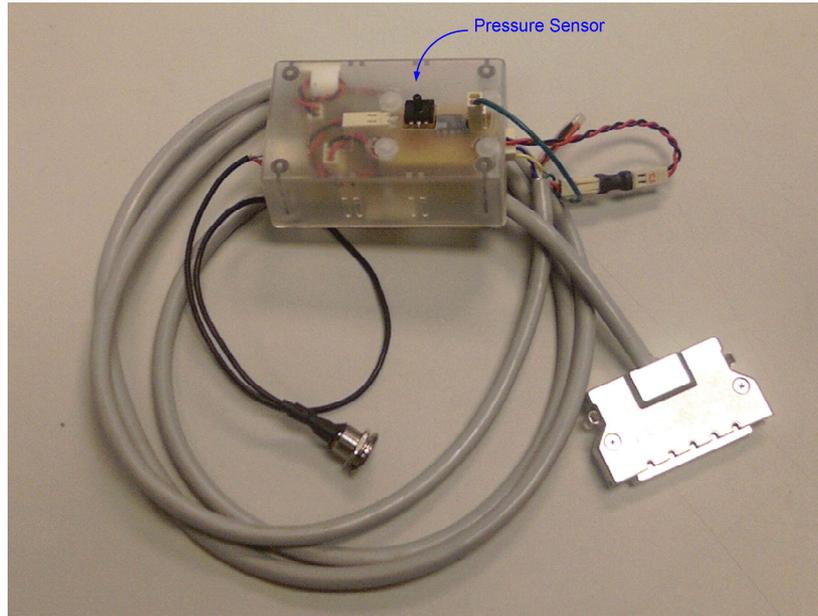
Figure 2.5.6 At the bottom of the box is the Operational Amplifier circuit board and the pressure sensor board is at the top of the box.

Figure 2.5.7 Welch Allyn® NIBP module.

Figure 2.5.8 Teaching stethoscope.

The bell mode side of a 3M^{TM} Littmann$^{®}$ dual-head teaching stethoscope was used to measure auscultatory blood pressure. This teaching stethoscope was used to allow two observers to listen simultaneously.

2.5.2 Cuff Pressure Calibration

The voltage output from the pressure transducer was calibrated against the aneroid sphygmomanometer over the range from 20 mmHg to 280 mmHg. The ADP1 semiconductor pressure sensor was used for cuff pressure detection. Using the amplifier designed by Pulsecor Ltd., the pressure sensor produced a linear characteristic between 0.5 to 3.5 output voltages. The calibration process has been repeated and carried out for the individual cuff. The small and medium cuffs were tested on a rigid metal cylinder tube with the diameter of 75 mm and the large cuff was tested on a solid lightweight formed concrete with the diameter of 100 mm. Figure 2.5.9 shows the actual material used. The cuff was manually pumped up to 300 mmHg. The pressure reading was based on the aneroid sphygmomanometer used in this research to keep the same reading from the observers. The cuff pressure was then gradually decreased to 280 mmHg. The LabView VI recorded 500 to 1000 samples of the output voltage from the OpAmp when the pressure was read at 280 mmHg. The LabView VI then recorded the output voltage at every 10 mmHg deflation on the cuff until the cuff pressure reached 20 mmHg. The cuff was then totally deflated and removed from the cylinder object. The cuff was replaced around the object three times to repeat the calibration procedure. Regular

calibration was performed throughout the data collection period to check the stability and the accuracy of the output signal. All the recorded data was used for the linear equation calculations.

Figure 2.5.9 Cylindrical tubes used for cuff pressure calibration. The lightweight form concrete on the right hand side of the picture was used for large cuff calibration. The metal cylinder on the left hand side was used for medium and small cuffs calibration.

It was discovered from the recorded data that the output voltage between 20 mmHg to 250 mmHg had a linear relationship and that the voltage above 250 mmHg became constant as shown in Figure 2.5.10(a). Therefore, a best fit line was fitted into the values between 20 mmHg to 250 mmHg to produce a linear curve that could be used to calculate the cuff pressure. The polynomial curve fitting function was used to minimize the total square error at the data points [51]. The linear regression line equation is shown in the equation (2.6)

$$V = aP + b \qquad\qquad (2.6)$$

where P is the pressure value and V is the output voltage. The slope of the line is a, and b is the intercept (the value of V when $P = 0$).

The function polyfit was used in MATLAB to calculate the polynomial coefficients (a and b), which is a row vector, that was saved and used for cuff pressure estimation in this research project. The result of the linear regression is plotted in Figure 2.5.10(b).

(a) Raw cuff pressure data collected from the output of the OpAmp.

(b) Best fitting curve line through the collected data.
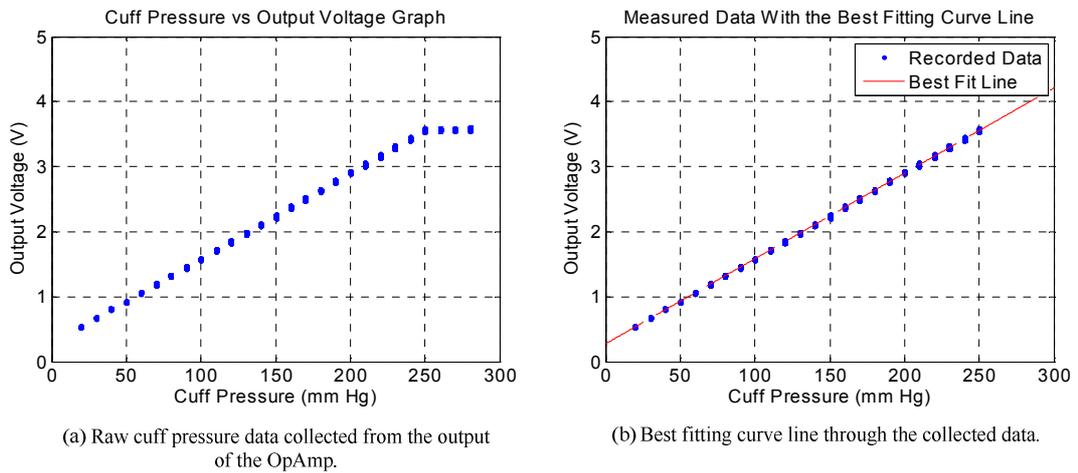
Figure 2.5.10 Cuff pressure calibration graphs.

2.5.3   Data Collection

Ethics approval was required before the data collection started since it involved human subjects. Application for ethics approval was submitted and approved by the Auckland University of Technology Ethics Committee (AUTEC) on 12 July 2006, AUTEC Reference number 06/126. All participants were required to sign a written consent form. The approved letter, Information Sheet and Consents Form can be found in Appendix II, Appendix III and Appendix IV, respectively.

According to the AAMI standard requirement, 85 subjects were required to take part in this study. All invited participants were staff and students at AUT Wellesley campus. All subjects were recruited by personal approach. A complete set of three consecutive measurements was obtained by two observers. The Auscultatory method has been used as the reference. The SP and DP readings from both observers were controlled within the difference of ±5 mmHg. For any subject that did not contribute 3 data sets within the difference of ±5 mmHg between observers, an additional measurement was carried out. For participants requiring additional measurement, a maximum of five measurements were recorded. The subject data will not be used in this research if the above requirements were not met after five measurements.

Only healthy subjects age 16 and above were invited for the study. For those subjects who had weak Korotkoff sounds, irregular heart rate and/or other problems found during the measurement process, the recorded data was destroyed and cannot be used in this research. Only those subjects who met the requirements as described above were

included in this study. Additional subjects have been measured to reach the total of 85 subjects as the AAMI standard required.

2.5.3.1 Procedures

Before Data Collection

Equipments were setup as described in Figure 2.5.2. The LabView VI program will be run so that settings can be made before data collection as shown in Figure 2.5.11. Four channels have been used to collect signal from the two inside sensors, the outside sensor and the pressure sensor. Input signal limits were set between the voltages from 0 to 5 volts. The sampling rate has been set at 250 samples per second. Each measurement has been given a specific name of *expXX_z* to store measured data, where *XX* is the number assigned to the individual subject and *z* represented the number of measurements taken from that subject.
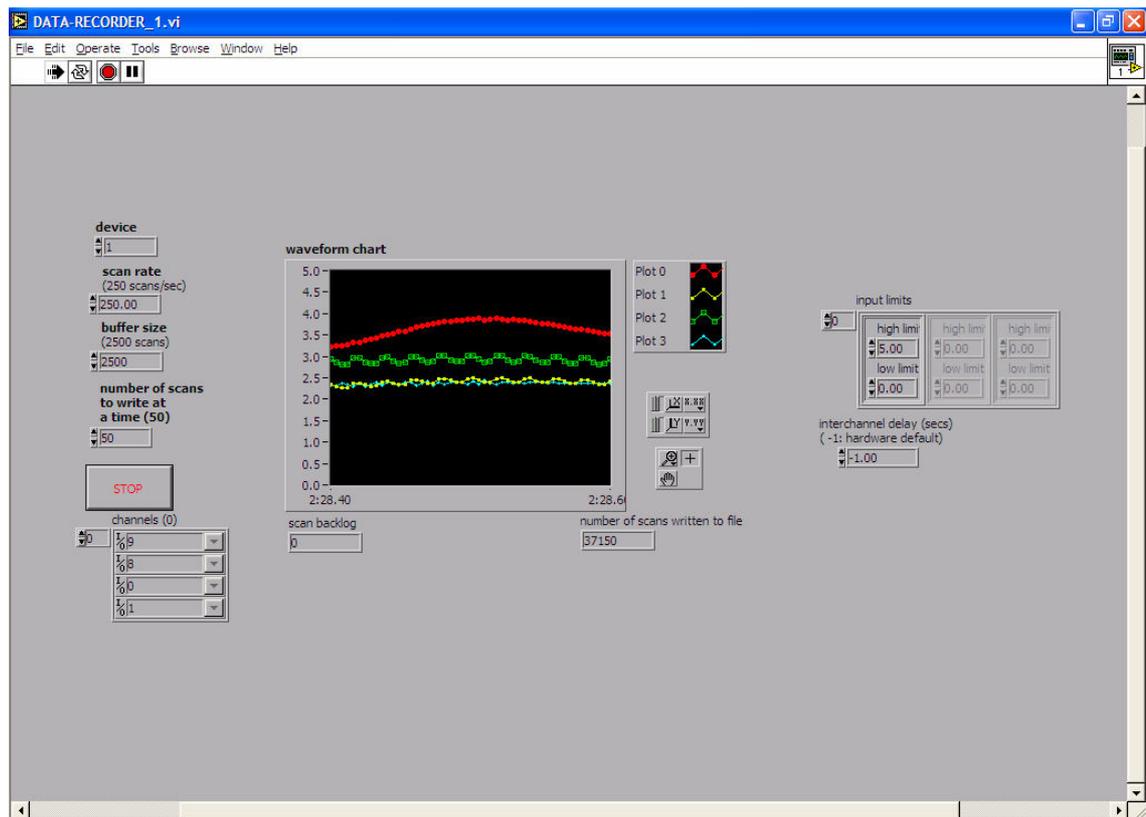
Figure 2.5.11 Graphical User Interface of the LabView VI data collection program.

The Poemtalk software program was used to control the Welch Allyn® NIBP module. This software ran under Windows Command Prompt environment. Commands were entered to set the service mode and the maximum target pressure.

All subjects were recruited by personal approach. Every subject received both information sheet and verbal explanations regarding the purpose of the research. Procedures were explained to them and time was given for them to decide to participate in this research project before written consent was obtained. Every subject was given a number instead of their name to ensure the confidentiality of the data collected. The subject's arm circumference was measured and recorded to ensure the correct cuff had been used. If the arm circumference was measured less than 25 cm, the small adult cuff was used. If the arm circumference was measured greater than 35 cm, the large adult cuff was used. Otherwise, the medium cuff was used. Before the cuff wrapped around the upper arm, observers would ensure the cuff was totally deflated and the aneroid sphygmomanometer was read at zero mmHg.

During Data Collection

The subject was allowed to rest on a chair for a while before the data started recording. The selected cuff was placed around the subject's right upper arm. The placement of the cuff was such that the subject's brachial artery lay within two inside sensors to obtain the best signal result. The subject had been asked to sit comfortably, relax, keep still and avoid deep breathing during the data collection.

Two observers sat beside the subject to take measurements using the ausculatory method. The teaching stethoscope was used to allow two observers to listen to the heart sound simultaneously. Observers recorded all subjects' blood pressure separately on a different log sheet to prevent readings read by another observer being influenced in any way.

The NIBP module inflated the cuff pressure to 150 mmHg for the first measurement. For subsequent measurements, the cuff pressure was inflated about 30 mmHg above the previous recorded SP. Then the deflation command was activated to deflate the cuff at a deflation rate of about 3 ~ 4 mmHg per second. The LabView VI was used to measure and record signal data from the cuff inflation until the cuff pressure was 30 mmHg below the recorded DP. Measured data was saved in a computer spreadsheet under the given file name for further analysis. After the LabView VI stopped recording, the fast

deflation command was used to totally deflate the cuff. The fast deflation command had to be entered after the LabView VI stopped recording to prevent large signal changes produced by the piezoelectric film elements.

After Data Collection

Two observers recorded the subject's blood pressure separately. The recorded SP and DP values were compared between two observers. If the difference of either SP or DP value was greater than 5 mmHg, the data could not be used and an extra measurement had to be carried out.

If all three pressure readings were within the limit range of 5 mmHg, an averaged blood pressure value was told to the subject. During the waiting period for the next measurement, time was given for the subject to ask questions. The collection date and time was recorded to ensure that more than a one minute break had been taken between the two measurements.

Chapter 3   Algorithm Development

3.1   Introduction

This chapter covers the application of the signal processing techniques described in the previous chapter and three different algorithm development procedures for BP measurement. Heart beat (HB) waveform and heart rate (HR) waveform detection methods will be covered in detail. The corresponding pressure value of each detected HB will be briefly explained. Developed algorithms were height-based algorithm, STFT algorithm and ANN classification algorithm. Results from each algorithm were included and discussed in this chapter. The best algorithm was selected for further development. Height-based algorithm was developed from the basic idea of the traditional Oscillometric height-based technique. STFT algorithm was developed based on signal magnitude and PSD found in the frequency domain. ANN classification algorithm was developed from the signal pattern changes through the measurement.

The new developed cuff sensors were capable to pickup all the significant reflected waveform from the brachial artery as shown in Figure 3.1.1. Figure 3.1.1(a) shows the waveform changes from the supra-systolic region to the systolic region. Figure 3.1.1(b) shows the waveform changes from the diastolic region to the sub-diastolic region. The top part of the waveform (red) represents the measured cuff pressure values. The bottom part of the waveform (blue) represents the voltage difference between the two inside sensors. The vertical dashed line represents the average pressure obtained by two observers. It is quite easy to identify the signal changes by visualizing the measured signal. An ANN classifier was developed to classify each HB into one of the pressure regions, Supra-Systolic, Systolic to Diastolic and Sub-Diastolic region.

MATLAB R2006a (The MathWorks, U.S.) software program was used to complete the signal processing and algorithm development design. Some MATLAB build in and self developed functions were used. In this report, the function name appeared in Arial font characters represented as build in functions in MATLAB, e.g. FFT. Whereas the

function name appeared in Italic font characters represented as self developed functions, e.g. *funMag*.
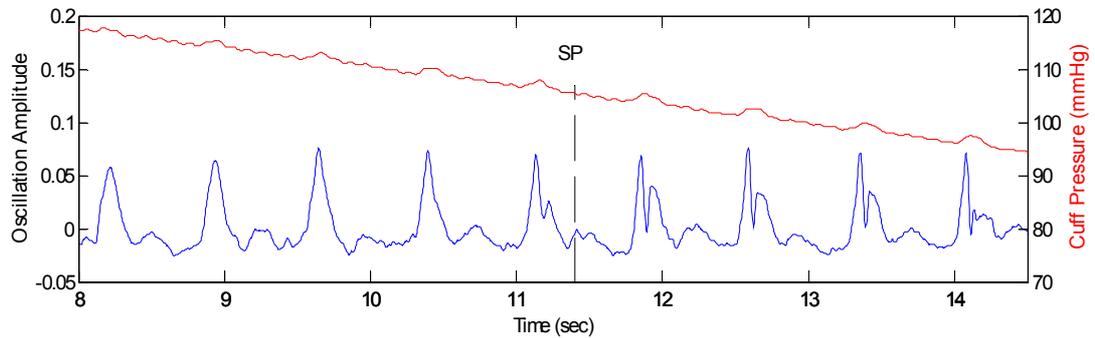
(a) Signal changes from supra-systolic pressure region to systolic pressure region.
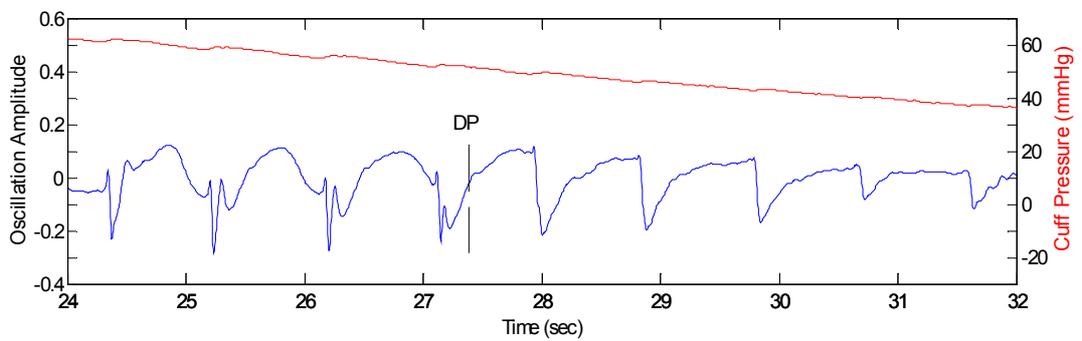
(b) Signal changes from diastolic pressure region to sub-diastolic pressure region.

Figure 3.1.1 Measured signal from the inside sensors (blue), pressure sensor (red) and the blood pressure estimated from observers (vertical dashed line).

3.2   Signal Processing

The purpose of raw data pre-processing was to eliminate unnecessary signals and strengthen the relevant features. As described in Chapter 2, the raw data was first measured from the subject's brachial artery through an amplifier to enlarge the measured signal before passing on to the data acquisition card to convert the analogue signal to digital. The data acquisition card was installed with its running software, LabView, in the computer. The measured data signals were stored in a text formatted spreadsheet which was created by LabView.

The MATLAB program was used for the rest of the signal analysis process. Typical signals measured from the new developed cuff are shown in Figure 3.2.1 for a cuff deflation cycle. The inside and outside sensor signals shown have been band-pass filtered using fourth order Butterworth digital filter with corner frequencies at 0.25 and

30 Hz and applied zero-phase digital filtering to ensure zero-phase distortion. The signal obtained from the inside sensors was the voltage difference between two sensors, DT2 and DT1. The pressure sensor signal shown in Figure 3.2.1(c) has been low-pass filtered in both directions using the third order Butterworth digital filter with corner frequency at 10 Hz. The pressure values were estimated by applying the polynomial coefficients generated from the cuff calibration linear regression.

(a) Inside Sensor

(b) Outside Sensor

(c) Pressure Snesor
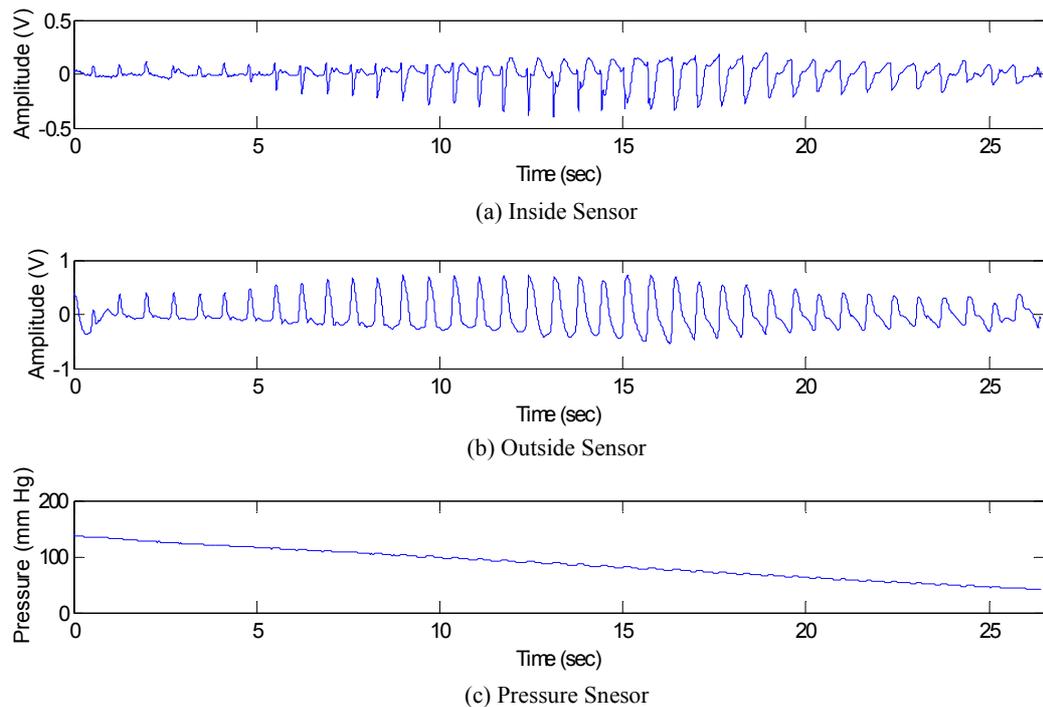
Figure 3.2.1 An example of signal data from subject 1, recording 2.

In all cases the outside sensor produced a signal similar to an oscillometric plot. The amplitude increased as cuff pressure approached the mean blood pressure and decreased thereafter. The outside sensor appears relatively noise free and provides a good signal for identifying each HB. The HB detection method will be described in the following section. Each individual HB was then windowed using the Hanning window, where the window length was about 70 ~ 80% of the mean beat rate from the start of the beat.

A number of algorithms were developed to estimate the blood pressure either from each HB waveform or the overall trend of the measured signals.

3.3   Heart Beat / Rate Determination

The outside sensor signal as shown in Figure 3.2.1(b) was used for identifying each HB. A second order Butterworth Low-Pass Filter (LPF) was applied initially to the signal to filter out high frequency signals at corner frequency of 2 Hz. All the minimum turning points (troughs) from the low-pass filtered signal were detected. Each HB starting point was then defined as the maximum point (peak) of the initial signal between two troughs. The HR was calculated as one over the time difference between two peaks in beats per second, i.e. equation (3.1). Mean HR was calculated as the total HR divided by the total number of beats, i.e. equation (3.2).

$$ HR = \frac{1}{Time(2) - Time(1)} \tag{3.1} $$

$$ mean\ HR = \frac{\sum HR}{No.\ of\ beats} \tag{3.2} $$

HB detection algorithm did not always perfectly match the peak point. Subjects with faster or slower HR would end up with error detection. Error checking was carried out and the LPF corner frequency was adjusted if the detected HB did not pass the error checking. The first stage of the error checking algorithm was to calculate the mean HR from the whole signal excluding the first and the last three HB. The algorithm checked through each detected HB, if any of the HR was either 1.6 times greater or 0.5 times lesser than the mean HR, the HB with the smaller amplitude value was eliminated. The new HR of each HB and the mean HR was calculated. The new HR and mean HR results passed to the next stage of the checking algorithm. This part only checked the length of the first and the last three HB. If any of the first three HR was either 1.6 times greater or 0.5 times lesser than the mean HR, the HB and the HB before it was eliminated, e.g. if the second HB failed the test condition, both the first and second HB was eliminated. The same principle applied to the last three HB. This procedure removed the noise signal obtained during the data collection. If the result had only one HB left, the corner frequency of the LPF applied on the outside signal was reduced by 0.1 Hz. Then the whole HB detection and the error checking procedures had to be carried out again. Otherwise, the program leads to the last stage of the checking algorithm which checked the length of the HB.

The shortest HB length must be greater than the length of the Hanning window. The Hanning window length was firstly set at 80% of the mean HB length. If the shortest HB length was lesser than the length of the Hanning window, the Hanning window length was decreased by 1%. The length of the Hanning window kept decreasing by 1% until the shortest HB length was greater or equal to the length of the Hanning window or the length of the Hanning window was lesser than 70% of the mean HB length. If the shortest HB length was greater than the Hanning window length, the HB detection was finalised. Otherwise, the corner frequency of the LPF applied on the outside signal was reduced by 0.1 Hz. Then the whole HB detection and the error checking procedures was carried out again. If the corner frequency decreased to 0.5 Hz and none of the HB detection procedures passed the error checking, an error message was displayed and the program stopped. When the HB detection finalised, each starting point of the HB was shifted by a fixed length forward to ensure the main pulse was covered within the detected HB. The shifting length was set at half of the mean HR. An example result is shown in Figure 3.3.1. The vertical straight lines in Figure 3.3.1(a) represent the final HB detected from the measurement. The dashed lines in Figure 3.3.1(b) represent the HB detected after shifting the starting point. The flowchart of the HB / HR determination program is shown in Figure 3.3.2.
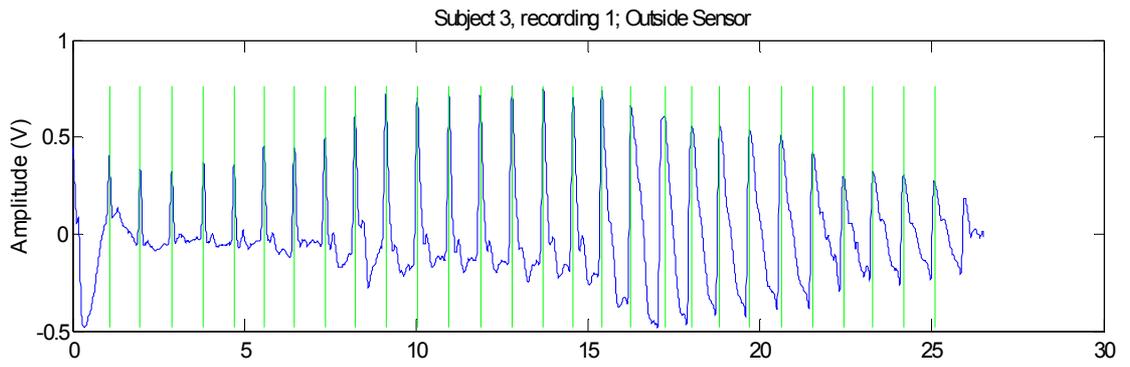
(a) HB detected before shift
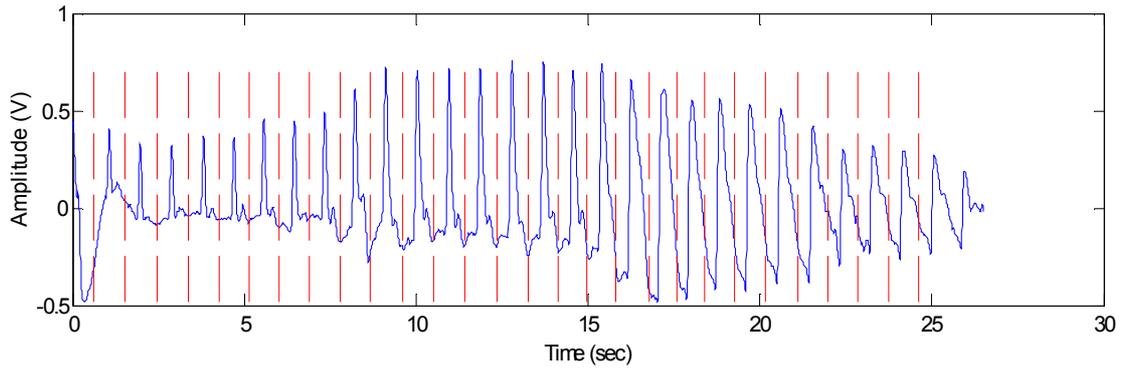
(b) HB detected after shift

Figure 3.3.1 HB determination result from outside sensor.
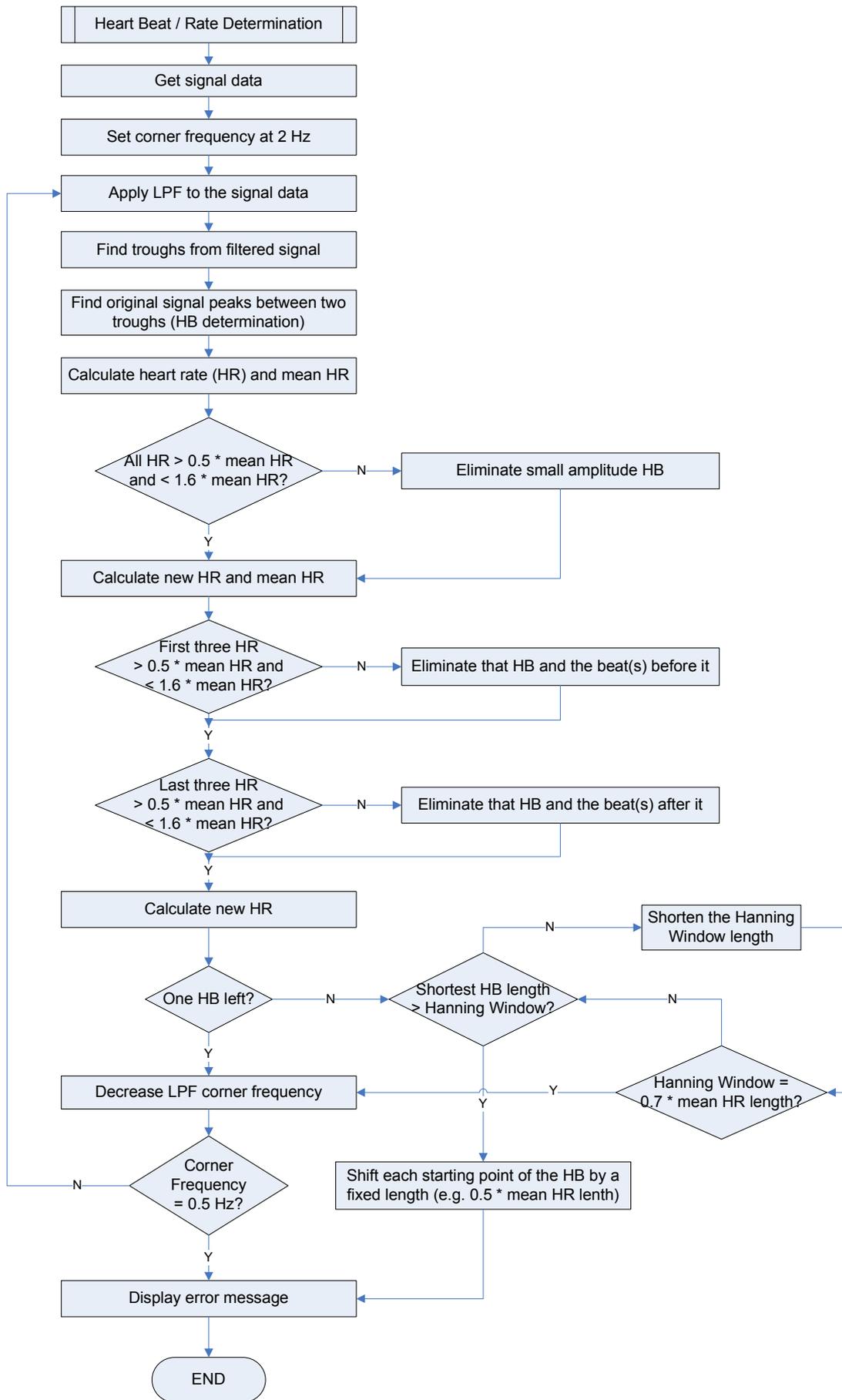
Figure 3.3.2 MATLAB development flowchart for heart beat / rate determination.

3.4   Pressure Selection

After the HB was defined, the corresponding pressure value was from each HB. The cuff pressure value was measured from the pressure sensor. The length of each HB corresponds to a range of pressure values. The pressure oscillation signal was similar to the outside sensor. A pressure was selected at the point on the upstroke of the pressure oscillation signal.

The selection program was implemented in MATLAB. The pressure sensor signal shown in Figure 3.4.1 has been band-pass filtered in both directions using the fourth order Butterworth digital filter with corner frequencies at 0.25 and 30 Hz. Pressure values have been divided based on the range of individual HB. All troughs from the first point of the HB range to the maximum point of each oscillation signal have been found. If only one trough has been found, that corresponding point would be selected as the pressure value. If more than one trough has been found and the minimum point of the oscillation signal was less than zero, the minimum point would be selected as the pressure value. Otherwise, the program would select the last trough point as the pressure value. The reason for selecting the minimum point will be discussed in the discussion section in Chapter 5, page 106. Figure 3.4.2 shows an example of the result of selected pressures which are given numerically.
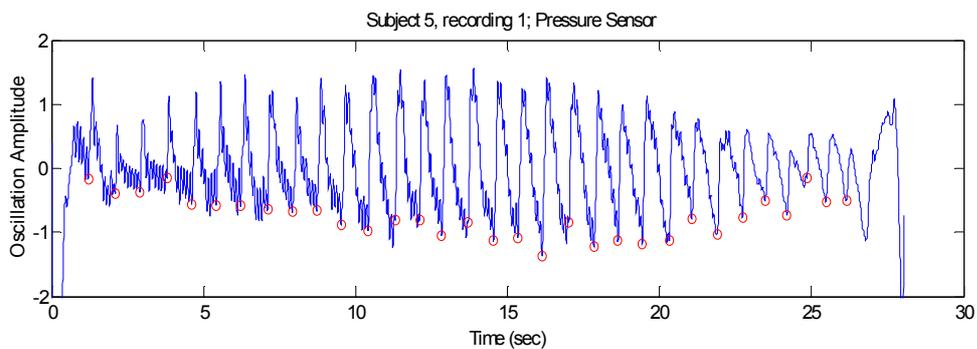
Figure 3.4.1 Blue line is the pressure sensor signal after band-pass filtered. Red circles indicate the pressure selected point of each HB.
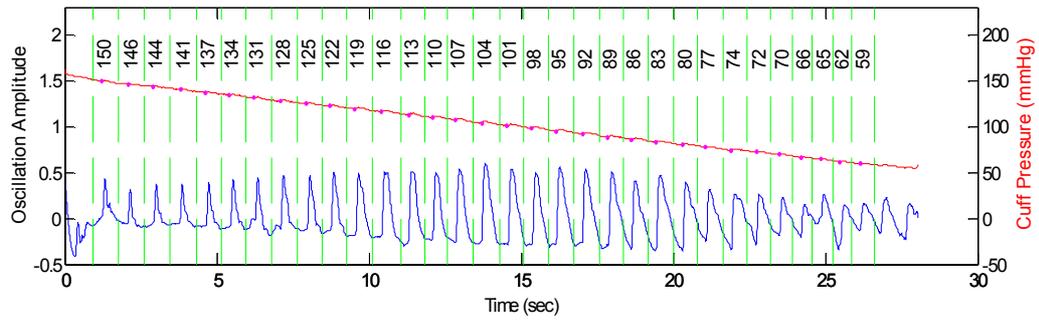
Figure 3.4.2 Blue line is the outside sensor oscillating signal. Green dashed lines indicate the start of each HB. Selected cuff pressures for each HB are given numerically.

3.5 Height-based Algorithm

Height-based algorithm was developed from the idea of the traditional method used in the Oscillometric method. This method has been discussed in Chapter 1. This algorithm calculated the amplitude of each HB. The amplitude was calculated by taking the difference between the maximum and minimum points of each HB. Amplitudes were then normalised to the maximum amplitude. SP and DP values were selected from the preset ratio compared to the maximum amplitude.

SP and DP ratios were defined from the measured results. Auscultatory results obtained from two observers were used to find the average ratio where the SP and DP occurred. The average ratio to its maximum amplitude was calculated and stored. Twenty-seven measurements were used to find the average ratios for this algorithm. Three signal data (inside, outside and pressure sensors) from each measurement were tested separately as the input source. SP was selected at the pressure where the oscillation amplitude was 60% of the maximum oscillation for the inside sensor signal and 70% for both the outside and the pressure sensor signals. DP was selected at the pressure where the oscillation amplitude was 40% of the maximum oscillation for the inside sensor signal and 60% for both the outside and the pressure sensor signals. The ratio taken for this algorithm was based on the Auscultatory results.

The same twenty-seven measurements were tested by applying the preset ratio for each signal to estimate the BP for each subject. The flowchart of this algorithm is shown in Figure 3.5.1. The results were compared with the AAMI and BHS standards to check the accuracy of this algorithm.

47
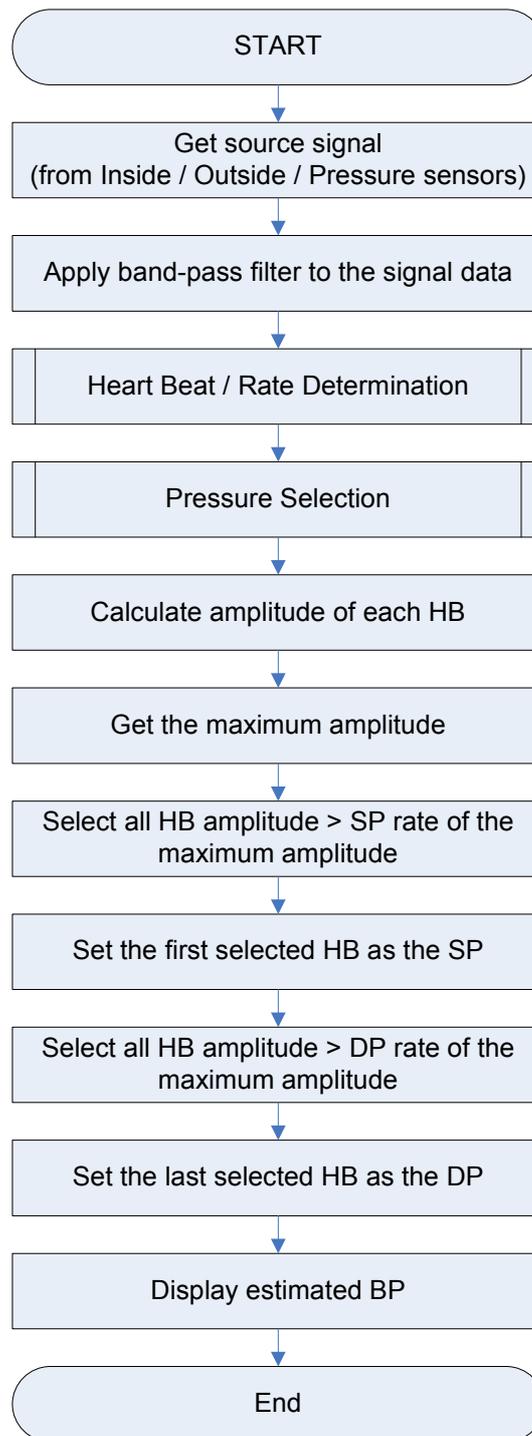
Figure 3.5.1 Height-based algorithm flowchart.

3.6 STFT Algorithm

The STFT provides the information of the magnitude of each segment signal at different time and frequency. Spectrogram function in MATLAB was used to calculate the magnitude, frequencies, time and PSD of each segment signal. Arguments in the function were set 100 for the length of Hamming window, 50% overlap between

segments, 256 for FFT length and the sampling frequency for the function. An example of 2D graph Time-Frequency graph is shown in Figure 3.6.1. The colour represents the magnitude of each segment signal at a different time and frequency. It clearly shows that during the time between 10 ~ 35 seconds the magnitudes were higher than at the other times at the frequencies between 10 ~ 30 Hz. BP was estimated at the edge where the magnitude changed significantly. 3D Time-Frequency, frequency between 10 ~ 35 Hz, and the magnitude graph is shown in Figure 3.6.2(a) and 3D Time-Frequency and PSD graph is shown in Figure 3.6.2(b). Both the average values at each time segment of frequency range between 10 ~ 35 Hz was calculated. The averaged values were normalised with the largest value to get a ratio plot as shown in Figure 3.6.3. It was clearer to see the changes of the PSD than the magnitude of the segment signal. A threshold was set at 30% of the signal and the first point above the threshold from the averaged PSD values as the systolic blood pressure and the last point as the diastolic blood pressure was selected. The flowchart of this algorithm is shown in Figure 3.6.4. This algorithm was tested on all sensor signals and the results were compared with the AAMI and BHS standards to check the accuracy.
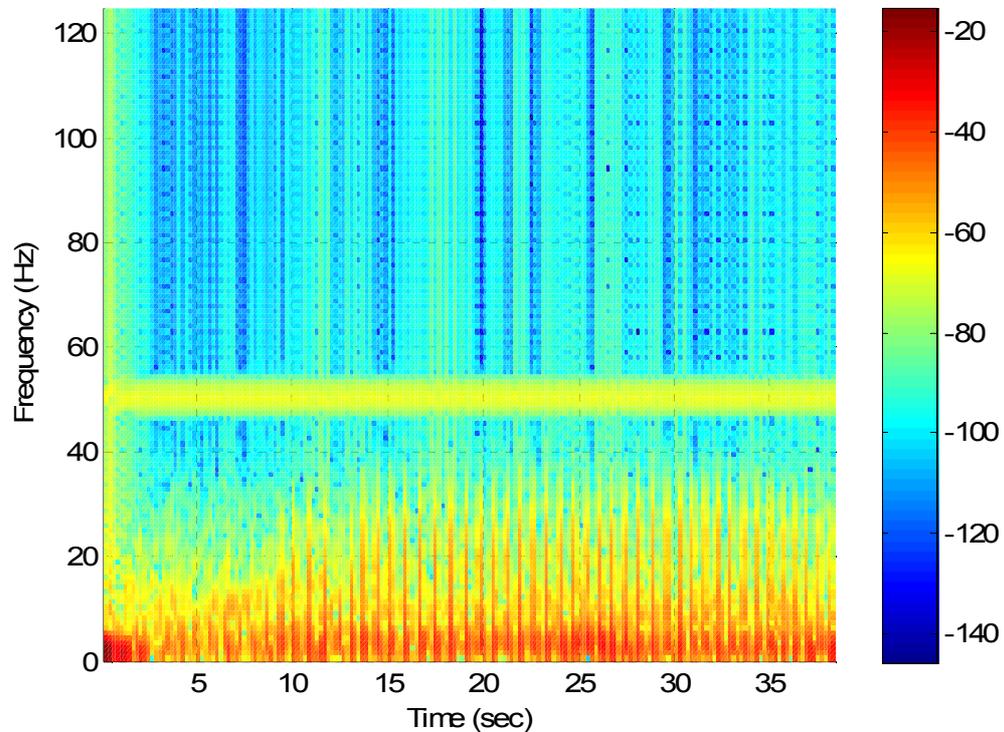
Figure 3.6.1 Two-dimensional Time-Frequency plot for subject 9, recording 1.
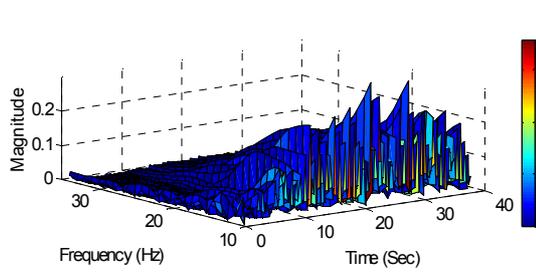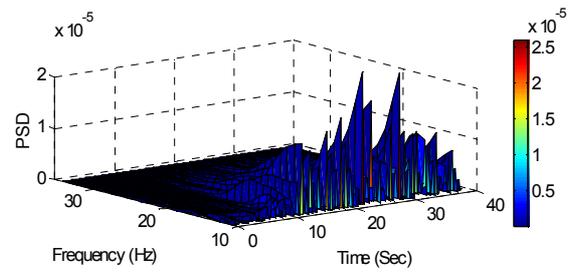
(a) 3D Time-Frequency vs Magnitude Graph (b) 3D Time-Frequency vs PSD Graph

Figure 3.6.2 Three-dimensional STFT plot for subject 9, recording 1. Frequency range from 10 ~ 35 Hz.
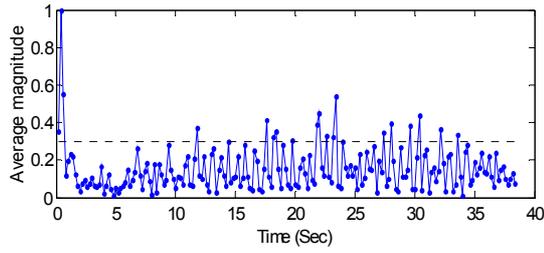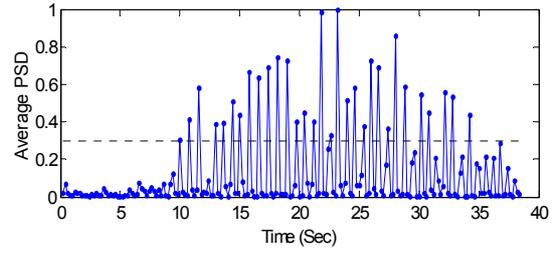
(a) 3D Time-Frequency vs Magnitude Graph (b) 3D Time-Frequency vs PSD Graph

Figure 3.6.3 Averaged values at each time segment at frequency range from 10 ~ 35 Hz.

```
                    ┌─────────────────────┐
                    │        START        │
                    └─────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │      Get source signal       │
                │ (from Inside / Outside /     │
                │      Pressure sensors)       │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │ Apply band-pass filter to    │
                │      the signal data         │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │ Apply spectromgram function  │
                │  to get time, frequencies,   │
                │ magnitude and PSD matrices   │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │ Set frequency range          │
                │     (e.g. 10 ~ 35 Hz)        │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │ Calculate mean PSD of each   │
                │ time segment within the      │
                │      frequency range         │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │ Normalised each mean PSD     │
                │          value               │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │    Set threshold (e.g. 0.3)  │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │ Select the first point where │
                │ the mean PSD was greater     │
                │ than threshold as the SP     │
                └─────────────────────────────┘
                              │
                              ▼
                ┌─────────────────────────────┐
                │ Select the last point where  │
                │ the mean PSD was greater     │
                │ than threshold as the DP     │
                └─────────────────────────────┘
                              │
                              ▼
                    ┌─────────────────────┐
                    │         END         │
                    └─────────────────────┘
```

Figure 3.6.4 STFT algorithm flowchart.

3.7 Artificial Neural Network Classification

As mentioned at the beginning of this chapter, the measured signals changed throughout the measurement from the supra-systolic pressure region to the sub-diastolic pressure region. An ANN classifier was developed to classify each detected HB.

Raw signal data was pre-processed by following the procedure described in section 3.2. Each HB and HB pressure was determined as described in section 3.3 and 3.4. Twenty-four features were extracted from each detected HB. The detail of individual features will be described in the next section. The extracted features were used as the input of

the ANN. The structure of the ANN and the training function was decided. Eighteen measurements were used for training the ANN and nine measurements were used for testing the trained ANN. SP and DP were selected from the output of the ANN. BP selection method will be described in detail. This algorithm was tested on all sensor signals and the results were compared with the AAMI and BHS standards to check the accuracy.

3.7.1 Input Features Extraction

Features were extracted from each detected HB and obtained from both frequency and time domain. Average magnitude and PSD was extracted from the frequency domain. Total amplitude of each turning point, area under the curve, peaks and rate of change (RoC) were extracted from the time domain. Individual feature extraction functions were developed in MATLAB. Functions were named by its function purpose. These functions returned results to the main program and saved results into an Excel spreadsheet for ANN training and testing purpose.

Features obtained from the frequency were similar to the STFT algorithm. An individual HB signal was sent to the *funMag* and the *funPSD* functions. The Hanning window was applied to each HB. FFT function was used to calculate magnitudes and PSD of the signal in different frequencies. The average magnitude and PSD values of each HB at frequency range between 5 ~ 35 Hz, 10 ~ 35 Hz, 15 ~ 35 Hz, 20 ~ 35 Hz and 25 ~ 35 Hz were calculated and returned to the main program. The *funMag* and the *funPSD* flowcharts are shown in Figure 3.7.2(a) and Figure 3.7.2(b), respectively.

FFT function was applied to each HB and then applied to the Hanning window to emphasise the main pulse and force the signal outside the pulse range to zero. The signal outside the detected HB range was set to zero too. A new signal waveform was formed. An absolute maximum value was found from the new signal. Every point was normalised to that maximum value, so that the negative signal remained negative. Features extracted from the time domain were then carried out afterward.

funAmp function was developed to calculate the total amplitude of each HB. Each turning point was found and the amplitudes from point to point were calculated. The sum of the amplitudes was calculated and returned to the main program. The flowchart of *funAmp* is shown in Figure 3.7.2(c).

funArea function was developed to calculate the area under the curve of each HB. Integration can be used to compute the area under a signal function [51] and the sum of the trapezoidal area approximates the integral of the function. The trapezoidal numerical integration (trapz) function was used and the result was returned to the main program. The flowchart of *funArea* is shown in Figure 3.7.2(d).

funPeak function was developed to count the peaks and troughs at different thresholds. Thresholds were set at ±0.1, ±0.3, ±0.5 and ±0.7. Total peaks were counted above each positive threshold value and total troughs were counted below each negative threshold. Eight values were obtained for each HB and returned to the main program. An example is shown in Figure 3.7.1. The counted peaks and troughs are shown at the right hand side of the plot. The flowchart of *funPeak* is shown in Figure 3.7.2(e).
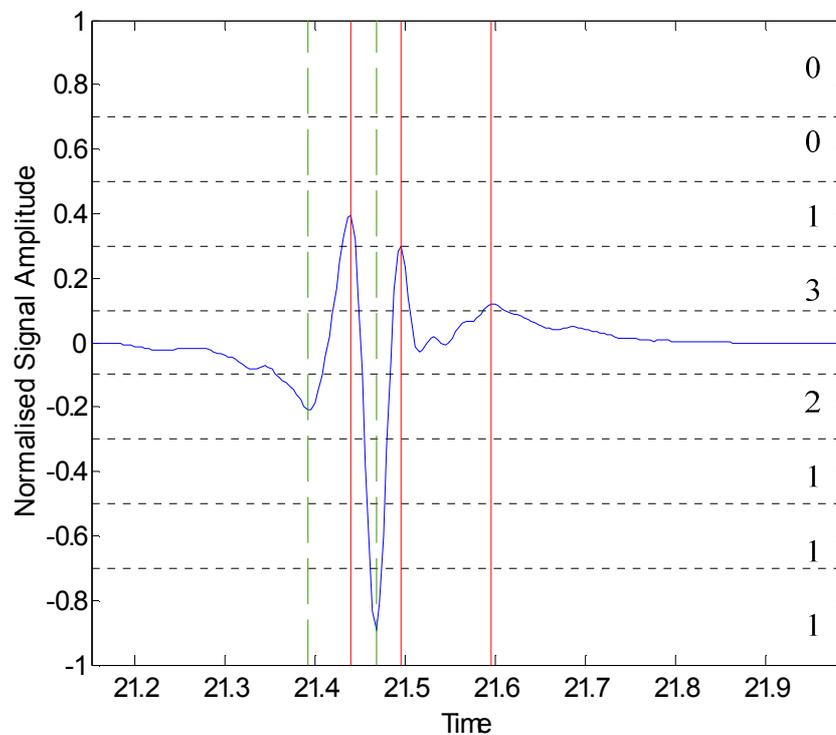
Figure 3.7.1 Peaks and troughs counting example. Blue line represents one HB signal. Red vertical lines are the peaks above the positive threshold 0.1 and green dashed lines are the troughs below the negative threshold -0.1. Counted peaks and troughs are given numerically at the right hand side.

funROC was developed to calculate the maximum positive and the maximum negative RoC of each HB. Each turning point was found and the amplitudes from point to point were calculated. RoC was calculated as the amplitude divided by the change of time as shown in equation (3.3). Amplitude calculated from peak minus trough gives a positive RoC and trough minus peak gives a negative RoC. The maximum positive and the

maximum negative RoC values were returned to the program. The flowchart of *funROC* is shown in Figure 3.7.2(f).

$$\text{Rate of Change} = \frac{\Delta y}{\Delta x} = \frac{\text{Change in amplitude}}{\text{Change in time}} \qquad (3.3)$$

fundPdt function was developed to calculate the maximum positive and the maximum negative slope of each HB. It was very similar to the *funROC* function. The only difference between them was the function *fundPdt* calculated the RoC at each data point instead of calculating the RoC at turning point to turning point. The maximum positive and the maximum negative slope values were returned to the program. The flowchart of *fundPdt* is shown in Figure 3.7.2(g).

A total of 24 feature values, 10 values from the frequency domain and 14 values from the time domain, were extracted from each HB and saved into an Excel spreadsheet for ANN training and testing purpose.
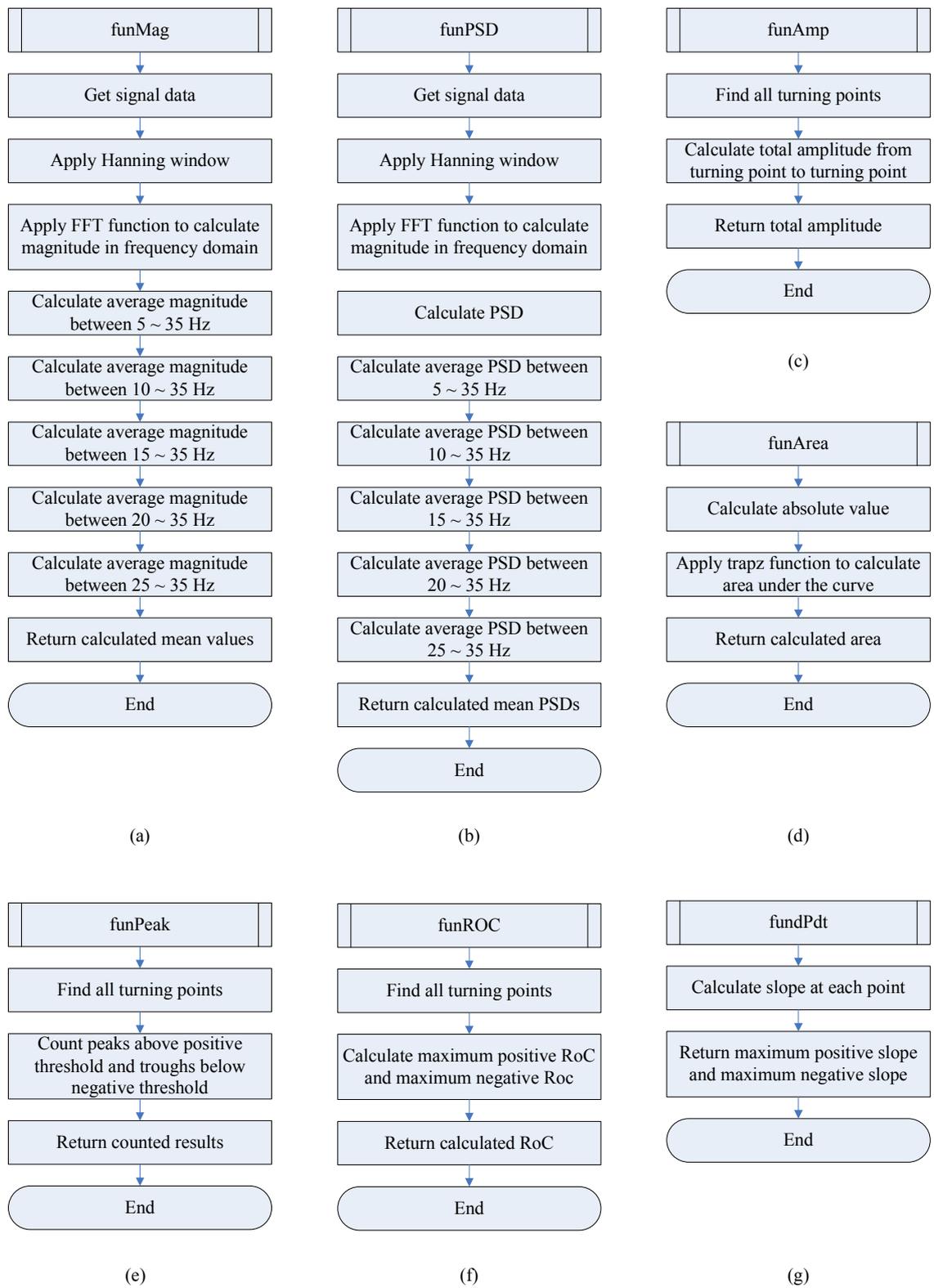
Figure 3.7.2 Feature extraction functions.

3.7.2 Design and Training of the Neural Network

As mentioned in 2.3.1, an ANN needs to go through the training process and adjust weights until the network output matches the target. There are four steps in the training process [56].

1 Assemble the training data – the feature inputs extracted from each HB

2 Design the network object – design and initialise the neural network

3 Train the network – modify weights

4 Simulate the network – compare the output and target values by applying new input data.

Data Assembling

The data were required to be organised according to the architectural structure of the neural network program. Input and output matrices were gathered for ANN training. For the input matrix, each HB with 24 features was gathered in a column vector. The total number of columns represented the total number of HB entered the ANN. Each HB has an output vector corresponded to its blood pressure region, either Supra-Systolic, Systolic to Diastolic or Sub-Diastolic region. The target vector of each region is shown in Figure 3.7.3 and the structure of the input and output matrices are shown in Figure 3.7.4.

$$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \qquad \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

(a) Supra-Systolic Region   (b) Systolic to Diastolic Region   (c) Sub-Diastolic Region

Figure 3.7.3 Target vector of each pressure region.

No	Features	Subject 1				Subject 2				...	Subject n			
	Time Domain	HB 1	HB 2	...	HB n	HB 1	HB 2	...	HB n	...	HB 1	HB 2	...	HB n
1	Total amplitude													
2	Area under the curve													
3	Max. + dP/dt													
4	Max. - dP/dt													
5	Peak < -0.1													
6	Peak < -0.3													
7	Peak < -0.5													
8	Peak < -0.7													
9	Peak > 0.1													
10	Peak > 0.3													
11	Peak > 0.5													
12	Peak > 0.7													
13	Max. + RoC													
14	Max. - RoC													
	Frequency Domain													
15	Mag. 5 ~ 35 Hz													
16	Mag. 10 ~ 35 Hz													
17	Mag. 15 ~ 35 Hz													
18	Mag. 20 ~ 35 Hz													
19	Mag. 25 ~ 35 Hz													
20	PSD 5 ~ 35 Hz													
21	PSD 10 ~ 35 Hz													
22	PSD 15 ~ 35 Hz													
23	PSD 20 ~ 35 Hz													
24	PSD 25 ~ 35 Hz													

Target output vector

1			0
0			0
0			1

....

1			0
0			0
0			1

Output vector
depends on the
pressure region
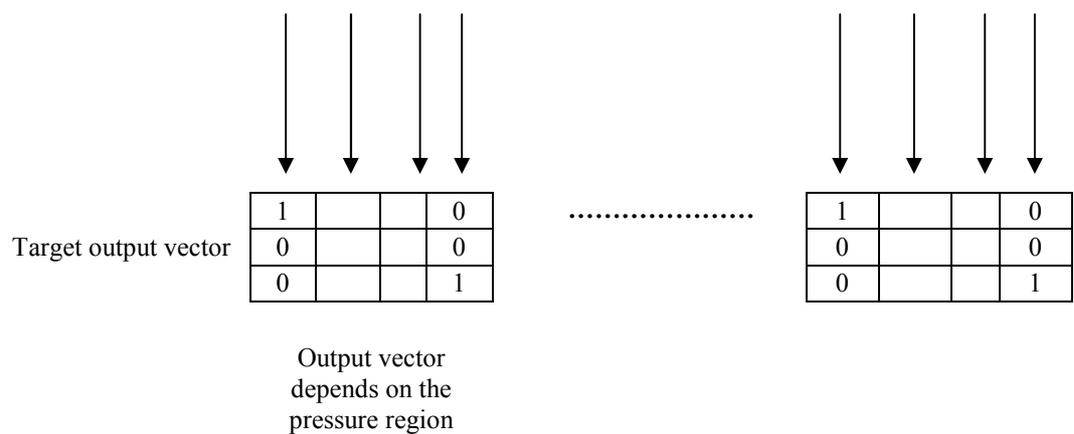
Figure 3.7.4 Input and output matrices for the ANN.

57

Designing and Training the Neural Network

An input layer, a hidden layer and an output layer was constructed as shown in Figure 3.7.5. Input matrix of all features extracted from each HB was inserted in the input layer. These values were the data used for training in the ANN. The hidden layer in the ANN consisted a number of neurons and a tan-sigmoid transfer function. The number of neurons (S^l) in the hidden layer was selected from 1 to 30. The tan-sigmoid transfer function calculated output value in the range between -1 and 1. The output values of the hidden layer were used as the input values for the output layer. A log-sigmoid transfer function was used to calculate output value in the range between 0 and 1. A random seed (0) was selected for the initialization of the weights and bias. newff function was used to create a trainable feedforward network and initialize the weights and biases of the network [56].
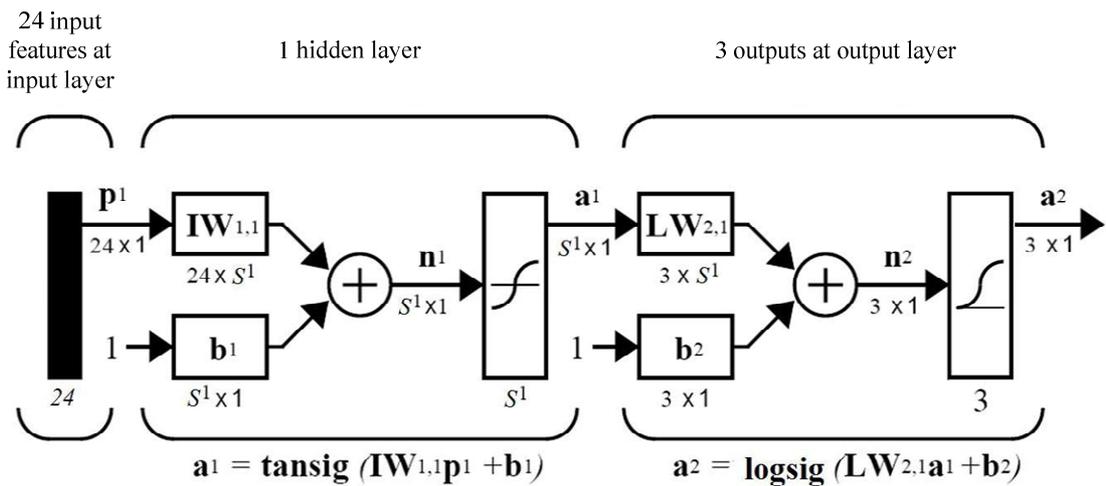
Figure 3.7.5 Network architecture built for ANN classification algorithms.

Once the network weights and biases have been initialized, the network is ready for training. Levenberg-Marquardt backpropagation training function was selected to train the ANN in a faster way. Termination criteria Parameters were set as follows:

- No. of epochs or iterations: 500
- Performance Error Goal (PEG):   0.001 (0.1% error)
- Maximum performance gradient: 1×10^{-10}

During the training process, the ANN iteratively trained the network and modified the weights until one of the termination parameter reached the set value. 30 different numbers of neurons were used to construct 30 different ANNs, which were trained on each sensor signals and saved for simulation. The output result was compared to the target and the error between them was calculated. Any difference between the output vector and the target output vector would be counted as one error. Such error counting would be continued over the entire data set. The sum of counted errors divided by the total number of vectors would be used to calculate the percentage error for both training and testing data sets. A good ANN should have both training and testing errors as low as possible and as close as possible.

3.7.3   Blood Pressure Selection Method

Blood pressures were selected from the output vector of the ANN. Every output contained values between 0 and 1. An example of the target and the output values simulated by an ANN is shown in Figure 3.7.6. The output values almost perfectly matched the target. Another example in Figure 3.7.7 shows that the Supra-systolic and Sub-diastolic output were not as clear as the output between SP to DP. Therefore, blood pressures were selected from the SP to DP output only. int8 function was used to convert the output vector to a vector of signed 8-bit integers [51]. It means all values above 0.5 will become 1 and others are 0. From the designed target all pressure between SP and DP should have second row of output close to 1. The algorithm selected all the 1s from the second row of the output. In some situations, large pulses caused by body motion were recorded at the beginning and/or at the end of the measurement. Therefore, the first three HB with consecutive output of 1s were picked and the first HB from that three was selected for SP. The last three HB with consecutive output of 1s were picked and the last HB from that three was selected for DP. An example is shown in Table 3.7.1 for algorithm BP selection.
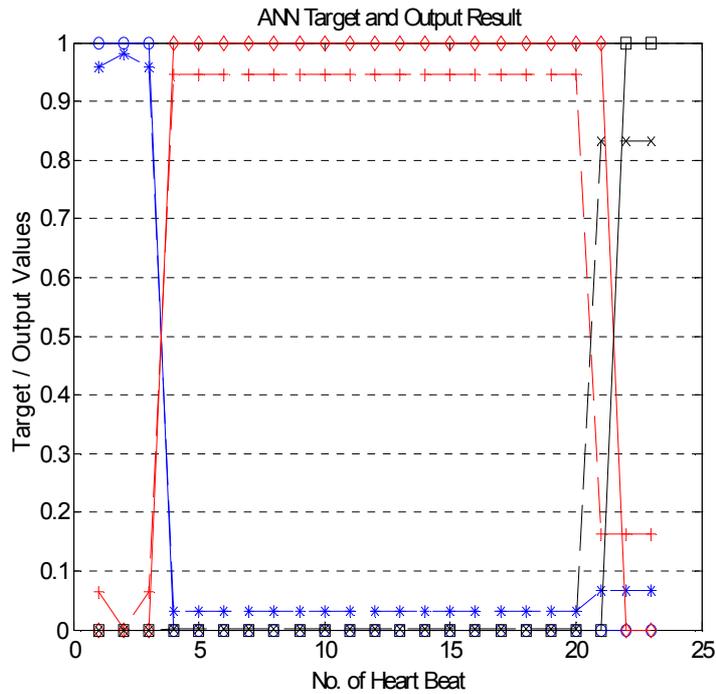
Figure 3.7.6 Example of an ANN simulated output for subject 4, recording 3. The symbol ○ with the blue solid line represents the target output for the Supra-SP region. The symbol * with the blue broken line represents the actual output from the ANN for the Supra-SP region. The symbol ◊ with the red solid line represents the target output between the SP and DP regions. The symbol + with the red broken line represents the actual output between the SP and DP regions. The symbol □ with the black solid line represents the target output for the Sub-DP region. The symbol × with the black broken line represents the actual output from the ANN for the Sub-DP region.
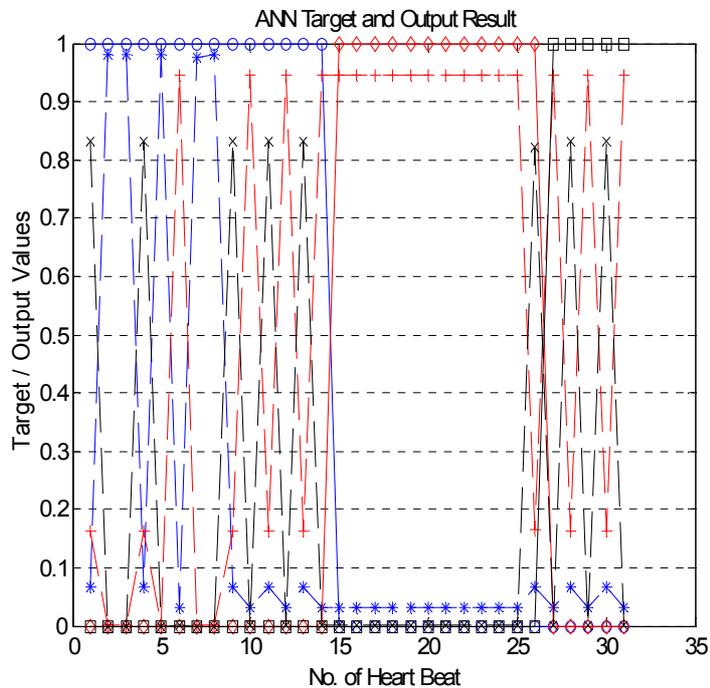
Figure 3.7.7 Example of an ANN simulated output for subject 7, recording 3. The symbol ○ with the blue solid line represents the target output for the Supra-SP region. The symbol * with the blue broken line represents the actual output from the ANN for the Supra-SP region. The symbol ◊ with the red solid line represents the target output between the SP and DP regions. The symbol + with the red broken line represents the actual output between the SP and DP regions. The symbol □ with the black solid line represents the target output for the Sub-DP region. The symbol × with the black broken line represents the actual output from the ANN for the Sub-DP region.

Table 3.7.1 ANN classification algorithm BP selection. Yellow boxes highlight the target HB for BP selection. Blue boxes highlight the algorithm selected HB for BP selection.

	Supra-Systolic Region			Systolic to Diastolic Region					Sub-Diastolic Region		
HB No.	1	2	3	4	5	6	7	8	9	10	11
Target Output	1	1	1	0	0	0	0	0	0	0	0
	0	0	0	1	1	1	1	1	0	0	0
	0	0	0	0	0	0	0	0	1	1	1
ANN Output	0.86	0.73	0.47	0.66	0.38	0.21	0.46	0.3	0.11	0.32	0.24
	0.52	0.4	0.53	0.46	0.78	0.89	0.67	0.51	0.65	0.23	0.01
	0.1	0.35	0.34	0.21	0.15	0.34	0.41	0.73	0.61	0.58	0.75
Converted ANN Output	1	1	0	1	0	0	0	0	0	0	0
	1	0	1	0	1	1	1	1	1	0	0
	0	0	0	0	0	0	0	1	1	1	1

3.8 Algorithm Validation Result Comparison

Result from each algorithm was presented in Bland and Altman plots and a table to show the mean, SD and the percentage of the measurement error. The validation result which Passed/Failed the AAMI protocol and the grades obtained according to the BHS protocol were also included in the table. The algorithm will be considered for further development if it passed the AAMI criteria and received a grade of A or B under the BHS protocol for both systolic and diastolic blood pressures.

3.8.1 Height-based Algorithm

Figure 3.8.1 to Figure 3.8.3 show the Bland Altman plot to compare Auscultatory and Height-based algorithm results by using different sensor signals. Table 3.8.1 demonstrates the result compared to the AAMI and BHS standard protocols.
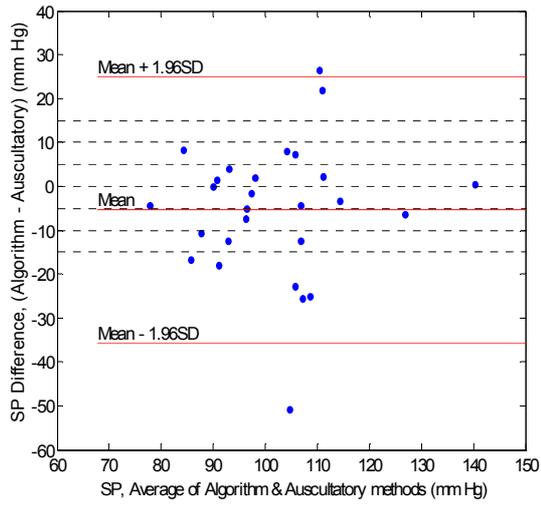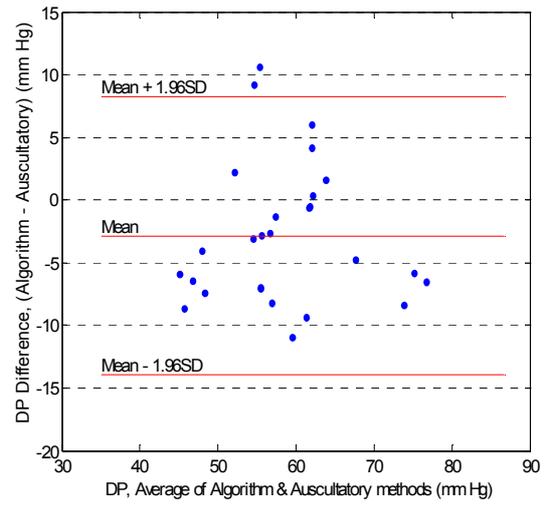
(a) Systolic Blood Pressure
(b) Diastolic Blood Pressure

Figure 3.8.1 Blood pressure estimation from inside sensor signal and Auscultatory result comparison.
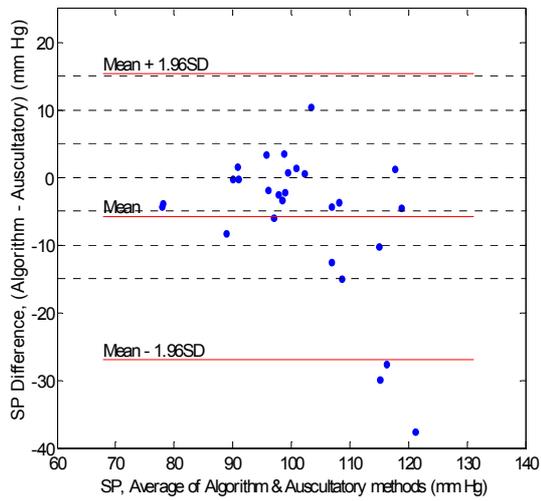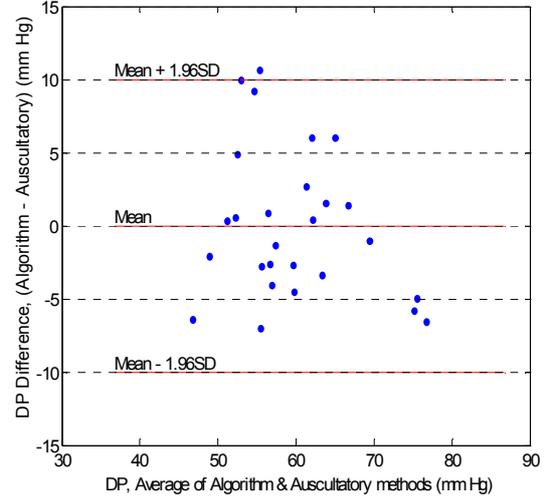
(a) Systolic Blood Pressure
(b) Diastolic Blood Pressure

Figure 3.8.2 Blood pressure estimation from outside sensor signal and Auscultatory result comparison.

(a) Systolic Blood Pressure   (b) Diastolic Blood Pressure
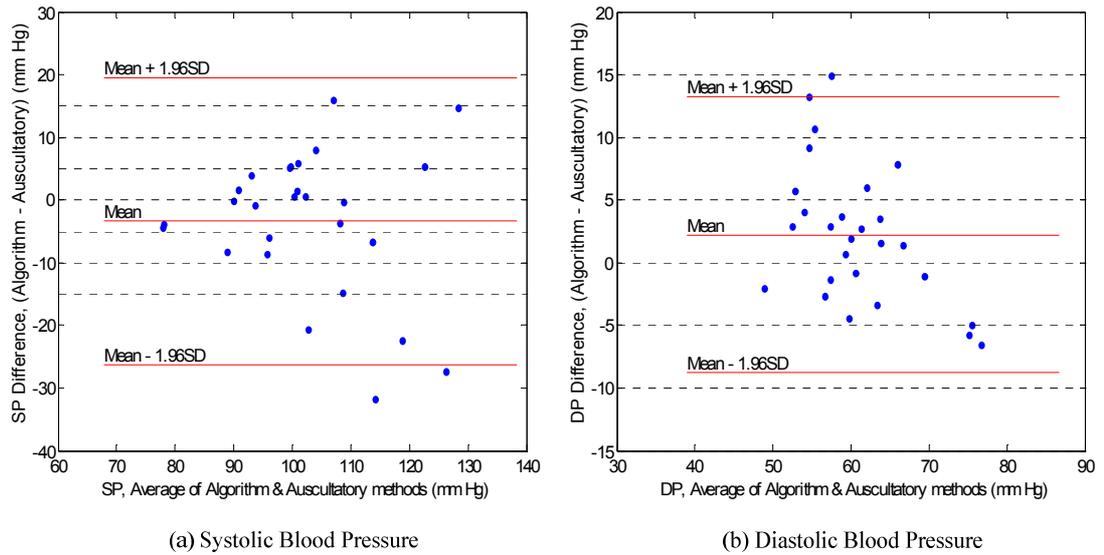
Figure 3.8.3 Blood pressure estimation from pressure sensor signal and Auscultatory result comparison.

Table 3.8.1 Height-based algorithm result compared to the standard protocols.

Height based	Systolic Pressure					Diastolic Pressure					Standard (SP / DP)	
	Measurement Error		Absolute difference (%)			Measurement Error		Absolute difference (%)			AAMI	BHS
Sensor	mean	SD	$\leq \pm 5$	$\leq \pm 10$	$\leq \pm 15$	mean	SD	$\leq \pm 5$	$\leq \pm 10$	$\leq \pm 15$	Pass/Fail	Grades
Inside	-5.33	15.44	37.04	59.26	70.37	-2.85	5.65	44.44	92.59	100	F/P	D/C
Outside	-5.73	10.79	66.67	74.07	88.89	-0.02	5.1	66.67	96.3	100	F/P	C/A
Pressure	-3.38	11.67	40.74	74.07	81.48	2.22	5.61	66.67	88.89	100	F/P	C/A

Result from Table 3.8.1 shows that Height-based BP measurement algorithm failed both the AAMI and BHS standard protocols. Only DP measurement from the outside and the pressure sensors passed these two standard protocols. This algorithm used fixed percentile rule to estimate SP and DP values. The selected ratios for SP and DP would not be able to fit all subjects. Therefore, this algorithm was not selected for future development. But the amplitude of each HB could be a useful feature in the future development.

3.8.2 STFT Algorithm

Figure 3.8.4 to Figure 3.8.6 show the Bland Altman plot to compare Auscultatory and STFT algorithm results by using different sensor signals. Table 3.8.2 demonstrates the result compared to the AAMI and BHS standard protocols.
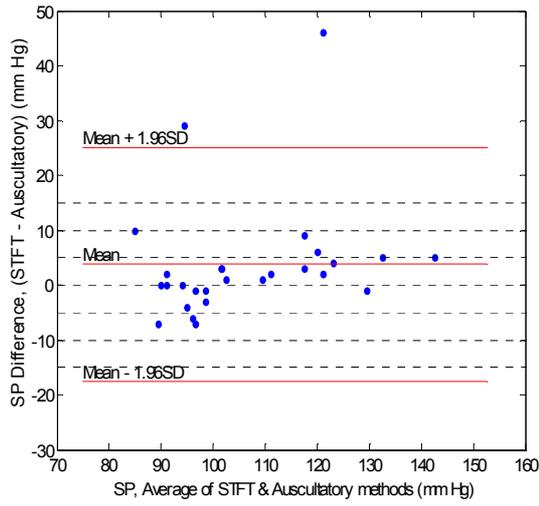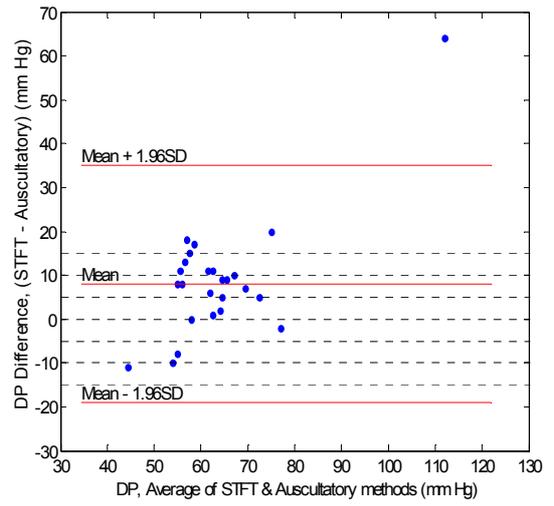
(a) Systolic Blood Pressure (b) Diastolic Blood Pressure

Figure 3.8.4 Blood pressure estimation from inside sensor signal and Auscultatory result comparison.
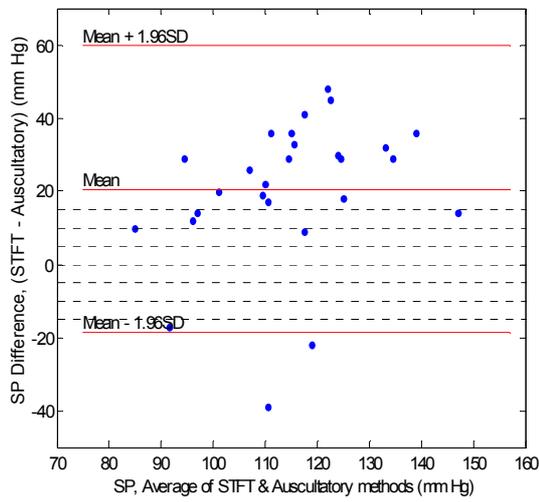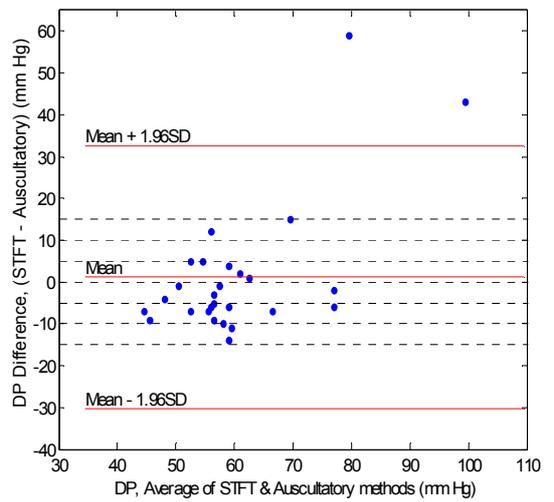
(a) Systolic Blood Pressure (b) Diastolic Blood Pressure

Figure 3.8.5 Blood pressure estimation from outside sensor signal and Auscultatory result comparison.

(a) Systolic Blood Pressure

(b) Diastolic Blood Pressure
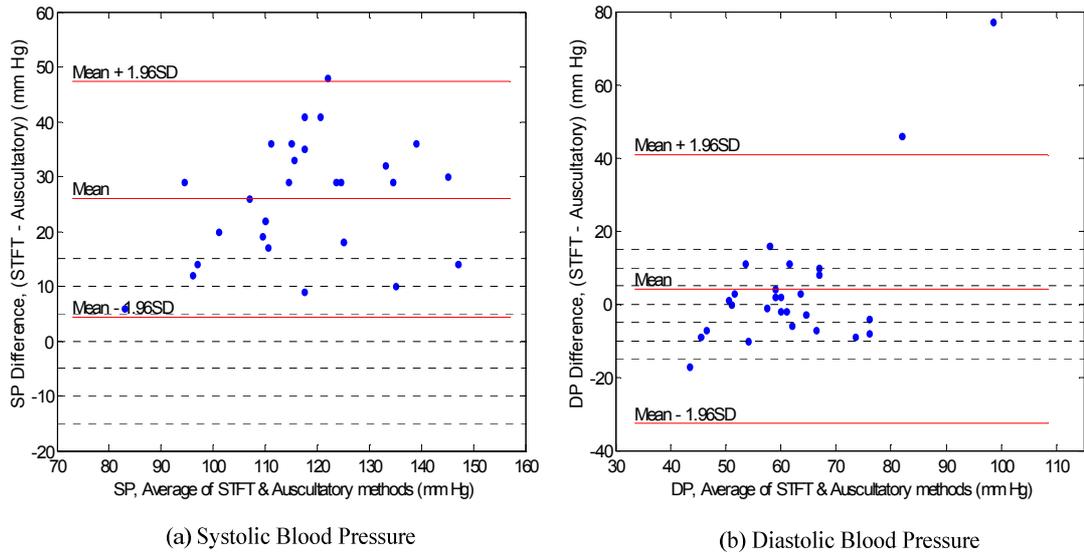
Figure 3.8.6 Blood pressure estimation from pressure sensor signal and Auscultatory result comparison.

Table 3.8.2 STFT algorithm result compared to the standard protocols.

STFT	Systolic Pressure					Diastolic Pressure					Standard (SP / DP)	
	Measurement Error		Absolute difference (%)			Measurement Error		Absolute difference (%)			AAMI	BHS
Sensor	mean	SD	$\leq \pm 5$	$\leq \pm 10$	$\leq \pm 15$	mean	SD	$\leq \pm 5$	$\leq \pm 10$	$\leq \pm 15$	Pass/Fail	Grades
Inside	3.74	10.86	70.37	92.59	92.59	8.04	13.79	29.63	62.96	85.19	F/F	B/D
Outside	20.59	19.97	0	7.41	18.52	1.15	16.02	40.74	77.78	92.59	F/F	D/C
Pressure	25.93	10.98	0	11.11	22.22	4.04	18.7	44.44	77.78	85.19	F/F	D/C

Result from Table 3.8.2 shows that STFT BP measurement algorithm failed both the AAMI and BHS standard protocols. Only SP measurement from the inside sensor signal passed the BHS standard protocol. Figure 3.8.5(a) and Figure 3.8.6(a) show that SP estimation from both outside and pressure sensors were overestimated for about 20 to 25 mm Hg. A possible reason of this might be the unstable signal recorded during the first few seconds of data collection. After the STFT algorithm was applied, a large value would be calculated and selected as the SP. An example is shown in Figure 3.6.3(a). Therefore, this algorithm was not selected for future development.

3.8.3 ANN Classification

Figure 3.8.7 shows the training and testing errors tested from different sensor signals with a different number of neurons in each ANN. Total of 629 HBs (83 HBs from the Supra-systolic region, 453 HBs from the systolic to diastolic region and 93 HBs from

the Sub-diastolic region) from 18 measurements were used for training the ANN and 273 HBs (51 HBs from the Supra-systolic region, 182 HBs from the systolic to diastolic region and 40 HBs from the Sub-diastolic region) from 9 different measurements were used for testing. A total of 30 ANNs were tested on each sensor signal. Table 3.8.3 shows a summary result of the number of ANNs that passed each standard protocol. Figure 3.8.8 to Figure 3.8.10 show the Bland Altman plot of the best ANN from different sensor signals to compare ANN classification and Auscultatory algorithm results. Table 3.8.4 is the best ANN result compared to the AAMI and BHS standard protocols.

(a) Inside Sensor Signal

(b) Outside Sensor Signal

(c) Pressure Sensor Signal

(d) Testing Errors Comparison
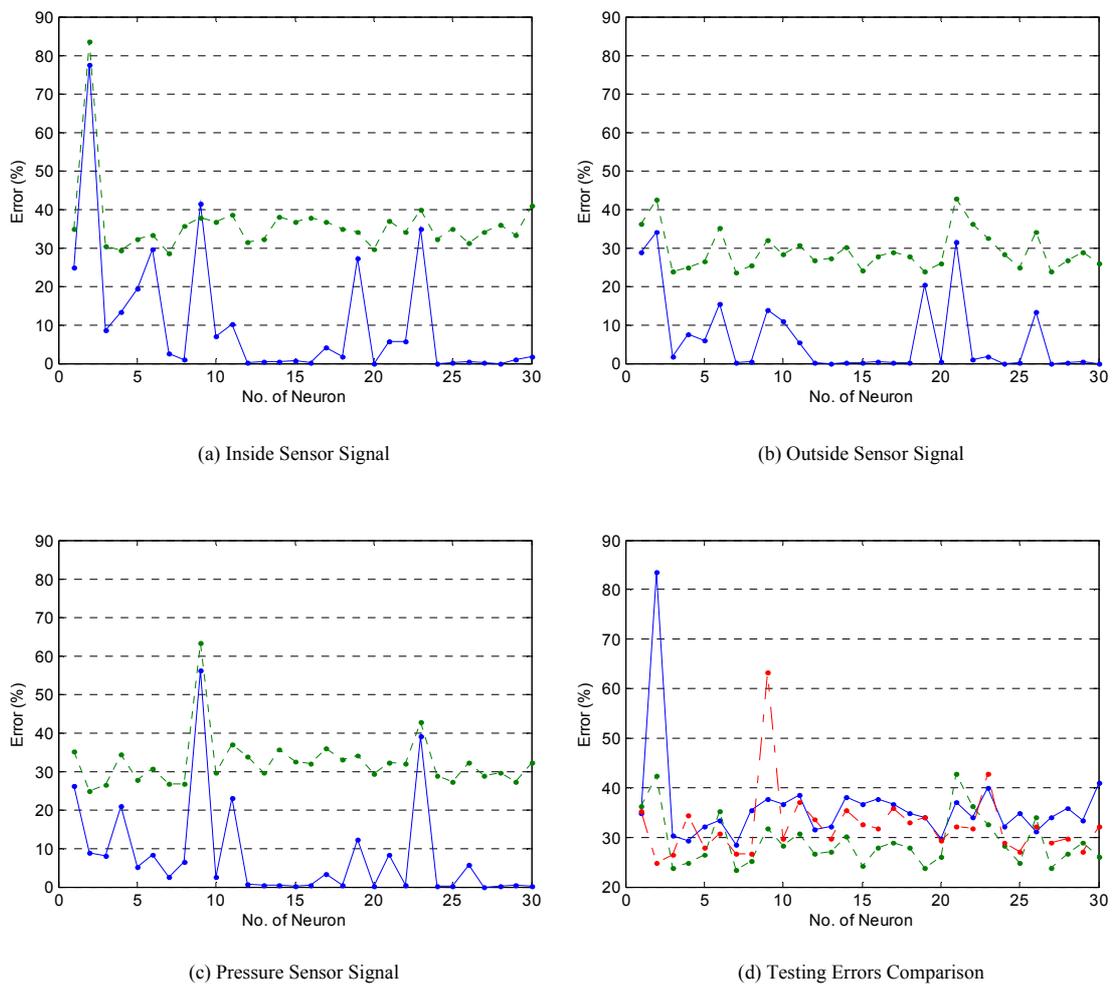
Figure 3.8.7 (a) to (d) show the training and testing errors from different sensor signals. The blue solid line represents the percentage of the training errors. The green dotted line represents the percentage of the testing errors. (d) shows the comparison of the testing error between different sensors. The blue solid line represents the percentage of the testing errors from the inside sensor signal. The green dotted line represents the percentage of the testing errors from the outside sensor signal. The red dash-dot line represents the percentage of the testing errors from the pressure sensor signal.

Table 3.8.3 Number of ANNs that passed the standard protocols.

Sensor	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B	
	SP	DP	SP	DP	SP	DP
Inside	0	10	0	4	0	13
Outside	1	13	0	4	0	12
Pressure	23	30	15	29	4	0

(a) Systolic Blood Pressure　　　　　(b) Diastolic Blood Pressure
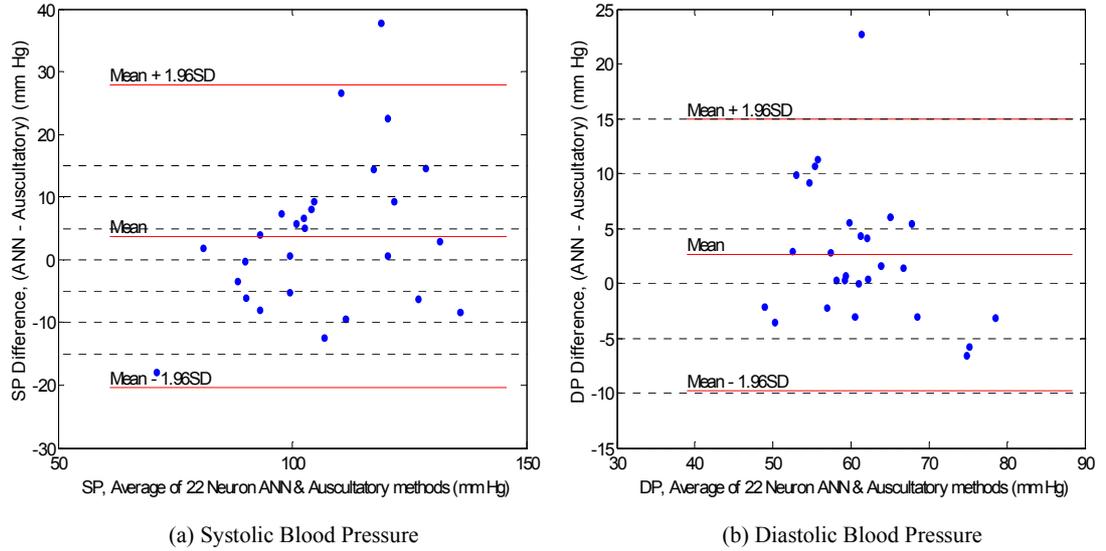
Figure 3.8.8 Blood pressure estimated from inside sensor signal. Bland and Altman plot comparing ANN with 22 neurons and Auscultatory result.

(a) Systolic Blood Pressure　　　　　(b) Diastolic Blood Pressure
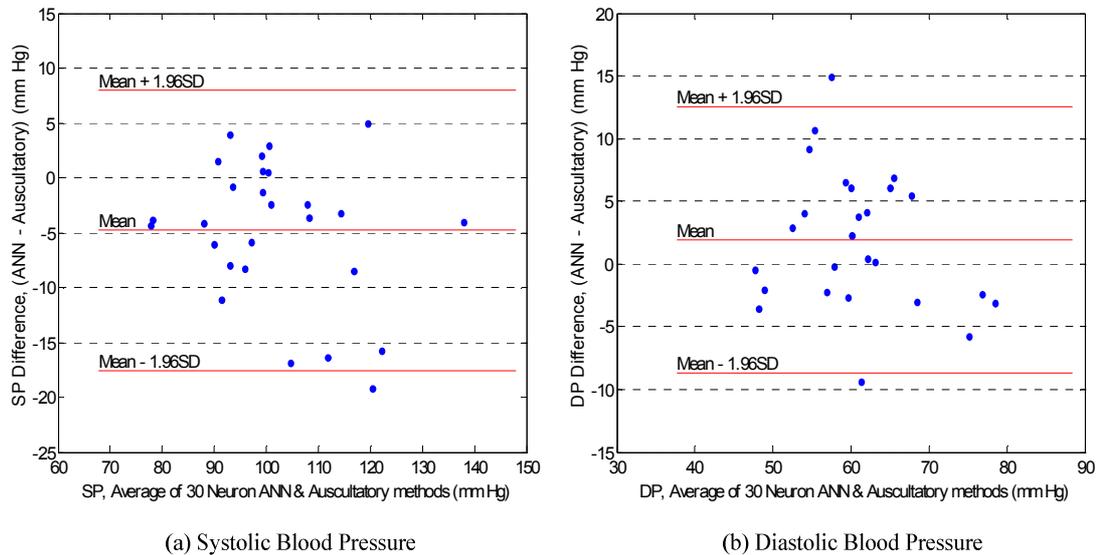
Figure 3.8.9 Blood pressure estimated from outside sensor signal. Bland and Altman plot comparing ANN with 30 neurons and Auscultatory result.

(a) Systolic Blood Pressure   (b) Diastolic Blood Pressure
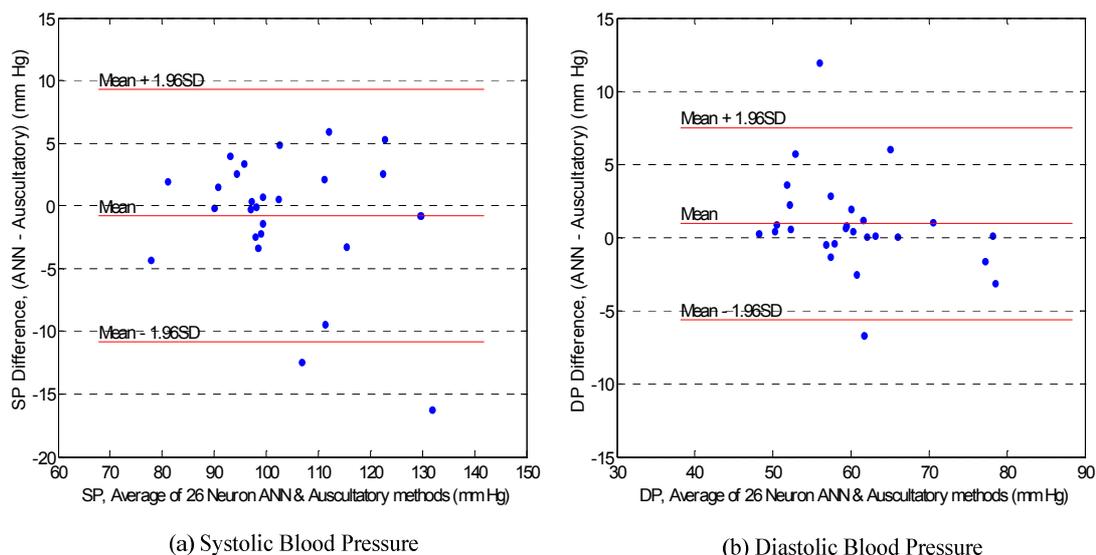
Figure 3.8.10 Blood pressure estimated from pressure sensor signal. Bland and Altman plot comparing ANN with 26 neurons and Auscultatory result.

Table 3.8.4 Best ANN algorithm result compared to the standard protocols.

Best ANN	Systolic Pressure					Diastolic Pressure					Standard (SP / DP)	
	Measurement Error		Absolute difference (%)			Measurement Error		Absolute difference (%)			AAMI	BHS
Sensor (No. of neurons)	mean	SD	$\leq \pm 5$	$\leq \pm 10$	$\leq \pm 15$	mean	SD	$\leq \pm 5$	$\leq \pm 10$	$\leq \pm 15$	Pass/Fail	Grades
Inside (22)	3.74	12.31	25.93	74.07	85.19	2.61	6.31	62.96	88.89	96.3	F/P	D/A
Outside (30)	-4.8	6.54	62.96	81.48	85.19	1.89	5.41	62.96	92.59	100	P/P	C/A
Pressure (26)	1.22	4.61	66.67	100	100	1.28	5.58	66.67	88.89	100	P/P	A/A

Figure 3.8.7 shows that all the differences between training and testing errors from each ANN are too large. This means that the ANNs were over-trained by the training data. ANNs tend to match the training data perfectly, so the ANNs cannot identify the new data. Result from Table 3.8.3 shows that many ANNs tested on the pressure sensor passed both the AAMI and BHS standard protocols. Also the best ANN tested result shown in Table 3.8.4 gives a good result compared to the other two algorithms, i.e. Height-based and STFT. Therefore, ANN classification algorithm was selected for future development.

3.9 Summary

Three algorithms, Height-based, STFT and ANN classification, were developed by applying signal processing techniques. Inside, outside and pressure sensor signals were tested on these three algorithms. A total of 3 measurements were taken from each of 9 subjects (27 measurements) and these were used to validate these algorithms. Results obtained by each algorithm were compared to the AAMI and BHS standard protocols. ANN classification algorithm was selected for further development. More measurements from new subjects were required to validate the ANN algorithm. The ANN algorithm was further modified for faster and more accurate classification.

Chapter 4 Algorithm Modification, Validation and

Finalisation

4.1   Introduction

This chapter covers the methods used to modify the selected algorithm (ANN classification) to get a better performance. Validations of the modified algorithms were tested using the new collected data before a final ANN was selected. Every modified algorithm was tested and compared to the standard protocols. The details of the modification, validation and finalisation procedures are described in this chapter.

From the results of the three algorithms presented in Chapter 3, outside sensor signals produced a better result than the other two signals. Although more ANNs passed the AAMI and BHS standard protocols on the test of the pressure sensor signals, the overall ANN testing error for the outside sensor signals were lower than the pressure sensor signals. In consideration of the time required for ANN classification procedures for training, testing and BP estimation to validate the result of each measured signals, the outside sensor signal was selected for further testing.

New measured data from 9 subjects, (each subject contributed 3 measurements,) were collected for the algorithm modification and validation. An algorithm which produced the best result would be selected for further testing. Then a total of 76 subjects, involving 228 measurements, was collected and tested on the selected algorithm. The BPM algorithm was kept modified until the result of the final algorithm met the requirements of the standard protocols. The final algorithm was tested on a total of 86 subjects, involving 258 measurements, to pass the AAMI minimum requirement of 85 subjects.

4.2 Algorithm Modification

This research project focused on three main modifications of the ANN classification algorithm. This section describes the changes and modifications of the algorithm in detail. The first modification focused on the improvement of the existing ANNs. From the simulated output shown in Figure 3.7.6 and Figure 3.7.7 in Chapter 3, most of the output values were either greater than 0.8 or lesser than 0.2. The goal of the first modification was to train the ANN so that the simulated output had some transitional values. The second modification of the algorithm was to generate a second ANN after the first trained ANNs. The second ANN was trained to select the right HB for the BP estimation. The last modification was designed to get a better result by changing some of the features in the first ANN and using a different combination of the training inputs for the second ANN.

4.2.1 ANN Improvement

Results from the ANN training and testing errors in Chapter 3 showed that all the ANNs were over-trained. Those errors on the training set had a very small value, but the error became large when new data was tested on the network. A network that is simple (such as less neurons, features, etc) yet can complete the task without overfitting the data and with less error would be ideal. A few solutions for the training problem of ANNs are listed below:

- ➢ Reset the initial network weights and biases with different values
- ➢ Reduce the number of hidden layer neurons
- ➢ Reduce number of training epochs (iterations)
- ➢ Increase performance error goal (PEG) value
- ➢ Decrease the number of input features
- ➢ Use different training functions
- ➢ Increase the number of training vectors

Uniformly distributed pseudorandom numbers (rand) function was used to generate pseudorandom numbers. It was used to set the initial network of weight and bias values. This function initialized the state of a generator by entering a scalar integer value from 0 to 2^{32}-1. A value of zero was used to set the generator to its default initial state. Numbers ranging from 0 to 100 were used to initialize 101 different weight and bias

matrices for ANNs.

Numbers used for hidden layer neurons were ranged from 1 to 15 to generate 15 different structures of ANN. The lesser the hidden layer neurons, the shorter the training time. The goal was to get the smallest number of hidden layer neurons that produced good training and testing results.

The training epoch was set at 100 to shorten the training iteration. The training epoch was adjusted to check the difference between the training and testing results. If the result did not change much, then the smaller epoch value was selected.

The PEG value was set between 0.001 and 0.2. Training and testing errors were calculated and compared. The PEG value was selected when the training and testing error values were low and the difference between them were small.

The numbers of input features was reduced from 24 to 6, 12 and 18 inputs. The weight values of each trained ANN were checked. The weight values of each ANN were shown on an m-by-n matrix, where m is the number of neurons and n is the number of inputs. The average weight values of each input from the trained ANNs were calculated. Six input features were selected from the 6 highest weight values compared to the other inputs. Those features were the total amplitude, the area under the curve, the positive peaks above threshold 0.1 and 0.3, the maximum negative RoCand the average PSD value at a frequency range of 25 ~ 35 Hz of each HB. 12 input features would include the 6 inputs plus another 6 inputs which were the maximum negative slope, the positive peaks above threshold 0.7, the negative peaks below threshold 0.3, the average magnitudes at a frequency range of 5 ~ 35 and 10 ~ 35 Hz and the average PSD value at a frequency range of 15 ~ 35 Hz of each HB. 18 input features would include the 12 inputs plus another 6 inputs which were the maximum positive slope, the positive peaks above threshold 0.5, the negative peaks below threshold 0.5, the average magnitudes at a frequency range of 15 ~ 35 and 20 ~ 35 Hz and the average PSD value at a frequency range of 10 ~ 35 Hz of each HB. New ANNs were trained by these new input data sets. Training and testing errors were recorded and compared to each other. The BP estimation results were compared to the standard protocols.

PCA was also used to reduce the dimension of the input data sets. As discussed in section 2.3.2, each data dimension needs to get a zero mean data set. The mapstd function was applied to calculate it. The processpca function was used to process

input data sets using PCA so that each row was uncorrelated. The rows which contributed least to the total variation in the data sets were removed. Figure 4.2.1 shows the number of components left after running the processpca function by selecting different contribution thresholds. The contribution threshold was selected at the point where the numbers of components dropped deepest but without losing too many important values. In this case the contribution threshold was set at 0.014. New ANNs were trained by the new input data sets. Training and testing errors were recorded and compared to other input data sets. The BP estimation results were compared to the standard protocols.
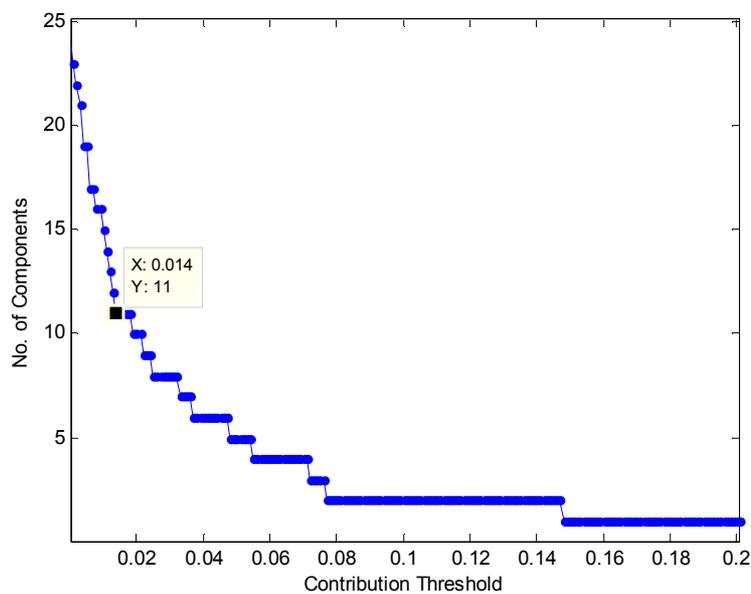
Figure 4.2.1 PCA dimension reduction result.

Different training functions were used to train the new ANNs. Levenberg-Marquardt backpropagation (Trainlm) training function was initially used. This function required less calculation speed and the performance function reduced at each iteration [56]. The training function Traingdx was used to update weight and bias values according to the gradient descent momentum and an adaptive learning rate. This function changes the learning rate to its optimal value during the training process. It takes more time than the other training functions. Another two training functions, Bayesian regularization (Trainbr) and BFGS Quasi-Newton backpropagation (Trainbfg), were used to improve the generalization of the ANNs. Trainbr is desirable to determine the optimal regularization parameters in an automated way. The Trainbfg algorithm modifies the typical performance function and the performance ratio to improve generalization for

training feedforward neural networks [56]. It generally converges in fewer iterations but requires more computation in each iteration. It is an efficient training function for smaller networks. New ANNs were trained by these training functions. Training and testing errors were recorded and compared to each other. The BP estimation results were compared to the standard protocols.

4.2.2   Second ANN

In all trained ANNs used so far, each HB has an output vector of three values representing the region of that HB. The BP estimation was decided from the second value of the output vector which represented the SP to DP region. The purpose of the second ANN was to utilise the output vector from the trained ANN as the input vector of the second ANN. The output of the second ANN has only one value to decide whether the HB lies within the SP to DP region. The pressure range of HB lay within and outside the SP to DP region represented by a target output of 1 and 0, respectively. Two networks, linear and feedforward backpropagation network, were tested and compared for the second ANN.

A single Adaptive Linear Neuron network (ADALINE) was used for the second ANN. The diagram for this network is shown in Figure 4.2.2. The weight matrix w has only one row. The network output is:

$$a = purelin(\mathrm{W}p + b) = w_{1,1}p_1 + w_{1,2}p_2 + w_{1,3}p_3 + b \qquad (4.1)$$

The output a was equal to the target output and the bias b equal to 0. The weight matrix w was calculated by taking the target output divided by the input matrix (output of the first ANN). newlin function was used to create a linear layer. The number of output was set to 1. The algorithm used was the same as BP selection method described in section 3.7.3 to select the HB for BP estimation. The results were compared to the standard protocols.

Input   Layer of linear neurons

$$a = purelin(\mathbf{W}p + b)$$
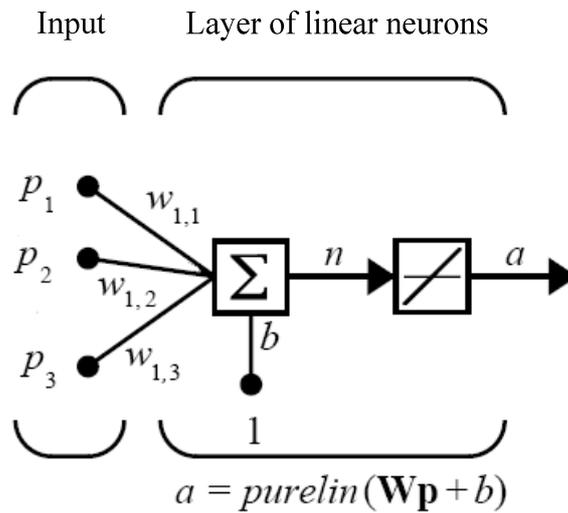
Figure 4.2.2 Adaptive Linear Neuron network architecture for 2nd ANN.

Another simple feedforward backpropagation network was used to construct the second ANN instead of a linear network. The network structure is similar to the structure as shown in Figure 16 in Chapter 3. The input matrix was constructed from the output of the first ANN. A single hidden neuron and a tan-sigmoid transfer function were used in the hidden layer. A log-sigmoid transfer function was used to calculate a single output value in the range between 0 and 1. A different random seed from the first ANN was selected for the initialization of the weight and bias. newff function was used to create the network and Trainbfg function was used for the training function.

4.2.3   Inputs Modification

Inputs for the first ANN

The original feature extractions for the first ANN inputs were described in section 3.7.1. The absolute maximum value of each measured signal was found from the processed signal. Every point was normalised to the maximum value for feature extractions in the time domain. If any one or two pulses were extremely larger than the other pulses, the significant features would not be able to be extracted after the normalisation procedure. Modifications of the inputs were made to improve this shortcoming.

Approximate upper and lower envelopes were generated for each signal by interpolating a piecewise polynomial (spline) function through the peaks and minimum points of each HB. The amplitude was calculated by taking the difference between the lower and

75

the upper envelopes of each point. The result of the envelope amplitudes is shown in Figure 4.2.3.
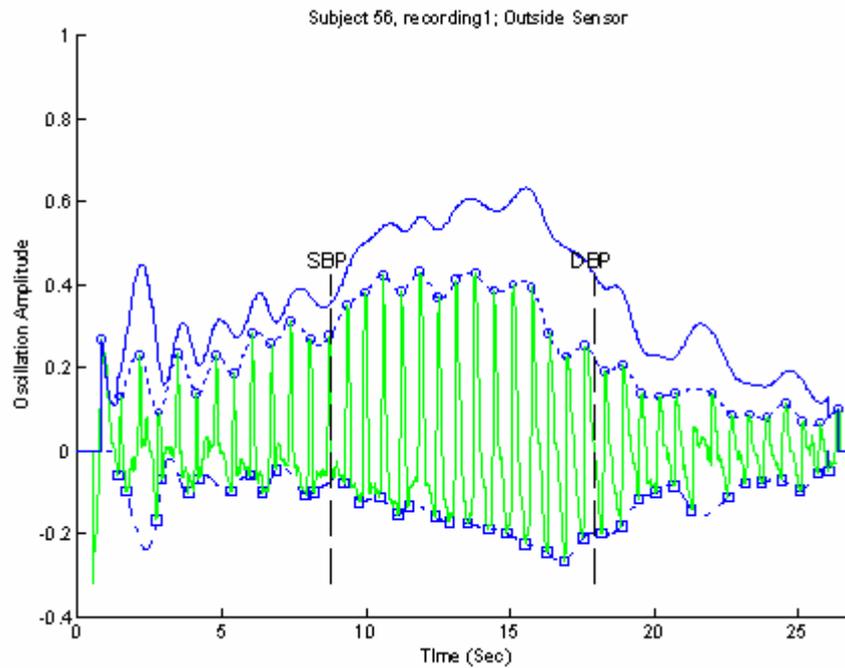
Figure 4.2.3 An example of envelope amplitude detection results from the outside sensor. The green solid line represents the measured signal. The vertical dashed lines represent the averaged SP and DP values measured by two observers. The symbol ○ represents the detected peaks of each HB. The blue dotted line represents the upper envelope through the peaks. The symbol □ represents the detected minimum points of each HB. The blue dash-dot line represents the lower envelope through the minimum points. The blue solid line represents the result of the envelope amplitudes.

Features extracted from the frequency domain were unchanged. Before the features were extracted from the time domain, the envelope amplitudes were calculated for each individual HB. The maximum envelope amplitude of each HB was used as one of the input features. The signal of each individual HB was then normalised by subtracting the minimum lower envelope and then divided by the maximum upper envelope of that HB. Then the Hanning window was applied and the rest of the features were extracted from the time domain.

After the normalisation of each HB, there were no more negative peaks existing. The features extracted from the time domain of each individual HB included the total amplitude, area under the curve, positive peaks from different thresholds, maximum positive and negative slopes, maximum positive and negative RoCand the envelope amplitude. The flowchart of features extraction is shown in Figure 4.2.4. ANNs were then trained and tested.
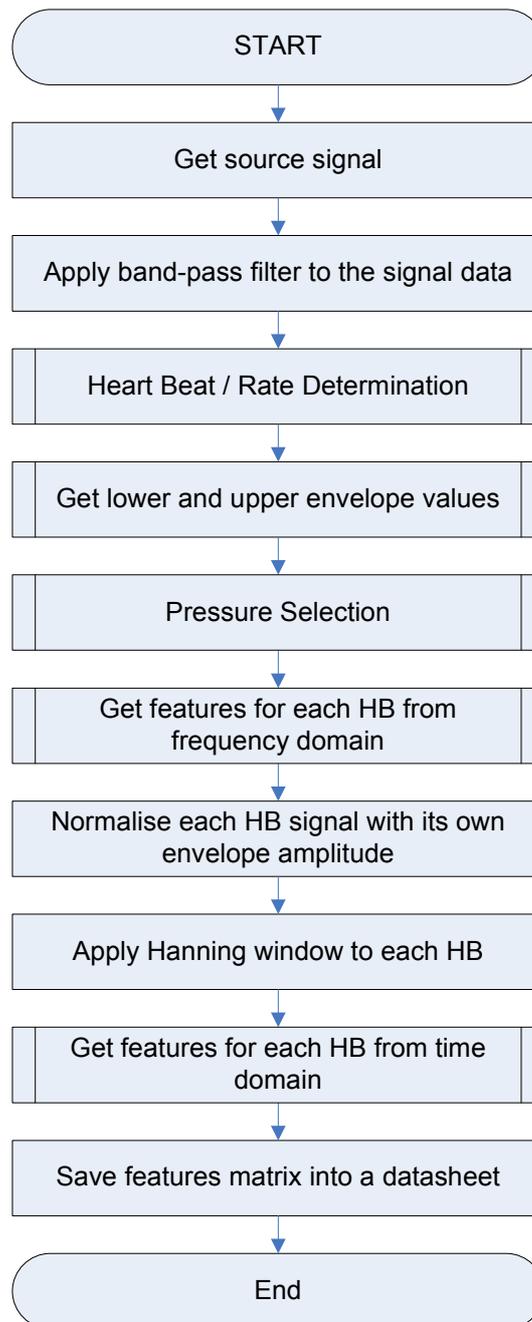
Figure 4.2.4 Features extraction flowchart

Inputs for the second ANN

The original input vector for the second ANN was obtained directly from the output vector of the first ANN. It was based on the decision of the first ANN of that HB. Modification was made to include the trend of the waveform by gather the three consecutive output vectors as the input for the modified second ANN. This resulted in a total of 9 inputs instead of 3. Table 4.2.1 shows the structure of the modified input vector.

Table 4.2.1 Input and output matrices for the 2nd ANN. The yellow colour represents the corresponding output for each HB. The blue, gray and green colours represent how the modified 2nd ANN input matrices have been constructed.

	Supra-Systolic Region			Systolic to Diastolic Region					Sub-Diastolic Region		
HB No.	1	2	3	4	5	6	7	8	9	10	11
1st ANN Target Output	1	1	1	0	0	0	0	0	0	0	0
	0	0	0	1	1	1	1	1	0	0	0
	0	0	0	0	0	0	0	0	1	1	1
1st ANN Output = 2nd ANN Input	0.86	0.73	0.47	0.66	0.38	0.21	0.46	0.3	0.11	0.32	0.24
	0.52	0.4	0.53	0.46	0.78	0.89	0.67	0.51	0.65	0.23	0.01
	0.1	0.35	0.34	0.21	0.15	0.34	0.41	0.73	0.61	0.58	0.75
Modified 2nd ANN Input	-	0.86	0.73	0.47	0.66	0.38	0.21	0.46	0.3	0.11	-
	-	0.52	0.4	0.53	0.46	0.78	0.89	0.67	0.51	0.65	-
	-	0.1	0.35	0.34	0.21	0.15	0.34	0.41	0.73	0.61	-
	-	0.73	0.47	0.66	0.38	0.21	0.46	0.3	0.11	0.32	-
	-	0.4	0.53	0.46	0.78	0.89	0.67	0.51	0.65	0.23	-
	-	0.35	0.34	0.21	0.15	0.34	0.41	0.73	0.61	0.58	-
	-	0.47	0.66	0.38	0.21	0.46	0.3	0.11	0.32	0.24	-
	-	0.53	0.46	0.78	0.89	0.67	0.51	0.65	0.23	0.01	-
	-	0.34	0.21	0.15	0.34	0.41	0.73	0.61	0.58	0.75	-
2nd ANN Target Output	-	0	0	1	1	1	1	1	0	0	-

The number of output from this modified algorithm was two HBs shorter than the original number of HBs. A value of zero was added to the first HB so that the selection of the pressure corresponded to the selected HB. The blood pressure selection method was the same method described in section 3.7.3. Another selection was made without a zero added to the first HB so that the blood pressure selection was one HB higher than the output of the ANN classification algorithm. The results were compared to the standard protocols.

4.3 Algorithm Validation

4.3.1 Validation On Nine Subjects

New measured data from 9 subjects, involving 27 measurements, was collected for the algorithm modification. First two measurements from each subject were used for training and the last measurement was used for testing purposes. A total of 429 HBs (139 HBs from the Supra-systolic region, 237 HBs from the systolic to diastolic region and 53 HBs from the Sub-diastolic region) from 18 measurements were used for

training the ANNs and 217 HBs (77 HBs from the Supra-systolic region, 114 HBs from the systolic to diastolic region and 26 HBs from the Sub-diastolic region) from 9 different measurements were used for testing. The algorithm which gave the best result was selected for further testing on more subjects.

4.3.1.1 ANN Improvement

101 different weight and bias matrices were initialised by using 101 different random seeds, 0 ~ 100. 15 different numbers of hidden layer neurons were combined with 101 different initialised matrices to generate a total of 1515 numbers of ANNs. 6, 12, 18 and 24 inputs were tested on these 1515 ANNs individually. The Trainlm training function was used to train the ANNs. The training epoch was set at 100 and the PEG was set at 0.001. Each input vector with 15 different neuron numbers had 101 different training and testing results. The mean and minimum testing errors from those 101 results and the different input values were calculated and presented in Figure 4.3.1. Results showed that the ANNs using 1 or 2 hidden layer neurons have the highest testing errors. ANNs using 3 to 5 hidden layer neurons have less numbers of hidden layer neurons and testing errors. ANNs with 3 to 5 hidden layer neurons were thus selected for future testing. ANNs using 6 inputs have a higher mean and minimum testing error than the other number of inputs. Table 4.3.1 and Table 4.3.2 show the result of the number of ANNs that passed the standard protocols. Values in Table 4.3.1 are the summary of results from 1515 ANNs. Table 4.3.2 summarises those ANNs that passed standard protocols out of 303 ANNs where only those ANNs with 3 to 5 hidden layer neurons were selected for further comparison. The highlighted values are the values used for comparison.
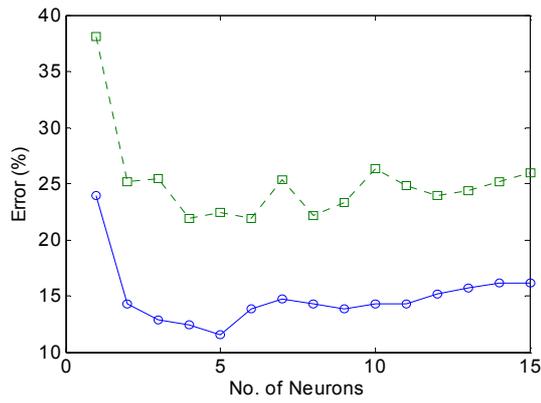
(a) 6 Input Data Sets
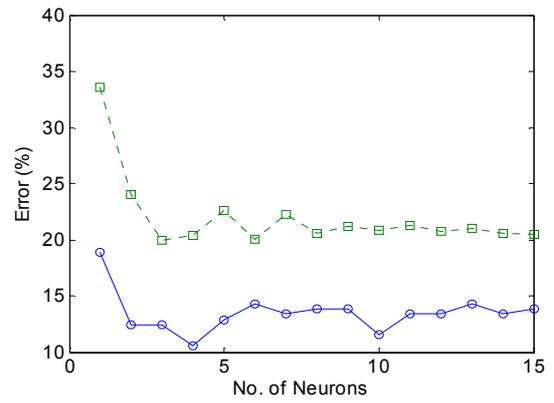
(b) 12 Input Data Sets
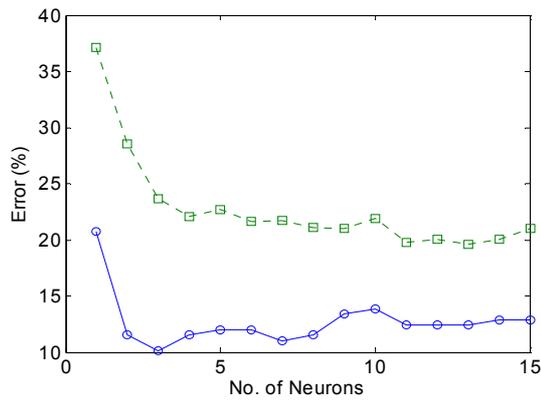
(c) 18 Input Data Sets
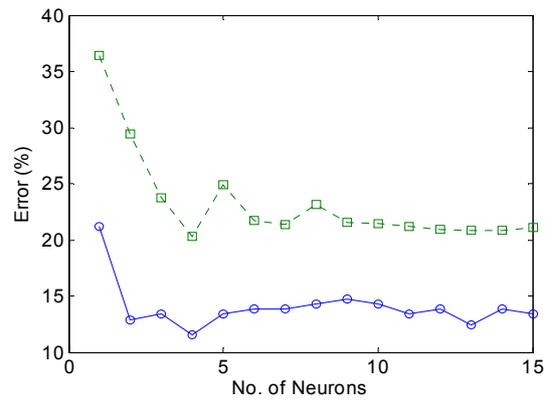
(d) 24 Input Data Sets
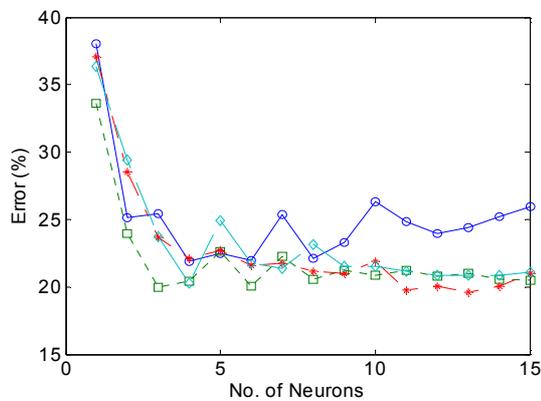
(e) Mean Testing Errors Comparison
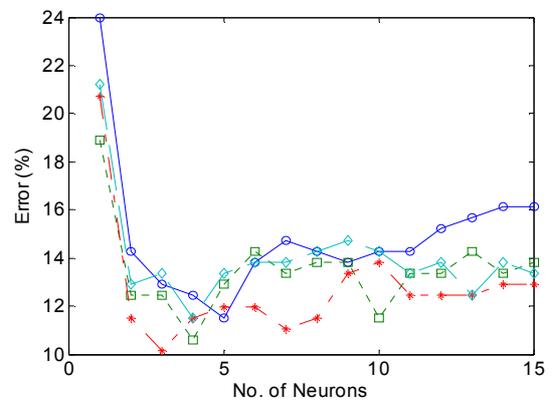
(f) Minimum Testing Errors Comparison

Figure 4.3.1 (a) to (d) show the mean and minimum testing error from 101 results of each input data set. The symbol ○ with the blue solid line represents the minimum testing errors from the same number of neurons with 101 different random seeds. The symbol □ with the green dotted line represents the mean testing errors from the same number of neurons with 101 different random seeds. (e) and (f) show the comparison of the mean and minimum testing error between input data sets. The symbol ○ with the blue solid line represents the result from the 6 input data sets. The symbol □ with the green dotted line represents the result from the 12 input data sets. The symbol * with the red dash-dot line represents the result from the 18 input data sets. The symbol ◊ with the cyan dashed line represents the result from the 24 input data sets.

Table 4.3.1 Number of ANNs passed the standard protocols by using Trainlm training function with 0.001 PEG.

No. of Inputs	No. of ANNs (15×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	1515	25	1360	0	1302	4	42	0	0	4	0
12	1515	298	1414	7	1332	57	54	7	0	52	1
18	1515	301	1389	26	1231	81	93	24	0	66	5
24	1515	276	1380	21	1165	52	125	16	4	43	6
Total	6060	900	5543	54	5030	194	314	47	4	165	12

Table 4.3.2 Number of 3 to 5 hidden layer neuron ANNs passed the standard protocols.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	1	275	0	267	0	8	0	0	0	0
12	303	42	286	3	273	7	9	0	0	10	0
18	303	63	276	10	239	14	18	9	0	11	0
24	303	60	281	6	226	9	31	4	1	7	1
Total	1212	166	1118	19	1005	30	66	13	1	28	1

Table 4.3.1 shows that most of the ANNs passed both the AAMI and BHS standard protocols for DP selection. Trainlm and Traingdx training functions were used to train the 4 different input sets (6, 12, 18 and 24) of ANNs. Training epochs were set at 100 and the PEGs were set at 0.1. Each input vectors were trained with 3 different numbers of neurons (3 to 5) and 101 different weight and bias matrices. The results of the number of ANNs that passed the standard protocols were shown in Table 4.3.3 and Table 4.3.4 by using Trainlm and Traingdx training functions, respectively. These results are not as good as those results shown in Table 4.3.2. A comparison of the training and testing errors tested from different training functions are shown in Figure 4.3.2(a) to Figure 4.3.2(d). These results were selected from the ANNs by using 24 inputs and 3 hidden layer neurons.

Table 4.3.3 Number of ANNs that passed the standard protocols by using Trainlm training function with 0.1 PEG.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	274	0	251	0	27	0	0	0	0
12	303	23	273	1	243	8	22	1	0	7	1
18	303	43	258	5	215	11	24	4	0	8	1
24	303	50	247	4	200	19	31	3	0	14	1
Total	1212	116	1052	10	909	38	104	8	0	29	3

Table 4.3.4 Number of ANNs that passed the standard protocols by using Traingdx training function with 0.1 PEG.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	301	0	290	0	12	0	0	0	0
12	303	17	269	2	242	3	16	2	0	2	0
18	303	33	265	4	218	8	34	2	2	7	0
24	303	33	277	3	232	7	31	3	0	6	1
Total	1212	83	1112	9	982	18	93	7	2	15	1

(a) Trainlm Training Function with Performance Error Goal of 0.001

(b) Trainlm Training Function with Performance Error Goal of 0.1

(c) Traingdx Training Function with Performance Error Goal of 0.1

(d) Testing Error Comparison
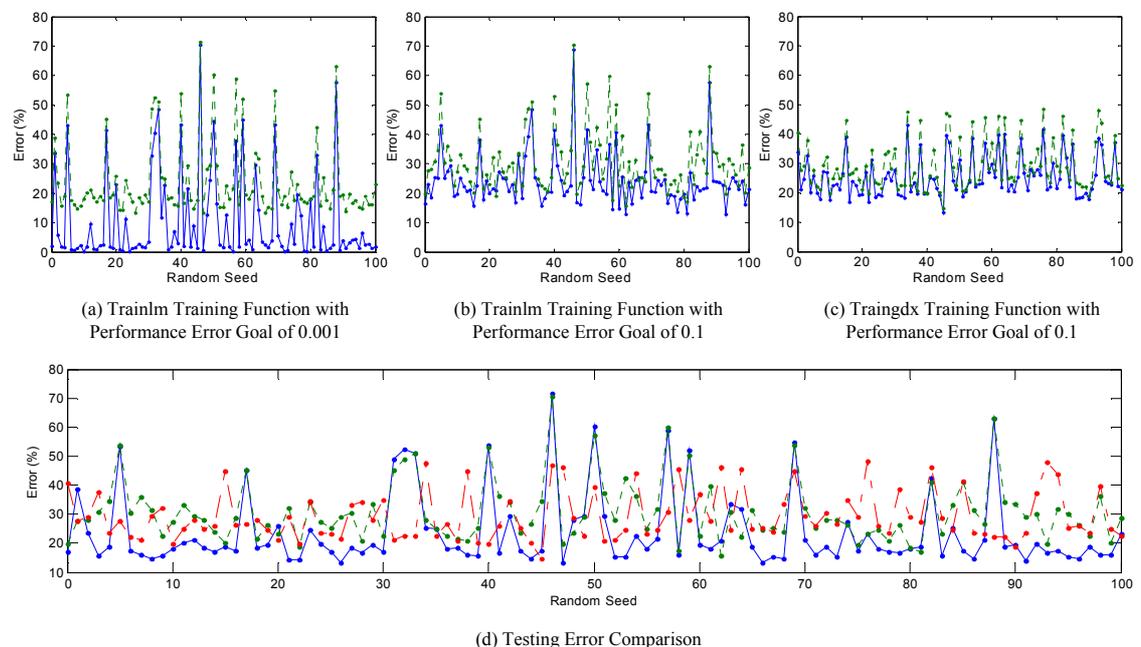
Figure 4.3.2 (a) to (c) show the training and testing errors tested by using different training functions. Plots were selected from ANNs using 24 input data sets with 3 hidden layer neurons. The blue solid line represents the percentage of the training errors. The green dotted line represents the percentage of the testing errors. (d) shows the comparison of the testing error between different training functions. The blue solid line represents the result from the Trainlm training function with PEG of 0.001. The green dotted line represents the result from the Trainlm training function with PEG of 0.1. The red dash-dot line represents the result from the Traingdx training function with PEG of 0.1.

Figure 4.3.2 (a) shows that the difference between the training and testing errors were too large. This meant that most of the ANNs were over-trained. Figure 4.3.2(b) and Figure 4.3.2(c) shows that a larger PEG reduced the difference between training and testing errors, but the overall errors were higher than the small PEG values as shown in Figure 4.3.2(d). The mean and minimum testing error from the 101 results on different random seeds and 4 different input sets by applying different training functions and PEGs are presented in Figure 4.3.3 to Figure 4.3.5.

(a) Mean Testing Errors   (b) Minimum Testing Errors
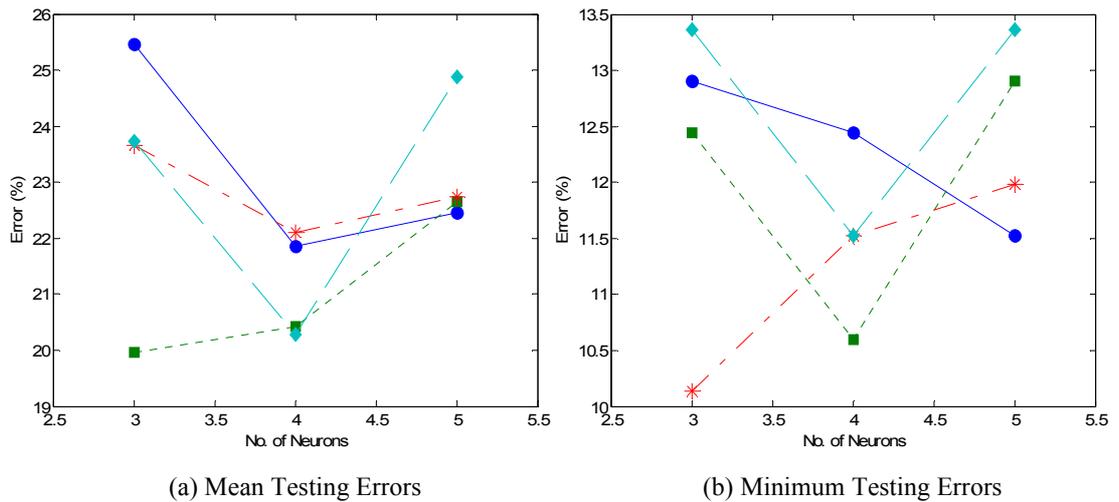
Figure 4.3.3 The mean and minimum testing error calculated from 101 results for each input data set by using Trainlm training function with 0.001 PEG. The symbol ● with the blue solid line represents the result from the 6 input data sets. The symbol ■ with the green dotted line represents the result from the 12 input data sets. The symbol * with the red dash-dot line represents the result from the 18 input data sets. The symbol ♦ with the cyan dashed line represents the result from the 24 input data sets.

(a) Mean Testing Errors   (b) Minimum Testing Errors
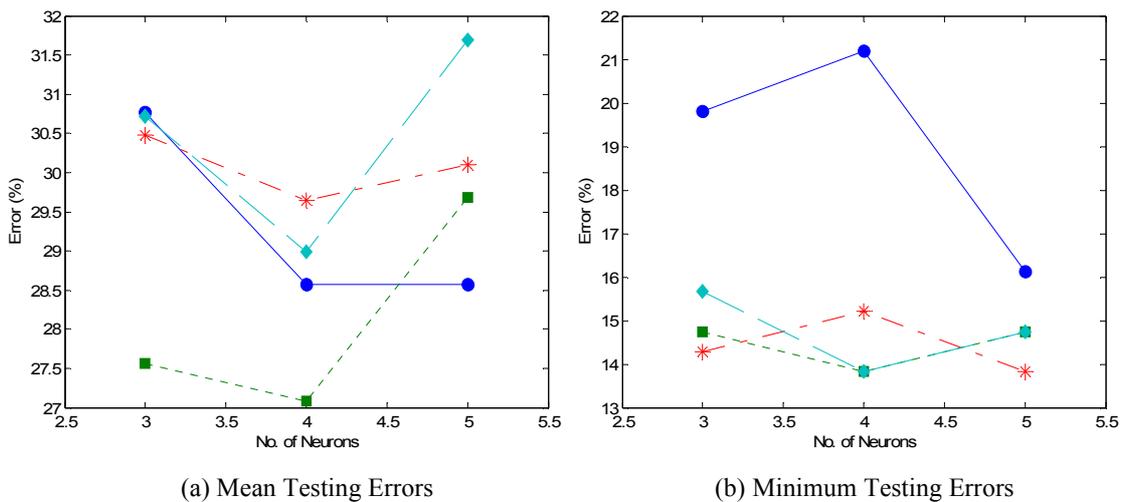
Figure 4.3.4 The mean and minimum testing error calculated from 101 results for each input data set by using Trainlm training function with 0.1 PEG. The symbol ● with the blue solid line represents the result from the 6 input data sets. The symbol ■ with the green dotted line represents the result from the 12 input data sets. The symbol * with the red dash-dot line represents the result from the 18 input data sets. The symbol ♦ with the cyan dashed line represents the result from the 24 input data sets.

(a) Mean Testing Errors	(b) Minimum Testing Errors
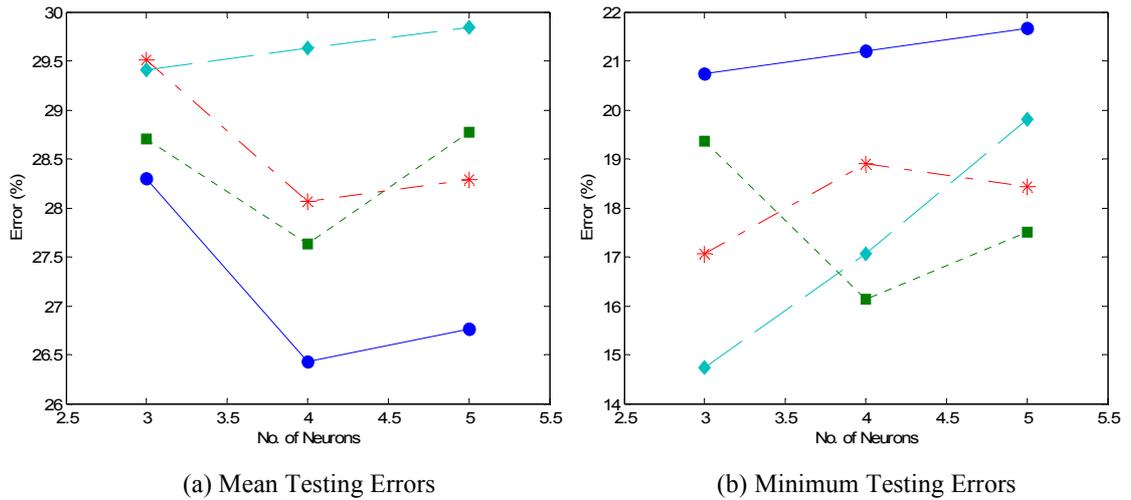
Figure 4.3.5 The mean and minimum testing error calculated from 101 results for each input data set by using Traingdx training function with 0.1 PEG. The symbol ● with the blue solid line represents the result from the 6 input data sets. The symbol ■ with the green dotted line represents the result from the 12 input data sets. The symbol * with the red dash-dot line represents the result from the 18 input data sets. The symbol ♦ with the cyan dashed line represents the result from the 24 input data sets.

To improve the generalization of the ANNs, Trainbr and Trainbfg training functions were used. Numbers of training epochs and PEGs were selected and tested to compare the training and testing error result. 100 and 300 training epochs and 0.1, 0.01 and 0.001 PEGs were set and tested. Random seed was set at 0 to reinitialise the weight and bias sets. 18 input data sets were selected and 3 different numbers of neurons were tested. The training and testing error result using the Trainbr training function is shown in Table 4.3.5. The results showed that the training errors were unchanged but the testing errors were slightly increased only when the training epoch increased to 300 with 3 hidden layer neurons or the PEG set at 0.001 with 4 hidden layer neurons. The msereg performance function was set and 0.5, 0.25 and 0.75 performance ratio was set and tested by using the Trainbfg training function. The training and testing error result is shown in Table 4.3.6. The results show that the training and testing errors were higher when the PEG was set at 0.1. The ratio set at 0.5 and 0.75 gave a similar result. Compared to others, the ratio set at 0.25 gave higher testing errors when using 4 hidden layer neurons but lower testing errors when using 3 and 5 hidden layer neurons in the ANN. The best selection of the PEG and epoch values was 0.01 and 100 respectively for both Trainbr and Trainbfg training functions. The ratio was set at 0.5 for the Trainbfg training function. Four different input sets (6, 12, 18 and 24) with 3 different numbers of neurons (3 to 5) and 101 different weight and bias matrices were tested. The

results of the number of ANNs that passed the standard protocols were shown in Table 4.3.7 and Table 4.3.8 by using Trainbr and Trainbfg training functions, respectively. A comparison of the training and testing errors tested from different training functions are shown in Figure 4.3.6(a) to Figure 4.3.6(c). These results were selected from the ANNs by using 24 inputs and 3 hidden layer neurons.

Table 4.3.5 The training and testing error result using the Trainbr training function with a different combination of PEG and Epoch values. Shaded part highlights the increased value of the testing errors.

Trainbr 18 Inputs RandSeed = 0		3 neurons		4 neurons		5 neurons	
PEG	Epoch	Training Errors (%)	Testing Errors (%)	Training Errors (%)	Testing Errors (%)	Training Errors (%)	Testing Errors (%)
0.1	100	0.93	18.43	0.00	16.59	0.23	19.82
0.01	100	0.93	18.43	0.00	16.59	0.23	19.82
0.001	100	0.93	18.43	0.00	17.05	0.23	19.82
0.1	300	0.93	18.89	0.00	16.59	0.23	19.82
0.01	300	0.93	18.89	0.00	16.59	0.23	19.82
0.001	300	0.93	18.89	0.00	17.05	0.23	19.82

Table 4.3.6 The training and testing error result using the Trainbfg training function with a different combination of Ratio, PEG and Epoch values. Shaded part highlights some different testing error results.

| Trainbfg 18 Inputs RandSeed = 0			3 neurons		4 neurons		5 neurons	
Ratio	PEG	Epoch	Training Errors (%)	Testing Errors (%)	Training Errors (%)	Testing Errors (%)	Training Errors (%)	Testing Errors (%)
0.5	0.1	100	20.51	21.66	21.91	20.28	19.58	24.42
0.5	0.01	100	9.09	14.75	8.86	13.82	8.86	15.67
0.5	0.001	100	9.09	14.75	8.86	13.82	8.86	15.67
0.5	0.01	300	9.09	14.75	8.86	13.82	9.09	15.21
0.25	0.01	100	9.09	14.29	8.62	14.75	8.39	15.21
0.25	0.01	300	9.09	14.75	8.86	14.75	6.99	14.29
0.75	0.01	100	9.09	14.75	8.86	13.82	8.62	15.67

Table 4.3.7 Number of ANNs that passed the standard protocols by using the Trainbr training function.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	3	282	0	282	1	0	0	0	1	0
12	303	61	295	1	269	11	20	1	0	10	0
18	303	137	294	8	264	22	13	7	1	18	1
24	303	162	296	6	249	32	21	6	0	28	1
Total	1212	363	1167	15	1064	66	54	14	1	57	2

Table 4.3.8 Number of ANNs that passed the standard protocols by using the Trainbfg training function.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	303	0	303	0	0	0	0	0	0
12	303	0	303	0	303	0	0	0	0	0	0
18	303	203	303	52	303	98	0	52	0	98	0
24	303	114	303	1	303	96	0	1	0	96	0
Total	1212	317	1212	53	1212	194	0	53	0	194	0

(a) Trainbr Training Function

(b) Trainbfg Training Function

(c) Combined Testing Result
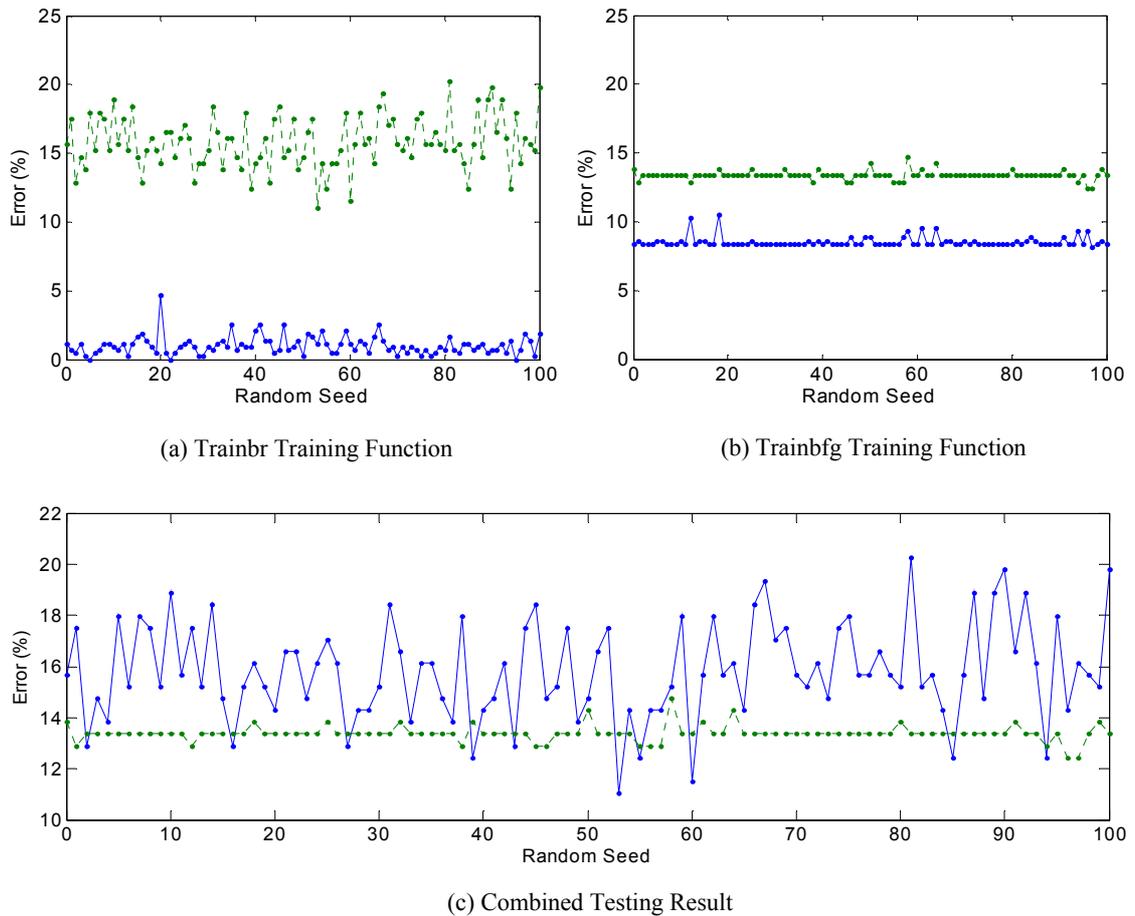
Figure 4.3.6 (a) and (b) show the training and testing errors tested by using Trainbr and Trainbfg training functions. Plots were selected from ANNs using 24 input data sets with 3 hidden layer neurons. The blue solid line represents the percentage of t nts the result from the Trainbr training function. The green dotted line represents the result from the Trainbfg he training errors. The green dotted line represents the percentage of the testing errors. (c) shows the comparison of the testing error between two training functions. The blue solid line represents the training function.

Figure 4.3.6(b) shows that the ANNs gave smaller testing errors and closer values between the training errors and testing errors by applying the Trainbfg training function. It also shows that the errors did not vary much using different random seeds. The mean and minimum testing errors from the 101 results on different random seeds and 4 different input sets by applying different training functions are presented in Figure 4.3.7 and Figure 4.3.8. The Trainbfg training function produced the lowest testing errors and more ANNs passed the standard protocols. This training function was thus selected for further validation.

(a) Mean Testing Errors   (b) Minimum Testing Errors
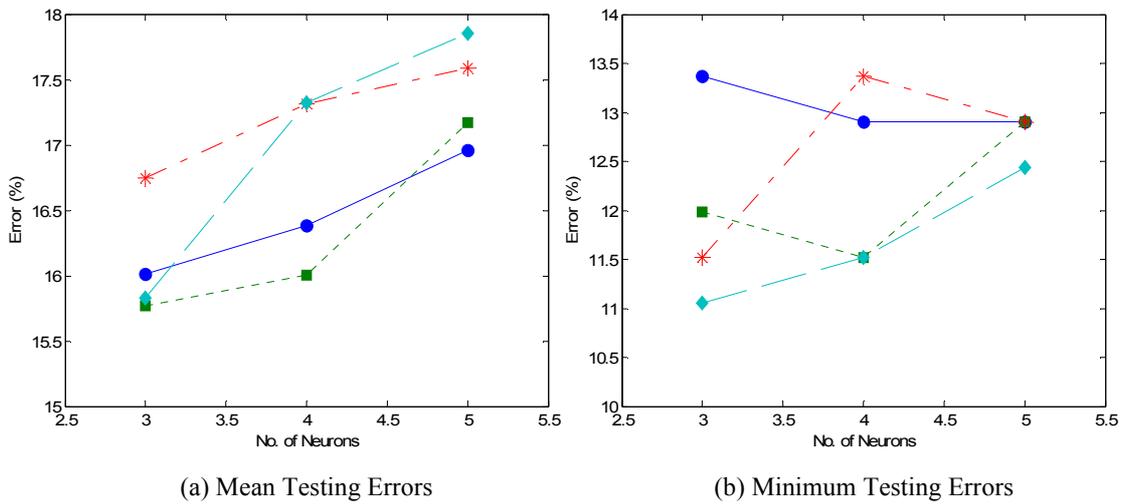
Figure 4.3.7 The mean and minimum testing error calculated from 101 results for each input data set by using Trainbr training function. The symbol ● with the blue solid line represents the result from the 6 input data sets. The symbol ■ with the green dotted line represents the result from the 12 input data sets. The symbol * with the red dash-dot line represents the result from the 18 input data sets. The symbol ♦ with the cyan dashed line represents the result from the 24 input data sets.

(a) Mean Testing Errors   (b) Minimum Testing Errors
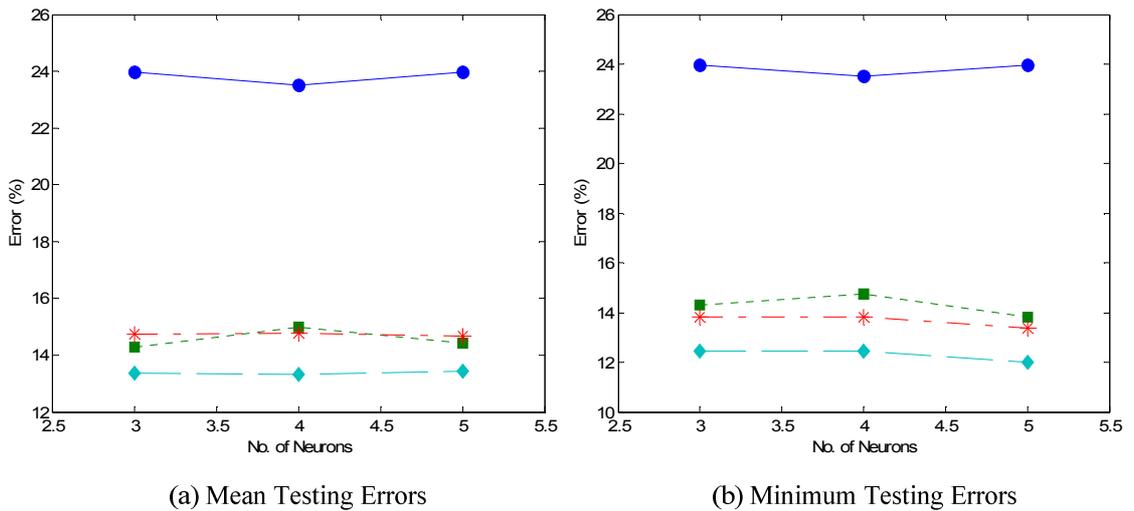
Figure 4.3.8 The mean and minimum testing error calculated from 101 results for each input data set by using Trainbfg training function. The symbol ● with the blue solid line represents the result from the 6 input data sets. The symbol ■ with the green dotted line represents the result from the 12 input data sets. The symbol * with the red dash-dot line represents the result from the 18 input data sets. The symbol ♦ with the cyan dashed line represents the result from the 24 input data sets.

PCA was applied to the 24 input sets to reduce the data set to 11 inputs. The Trainbfg training function was used and the result of the number of ANNs that passed the standard protocols is shown in Table 4.3.9. The result shows that most of the ANNs passed the AAMI standard protocol but the DP selection only passed the BHS standard protocol. Figure 4.3.9 shows the Bland Altman plot of the best ANN BP estimation

result compared to Auscultatory algorithm result. It shows that most of the SPs were underestimated by the trained ANN than the results obtained by the observers.

Table 4.3.9 Number of ANNs that passed the standard protocols by using PCA method as the input data sets. A total number of ANNs passed the standard protocols with 4 other input data sets were included.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
11 (PCA)	303	194	303	0	303	0	0	0	0	0	0
Total (5 inputs)	1515	511	1515	53	1515	194	0	53	0	194	0

(a) Systolic Blood Pressure   (b) Diastolic Blood Pressure
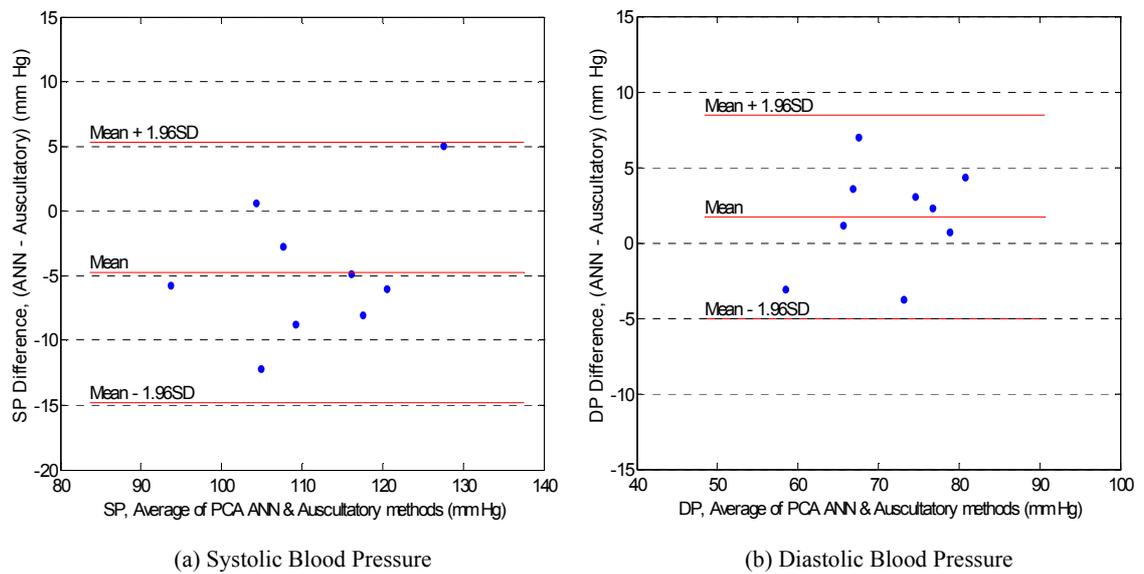
Figure 4.3.9 Blood pressure estimated from the ANN used PCA input data sets. Bland and Altman plot of the ANN and Auscultatory result comparison from 9 testing measurements.

4.3.1.2 Second ANN

The ANNs employing the Trainbfg training function produced the best results. Those networks were used for the 1st ANNs. Five different input sets (6, 12, 18, 24 and PCA) with 3 different numbers of neurons (3 to 5) and 101 different weight and bias matrices were tested for the 1st ANNs. The newlin function was used to create a linear layer for 2nd ANN after each ANN training and used the output vector from the 1st ANN as the input. The results of the number of ANNs that passed the standard protocols are shown in Table 4.3.10.

Table 4.3.10 Number of ANNs that passed the standard protocols by using newlin function as the 2nd ANNs.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	303	0	303	0	0	0	0	0	0
12	303	0	303	0	303	0	0	0	0	0	0
18	303	205	303	52	303	113	0	52	0	113	0
24	303	114	303	1	303	100	0	1	0	100	0
11 (PCA)	303	0	303	0	303	0	0	0	0	0	0
Total	1515	319	1515	53	1515	213	0	53	0	213	0

The 1st ANN results were unchanged and the Trainbfg training function was used for 2nd ANN. 101 different weight and bias matrices were initialised by using another 101 different random seeds, 500 ~ 600. The settings of the PEG, epoch and ratio values were 0.01, 100 and 0.1, respectively. The results of the number of ANNs that passed the standard protocols are shown in Table 4.3.11.

Table 4.3.11 Number of ANNs that passed the standard protocols by using Trainbfg training function as the 2nd ANNs.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	291	0	291	0	0	0	0	0	0
12	303	0	279	0	279	0	0	0	0	0	0
18	303	39	281	0	281	4	0	0	0	4	0
24	303	240	277	0	277	41	0	0	0	41	0
11 (PCA)	303	25	281	0	281	0	0	0	0	0	0
Total	1515	304	1409	0	1409	45	0	0	0	45	0

2nd ANN applied the newlin function and produced a better BP estimation than the Trainbfg training function. However, both of the functions had lesser ANNs that passed the AMMI standard protocol than the trained 1st ANNs.

4.3.1.3 Inputs Modification

Since the 2nd ANNs did not give better result than the 1st ANNs, a modification of the input of the 2nd ANNs was made. Nine inputs were used instead of 3 as described in

section 4.2.3. The Trainbfg training function with the same settings as in last section (4.2.3) was used for 2nd ANN. Two different methods of selecting the HB for BP estimations were carried out. The first method used the original blood pressure selection method and another selected the pressure before the selected HB as described in the last part of section 4.2.3. The results of the number of ANNs that passed the standard protocols are shown in Table 4.3.12 and Table 4.3.13.

Table 4.3.12 Number of ANNs that passed the standard protocols by using the original BP selection method.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	303	0	289	0	0	0	0	0	0
12	303	3	303	1	278	0	0	1	0	0	0
18	303	279	303	1	280	25	2	1	0	25	0
24	303	173	303	12	289	5	0	12	0	5	0
11 (PCA)	303	20	303	0	282	1	0	0	0	1	0
Total	1515	475	1515	14	1418	31	2	14	0	31	0

Table 4.3.13 Number of ANNs that passed the standard protocols by using one HB shifting method as the BP selection method

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	302	0	301	0	2	0	0	0	0
12	303	3	286	3	25	0	274	0	1	0	0
18	303	280	49	279	22	0	17	1	17	0	0
24	303	264	38	192	16	39	3	0	3	0	0
11 (PCA)	303	280	58	22	22	171	1	0	1	0	0
Total	1515	827	733	496	386	210	297	1	22	0	0

Table 4.3.12 and Table 4.3.13 show that neither BP selection method met the BHS standard protocol requirements. Table 4.3.13 shows that selecting one HB before the original selection from the algorithm gave a better SP estimation than all the other tested algorithms. Therefore, a new blood pressure selection method was made by selecting the SP using the pressure value which is one HB before the original selection while keeping DP selection the same as the blood pressure selection method. The results of the number of ANNs that passed the standard protocols are shown in Table 4.3.14.

Table 4.3.14 Number of ANNs that passed the standard protocols by using one HB shifting method for SP selection and original BP selection method for DP selection.

No. of Inputs	No. of ANNs (3×101)	Pass AAMI		Pass BHS Grade A		Pass BHS Grade B		BHS Recommended Grades (SP/DP)			
		SP	DP	SP	DP	SP	DP	A/A	A/B	B/A	B/B
6	303	0	303	0	289	0	0	0	0	0	0
12	303	3	303	3	278	0	0	3	0	0	0
18	303	280	303	279	280	0	2	279	0	0	0
24	303	264	303	192	288	39	1	191	1	39	0
11 (PCA)	303	280	303	22	282	0	0	22	0	0	0
Total	1515	827	1515	496	1417	39	3	495	1	39	0

Table 4.3.14 shows that most of the trained ANNs with 18 and 24 input sets passed both the AAMI and BHS standard protocols with grade A. This method was therefore selected for further tests on more subjects.

4.3.2 Algorithm Validation on 76 subjects

New measured data from 76 subjects, involving 228 measurements, were collected for the algorithm validation. A total of 4694 HBs (1370 HBs from Supra-systolic region, 1921 HBs from systolic to diastolic region and 1403 HBs from Sub-diastolic region) from 152 measurements were used for training the ANNs and 2388 HBs (713 HBs from Supra-systolic region, 928 HBs from systolic to diastolic region and 747 HBs from Sub-diastolic region) from 76 different measurements were used for testing. The algorithm was modified to meet the requirements of the standard protocols. Final validation on 86 subjects was later carried out.

The Trainbfg training function was used for both 1st and 2nd ANNs. All the results have shown that 18 and 24 input sets gave best BP estimations. New measured data were used to extract 18 and 24 features from each HB as the input sets for the ANNs. Three different numbers of neurons (3 to 5) were applied in ANN constructions. Figure 4.3.6(b) shows that the training and testing errors did not have much variation by using Trainbfg training function with different initialised weight and bias matrices. Therefore, 11 different weight and bias matrices with random seeds from 0 ~ 10 and 500 ~ 510 were used for the 1st and 2nd ANNs, respectively. The settings of the PEG, epoch and ratio values were 0.01, 100 and 0.1, respectively. The training and testing errors for 1st ANNs and 2nd ANNs are shown in Figure 4.3.10 and Figure 4.3.11, respectively.
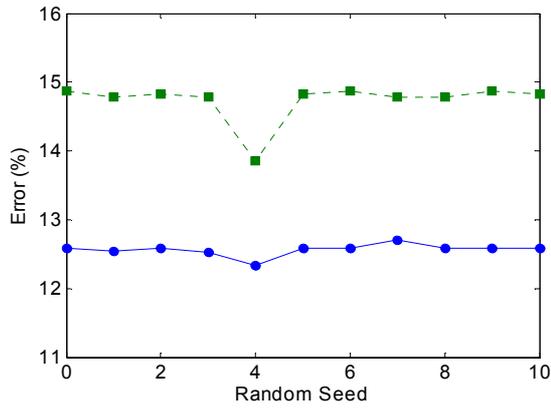
(a) 3 neurons; 18 Input Data Sets

(b) 3 neurons; 24 Input Data Sets
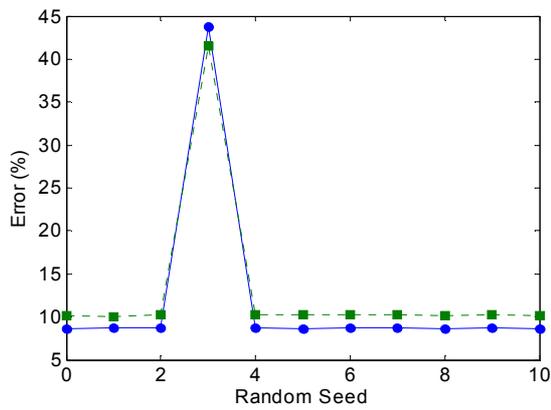
(c) 4 neurons; 18 Input Data Sets
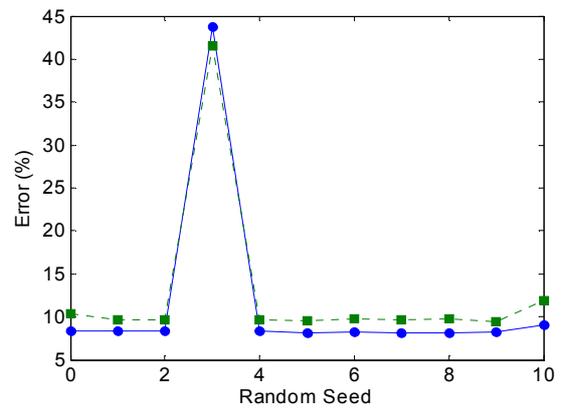
(b) 4 neurons; 24 Input Data Sets
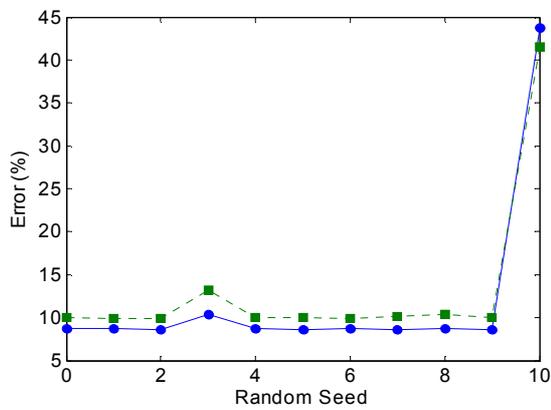
(e) 5 neurons; 18 Input Data Sets

(f) 5 neurons; 24 Input Data Sets
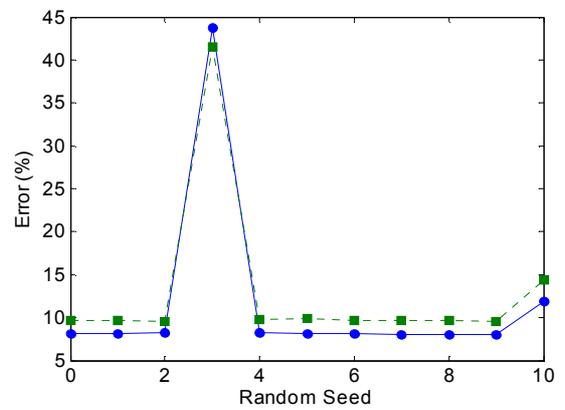
Figure 4.3.10 Training and testing error results from 1st ANNs. The symbol ● with the blue solid line represents the percentage of the training errors. The symbol ■ with the green dotted line represents the percentage of the testing errors.

Figure 4.3.11 Training and testing error results from 2nd ANNs. The symbol ● with the blue solid line represents the percentage of the training errors. The symbol ■ with the green dotted line represents the percentage of the testing errors.

Figure 4.3.10 and Figure 4.3.11 show that all ANNs gave similar training and testing errors on the same input data sets except that a few ANNs in the 2nd ANNs gave extremely large errors. This means that the trained weights and biases were similar in each ANN. The ANN structure with 3 neurons and random seed of 0 for 1st ANN and

random seed of 500 for 2nd ANN were selected for validating the new measured 76 subjects. Table 4.3.15 and Table 4.3.16 show the ANN results compared to the AAMI and BHS standard protocols for 18 and 24 input data sets. "Net 1" is the BP selection result from 1st ANN; "Net 2_1" is the BP selection result from 2nd ANN with original BP selection method; "Net 2_2" is the BP selection result from 2nd ANN with one HB shifting method and "Net 2_3" is the BP selection result from 2nd ANN with one HB shifting method on the SP selection and original selection on the DP selection. Figure 4.3.12 and Figure 4.3.13 show the Bland Altman plot of the BP estimation from different input data sets to compare ANN classification results from Net 2_3 and Auscultatory algorithm results.

Table 4.3.15 Results from 18 input data sets compared to the standard protocols by using different BP selection methods.

18 Inputs	Systolic Pressure					Diastolic Pressure					Standard (SP / DP)	
	Measurement Error		Absolute difference (%)			Measurement Error		Absolute difference (%)			AAMI	BHS
Net	mean	SD	≤ ±5	≤ ±10	≤ ±15	mean	SD	≤ ±5	≤ ±10	≤ ±15	Pass/Fail	Grades
1	-3.65	12.54	75.00	88.16	92.11	-2.21	10.28	63.16	89.47	94.74	F/F	B/B
2_1	-3.33	12.62	72.37	89.47	94.74	-1.64	10.53	61.84	88.16	93.42	F/F	B/B
2_2	0.02	12.83	63.16	92.11	96.05	1.40	10.79	55.26	82.89	97.37	F/F	A/B
2_3	0.02	12.83	63.16	92.11	96.05	-1.64	10.53	61.84	88.16	93.42	F/F	A/B

Table 4.3.16 Results from 24 input data sets compared to the standard protocols by using different BP selection methods.

24 Inputs	Systolic Pressure					Diastolic Pressure					Standard (SP / DP)	
	Measurement Error		Absolute difference (%)			Measurement Error		Absolute difference (%)			AAMI	BHS
Net	mean	SD	≤ ±5	≤ ±10	≤ ±15	mean	SD	≤ ±5	≤ ±10	≤ ±15	Pass/Fail	Grades
1	-3.74	12.42	73.68	89.47	93.42	-1.97	10.30	63.16	89.47	96.05	F/F	B/A
2_1	-3.64	12.41	71.05	90.79	94.74	-2.81	10.20	59.21	90.79	96.05	F/F	B/B
2_2	-0.24	12.68	68.42	93.42	96.05	0.26	10.51	64.47	84.21	96.05	F/F	A/B
2_3	-0.24	12.68	68.42	93.42	96.05	-2.81	10.20	59.21	90.79	96.05	F/F	A/B

(a) Systolic Blood Pressure   (b) Diastolic Blood Pressure
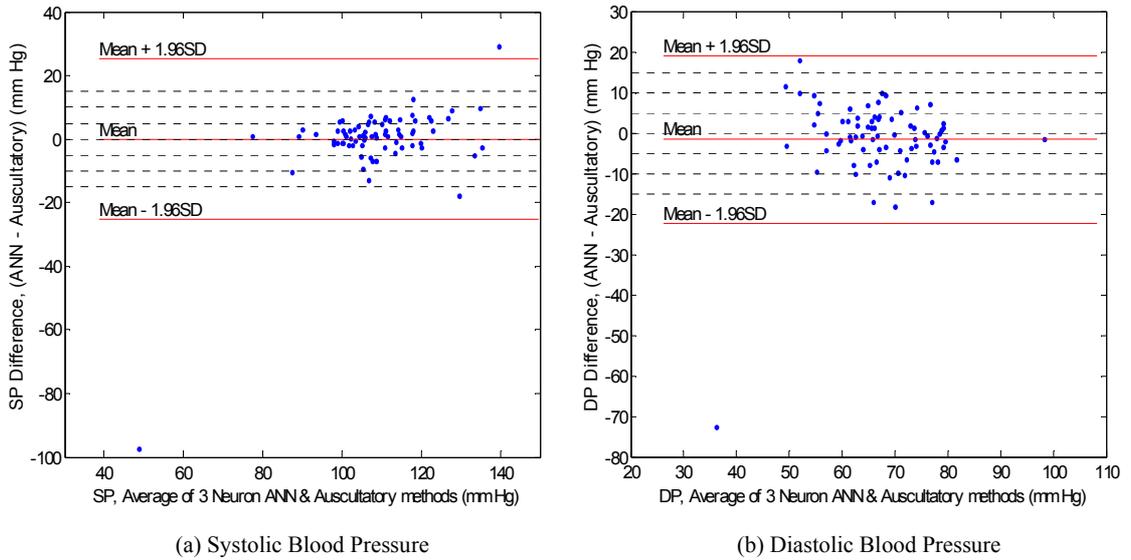
Figure 4.3.12 Bland and Altman plot of 18 input data sets with 3 hidden layer neuron ANN and Auscultatory result comparison from 76 testing measurements.

(a) Systolic Blood Pressure   (b) Diastolic Blood Pressure
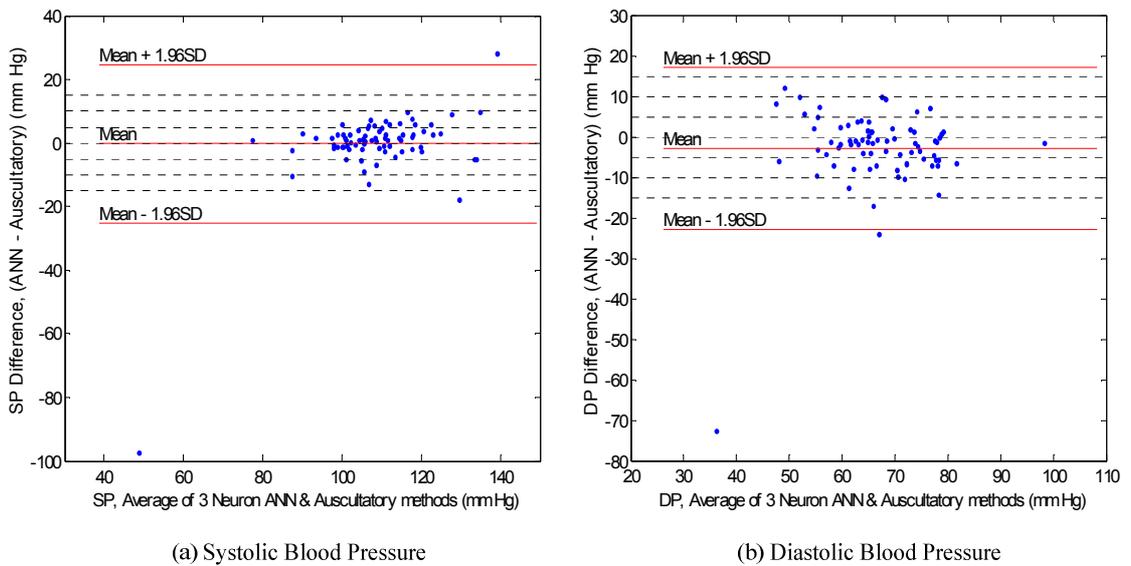
Figure 4.3.13 Bland and Altman plot of 24 input data sets with 3 hidden layer neuron ANN and Auscultatory result comparison from 76 testing measurements.

Table 4.3.15 and Table 4.3.16 show that the testing results met the standard protocol requirements except the SDs of measurement error. Figure 4.3.12 and Figure 4.3.13 show that one measurement was undetected from the designed algorithm. The measurement was checked and the measured signal from the outside sensor is plotted in Figure 4.3.14. The target and the output values simulated by both 1st and 2nd ANN with 18 and 24 input data sets were shown in Figure 4.3.15 and Figure 4.3.16, respectively.
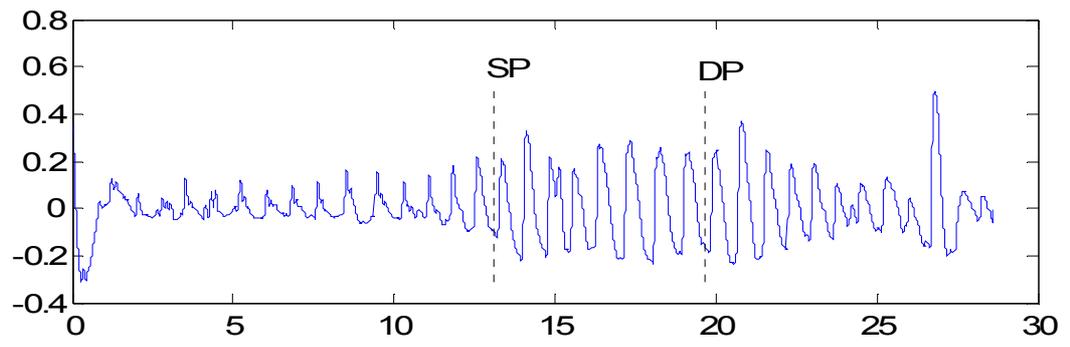
Figure 4.3.14 Signal measured from the outside sensor on subject 45, reading 3.

(a) 1st ANN Output. The symbol ○ with the blue solid line represents the target output for the Supra-systolic pressure region. The symbol ∗ with the blue broken line represents the actual output from the ANN for the Supra-systolic pressure region. The symbol ◇ with the red solid line represents the target output between the systolic and diastolic pressure regions. The symbol + with the red broken line represents the actual output between the systolic and diastolic pressure regions. The symbol □ with the black solid line represents the target output for the Sub-diastolic pressure region. The symbol × with the black broken line represents the actual output from the ANN for the Sub-diastolic pressure region.

(b) 2nd ANN Output. The symbol ○ with the blue solid line represents the target output between the systolic and diastolic pressure regions. The symbol ∗ with the blue broken line represents the actual output between the systolic and diastolic pressure regions.

Figure 4.3.15 1st and 2nd ANNs simulated output for subject 45, recording 3, by using 18 input data sets.

(a) 1st ANN Output. The symbol ○ with the blue solid line represents the target output for the Supra-systolic pressure region. The symbol * with the blue broken line represents the actual output from the ANN for the Supra-systolic pressure region. The symbol ◇ with the red solid line represents the target output between the systolic and diastolic pressure regions. The symbol + with the red broken line represents the actual output between the systolic and diastolic pressure regions. The symbol □ with the black solid line represents the target output for the Sub-diastolic pressure region. The symbol × with the black broken line represents the actual output from the ANN for the Sub-diastolic pressure region.

(b) 2nd ANN Output. The symbol ○ with the blue solid line represents the target output between the systolic and diastolic pressure regions. The symbol * with the blue broken line represents the actual output between the systolic and diastolic pressure regions.

Figure 4.3.16 1st and 2nd ANNs simulated output for subject 45, recording 3, by using 24 input data sets.

Figure 4.3.14 shows that the last HB gives the largest amplitude from the entire measured signal. Such a signal might be caused by an artificial motion. Every point was normalised to that maximum value for feature extractions in the time domain. It caused the pulses between the SP and DP to become extremely small, therefore, making ANNs unable to classify those pulses into the right region. To solve this problem, envelope amplitudes were calculated for each individual HB and used as one of the input features. Therefore, the individual HB signals would not be affected by the artificial motion. Input features were modified for the 1st ANN as described in section 4.2.3. A total of 21 features were extracted from each HB. The Trainbfg training function was used for both 1st and 2nd ANN. Three neurons and a random seed of 0 were applied in 1st ANN and a random seed of 500 was applied for 2nd ANN. The setting of the PEG, epoch and ratio values were 0.01, 100 and 0.1, respectively. The training and testing errors from 1st ANN were 13.36% and 14.99%, respectively. The training and testing errors from 2nd ANN were 8.93% and 8.18%, respectively. Table 4.3.17 summarises the ANN results compared to the AAMI and BHS standard protocols. Figure 4.3.17 shows the Bland Altman plot of the BP estimation to compare ANN classification results from Net 2_3 and Auscultatory algorithm results. The undetected signal (subject 45, reading 3) from the previous algorithm was now detected. The target and the output values simulated by both 1st and 2nd ANN are shown in Figure 4.3.18.

Table 4.3.17 Results from 21 input data sets compared to the standard protocols by using different BP selection methods.

| 21 Inputs	Systolic Pressure					Diastolic Pressure					Standard (SP / DP)	
	Measurement Error		Absolute difference (%)			Measurement Error		Absolute difference (%)			AAMI	BHS
Net	mean	SD	≤ ±5	≤ ±10	≤ ±15	mean	SD	≤ ±5	≤ ±10	≤ ±15	Pass/Fail	Grades
1	-3.15	6.01	63.16	90.79	94.74	-0.53	5.53	68.42	96.05	98.68	P/P	B/A
2_1	-2.97	5.45	65.79	92.11	97.37	0.69	4.96	68.42	96.05	100	P/P	A/A
2_2	0.46	5.46	73.68	97.37	97.37	3.77	5.05	56.58	86.84	100	P/P	A/B
2_3	0.46	5.46	73.68	97.37	97.37	0.69	4.96	68.42	96.05	100	P/P	A/A
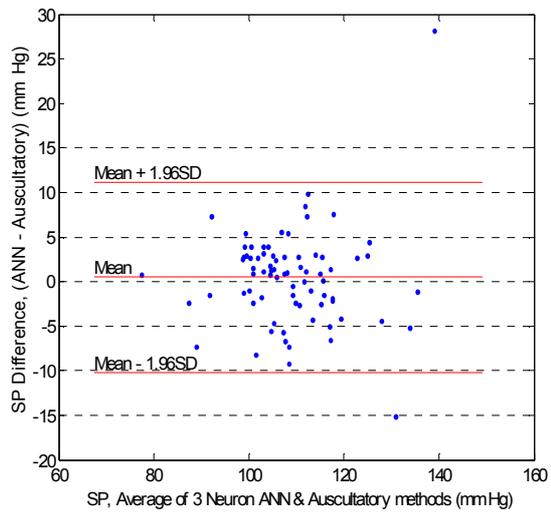
(a) Systolic Blood Pressure
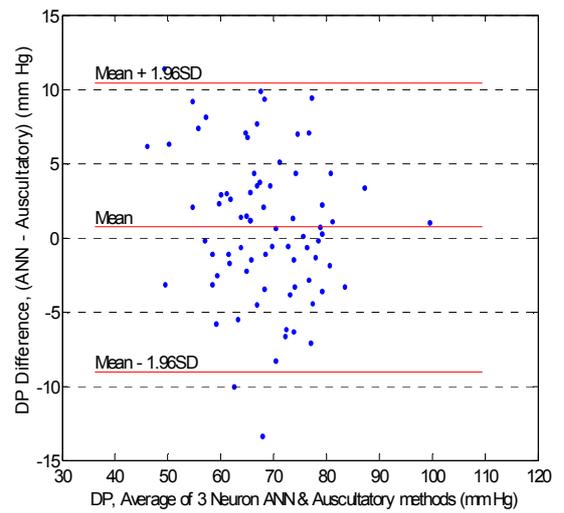
(b) Diastolic Blood Pressure

Figure 4.3.17 Bland and Altman plot of 21 input data sets with 3 hidden layer neurons in the ANN and Auscultatory result comparison from 76 testing measurements.

(a) 1st ANN Output. The symbol ○ with the blue solid line represents the target output for the Supra-systolic pressure region. The symbol * with the blue broken line represents the actual output from the ANN for the Supra-systolic pressure region. The symbol with the red solid line represents the target output between the systolic and diastolic pressure regions. The symbol + with the red broken line represents the actual output between the systolic and diastolic pressure regions. The symbol □ with the black solid line represents the target output for the Sub-diastolic pressure region. The symbol with the black broken line represents the actual output from the ANN for the Sub-diastolic pressure region.


(b) 2nd ANN Output. The symbol ○ with the blue solid line represents the target output between the systolic and diastolic pressure regions. The symbol * with the blue broken line represents the actual output between the systolic and diastolic pressure regions.

**Figure 4.3.18** 1st and 2nd ANNs simulated output for subject 45, recording 3, by using 21 input data sets.

102

Table 4.3.17 shows a good result that all of the networks passed both the AAMI and BHS standard protocols. The final validation on 86 subjects was then carried out.

4.4 Final Validation of Algorithm

New measured data from 10 more subjects, involving 30 measurements, was collected to add up to 86 subjects, involving 258 measurements, for the algorithm final validation. The finalised ANNs were kept similar without using the new measured data for further training. Table 4.4.1 shows the ANN results from the total of 258 measurements and compared to the AAMI and BHS standard protocols. Figure 4.4.1 shows the Bland Altman plot of the BP estimation to compare ANN classification results from Net 2_3 and Auscultatory algorithm results.

Table 4.4.1 Results from 21 input data sets compared to the standard protocols by using different BP selection methods on 86 subjects, 258 measurements.

21 Inputs	Systolic Pressure					Diastolic Pressure					Standard (SP / DP)	
	Measurement Error		Absolute difference (%)			Measurement Error		Absolute difference (%)			AAMI	BHS
Net	mean	SD	≤ ±5	≤ ±10	≤ ±15	mean	SD	≤ ±5	≤ ±10	≤ ±15	Pass/Fail	Grades
1	-3.17	8.33	64.73	91.09	96.51	0.12	7.30	66.67	89.15	96.90	F/P	A/A
2_1	-2.06	5.21	72.48	92.25	98.45	1.77	6.17	63.95	89.53	96.12	P/P	A/A
2_2	1.44	5.27	71.32	96.51	98.06	5.02	6.33	45.35	81.40	94.96	P/F	A/C
2_3	1.44	5.27	71.32	96.51	98.06	1.77	6.17	63.95	89.53	96.12	P/P	A/A

(a) Systolic Blood Pressure (b) Diastolic Blood Pressure
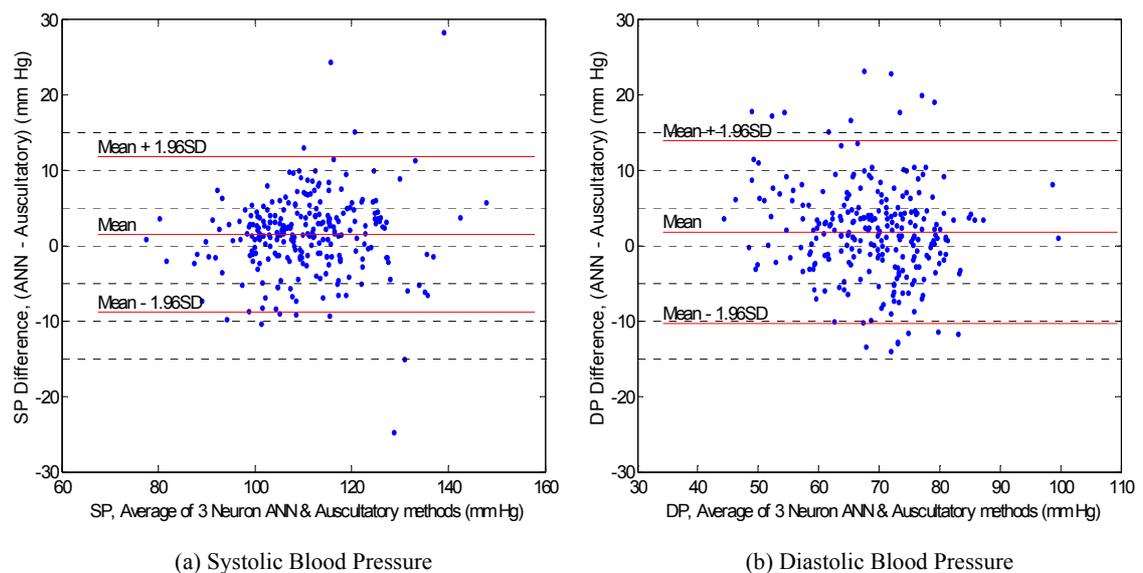
Figure 4.4.1 Bland and Altman plot of 21 input data sets with 3 hidden layer neurons in the ANN and Auscultatory result comparison from 86 subjects, 258 measurements.

4.5 Summary

ANN classification was selected for algorithm modifications and validations. The first two measurements from 9 subjects were used for training and the last measurement from the same subjects was used for testing ANNs. Trainlm, Traingdx, Trainbr and Trainbfg training functions were tested with different combinations of initial network weight and bias values, number of hidden layer neurons, number of training epochs and PEG values. A second ANN was added to the network structure. Input features and blood pressure selection methods were modified. The final HB / HR detection function, the feature extraction functions, other used functions and the final algorithm program codes can be found in Appendix V, Appendix VI, Appendix VII and Appendix VIII, respectively. The final algorithm was tested on 258 measurements from 86 subjects. It fulfilled the AMMI criteria and also achieved an overall grade A for both SP and DP according to the BHS standard protocol.

Chapter 5 Discussion and Conclusion

5.1 Introduction

This chapter covers the discussions, conclusion and possible future works for this research. Discussions on all of the stages undertaken in this research were described in detail. Conclusion on the works that had been done during the research period and the results of the developed algorithm were included. A few possible future works were suggested in this chapter to further develop a better system.

5.2 Discussion

5.2.1 Data Collection

The collection of the data was a massive workload to undertake. Ethical approval was required before the data collection as it involved human subjects as the participants. Application for ethics approval was approved by the AUTEC and the participants invited were focused on those staff and students in AUT Wellesley campus. The collection requirements and procedures stated in the AAMI protocol were carefully studied and followed during the data collection process. This ensured the consistency between observers and the participants to minimise any human error that could occur during the data collection process.

Apparatus used in this research was supplied by Pulsecor Ltd. except the laptop used for data collection. A second laptop was used because the original laptop broke down during the data collection period due to non-project related events. Regular calibration of the cuff pressure output was performed throughout the data collection period. Although different laptops were used during data the collection process, the stability and the accuracy of the output signal was checked constantly.

A total of 94 subjects participated in this research but 8 were excluded due to predefined criteria failure from the research requirements. Two subjects were excluded because

they did not contribute 3 data sets within the difference of ±5 mmHg between observers. One subject withdrew from the procedure due to the discomfort of cuff pressure. One subject was excluded because the Korotkoff sounds heard fell below 20 mmHg by the observers. One subject was excluded because the subject had an irregular HR which was discovered by the observers during the collection. Two subjects were excluded because these subjects were under medication and another subject had had heart valve surgery in the past. Therefore, a total of 86 subjects were left for BPM algorithm testing.

The AAMI standard recommended that when using the auscultatory monitoring method for comparison, at least 10% of SP and DP values should fall outside the range from 100 to 160 mmHg and 60 to 100 mmHg respectively. 10% of the total subjects should have an arm size above 35 cm and below 25 cm in circumference. None of the 86 subjects had SP greater than 160 mmHg and DP greater than 100 mmHg. 14.7% of subjects had SP lesser than 100 mmHg and 17.8% of subjects had DP lesser than 60 mmHg. In this research, there was only 1 subject (1.2%) that had an arm size greater than 35 cm and 11 subjects (12.8%) had an arm size less than 25 cm. This research focused on the healthy people in AUT Wellesley campus. None of them measured had hypertension. Not many people with an arm size greater than 35 cm were available on the campus and some of them did not want to participate in this research. Therefore, this requirement was not met.

The AAMI standard recommended that for the auscultatory measurement, two trained observers should have 100% of simultaneous measurements within a difference of 10 mmHg, and 90% or more within 5 mmHg. In order to enhance the credibility of the data since the two observers in this research were not professionally trained, the difference agreement between observers was tightened to 100% of measurements within 5 mmHg. At the end of this research, a total of 294 measurements had been taken from 86 subjects. A total of 36 measurements had a difference of more than 5 mmHg between observers and the rest of the 258 measurements were used for testing the final algorithm.

5.2.2 Algorithm Development

The software development stage used MATLAB as the developing program. It was crucial in processing and analysing informaion. All efforts have been made to learn the MATLAB programming codes in order to build the signal processing tools for analysing, extracting features and classification of the measured signals.

Signal Processing

The cutting line of each HB might not be the lower part of the pulse signal. Sometimes the pulse signal could be very small and / or fell into the negative region as shown in Chapter 3 Figure 3.1.1. When the amplitude at the cutting line is larger than the amplitude of the desired pulse, the calculated values for the height and area under the pulse would be incorrect. Therefore, each HB was windowed by using the Hanning window to bring down any signal outside the chosen range to zero. Although the band-pass filter and FFT functions were also tried to perform the same purpose, the Hanning window produced the best result for the requirement of this research.

Heart Beat / Heart Rate Determination

The original HB and HR determinations were much simpler than the detail described in section 3.3. The first part of the measurements from 9 subjects did not have much variation in the measured signal. HBs were easily detected by applying a second order Butterworth LPF with a corner frequency at 1.3 Hz. When measurements increased to 76 subjects, the developed HB detection algorithm was not able to detect the HB at the same corner frequency for all cases. Therefore, variable corner frequencies from 2 Hz to 0.5 Hz were designed to perform the HB detection for all subjects . This variation worked well for all healthy subjects involved in this research. However, it is understood that measurements from diseased subjects such as bradycardia (slow heart) or tachycardia (rapid beating) and subjects not in a resting situation may need another HB determination method.

Pressure Selection

A pressure value for each HB was selected after the HB was defined. The pressure value selected for each HB was the pressure at the point on the upstroke of the oscillation signal. An example of a measured pressure signal is shown in Figure 5.2.1. The Korotkoff sounds should be heard when the pressure dropps to the BP. The BP could not occur between point B and C or C and D of each pulse because the same pressure value occurred between point A and B. The possible pressure occurred between point D and E. It is also hard to define the point for pressure value at that range. Therefore, the upstroke of the pressure was defined as the pressure value for that HB for uniformity.
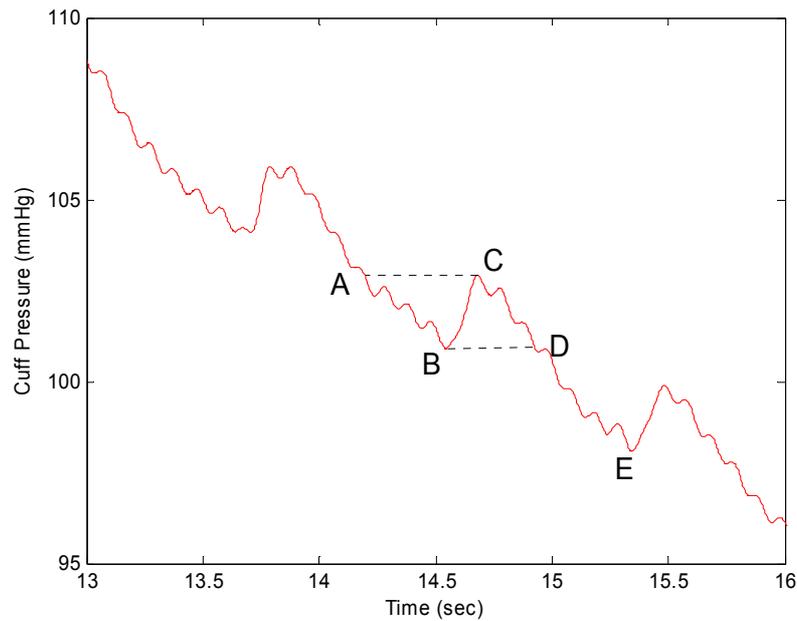
Figure 5.2.1 An example of pressure signal measured from subject 5, recording 1.

Height-based Algorithm

Height-based algorithm was developed based on the comparison of the amplitude of each HB. SP and DP obtained from observers were used to determine the ratio for all 27 cases. An average ratio was obtained from 27 ratios calculated above. The average ratio was set as the preset ratio of the Height-based algorithm.

The actual ratio varied from measurement to measurement. For the inside sensor signal, the minimum ratio for the SP was about 30% of its maximum amplitude and 3 measurements had their maximum amplitude at the SP. The average preset ratio selected for SP was 60% of its maximum amplitude. The minimum and maximum ratios for the DP were about 10% and 90% respectively of its maximum amplitude. The average preset ratio selected for DP was 40% of its maximum amplitude. For the outside sensor signal, the minimum and maximum ratios for the SP were about 35% and 85% respectively of its maximum amplitude. The average preset ratio selected for SP was 70% of its maximum amplitude. The minimum and maximum ratios for the DP were about 40% and 100% respectively of its maximum amplitude. The average preset ratio selected for DP was 60% of its maximum amplitude. For the pressure sensor signal, the minimum and maximum ratios for the SP were about 35% and 100% respectively of its maximum amplitude. The average preset ratio selected for SP was 70% of its

maximum amplitude. The minimum and maximum ratios for the DP were about 40% and 90% respectively of its maximum amplitude. The average preset ratio selected for DP was 60% of its maximum amplitude. The distributions of SP and DP ratio for each sensor are shown in Figure 5.2.2 to Figure 5.2.4.
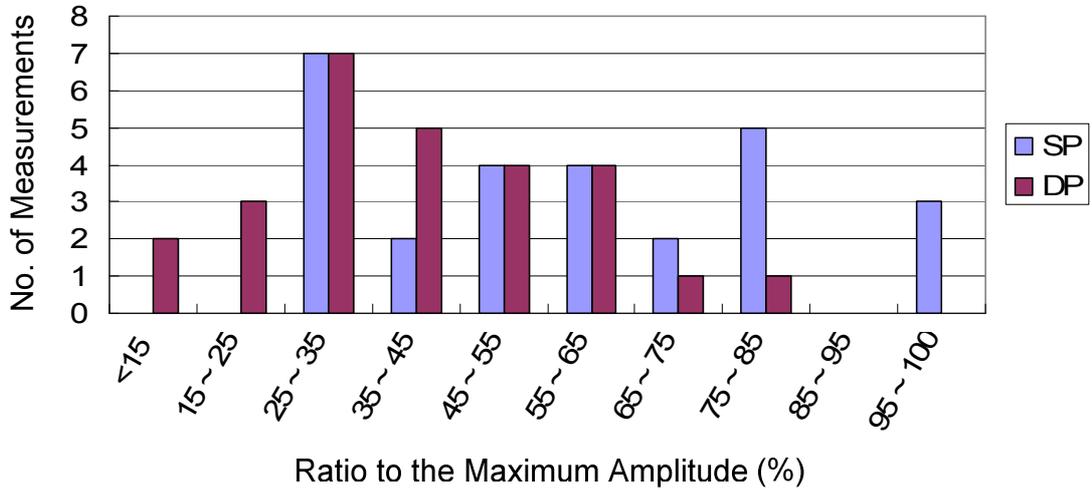
Figure 5.2.2 Inside sensor height ratio distribution graph.
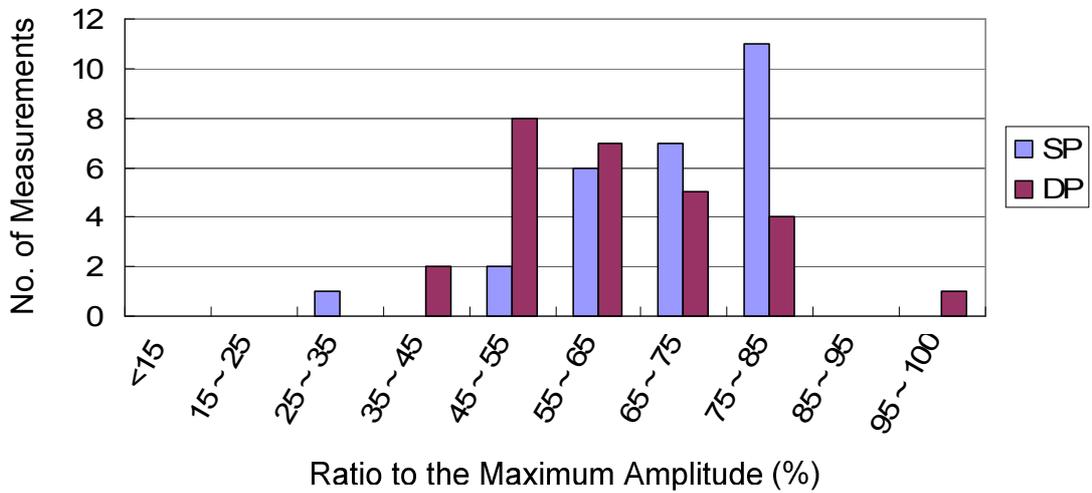
Figure 5.2.3 Outside sensor height ratio distribution graph.
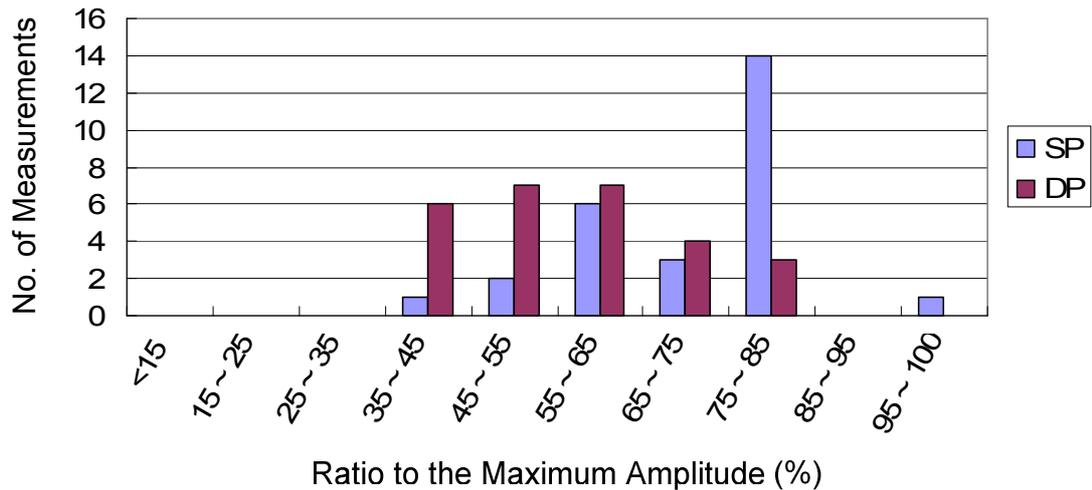
Figure 5.2.4 Pressure sensor height ratio distribution graph.

A wide range of ratios were distributed among 9 subjects, involving 27 measurements. The height-based algorithm could not detect the right BP by only setting average values for SP and DP ratios. Therefore, the height-based algorithm was not used for BP selection.

STFT Algorithm

STFT algorithm used the MATLAB built-in Spectrogram function to calculate the magnitude and PSD for each measurement. The algorithm divides the original signal into small segments as described in section 2.2.1.2. If the segments were small this would give a good time resolution but a poor frequency resolution. Increases in the segment size give a better frequency resolution but poor time resolution. The resolution selected for this research was the segment length lesser than the length of a pulse. The sampling frequency of the measured signals was 250 samples per second. The length of each segment, Hamming window, was set at 100 points with 50% overlap between segments, so that each segment covered 0.4 sec signal with the time interval at 0.2 sec.

Inside sensors were more sensitive to pick up the reflection wave from the brachial artery. This algorithm gave better results for SP estimation from the inside sensor as shown in Chapter 3 Table 3.8.2. DP was not easy to determine by using this algorithm. Most of the cases were estimated outside the range of ± 10 mmHg compared to the reference values. Therefore, this algorithm was not selected for further development. However this algorithm gave a good analysis in the frequency domain. This method was

used in the ANN classification algorithm to extract input features from the frequency domain.

Artificial Neural Network Classification

Feature extraction of significant information from the measured signal data is the most difficult but crucial part for the ANN classification algorithm. It can affect the success or failure in the analysis. The selected features were based on those visualised signal changes among pulses.

Results tested in this research showed that too small a number of inputs (6 or 12) or a large number of inputs (24) would not give good results. PCA was applied to reduce the number of inputs. Since the total number of inputs in this case were not in the range of hundreds or thousands, even significant inputs would be eliminated by the application of PCA during the analysis process. However, the training results obtained using the PCA input did not give as good a result as expected.

For re-validation of the modified algorithm, there were only 53 HBs in the sub-diastolic pressure region for training and 26 HBs for testing the ANNs as described in section 4.3.1. Because there were so few HBs in the sub-diastolic pressure region it was difficult to get results which showed the accuracy of the algorithm. The reason for that small number of HBs in the sub-diastolic pressure region was because the software chopped off the last few seconds of measured signal. Most of the measured signals found that there was some noise signals in the last few seconds. Some noise signals were small whereas some were larger than the significant signals. It was presumed that the noise signals were caused by the movement of the subjects. Every time the observers noted the DP from the subject, observers would proceed to record the BP for that subject before stopping the software recording. In addition, subjects would start to move their arms or clenched their fist during that time. These movements were easily picked up by the sensor, especially the inside sensors. Therefore, the software was designed to chop off those noise signals before the analysis began. After this problem was discovered, observers were advised to record BP after the software stopped recording to minimise the noise signals. The algorithm validation for 76 subjects therefore used the whole recorded signal without chopping any of the noise signals at the end. This made the training and testing of HBs in the sub-diastolic pressure region more balanced like the other two regions as described in section 4.3.2.

The outside sensor signal was selected for the final analysis. This sensor gave a clear signal and a similar pattern for most of the measurements. As mentioned above, inside sensors were closer to the brachial artery and thus more sensitive to the reflection waves as shown in Chapter 3 Figure 3.1.1. In many cases the inside sensors did not record the data signal as clearly as shown in the figure. This might be caused by the misplacement of the cuff or the depth of the brachial artery which depended on the individual subject. The actual problem could be discovered in future work. However, these sensors were too sensitive to pick up all the noise signals from any movements of the subject as shown in Figure 5.2.5. It was so challenging for the algorithm to select the BP from the trained ANNs. Signals measured from the pressure sensor also contained more noise signals after filtering than the signals measured from the outside sensor as shown in Chapter 3 Figure 3.4.1. Therefore, signals measured from both inside sensor and pressure sensor were not used for the final analysis.
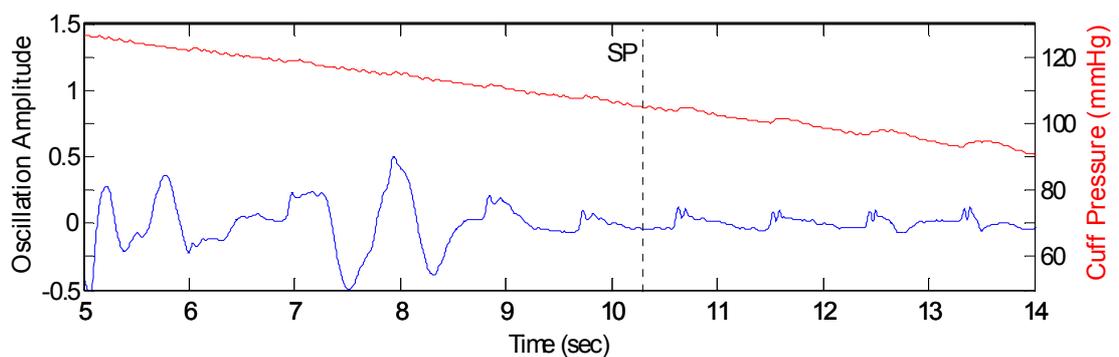
(a) Signal changes from supra-systolic pressure region to systolic pressure region.
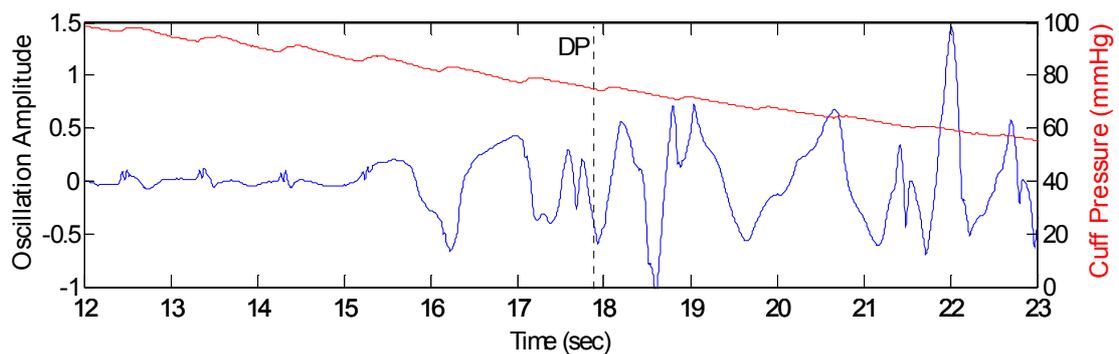
(b) Signal changes from diastolic pressure region to sub-diastolic pressure region.

Figure 5.2.5 Measured signal from the inside sensors (blue), pressure sensor (red) and the blood pressure estimated from observers (vertical dashed line).

The newlin function and Trainbfg training function were tested on 2nd ANN. Results from Chapter 4 Table 4.3.10 and Table 4.3.11 show that 2nd ANN used the newlin

function gave a better BP estimation than the Trainbfg training function. However, the newlin function is used for linear approximation which gives output on any values. If more untrained signals entered the ANN classifier and the output values fell outside the range between 0 and 1, then the algorithm was not able to select the HB for the SP and DP estimation. Therefore, Trainbfg training function was selected for the final algorithm on both the 1st and 2nd ANN.

5.3 Conclusions

In summary, this research investigated aspects of the digital signal processing, statistical analysis techniques and developed various algorithms used for blood pressure estimation. Standard Auscultatory BP measurement procedures were performed based on the AAMI requirement. Algorithm developments were completed for signal processing, HB / HR detection and cuff pressure selection for each HB. Three algorithms, height-based algorithm, STFT algorithm and ANN classification algorithm, were initially developed and compared. ANN classification algorithm was selected for modifications and validations. The final algorithm used two ANNs in series to select blood pressures. This algorithm achieved a grade A for both SP and DP according to the BHS protocol. The mean differences (SD) between the observers and the developed algorithm were 1.44 (5.27) mmHg and 1.77 (6.17) mmHg for SP and DP, respectively, which also fulfilled the AAMI criteria. In conclusion, this algorithm was successfully developed and recommended for further clinical trials with the wider adult population.

5.4 Recommendation and Future Work

This research was successfully completed with good results, but there are some shortcomings that can be improved for future development.

The reference values are very important during the development stage. These were treated as a perfect result for comparison with the newly developed devices or algorithms. If the reference values were not accurate enough, then the algorithm was inappropriately trained and thus would cause incorrect determination of SP and DP. The invasive method is the most accurate method to compare the new algorithms but it is inconvenient and may be dangerous to subjects. If the Auscultatory method was selected to obtain the reference value, then professionally trained observers should be

employed for the blood pressure data collection to minimise the errors due to untrained observers.

The ANN is a strong mathematical model which is trying to model a system similar to the human brain. Feature selection is the most crucial part for the input of the ANNs. In this research, extracted features from each HB of the processed signal were used as the inputs of the ANNs. Those selected features were based on the visualised feature changes of signals in both time and frequency domains. Features can be also selected from Wavelet analysis, Orthogonal transformation or PCA selection methods. ANN can also be applied for pattern recognition purposes. These are the other good methods which can be selected for future testing.

This research discovered that the inside sensor has the ability to pick up the reflection wave from the brachial artery. Further development can be made to improve the sensitivity and noise rejection. The measured signal can be tested and checked by changing the placement of the cuff on the upper arm. The placement of the sensors inside the cuff can also be modified to improve the whole measuring system.

Lastly, if the inside sensor can measure the reflection wave with not much noise signal easily, then this newly developed device has great potential to measure blood pressure signals from different sub-groups, e.g. pregnant women, arrhythmia, diabetics and other subjects with diseases. New algorithms may be needed to accurately measure the blood pressure for these subjects or a tuneable algorithm can be developed for different purposes of measurement.

REFERENCES

1. Holejšovská P, Peroutka Z, Cengery J. Non-invasive monitoring of the human blood pressure. Proceedings 16th IEEE Symposium on Computer-Based Medical Systems; 2003; 2003. p. 301-6.

2. Ball-Llovera A, Del Rey R, Ruso R, Ramos J, Batista O, Niubo I. An Experience in Implementing the Oscillometric Algorithm for the Non-Invasive Determination of Human Blood Pressure. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 2003; 2003. p. 3173-5.

3. Nissilä S, Sorvisto M, Sorvoja H, Vieri-Gashi E, Myllylä R. Non-invasive blood pressure measurement based on the electronic palpation method. Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 1998; 1998. p. 1723-6.

4. Northrop RB. Noninvasive Instrumentation and Measurement in Medical Diagnosis. Boca Raton, Fla. : London: CRC ; Chapman & Hall; 2001.

5. O'Brien E, Waeber B, Parati G, Staessen J, Myers MG. Blood pressure measuring devices: Recommendations of the European Society of Hypertension. British Medical Journal. 2001;322(7285):531-6.

6. O'Brien E, Petrie J, Littler W, de Swiet M, Padfield PL, Altman DG, et al. Short report: An outline of the revised British Hypertension Society protocol for the evaluation of blood pressure measuring devices. Journal of Hypertension. 1993 1993;11(6):677-9.

7. Association for the Advancement of Medical Instrumentation. American National Standard. Manual, electronic or automated sphygmomanometers. 2003;ANSI/AAMI SP10:2002.

8. Guyton AC, Hall JE. Textbook of Medical Physiology. 10th ed. Philadelphia, Pa. ; London: W. B. Saunders; 2000.

9. Dulbecco R. Encyclopedia of Human Biology. 2nd ed. San Diego, Calif.: Academic Press; 1997.

10. Des Jardins TR. Cardiopulmonary Anatomy & Physiology : Essentials for Respiratory Care. 4th ed. Australia ; Albany: Delmar/Thomson Learning; 2002.

11. Ward JPT, Linden RWA, Clarke RW. Physiology at a glance. Malden, Mass.: Blackwell Pub.; 2005.

12. Noble A, Johnson R, Thomas A, Bass P. The Cardiovascular System. Edinburg ; New York: Elsevier Churchill Livingstone; 2005.

13.   Tiger S, Kirk JK, Solomon RJ. MATHEMATICAL CONCEPTS IN CLINICAL SCIENCE. New Jersey: Prentice Hall, Inc.; 2000.

14.   Ng K-G. Oscillometric blood pressure measurement and simulation [NQ20572]. Canada: Queen's University at Kingston (Canada); 1997.

15.   Lin C-T, Liu S-H, Wang J-J, Wen Z-C. Reduction of interference in oscillometric arterial blood pressure measurement using fuzzy logic. IEEE Transactions on Biomedical Engineering. 2003;50(4):432-41.

16.   Wang J-J, Lin C-T, Liu S-H, Wen Z-C. Model-based synthetic fuzzy logic controller for indirect blood pressure measurement. IEEE Transactions on Systems, Man and Cybernetics. 2002;32(3):306-15.

17.   Sebald DJ, Bahr DE, Kahn AR. Narrowband auscultatory blood pressure measurement. IEEE Transactions on Biomedical Engineering. 2002;49(9):1038-44.

18.   Watson S, Wenzel RR, di Matteo C, Meier B, Luscher TF. Accuracy of a new wrist cuff oscillometric blood pressure device: Comparisons with intraarterial and mercury manometer measurements. American Journal of Hypertension. 1998 1998/12;11(12):1469-74.

19.   Drzewiecki G, Hood R, Apple H. Theory of the oscillometric maximum and the systolic and diastolic detection ratios. Annals of Biomedical Engineering. 1994 1994;22(1):88-96.

20.   Baker PD, Westenskow DR, Kück K. Theoretical analysis of non-invasive oscillometric maximum amplitude algorithm for estimating mean blood pressure. Medical and Biological Engineering and Computing. 1997 1997;35(3):271-8.

21.   Lee JY, Kim JK, Yoon G. Digital envelope detector for blood pressure measurement using an oscillometric method. Journal of Medical Engineering & Technology. 2002;26(3):117-22.

22.   Sapinski A. Theoretical basis for proposed standard algorithm of blood pressure measurement by the sphygmooscillographic method. Journal of Clinical Engineering. 1997;22(3):171-4.

23.   Sapinski A. Standard algorithm of blood-pressure measurement by the oscillometric method. Medical and Biological Engineering and Computing. 1994 1994;32(5):599-600.

24.   Geddes LA. Handbood of Blood Pressure Measurement: The Humana Press Inc.; 1991.

25.   Kim-Gau N. Blood pressure measurement. Medical Electronics. 1997 Feb.

26.   Dorsett TJ. Application Of A Prediction And Smoothing Algorithm To Non-invasive blood Pressure Measurement. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 1991; 1991. p. 468-9.

27. Yamakoshi K, Rolfe P, Murphy C. Current developments in non-invasive measurement of arterial blood pressure. Journal of Biomedical Engineering. 1988 1988;10(2):130-7.

28. Tanaka S, Gao S, Nogawa M, Yamakoshi K. A New Non-invasive Device for Measuring Instantaneous Blood Pressure in Radial Artery Based on the Volume-Compensation Method. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 2003; 2003. p. 3149-52.

29. Tanaka S, Gao S, Nogawa M, Yamakoshi K-I. Noninvasive measurement of instantaneous radial artery blood pressure. IEEE Engineering in Medicine and Biology Magazine. 2005 2005;24(4):32-7.

30. Sato T, Nishinaga M, Kawamoto A, Ozawa T, Takatsuji H. Accuracy of a continuous blood pressure monitor based on arterial tonometry. Hypertension. 1993;21(6 I):866-74.

31. Hori C, Itakura K, Nogawa M, Shirakabe M, Kubota I, Tomoike H, et al. Estimation of aortic BP waveform from noninvasive radial tonometry; validation of FFT and ARX methods. 1997; 1997. p. 1142-5 vol.3.

32. Liu S-H, Lin C-T. A model-based fuzzy logic controller with Kalman filtering for tracking mean arterial pressure. IEEE Transactions on Systems, Man & Cybernetics. 2001;31(6):676-86.

33. Colak S, Isik C. Fuzzy pulse qualifier. Annual Conference of the North American Fuzzy Information Processing Society; 2004; 2004. p. 850-3.

34. Colak S, Isik C. Blood pressure estimation using neural networks. 2004 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications; 2004; 2004. p. 21-5.

35. Brinton B, Todd J., Cotter M, Bruno, Kailasam M, Mala T., Brown M, David L., Chio P, Shiu-Shin. Development and Validation of a Noninvasive Method to Determine Arterial Pressure and Vascular Compliance. The American Journal of Cardiology. 1997 1997/8/1;80(3):323-30.

36. Ursino M, Cristalli C. Mathematical analysis of the oscillometric technique for indirect blood pressure evaluation. Proceedings of the 16th Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 1994; 1994. p. 1286-7.

37. Ursino M, Cristalli C. A mathematical study of some biomechanical factors affecting the oscillometric blood pressure measurement. Biomedical Engineering. 1996;43(8):761-78.

38. Colak S, Isik C. Fuzzy oscillometric blood pressure classification. 22nd International Conference of the North American Fuzzy Information Processing Society - NAFIPS Proceedings; 2003; 2003. p. 208-13.

39. Colak S, Isik C. Systolic Blood Pressure Classification. Proceedings of the International Joint Conference on Neural Networks; 2003; 2003. p. 627-30.

40. Colak S, Isik C. Feature subset selection for blood pressure classification using orthogonal forward selection. Proceedings of the IEEE 29th Annual Northeast Bioengineering Conference; 2003; 2003. p. 122-3.

41. Lee JY, Kim JK, Yoon G. A digital envelope detection filter for blood pressure measurement. Engineering in Medicine and Biology Society, 2001 Proceedings of the 23rd Annual International Conference of the IEEE; 2001; 2001. p. 226-8 vol.1.

42. Zong W, Heldt T, Moody GB, Mark RG. An open-source algorithm to detect onset of arterial blood pressure pulses. Computers in Cardiology; 2003; 2003. p. 259-62.

43. Perfetto JC, Ruiz A, Mazzeo JR, D'Attellis C. Robust algorithm to extract systoles and diastoles from continuous blood pressure signal. Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 2003; 2003. p. 279-82.

44. Gratze G, Fortin J, Holler A, Grasenick K, Pfurtscheller G, Wach P, et al. A Software package for non-invasive, real-time beat-to-beat monitoring of stroke volume, blood pressure, total peripheral resistance and for assessment of autonomic function. Computers in Biology and Medicine. 1998;28(2):121-42.

45. Moraes JCTB, Cerulli M, Ng PS. Development of a new oscillometric blood pressure measurement system. Computers in Cardiology; 1999; 1999. p. 467-70.

46. Moraes JCTB, Cerulli M, Ng PS. A strategy for determination of systolic, mean and diastolic blood pressures from oscillometric pulse profiles. Computers in Cardiology; 2000; 2000. p. 211-4.

47. Devasahayam SR. Signals and Systems in Biomedical Engineering: Signal Processing and Physiological Systems Modeling. New York: Kluwer Academic/Plenum Publishers; 2000.

48. Bruce EN. Biomedical Signal Processing and Signal Modeling. New York: Wiley; 2001.

49. Peters TM, Williams JC, Bates JHT. The Fourier Transform in Biomedical Engineering. Boston: Birkhaìˆuser; 1998.

50. Carlson GE. Signal and linear system analysis. 2nd ed. New York: John Wiley; 1998.

51. Hanselman DC, Littlefield B. Mastering MATLAB 7. Upper Saddle River, NJ: Pearson/Prentice Hall; 2005.

52. Denbigh P. System Analysis and Signal Processing : with emphasis on the use of MATLAB. HArlow, England ; Reading, Mass.: Addison-Wesley; 1998.

53. Van de Vegte J. Fundamentals of digital signal processing. Upper Saddle River, NJ: Prentice Hall; 2001.

54.   Zaknich A. Neural networks for intelligent signal processing. River Edge, NJ World Scientific Pub.

55.   Negnevitsky M. Artificial intelligence : a guide to intelligent systems. 1st ed. New York: Addison Wesley; 2002.

56.   Demuth H, Beale M. Neural Network Toolbox. Natick, MA: The MathWorks, Inc.; 2006.

57.   Diamantaras KI, Kung SY. Principal component neural networks : theory and applications. New York: Wiley; 1996.

58.   Hu YH, Hwang J-N. Handbook of neural network signal processing. Boca Raton, FL: CRC Press; 2002.

59.   Bland JM, Altman DG. Measuring agreement in method comparison studies. Statistical Methods in Medical Research. 1999;8(2):135-60.

60.   Scherer, R.C., Shinwari, D., Kenneth J. De Witt, Zhang, C., Kucinschi, B.R., and Afjeh, A.A. Intraglottal pressure profiles for a symmetric and oblique glottis with a divergence angle of 10 degrees. Journal of Acoustical Society of America. 2001;109(4):1616-30.

61.   Mantha, S., Mongeau, L., and Siegmund, T. Dynamic digital image correlation of a dynmaic physical model of the vocal folds. International Mechanical Engineering Congress and Exposition; 2005; Orlando, Florida USA: ASME; 2005. p. 1-2.

APPENDIX

APPENDIX I   Operational Amplifier Circuit Configuration

CONNECTIONS

The negative pins of the sensor are connected together and then to ground.

Capacitance of the sensors (Cp) is taken between the positive pins and the common negative one (the pins connected together).

An external capacitor is connected in series (C1.1, C1.2, C1.3, C1.4 in the circuit above), to create an equivalent more constant capacitance.

It is used a resistor R = 500 Mohm connected to ground, and the cut-off frequency it is creating with the equivalent capacitance is specified by Fc.

Sensor	Buffer / Filter board							National Instruments		
	Wire colour	Input pin (channel)	Output pin	C1 [nF]	C_p (Cap. Of Piezo) [nF]	Ceq [nF]	Fc [Hz]	Chnl	Wire colour	Pin
Negative pins	Screen	1	-							
External	Red	3 (1)	1	0.82	~ 5.00	0.70	0.45	9	Yellow	66
Internal Large	Blue	4 (2)	2	1	~ 2.70	0.70	0.46	8	Blue	34
Internal Small	White	5 (3)	3	33	~ 0.70	0.68	0.47	0	White	68
NC	NC	6 (4)	4	-	-	-	-			
Pressure	-	-	-	-	-	-	-	1	Green	33
Ground									Grey	24
+ 5 V									Red	8

NOT USED CHANNELS FROM NATIONAL INSTRUMENTS CARD

Wire	Chnl	Pin
Orange	2	65
Pink	3	30
Black	11	63

PIEZO FILM SENSORS AND PRESSURE SENSOR BOARDS AND INTERFACE

CIRCUIT
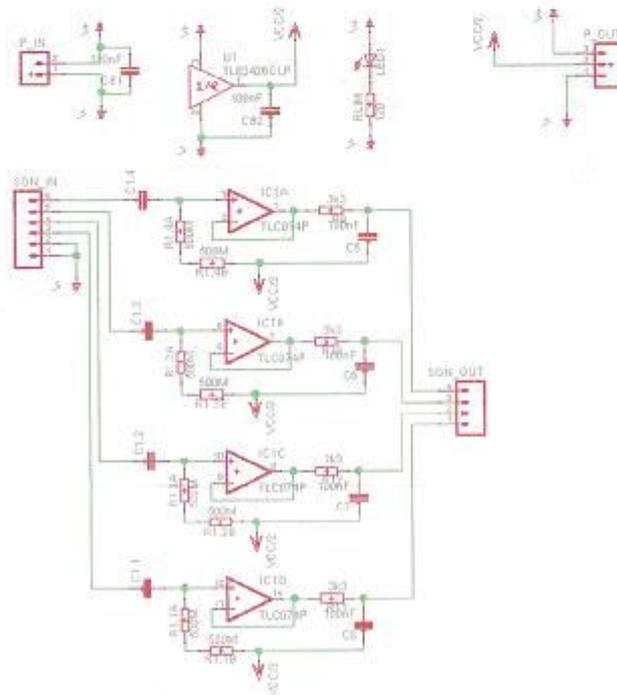
APPENDIX II   Ethics Approval Letter

AUT
U N I V E R S I T Y
TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

M E M O R A N D U M

To:	Ahmed Al-Jumaily
From:	**Charles Grinter** Ethics Coordinator
Date:	12 July 2006
Subject:	Ethics Application Number 06/126 **Specialised non-invasive blood pressure measurement algorithm.**

Dear Ahmed

Thank you for providing written evidence as requested. I am pleased to advise that it satisfies the points raised by the Auckland University of Technology Ethics Committee (AUTEC) at their meeting on 21 June 2006 and, acting for the Executive Secretary of AUTEC, I have approved your ethics application. This delegated approval is made in accordance with section 5.3.2.3 of AUTEC's *Applying for Ethics Approval: Guidelines and Procedures* and is subject to endorsement at AUTEC's meeting on 14 August 2006.

Your ethics application is approved for a period of three years until 12 July 2009.

I advise that as part of the ethics approval process, you are required to submit to AUTEC the following:

- A brief annual progress report indicating compliance with the ethical approval given using form EA2, which is available online through *http://www.aut.ac.nz/research/ethics*, including a request for extension of the approval if the project will not be completed by the above expiry date;

- A brief report on the status of the project using form EA3, which is available online through *http://www.aut.ac.nz/research/ethics*. This report is to be submitted either when the approval expires on 12 July 2009 or on completion of the project, whichever comes sooner;

You are reminded that, as applicant, you are responsible for ensuring that any research undertaken under this approval is carried out within the parameters approved for your application. Any change to the research outside the parameters of this approval must be submitted to AUTEC for approval before that change is implemented.

Please note that AUTEC grants ethical approval only. If you require management approval from an institution or organisation for your research, then you will need to make the arrangements necessary to obtain this.

To enable us to provide you with efficient service, we ask that you use the application number and study title in all written and verbal correspondence with us. Should you have any further enquiries regarding this matter, you are welcome to contact me by email at charles.grinter@aut.ac.nz or by telephone on 921 9999 at extension 8860.

On behalf of the Committee and myself, I wish you success with your research and look forward to reading about it in your reports.

Yours sincerely

Charles Grinter
Ethics Coordinator

On behalf of Madeline Banda, Executive Secretary, AUTEC

Cc: Han-Chun (Vivien) Lin hanlin06@aut.ac.nz

From the desk of …	Private Bag 92006, Auckland 1020	Tel: 64 9 921 9999
Madeline Banda	New Zealand	ext 8044
Academic Services	E-mail: madeline.banda@aut.ac.nz	Fax: 64 9 921 9812
Student Services		page 1 of 1

122

Participant Information Sheet

AUT
U N I V E R S I T Y
TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

Date Information Sheet Produced:

29 May 2006

Project Title

Specialised Non-Invasive Blood Pressure Measurement Algorithm

An Invitation

My name is Vivien Lin and I am currently studying for my Masters degree in Engineering at the Diagnostics and Control Research Centre, Auckland University of Technology.

For many people, a visit to their GP generally involves having their blood pressure measured. Although the traditional mercury blood-pressure measurement device (sphygmomanometer) is fairly accurate, it is also potentially quite dangerous and inconvenient to handle.

This research is involved with the testing of a new, low-risk device that can accurately measure blood pressure. My project is specifically focussed on the development of an algorithm for this device. There are two phases in the current project: the first phase is to develop the algorithm and the second phase will be algorithm verification.

The first phase is now complete, and so I am moving into the second phase of verifying the algorithm. I need volunteers to help with this research and I would very much like you to consider participating in this.

What is the purpose of this research?

This research intends to create a tunable blood-pressure measurement algorithm for an existing measurement device developed by Pulsecor Ltd. It is funded through a Technology for Industry Fellowship (TIF) which has been awarded by the Foundation for Research, Science and Technology (FRST).

This part of my project involves the collection of data which will be used to verify the new blood pressure device algorithm.

How was I chosen for this invitation?

All people with systolic blood pressure between 80 to 200 mmHg and diastolic blood pressure between 40 to 120 mmHg are welcome to participate in this study. These limits have been set by the Association for the Advancement of Medical Instrumentation (AAMI) in order to guarantee the accuracy of the device.

I believe your participation in this research will contribute to the success of the algorithm development and thus produce a new device that will benefit the general public in terms of cardiovascular health.

What will happen in this research?

This part of the project aims to verify the algorithm being developed. Therefore, your blood pressure will be measured and your general age group recorded. The measurement process is complete pain-free, and only consumes a couple of minutes of your time.

How will my privacy be protected?

You will be given a number which will be used instead of your name to ensure the confidentiality of the data collected. Data will be stored in a locked cabinet, and only I will have access to the data.

What opportunity do I have to consider this invitation?

You will be given as much time as you like to consider this invitation. I would also be happy to answer any questions you may have. Furthermore, since your participation is voluntary, you may withdraw from this program at any time and for any reason without penalty.

How do I agree to participate in this research?

Please sign the consent form provided with this information sheet in order to participate. Your blood pressure and general age group will then be recorded. You may also keep this participant information sheet if you wish and a copy of the consent form.

Will I receive feedback on the results of this research?

Yes, I will report back on your blood pressure readings and/or provide access to a summary of the findings if you are interested.

What do I do if I have concerns about this research?

Any concerns regarding the nature of this project should be notified in the first instance to the Project Supervisor, Professor Ahmed Al-Jumaily, *ahmed.aljumaily@aut.ac.nz*, 921 9777.

Concerns regarding the conduct of the research should be notified to the Executive Secretary, AUTEC, Madeline Banda, *madeline.banda@aut.ac.nz* , 921 9999 ext 8044.

Whom do I contact for further information about this research?

Researcher Contact Details:

Name: Han-Chun Lin   e-mail: hanlin06@aut.ac.nz   Phone: 921-9999 ext 8617

Project Supervisor Contact Details:

Name: Professor Ahmed Al-Jumaily   e-mail: ahmed.aljumaily@aut.ac.nz   Phone: 921-9777

APPENDIX IV   Consent Form

<table>
<tr>
<td></td>
<td>

Consent Form

</td>
<td>

AUT
U N I V E R S I T Y
TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

</td>
</tr>
</table>

Project title: Specialised Non-Invasive Blood Pressure Measurement Algorithm

Project Supervisor: Professor Ahmed Al-Jumaily

Researcher: Han-Chun (Vivien) Lin

○   I have read and understood the information provided about this research project in the Information Sheet dated 29 May 2006.

○   I have had an opportunity to ask questions and to have them answered.

○   I understand that I may withdraw myself or any information that I have provided for this project at any time prior to completion of data collection, without being disadvantaged in any way.

○   I understand that if I did not request to withdraw myself after the end of the data collection, my data will be used in the research and cannot be withdrawn.

○   I agree to take part in this research.

○   My data collected will be given a number instead of a name to ensure my confidentiality.

Participant's signature: ....…………….………….………………………………

Participant's name: …..……………………………………………….…………

Date: ....………………………………….………………………….…………..

Approved by the Auckland University of Technology Ethics Committee on 12 July 2006 AUTEC Reference number 06/126

Note: The Participant should retain a copy of this form.

APPENDIX V   Heart Beat / Rate Determination Function Codes

```matlab
%% Heart beat detection

function [ind,HBind,N]=funHB(out,outfilt,Fs)

lowFc=2;
while 1
    %low pass filter
    [b3,a3] = butter(2,lowFc/Fs*2,'low');
    outlow=filtfilt(b3,a3,out);

    %find min pt from outside sensor
    MinPeakInd=funFindMin(outlow)+1;
    HBind=[];
    Beat=[];
    for i=1:length(MinPeakInd)-1
        %find max point from outside sensor.
        [MaxOut indices]=max(outfilt(MinPeakInd(i):MinPeakInd(i+1)));
        %find min point from outside sensor.
        [MinOut MinInd]=min(outfilt(MinPeakInd(i):MinPeakInd(i+1)));
        indices=indices+MinPeakInd(i)-1;
        HBind(i)=indices;

        if i>1
            %calculate beat rate in beats per second
            Beat(i-1)=Fs/(HBind(i)-HBind(i-1));
        end
    end

    %Check all HB detection
    meanBeat=mean(Beat(3:end-3));
    i=1;
    while i<=length(Beat)
        if (Beat(i)<meanBeat/1.95)||(Beat(i)>meanBeat*1.6)
            if i==1
                if outfilt(HBind(1))> outfilt(HBind(2))
                    HBind=[HBind(1) HBind(3:end)];
                else
                    HBind=HBind(2:end);
                end
            else
                if outfilt(HBind(i))> outfilt(HBind(i+1))
                    HBind=[HBind(1:i) HBind(i+2:end)];
                else
                    HBind=[HBind(1:i-1) HBind(i+1:end)];
                end
            end
            %Calculate new Beat Rate
            Beat=[];
            for i=1:length(HBind)-1
                Beat(i)=Fs/(HBind(i+1)-HBind(i));
            end
        end
        i=i+1;
    end

    %Check first and last 3 HB detection
    meanBeat=mean(Beat(3:end-3));
    for i=1:3
        while (Beat(i)<meanBeat/1.95)||(Beat(i)>meanBeat*1.6)
            Beat=Beat(i+1:end);
            HBind=HBind(i+1:end);
        end
    end

    for i=0:2
        while (Beat(end-i)<meanBeat/1.95)||(Beat(end-i)>meanBeat*1.6)
```

```matlab
            Beat=Beat(1:end-i-1);
            HBind=HBind(1:end-i-1);
        end
    end

    %Calculate new Beat Rate
    Beat=[];
    for i=1:length(HBind)-1
        Beat(i)=Fs/(HBind(i+1)-HBind(i));
    end

    if length(HBind)==1
        rate=0.5;
    else
        rate=0.8;
        N=round(mean(diff(HBind))*rate);  %calculate number of Hann window
        while min(diff(HBind))<N && rate>0.5
            rate=rate-0.01;
            N=round(mean(diff(HBind))*rate);
        end
    end
    if rate >= 0.7
        break;
    else
        lowFc=lowFc-0.1
        HBind=[];
        Beat=[];
    end %end
    if lowFc<0.5
        disp(['Heart Rate is outside the calculation range.']);
        break;
    end
end %end while 1

meanHR=mean(Beat)*60;
disp(['Average Heart Rate: ' num2str(round(meanHR))]);
HalfBeat=0.3/mean(Beat);    %second per half beat
ind=HBind-round(HalfBeat*Fs); %shift average half beat forward
```

APPENDIX VI   Feature Extraction Function Codes

```matlab
%*********************************************************************
%% Envelope detection

function [MinEnv,Env1]=funEnv(HBind,ind,outfilt,t)

%develope an envelope of the outside sensor
for i=1:length(HBind)
    splx(i)=outfilt(HBind(i));
    splt(i)=t(HBind(i));
    if i>1
        [splx2(i-1) mInd]=min(outfilt(HBind(i-1):HBind(i)));
        splt2(i-1)=t(mInd+HBind(i-1)-1);
    end
end
% Positive envelope use spline function
splxx=spline(splt,splx,t);
splxx(1:HBind(1))=0;
splxx(HBind(end):end)=0;

splxx2=spline(splt2,splx2,t);
splxx2(1:find(splt2(1)==t))=0;
splxx2(find(splt2(end)==t):end)=0;

for i=1:length(ind)-1
    MaxEnv(i)=max(splxx(ind(i):ind(i+1)));
    MinEnv(i)=min(splxx2(ind(i):ind(i+1)));
    Env1(i)=MaxEnv(i)-MinEnv(i);
End
%*********************************************************************


%*********************************************************************
%%function to find mean magnitude for different frequency range

function Mag=funMag(xx,tt,N,Fs)

%Hanning Window
Window = hann(N);
xxWin = xx(1:N).*Window;

%fft calculation
xxfft=fft(xxWin);
NoX=ceil(length(xxfft)/2);
HalfX=xxfft(1:NoX);
F=Fs*(0:NoX-1)/length(xxfft);

Y=HalfX;

ind5=min(find(F>5));
ind10=min(find(F>10));
ind15=min(find(F>15));
ind20=min(find(F>20));
ind25=min(find(F>25));
ind35=max(find(F<35));

%calculate mean magnitude
Mag(1)=mean(Y(ind5:ind35));    %freq between 5~35Hz
Mag(2)=mean(Y(ind10:ind35));
Mag(3)=mean(Y(ind15:ind35));
Mag(4)=mean(Y(ind20:ind35));
Mag(5)=mean(Y(ind25:ind35));
%*********************************************************************
```

```matlab
%************************************************************************
%%function to find mean PSD for different frequency range

function PSD=funPSD(xx,tt,N,Fs)

%Hanning Window
Window = hann(N);
xxWin = xx(1:N).*Window;

%fft calculation
xxfft=fft(xxWin);
NoX=ceil(length(xxfft)/2);
F=Fs*(0:NoX-1)/length(xxfft);

%PSD
Pxx=xxfft.*conj(xxfft)/length(xxfft);
HalfPxx=Pxx(1:NoX);

P=HalfPxx;

ind5=min(find(F>5));
ind10=min(find(F>10));
ind15=min(find(F>15));
ind20=min(find(F>20));
ind25=min(find(F>25));
ind35=max(find(F<35));

%calculate mean PSD
PSD(1)=mean(P(ind5:ind35,:));
PSD(2)=mean(P(ind10:ind35,:));
PSD(3)=mean(P(ind15:ind35,:));
PSD(4)=mean(P(ind20:ind35,:));
PSD(5)=mean(P(ind25:ind35,:));
%************************************************************************




%************************************************************************
%%function to find total amplitude of each peak

function Amp=funAmp(x)

%find turning pt
TPInd=funTurnPt(x)+1;

FirstInd=max(find(x(1:TPInd(1))==0));
LastInd=min(find(x(TPInd(end):end)==0))+TPInd(end)-1;
TPInd=[FirstInd TPInd LastInd];

Amp=0;
for j=1:length(TPInd)-1
    Amp=Amp+abs(x(TPInd(j+1))-x(TPInd(j)));
End
%************************************************************************




%************************************************************************
%%function to find area under the curve

function Area=funArea(x)

xx=abs(x);
%Calculate area under the curve
Area=trapz(xx);
%************************************************************************
```

```matlab
%***********************************************************************
%%function to find dPdt for each point
%return two values; max/min dPdt
function dPdt=fundPdt(x,t)

dpdt=[];
dpdt(1:length(x))=0;
for j=2:length(x)-1
    %calculate dP/dt at each point
    dpdt(j)=(x(j+1)-x(j-1))/(t(j+1)-t(j-1));
end

dPdt(1)=max(dpdt);   %max dPdt
dPdt(2)=min(dpdt);   %min dPdt
%***********************************************************************




%***********************************************************************
%function to count peaks above different thresholds

function Peak=funPeak(Norxx)

%find turning pt
MinTPInd=funFindMin(Norxx)+1;
MaxTPInd=funFindMax(Norxx)+1;

Peak(1)=length(find(Norxx(MaxTPInd)>0.1));
Peak(2)=length(find(Norxx(MaxTPInd)>0.3));
Peak(3)=length(find(Norxx(MaxTPInd)>0.5));
Peak(4)=length(find(Norxx(MaxTPInd)>0.7));
%***********************************************************************




%***********************************************************************
%function to find the max. positive/negative rate of change

function ROC=funROC(xx,tt)

TPInd=funTurnPt(xx)+1;
FirstInd=max(find(xx(1:TPInd(1))==0));
LastInd=min(find(xx(TPInd(end):end)==0))+TPInd(end)-1;
TPInd=[FirstInd TPInd LastInd];

roc=[];
for j=1:length(TPInd)-1
    %calculate rate of change
    roc(j)=(xx(TPInd(j+1))-xx(TPInd(j)))/(tt(TPInd(j+1))-tt(TPInd(j)));
end

ROC(1)=max(roc);
ROC(2)=min(roc);
%***********************************************************************
```

130

APPENDIX VII   Other Used Function Codes

```matlab
%**********************************************************************
%%function to find each turning point
%return indices
function ind=funTurnPt(x)

xDiff=diff(x);
j=1;
ind=[];
for i=1:length(xDiff)-1
    if (xDiff(i)>0 && xDiff(i+1)<0) || (xDiff(i)<0 && xDiff(i+1)>0)
        ind(j)=i;
        j=j+1;
    end
end
if isempty(ind)
    ind=1;
end
%**********************************************************************




%**********************************************************************
%%function to find maximum turning point
%return indices
function ind=funFindMax(x)

xDiff=diff(x);
j=1;
ind=[];
for i=1:length(xDiff)-1
    if xDiff(i)>0 && xDiff(i+1)<0
        ind(j)=i;
        j=j+1;
    end
end
if isempty(ind)
    ind=1;
end
%**********************************************************************




%**********************************************************************
%%function to find minimum turning point
%return indices
function ind=funFindMin(x)

xDiff=diff(x);
j=1;
ind=[];
for i=1:length(xDiff)-1
    if xDiff(i)<0 && xDiff(i+1)>0
        ind(j)=i;
        j=j+1;
    end
end
if isempty(ind)
    ind=1;
end
%**********************************************************************
```

APPENDIX VIII Final Algorithm Codes

```matlab
%% BP Estimation Program

% Get source signal (Raw Data)
filename=['exp1_1.xls'];
d.data=dlmread(filename,'\t',1,0);
d.Fs=dlmread(filename,'\t',[0,0,0,0]);

%find Max cuff pressure
MaxInd=find(d.data(:,4)==max(d.data(:,4)));
%ignore inflating part
d.data=d.data(MaxInd:end,:);

% frequency domain filter
[b2,a2]=butter(3, 10/d.Fs*2); % filter for cuff pressure
p=(filtfilt(b2,a2,d.data(:,4))-0.2703)/0.0131;

x=[d.data(:,1) d.data(:,2) d.data(:,3)];
t=(0:size(d.data,1)-1)/d.Fs;

%filtering
[b4,a4]=butter(4,[0.5 30]/d.Fs*2);
xfilt=filtfilt(b4,a4,x);
pfilt=filtfilt(b4,a4,p);
infilt=xfilt(:,2)-xfilt(:,3);
outfilt=xfilt(:,1);

% Get each heart beat
[ind,HBind,N]=funHB2(x(:,1),outfilt,d.Fs);
NegInd=find(ind<0);
if isempty(NegInd)==0 ind=ind(max(NegInd)+1:end); end;
outfilt(1:ind(1))=0;
outfilt((end-round(ind(end)))/2+ind(end):end)=0;

% Get envelope
[MinEnv,Env1]=funEnv(HBind,ind,outfilt,t);

for i=1:length(ind)-1
    xx=outfilt(ind(i):ind(i+1)-1);
    tt=t(ind(i):ind(i+1)-1);
    pp=pfilt(ind(i):ind(i+1)-1);

    %Find starting pulse pressure of each heart beat
    MaxPInd=find(pp==max(pp));       %find max pt for each heart beat
    MinPeakPInd=funFindMin(pp(1:MaxPInd))+1;
    if length(MinPeakPInd)==1
        BPInd(i)=ind(i)+MinPeakPInd;    %find index for beat pressure
        BP(i)=p(ind(i)+MinPeakPInd);    %find pressure for each heart beat
    elseif min(pp(MinPeakPInd))<0
        ppMinInd=find(pp(MinPeakPInd)<0);
        BPInd(i)=ind(i)+MinPeakPInd(max(ppMinInd)); %find index for beat pressure
        BP(i)=p(ind(i)+MinPeakPInd(max(ppMinInd))); %find pressure for each HB
    else
        BPInd(i)=ind(i)+max(MinPeakPInd); %find index for beat pressure
        BP(i)=p(ind(i)+max(MinPeakPInd)); %find pressure for each HB
    end

    %Get Mag and PSD features
    mag(i,:)=funMag(xx,tt,N,d.Fs);   %mag=[m5 m10 m15 m20 m25]
    psd(i,:)=funPSD(xx,tt,N,d.Fs);   %psd=[p5 p10 p15 p20 p25]

    %Normalise each heart beat
    shiftValue=MinEnv-0;
    shiftmEnv=MinEnv-shiftValue;
    shiftxx=xx-shiftValue(i);
    Norxx=shiftxx/Env1(i);   %Normalise
```

```matlab
    %Hanning Window
    Window = hann(N);
    xxWin = Norxx(1:N).*Window;
    xxWin(N+1:length(Norxx))=0;

    %Get features from time domain
    TotAmp(i)=funAmp(xxWin);  %Calculate total amplitude for this heart beat
    Area(i)=funArea(xxWin);
    dpdt(i,:)=fundPdt(xxWin,tt);    %dPdt=[maxPosdPdt maxNegdPdt]
    peak(i,:)=funPeak2(xxWin);      %peak=[M1 M3 M5 M7]
    roc(i,:)=funROC(xxWin,tt);      %ROC=[maxPosROC maxNegROC]
end

%display feature matrix
Feature=[[TotAmp]' [Area]' abs(dpdt) peak abs(roc) abs(mag) abs(psd) [Env1]'];
%Normalise Feature
[m,n]=size(Feature);
Max=ones(m,1)*max(Feature);
Feature=[Feature./Max]';
Feature(isnan(Feature))=0;      %replace NaN = 0; happend when max(peak)=0;
%calculate all values into the range between -1~1
Feature=2*Feature-1;

%ANN simulation
Ip=Feature;
load('ANN21_3_0','net1')      %load weight and bias values
a=sim(net1,Ip);

%2nd ANN simulation
for i=1:length(a)-2
    Ip2(:,i)=[a(:,i);a(:,i+1);a(:,i+2)];      %get 2nd layer input
end
load('Net2ANN21_3_0', 'net3')    %load weight and bias values
b=sim(net3,Ip2);
b8=int8(b); %convert b to intergers
OutPutInd=[];
OutPutInd=find(b8==1);
%find SBP
SBP=0;
if isempty(OutPutInd) disp(['NO Blood Pressure Detected']);
else
    for i=1:length(OutPutInd)-2
        if (OutPutInd(i+1)==OutPutInd(i)+1)&&(OutPutInd(i+2)==OutPutInd(i)+2)
            SBP=BP(OutPutInd(i));
            break;
        end
    end
    if SBP==0 disp(['NO Systolic Blood Pressure Detected ']); end;
end

b8=[0 b8 0];
OutPutInd=[];
OutPutInd=find(b8==1);
%find DBP
DBP=0;
if isempty(OutPutInd) disp(['NO Blood Pressure Detected ']);
else
    DBP=0;
    for i=length(OutPutInd):-1:3
        if (OutPutInd(i-1)==OutPutInd(i)-1)&&(OutPutInd(i-2)==OutPutInd(i)-2)
            DBP=BP(OutPutInd(i));
            break;
        end
    end
    if DBP==0 disp(['NO Diastolic Blood Pressure Detected ']); end;
end

%Display estimated BP
disp(['BP = ' num2str(round(SBP)) '/' num2str(round(DBP))]);
```