

# **Unsupervised Segmentation of Heart Sounds**

submitted by

**Patrick Beirne, B.A.Sc.**

A thesis submitted to the  
Faculty of Graduate Studies and Research  
in partial fulfillment of the  
requirements for the degree of

**Masters of Applied Science**

Ottawa-Carleton Institute for Electrical and Computer Engineering

Faculty of Engineering

Department of Computer and Systems Engineering

Carleton University

Ottawa, Ontario, Canada K1S 5B6

September, 2006

© 2006 Patrick Beirne



Library and  
Archives Canada

Bibliothèque et  
Archives Canada

Published Heritage  
Branch

Direction du  
Patrimoine de l'édition

395 Wellington Street  
Ottawa ON K1A 0N4  
Canada

395, rue Wellington  
Ottawa ON K1A 0N4  
Canada

*Your file* *Votre référence*  
*ISBN: 978-0-494-23327-6*  
*Our file* *Notre référence*  
*ISBN: 978-0-494-23327-6*

#### NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

#### AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

---

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

  
**Canada**

# Abstract

Digitized and recorded heart sounds contain significant diagnostic information, and are easily acquired in clinical practice. Part of a hypothetical diagnostic system is the *segmentation subsystem* which locates the boundaries between heartbeats. This thesis examines the design and performance of two approaches to segmentation: *peak energy detection* and *sliding window autocorrelation*. Emphasis is placed on synchronous detection of the heartbeats, so that subsequent subsystems can superimpose the heartbeats.

An experimental database of heart sounds was compiled to assess the performance of the proposed algorithms. 206 sound files were gathered from 4 sources; the database includes 2709 heartbeats and covers a range of sound quality and complexity.

A *heart rate estimator* is presented and tested, yielding a meaningful beat period in 90-96% of typical heart sounds.

The *peak energy* segmentation algorithm compares the signal energy against a threshold proportional to the signal's local average energy. It correctly segments 64% of the heartbeats from the experimental database, indicating the difficulty of segmentation in general.

The novel *sliding window autocorrelation* segmenter is designed to take advantage of the self consistency of heart sounds. The algorithm operates by calculating multiple local autocorrelations in order to estimate the time offset to the next heart beat. A change in the time offset marks the boundary between heartbeats. It correctly segmented 83% of the heartbeats.

# Acknowledgments

I would like to sincerely thank my supervisors, Dr. Rafik Goubran and Dr. Adrian Chan for their encouragement, guidance and patience during the course of this work, and for their assistance in editing this thesis.

Dr. Jane Lougheed at the Children's Hospital of Eastern Ontario contributed the original requirements for a sound analysis system, and stimulated the ideas presented here. In addition, she contributed several sounds used in the experimental database (chapter 3).

# Table of Contents

<b>Abstract</b> .....	<b>iii</b>
<b>Acknowledgments</b> .....	<b>iv</b>
<b>Table of Contents</b> .....	<b>v</b>
<b>Index of Tables</b> .....	<b>vii</b>
<b>Index of Figures</b> .....	<b>vii</b>
<b>List of Abbreviations</b> .....	<b>ix</b>
<b>Chapter 1 Introduction</b> .....	<b>1</b>
1.1 The Importance of Synchronous Segmentation.....	2
1.2 Objectives.....	3
1.3 Thesis Structure.....	4
1.4 Thesis Contributions.....	5
<b>Chapter 2 Background</b> .....	<b>6</b>
2.1 Heart Murmurs.....	9
2.2 Medical Uses.....	13
2.3 A Murmur Analysis System.....	14
2.4 Segmentation Methods.....	16
2.4.1 Existing Methods.....	16
2.4.2 Segmentation in the Literature.....	18
2.4.3 Peak Detection.....	22
2.4.4 Sliding Window Autocorrelation.....	23
<b>Chapter 3 Experimental Setup</b> .....	<b>24</b>
3.1 Heart Sounds.....	24
3.2 Experimental Setup.....	29
3.3 Processing Techniques.....	31
3.3.1 Similarity Correlation.....	31
3.3.2 Matching Correlation Curves to Gaussian Template.....	35
<b>Chapter 4 Heart Rate Estimation</b> .....	<b>49</b>
4.1 Implementation.....	53
4.2 Experimental Results.....	54
<b>Chapter 5 Peak Detecting Segmentation</b> .....	<b>59</b>
5.1 Introduction.....	59
5.2 Block diagram.....	60
5.3 Implementation.....	63
5.4 Results & Discussion.....	64
<b>Chapter 6 Sliding Window Autocorrelating Segmentation</b> .....	<b>71</b>
6.1 Physiology of Heartbeat Timing.....	72
6.2 Justification for Energy/Envelope Detection.....	74
6.3 Waveform Correlation.....	76
6.4 Algorithm Summary.....	78
6.4.1 Phase 0: Normalize.....	79
6.4.2 Phase 1: Heart Rate Estimation.....	79

6.4.3 Phase 2: Sliding Correlation .....	80
6.4.4 Phase 3: Peak Detection.....	82
6.4.5 Phase 4: Left and Right Beat Search.....	83
6.5 Details and Justification.....	84
6.6 Experimental Results.....	87
6.7 Factors Affecting Segmentation.....	92
6.8 Conclusion.....	93
<b>Chapter 7 Conclusions.....</b>	<b>94</b>
7.1 Future Work.....	95
7.1.1 Rate Estimation.....	95
7.1.2 Segmentation.....	96
<b>References.....</b>	<b>98</b>

# List of Tables

Table 3.1: Database of heart sounds, broken down by murmur type.....	25
Table 3.2: Database of heart sounds, broken down complexity.....	28
Table 3.3: Comparison of the size of experimental databases.....	29
Table 3.4: Curve matching output (sound file: pda5.wav).....	44
Table 3.5: Curve matching output (sound file: pda.wav).....	45
Table 3.6: Peaks of autocorrelation (sound file: pda5.wav).....	47
Table 3.7: Peaks of autocorrelation (sound file: pda.wav).....	48
Table 4.1: Performance of the rate estimator against the experimental data.....	55
Table 5.1: Performance of the peak energy segmenter.....	65
Table 6.1: Performance of the sliding window autocorrelation segmenter .....	87

# List of Figures

Figure 1.1: Peak energy segmentation system.....	2
Figure 1.2: Sliding window autocorrelation segmentation system.....	2
Figure 2.1: Heart valves.....	7
Figure 2.2: Typical heart sound.....	8
Figure 2.3: Heart sound showing S3 and S4.....	9
Figure 2.4: Mid systolic murmur, with a rise and fall profile, typical of aortic stenosis.....	9
Figure 2.5: Continuous murmur, typical of PDA.....	10
Figure 2.6: Diastolic murmur, constant throughout, typical of mitral stenosis.....	10
Figure 2.7: Early systolic murmur, typical of mitral regurgitation and atrial septal defect .	11
Figure 2.8: Holosystolic murmur of mitral regurgitation.....	12
Figure 2.9: Holodiastolic murmur of mitral stenosis.....	12
Figure 2.10: Block diagram of a complete analyzer for heart sounds.....	14
Figure 2.11: S2 louder than S1.....	17
Figure 2.12: S1 louder than S2.....	17
Figure 2.13: S1 and S2 indistinct.....	17
Figure 3.1: Typical waveform of complexity "1"; clear S1 and S2 sounds.....	27
Figure 3.2: Typical sound of complexity "2"; S1 and S2 not obvious, but no large interference.....	27
Figure 3.3: Example of complexity "3", note the irregular beats.....	27
Figure 3.4: Example of complexity "3"; heartbeats is buried under breathing noises.....	28
Figure 3.5: Typical segmented sound file.....	30
Figure 3.6: Example of the autocorrelation process.....	32
Figure 3.7: Breakdown of the similarity calculation.....	34
Figure 3.8: A typical autocorrelation and the components of a curve match.....	36
Figure 3.9: The autocorrelation of the raw signal, with curve match.....	42
Figure 3.10: Autocorrelation of $x^2$ .....	42
Figure 3.11: Autocorrelation of $ \text{Hilbert}(w(x)) $ , with curve match.....	43
Figure 3.12: Autocorrelation of $ x $ , with curve match.....	43

Figure 3.13: Clean heart sound of patent ductus artereosis (pda5.wav, first 5 s).....	44
Figure 3.14: Waveform of patent ductus arteriosus (pda.wav, first 5 s).....	45
Figure 3.15: Waveform of patent ductus arteriosus (pda.wav, second 5 s).....	46
Figure 4.1: Typical heart sound and its autocorrelation.....	50
Figure 4.2: Autocorrelation of the signal energy, $Re(j)$ .....	51
Figure 4.3: Windows used on the autocorrelation to isolate the most likely rate peaks.....	52
Figure 4.4: Heart sound with emphasis on alternate beats.....	57
Figure 4.5: Heart sound with alternate sounds emphasized.....	57
Figure 4.6: Scatter graph of the performance of the rate estimation algorithm.....	58
Figure 5.1: Typical heart sound showing variation in magnitude over time.....	59
Figure 5.2: Block diagram of the peak energy segmenter.....	61
Figure 5.3: Peak energy detector on heart sound with distinct S1 and S2.....	61
Figure 5.4: Peak energy detector on heart sound with holosystolic murmur.....	62
Figure 5.5: Peak energy detector operating on a dominant S2.....	62
Figure 5.6: Block diagram of final peak energy-based segmenter.....	64
Figure 5.7: Peak energy segmenter operating on a sample with distinct S1 and S2.....	66
Figure 5.8: Peak energy segmenter operating on a sound with holosystolic murmur.....	67
Figure 5.9: Peak energy segmenter with sample that includes a missing beat.....	68
Figure 5.10: Peak energy segmenter with inconsistent presystolic murmur.....	68
Figure 5.11: Peak energy segmenter with a strong S2 waveform.....	69
Figure 6.1: intracellular potential of SA node.....	73
Figure 6.2: Two complete beats of a patient with chronic atrial regurgitation.....	75
Figure 6.3: Single heartbeat, details at time=0.9 seconds.....	75
Figure 6.4: Single heartbeat, details at time=1.6 seconds.....	76
Figure 6.5: Autocorrelation of the waveform in Figure 6.1.....	76
Figure 6.6: Heart sound, showing local variations in beat period.....	77
Figure 6.7: Heartbeat magnitude, showing a template and search field.....	80
Figure 6.8: Sliding window correlations.....	81
Figure 6.9: Sliding window autocorrelation maximums overlaid in the original signal magnitudes.....	82
Figure 6.10: sliding window autocorrelation with the best beat identified.....	83
Figure 6.11: Diastolic murmur of aortic regurgitation, segmented and marked correctly by the SWA algorithm.....	88
Figure 6.12: Diastolic murmur of aortic regurgitation, segmented and marked incorrectly by the peak energy algorithm (chapter 5).....	88
Figure 6.13: Diastolic murmur of aortic regurgitation, properly segmented.....	89
Figure 6.14: Heart sound including breathing noises, segmented incorrectly by the peak energy algorithm.....	89
Figure 6.15: Heart sound including breathing noises, segmented correctly by the sliding window autocorrelation algorithm.....	90
Figure 6.16: Stills murmur, marked correctly by a modified SWA algorithm.....	91
Figure 6.17: Stills murmur, marked incorrectly by the SWA segmentation algorithm.....	91
Figure 6.18: Pulmonary valve stenosis, marked incorrectly by the SWA algorithm.....	92

# List of Abbreviations

AGC	Automatic Gain Control
CHEO	Children's Hospital of Eastern Ontario
DSP	Digital Signal Processing
ECG	Electrocardiogram
PCG	Phonocardiogram
RMS	Root Mean Square
SWA	Sliding Window Autocorrelation

## Chapter 1

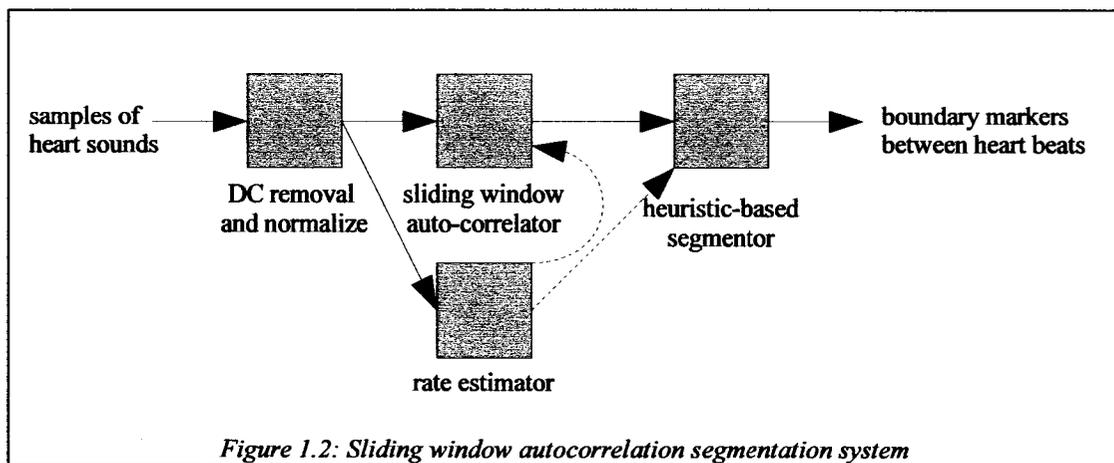
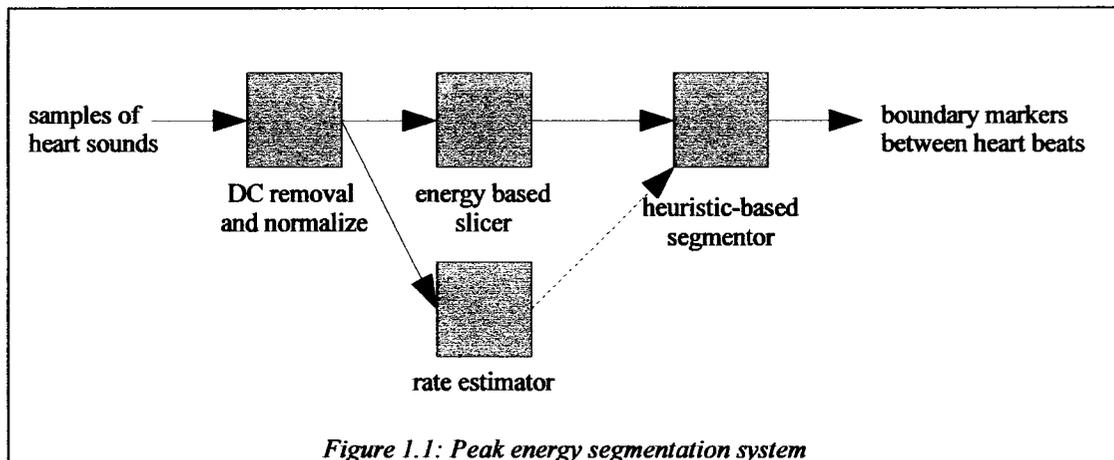
# Introduction

Heart sounds, or phonocardiograms (PCG) contain many clues to the health of the heart. In standard medical practice, listening to heart sounds is the principal technique used to detect heart disease. The skills associated with interpreting heart sounds are not obvious, usually requiring professional services.

An automated system for the analysis of PCG data could help in clinical situations, as a screening device or as an aid to diagnosis. An early step in any analysis is to locate the time boundaries of the heartbeats, i.e. to *segment* the heart sounds. Using the results of segmentation, the beats themselves may be analyzed or aggregated.

This thesis examines the design of two segmentation methods, quantifies the performance of these methods, identifies heart sounds which have proven difficult to work with, and suggests future research areas.

Specifically, this thesis compares the performance of two segmentation systems: a *peak energy* algorithm and a *sliding window autocorrelation* algorithm, as illustrated in Figures 1.1 and 1.2, respectively. These algorithms are then tested against several thousand heartbeats from an experimental database created for this work. Both systems are capable of marking 65-90% of the beats in the database.



## 1.1 The Importance of Synchronous Segmentation

The detection of individual heart beats is an important tool for many kinds of work. Most current methods of heart beat detection simply trigger on the presence of energy in the sound. For instance, the classic Holter monitor records heart beats during the day, noting missing beats and the average heart rate. For this kind of analysis, the exact phase of the heart sounds is not critical, as the analysis is mostly concerned with missing beats and timing.

Other sorts of analysis can benefit from *synchronous* detection of the beat boundary. That

is, the segmentation process splits the sound at a consistent point in the cycle, just before the S1 sound (see chapter 2).

For example, with synchronous segmentation, multiple heart beats may be superimposed. Significant elements of the sound will be reinforced and uncorrelated noises, such as room noise or movement will be suppressed. This may allow the processing of PCG signals with much lower signal to noise ratios, potentially even through layers of clothing, or from non-traditional locations on the body.

Heart rate variability studies can be more precise if the boundary of the heart sound is known.

Some diseases cause loud murmurs between heart sounds (holodiastolic), while others have loud murmurs in the middle of the sound (holosystolic). A classic peak detector can easily count the beats, but not separate these two types. Murmur detection and classification depends on knowing the difference between these, and synchronous segmentation distinguishes them.

## **1.2 Objectives**

- Develop a robust segmentation algorithm that
  - ◆ runs unsupervised
  - ◆ handles input data of varying quality
  - ◆ yields segmentation marks that are synchronous with the heart sound
  - ◆ may be used as an element of a complete murmur analysis system

- Develop a measurement system to quantify the performance of the algorithm

The resulting system can be used standalone as a device for studying heart rate variability and heart beat variability. As part of a system, it may be used to aggregate multiple heart beats that are affected by uncorrelated noise, thereby increasing the signal to noise ratio. Or it may be used to isolate heart beat irregularities.

### 1.3 Thesis Structure

In Chapter 2 the medical background behind heart sounds is presented along with an overview of the previous work in heart sound segmentation.

Chapter 3 explains the experimental setup, and discusses the corpus of heart sounds that was created as a test vehicle for this work. Two processing techniques that were developed for this thesis are also presented.

In Chapter 4 the *heart rate estimator* building block is disclosed. This is a common building block to both of the proposed heart segmentation systems, and may be considered as a reusable block onto itself. The chapter ends with a report of the performance of this estimator against the experimental data.

Chapter 5 introduces the *peak energy based* segmentation algorithm, using energy detection and slicing, followed by some simple heuristics. The chapter ends with a discussion of its performance against the experimental data. This algorithm is primarily included here as an indication of the performance of existing methods.

Chapter 6 proposes a *sliding window autocorrelation* segmentation algorithm, based on a sliding window of autocorrelation followed by some simple heuristics. It ends with some

results against the sample data. It is this algorithm that is the significant contribution of this thesis.

Chapter 7 concludes the thesis, and presents some opportunities for future work.

#### **1.4 Thesis Contributions**

This work expands the literature of available techniques for segmenting heartbeats, introducing an alternative which may complement or supplement existing techniques. The *sliding window autocorrelation* algorithm may be used as a building block in a full diagnostic system. Its performance is quantified against the *peak energy* algorithm which is typical of current techniques.

Secondly, a *heart rate estimation* algorithm is described and its performance measured.

Thirdly, this work has produced a cataloged and annotated collection of heart sounds (206 files and 2709 heartbeats) which may be used as a standard against which to measure future work.

Fourthly, the thesis describes a method for comparing cyclostationary signals that may vary in magnitude (fading) over time.

Finally, the thesis includes a method for characterizing the information contained in correlation curves.

## Chapter 2

# Background

The sound of the human heart has been used as a means for measuring the condition of the heart since ancient times. The practice has been formalized in the last 150 years with the introduction of the stethoscope (1816, basic model; 1852, rubber tube version [3]). The sound is detected at the surface of the skin, in a non-invasive manner, and a conclusive diagnosis can often be stated immediately by a skilled practitioner.<sup>1</sup>

The use of heart sounds as a manual diagnostic tool (auscultation) has waned in recent years in high technology environments [4][5], but it continues to be a primary diagnostic in many practical situations.

With the advent of high speed computing, the recorded and digitized heart sound, the phonocardiogram (PCG) can be analyzed using DSP techniques. After the simple measures of *rate* and *steadiness* of the beat, the *detection* and *quantification* of murmurs can give the physician vital clues to the condition of the heart. Several researchers are building systems to detect and automatically quantify murmurs [6]-[17].

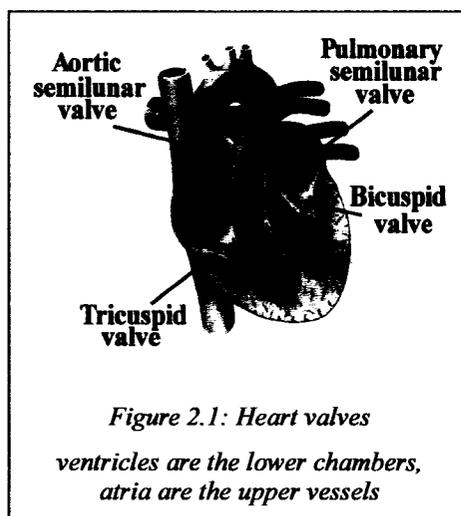
Heart sounds are cyclic (short term cyclostationary), with a period of 300-1500ms [18]. The sounds arise from movements in and around the heart. A cycle of heart sounds almost always has two principal components<sup>2</sup>, named S1 and S2, and possibly two other synchronous components, named S3 and S4, and possibly other sounds: knocks and

---

<sup>1</sup> For further reading on auscultation, see [1][2]

<sup>2</sup> In the presence of extremely large systolic murmur, the S1 and S2 sounds may appear to merge

murmurs. The percussive components are due to the movement of the valves of the heart (S1 and S2), and from sudden pressure changes (S3, S4 and knocks/snaps). The sibilant components arise from turbulent or restricted blood flow (murmurs).



S1 occurs at the beginning of systole. The ventricles are contracting and the mitral (bicuspid) and tricuspid valves (known together as the arterioventricular or AV valves, see Figure 2.1) close, causing a percussive pressure wave. This wave travels through the tissue to the skin surface forming a wavefront that emerges as S1. These valves are larger than the semilunar valves, causing sounds that have a low frequency profile, with the dominant components in the 10-140Hz range. The details of the creation of the sound are still under investigation; it is thought that the sound is created by a combination of impact of the valve surfaces and the local pressure changes in the blood volume, both principal and oscillatory[19][20].

After the ventricles have finished compressing, the built up pressure in the upper chambers closes the aortic and pulmonary valves (semilunar valves), creating a similar pressure wave, S2, the sound of the aortic and pulmonary valves. S2 is typically slightly higher in

frequency content than the S1 sound, due to both smaller valve structures and higher dynamic pressures [20]. The frequency components range from 10-400Hz, and are almost always higher than those in the S1 sound (on any particular patient). Figure 2.2 shows a typical sequence of heart sounds.

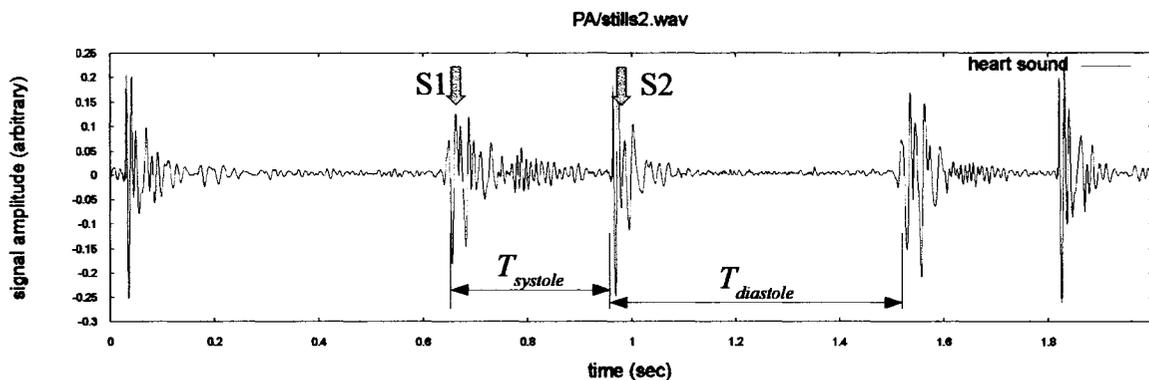


Figure 2.2: Typical heart sound

Both S1 and S2 are created by two valves. It is quite common that these valves will not operate at exactly the same time, resulting in a "split" sound for S1 and/or S2.

The time between S1 and S2 is *systole* and the time between S2 and S1 is *diastole*. The terms  $T_{S1-S2}$  and  $T_{systole}$  are used interchangeably in this work, as are  $T_{S2-S1}$  and  $T_{diastole}$ .

S3 and S4 are unusual sounds, similar in quality to S1 and S2, but smaller in magnitude. If an S3 is present, it occurs between S2 and S1 (in diastole), usually right after the S2. It is thought to be caused by the filling ventricle suddenly reaching its capacity [21]. If an S4 is present, it precedes S1 by a hundred milliseconds or so; it is caused by the atrial contractions, which occur shortly before S1 and which are usually silent. The atrial contraction pours blood into the ventricle, which may suddenly reach its capacity, causing the thump sound. Both S3 and S4 are very low frequency sounds, associated with much

larger blood volumes and surfaces. Both sounds are rare, and more often heard in the hearts of children.

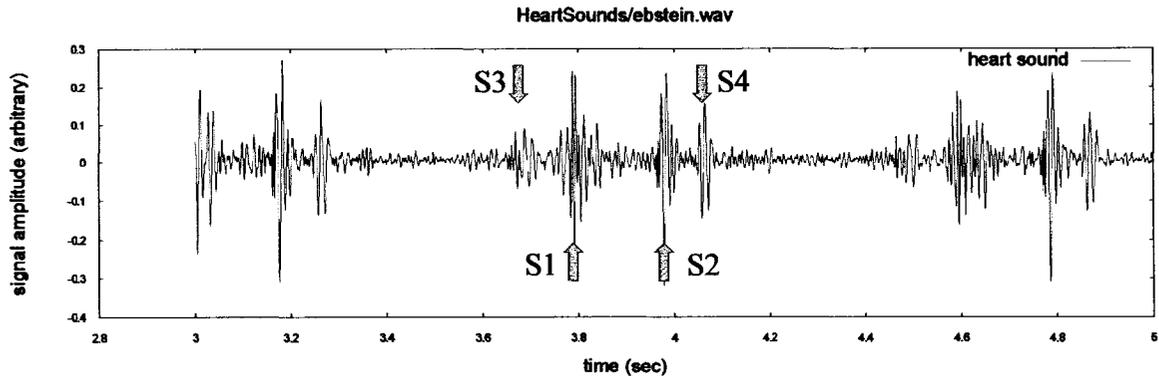


Figure 2.3: Heart sound showing S3 and S4

## 2.1 Heart Murmurs

Murmurs are the various morphological and functional abnormalities that are associated with sounds other than the S1 and S2 (first and second normal heart sounds). Physicians have traditionally listened to these sounds to diagnose complex function problems in the working heart in a non-invasive manner.

Murmurs are primarily classified according to when they occur, with respect to systole and diastole.

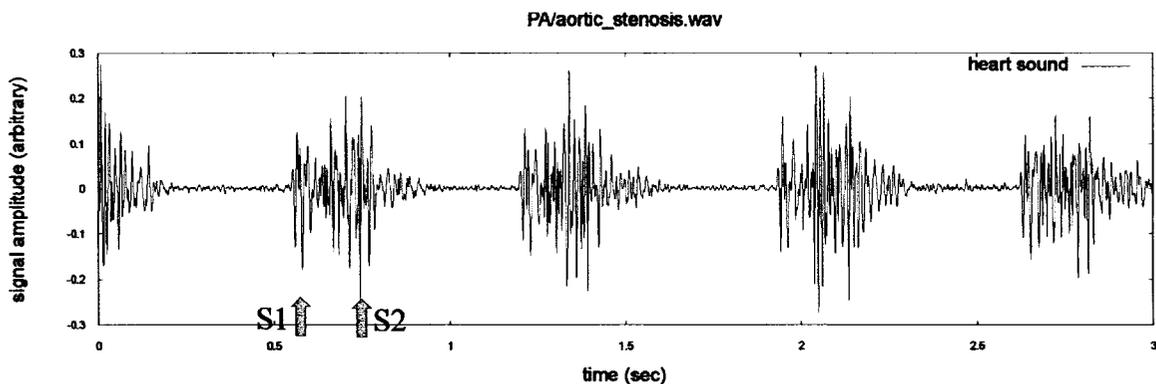


Figure 2.4: Mid systolic murmur, with a rise and fall profile, typical of aortic stenosis

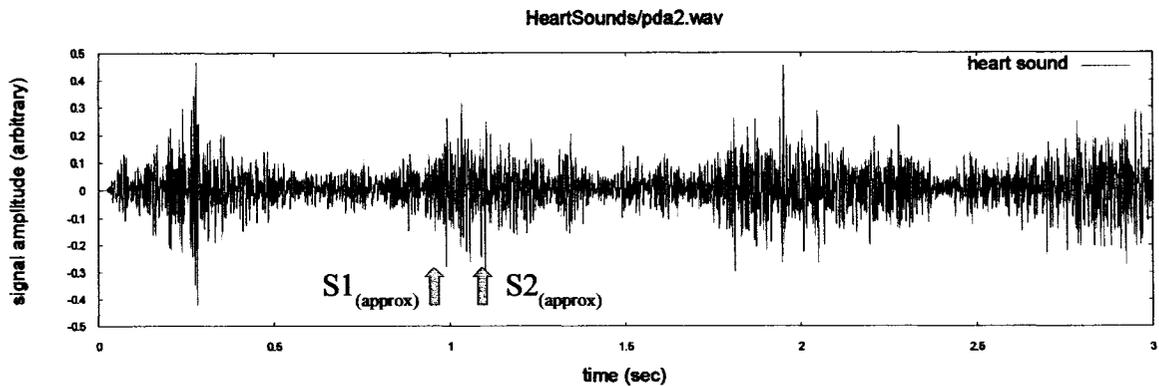


Figure 2.5: Continuous murmur, typical of PDA

The sound level of murmurs is usually much lower than that of S1 and S2. If there is any aspect of a murmur which occurs during S1 and S2 it is effectively masked, and not discussed. Therefore, audible murmurs are classified by whether they are between S1 and S2 (systolic murmur, Figure 2.4) or between S2 and S1 (of the next beat) (diastolic murmur, Figure 2.6) or continuous (Figure 2.5). Modifiers can be used to describe the character of the murmur. For instance, a *holodiastolic* murmur exists throughout the diastolic period. The terms *early* (Figure 2.7) and *late* are also used to describe shorter murmurs.

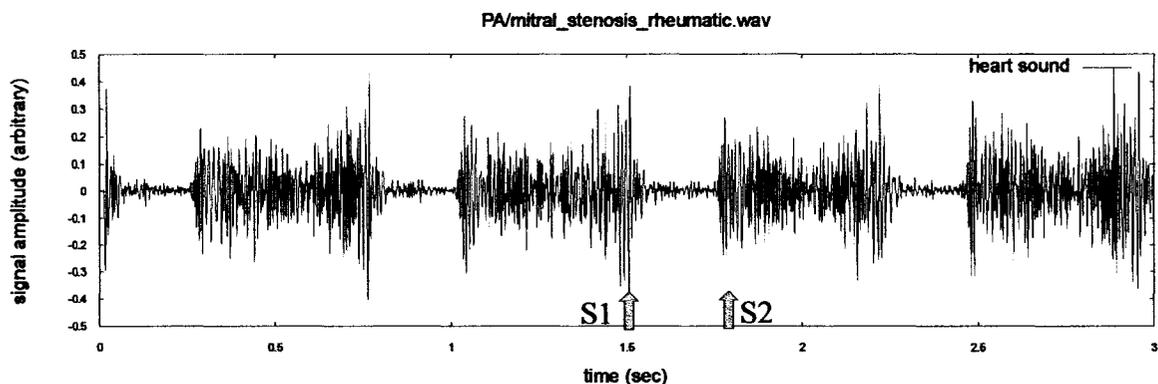
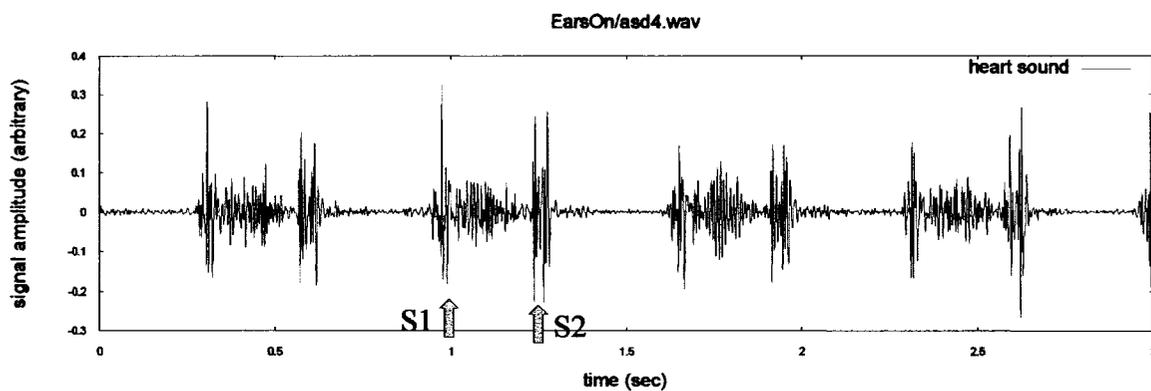


Figure 2.6: Diastolic murmur, constant throughout, typical of mitral stenosis

Note that systole is the quiet time in this particular sample



*Figure 2.7: Early systolic murmur, typical of mitral regurgitation and atrial septal defect*

The third level of classification describes the pitch or frequency of the murmur, and whether it has a dominant tone.

Other features may be applied characterize a murmur, such as:

- the sound rises and falls in amplitude
- occurs during lung inflation/deflation
- occurs when the patient is supine
- occurs during/after exercise

A physician can then use the description constructed above to diagnose several heart conditions such as:

- valves which are stiff may close late, or incompletely,  
or which may not fully open (stenosis)
- valves which leak (regurgitate), or which fold backward (prolapse)

-unusual propagation of the ventricular contraction

For example, mitral regurgitation is characterized by a tonal murmur which is constant amplitude during the whole systolic period (*holosystolic*, Figure 2.7). On the other hand, aortic stenosis is characterized by a tonal murmur in the middle of the systolic period, rising and falling in amplitude (Figure 2.8).

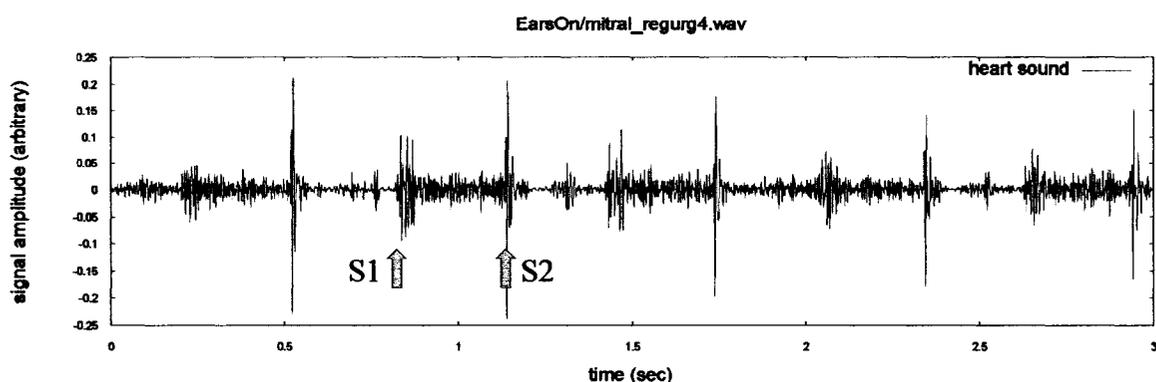


Figure 2.8: Holosystolic murmur of mitral regurgitation

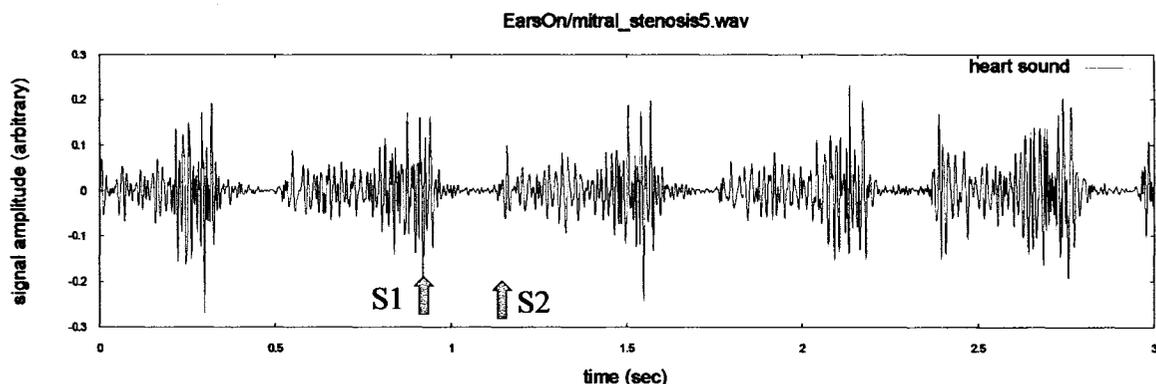


Figure 2.9: Holodiastolic murmur of mitral stenosis

*Note that the systolic time is the quiet period*

Mitral stenosis is characterized by a steady murmur throughout the diastole (Figure 2.9).

In addition to the murmurs, there may be other cardiac sounds, called “clicks” or “snaps”.

These occur synchronous with the beat, and are lumped in with murmurs in this work.

There may be other sounds that are collected by the stethoscope or microphone such as breathing, motion sounds, room noises, speech and gastrointestinal noises. These sounds tend to be uncorrelated to the heart sound, and their effect should be suppressed by techniques which take advantage of the cyclic nature of the heart sounds (chapter 6).

## **2.2 Medical Uses**

While it is premature to expect a computer system to replace human diagnostic work, a computerized heartbeat analyzer might be used for

- teaching
- diagnostic screening
- confirming diagnosis
- remote or telemedicine

One specific application is the detection of *patent ductus arteriosus* (PDA) in premature newborns. This disease is characterized by a background murmur, but the detection is obfuscated because:

- the murmur is intermittent, depending on the fatigue of the patient
- the murmur has a spectrum similar to the sounds of a respirator, or air conditioning
- the murmur sound is fairly static, similar to background noise
- the nursing staff assigned to monitor these patients is often untrained

An automatic system may assist in monitoring these patients.

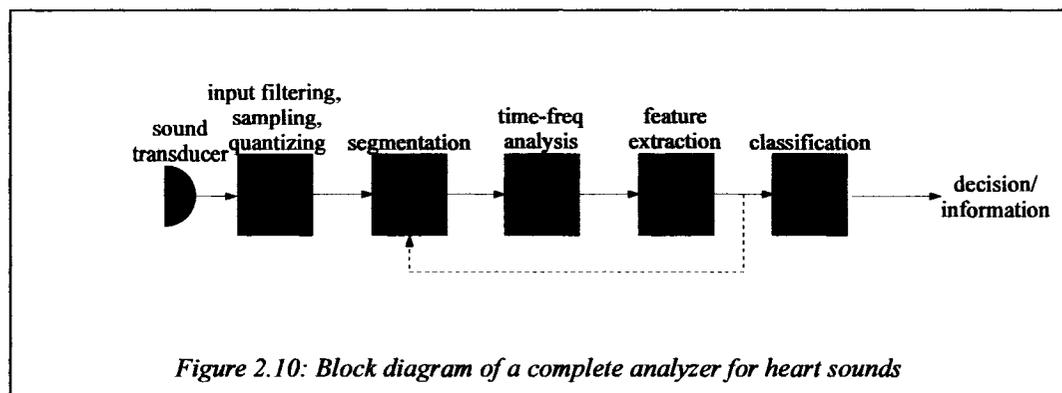
## 2.3 A Murmur Analysis System

This section outlines why a segmentation system is required and how it is used.

The detection of beat period is important for the study of heart endurance, exercise capacity and the detection of pace-maker function. The beat period can be determined easily by examining the envelope of the PCG. (see chapter 4)

However, in order to detect murmurs and deeper heart function problems, it is important to also analyze within the heartbeats themselves.

One possible automated system is shown in Figure 2.10.



The sound capture and front end processing is well understood and not dealt with here. For this work in heart sounds, a sampling rate of at least 1000Hz at 12 bits of resolution is expected, although the experimental setup uses 16 bit samples taken at 4000Hz.

The center part of the block diagram is quite flexible, but often begins with a process that determines the boundaries of the individual heartbeats, named *segmentation* in the literature. This is followed by a time-frequency transform, introducing a 2<sup>nd</sup> dimension to

the data. The beats are then folded over each other according to the segmentation results and a statistical method is used to merge the 2D data, feeding it to the feature extractor. The extracted features are fed to the classifier, such as a neural network.

*Segmentation* is a principal stage, because the accuracy of the alignment determines how well the beats can be aligned and averaged.

Many variations are possible to this block diagram. For instance, the segmentation process may be performed *after* the time-frequency transform, possibly yielding better results.

The output of the feature extractor may be fed back into the segmentation block, to iteratively refine the segmentation.

The problem of the extraction and classification of significant features from a series of heart sounds is tied to a definitive alignment of the heart sound.

Once the heartbeats are aligned, they may be aggregated in order to suppress uncorrelated noise and improve the signal to noise ratio. Alternatively, one could examine the variability of the murmur with respect to the S1/S2 features.

Previous work has been done in this field (see 2.4.2), including several projects that use a block diagram similar to Figure 2.10 [10][17]. Segmentation is sometimes performed by hand, or using algorithms which perform well on good quality PCGs. This thesis investigates a new approach to the problem of synchronous segmentation (chapter 6) of a range of PCG types, always with the intention of contributing to a total murmur detection system.

## 2.4 Segmentation Methods

### 2.4.1 Existing Methods

The detection of the heartbeat is a surprisingly difficult task using PCG data. In fact, most research work (outside of segmentation research) captures a simultaneous electrocardiogram (ECG), and uses it to synchronize and detect the heartbeat. The ECG is simple to work with because the QRS spike (the large peak in an ECG corresponding to systole) is a single dominant pulse, usually 5 times larger than the other aspects of the ECG, including the other pulses and the background noise.

Similarly, the clinical procedure for auscultation recommends that the practitioner probe for a pulse at the wrist, in order to feel the heartbeats; S1 occurs at the start of the pulse wave, and S2 at the end<sup>1</sup>.

The timing information from a concurrent ECG would obviate the need for segmentation at all. However, gathering ECG information in a clinical situation is often inconvenient. A proper ECG recording requires access to at least two open patches of skin on the patient, and often involves abrading the patient's skin to assure good electrical contact. A system that only involves the placement of the stethoscope bell to gather the PCG is better suited to clinical work, telemedicine or offline analysis.

The PCG-only beat detection (without a record of the pulse or ECG) is difficult because a simple time-domain envelope is ambiguous: sometimes the S1 sound is louder than the S2, and sometimes the opposite; sometimes the S1 and S2 sounds are not distinct, being joined by a murmur sound [Figures 2.11, 2.12 & 2.13]. Similarly, the time from S1 to S2 ( $T_{\text{systole}}$ )

---

<sup>1</sup> Practical diagnosis is often performed without pulse information, because the S1 sound is often obvious. In [22], the authors simply use a .wav file for their testing.

is usually half the time of the S2 to S1 time ( $T_{diastole}$ ), but this is not always the case; a rapid heartbeat can reverse this ratio.

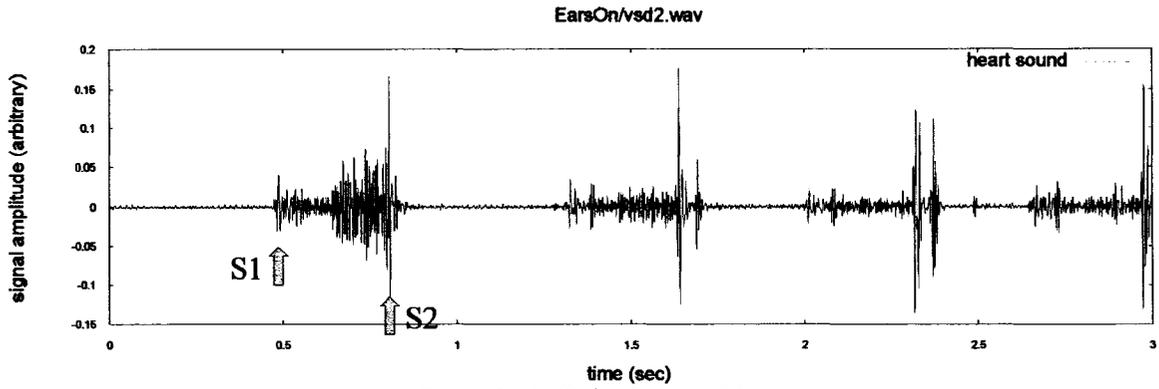


Figure 2.11: S2 louder than S1

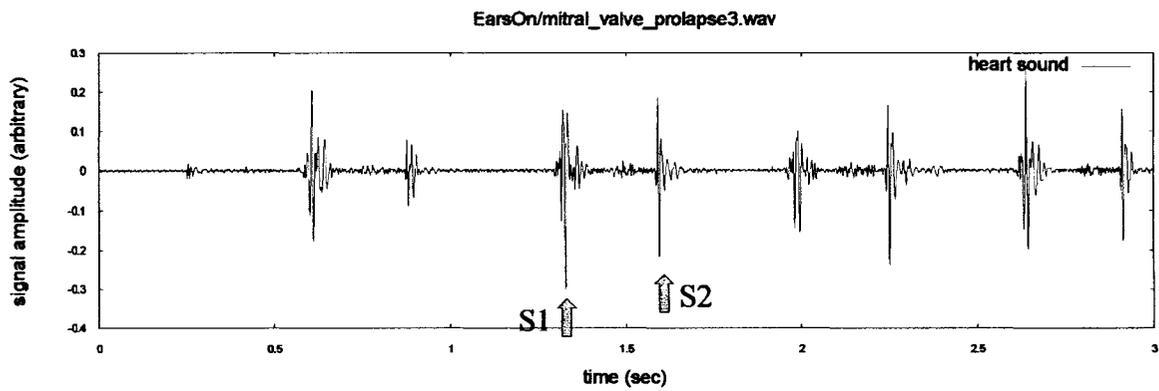


Figure 2.12: S1 louder than S2

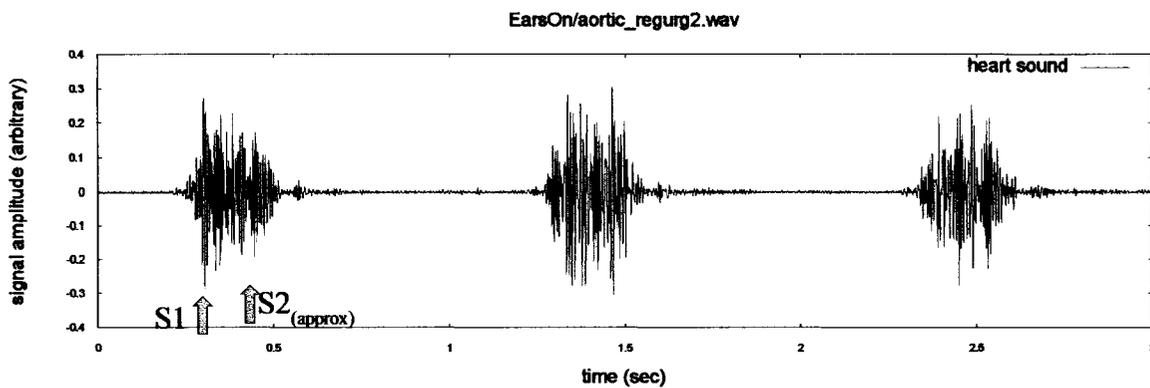


Figure 2.13: S1 and S2 indistinct

In summary, existing methods either take advantage of concurrent ECG information, or suffer from sensitivities to noise and signal shape.

#### **2.4.2 Segmentation in the Literature**

The tasks of murmur detection have been investigated for many years. Much of the previous research only mentions the problem of segmentation incidentally. Often the researchers will use an ECG recording to align the sounds or perform the segmentation task manually. Recently, some work has been done on using time-frequency methods to separate the components of S1 and S2, in order to extract either features of the heart sound directly, or of the adjacent murmur [23][24].

Liang [6] has built a system based on the normalized Shannon energy, and its envelope. The peaks of the signal (magnitude) are compared to a rolling average. The peaks are then processed with heuristics such as “when two peaks appear within 50ms...and the energy of the first peak is not too small compared to that of the second one, pick the first one” (peak rejection). Once the peaks have been located, a second pass determines which are S1 and S2, using the fact that  $T_{\text{systole}} < T_{\text{diastole}}$  (usually) and that  $\Delta t_{\text{systole}} < \Delta T_{\text{diastole}}$  (always). He reports accuracies of 93% on their corpus of sounds (515 heartbeats), with 6% false negatives and 1% false positives. The main process described in this work is similar to the peak segmenter described in chapter 5, although the S1/S2 classifier does take advantage of the consistency of  $T_{\text{systole}}$ .

Liang [25] also built a system that follows on from the first one, taking a spectrogram and using the energy in the 50-150Hz range to refine the location of the beats. This method often refines the leading edge of both S1 and S2, moving the detection point earlier than a

simple energy-based detector could. This work simply explains the method without any experimental validation.

Liang [26] continues with a third system based on wavelet decomposition, taking advantage of the fact that S1 consists of frequency components in the 10-50Hz range, and S2 contains components in a wider range (10-400Hz). The wavelet is a 6<sup>th</sup> order Daubechies, with sampling at 2205Hz. The actual S1 and S2 detection is mostly a peak detector on the 4<sup>th</sup> and 5<sup>th</sup> decomposition components (4<sup>th</sup> is ~70-140Hz, and 5<sup>th</sup> is ~35-70Hz). He reports detection success of over 93%, based on 1165 heard beats.

Omran [7] built a system which decomposes the PCG into wavelets using a Daubechies db6. He uses the 3<sup>rd</sup>, 4<sup>th</sup> and 5<sup>th</sup> levels as measures of power and uses several heuristics to determine features of the waveform. The features are passed to a neural net (13 neurons) for analysis. After training with 60 samples of heart sounds, the system was able to classify 20-30 unknown with an accuracy of 98%. One might expect that this method might have trouble in the presence of large background interference, because of the dependency on the second-stage heuristics.

Saha [8] uses a system based on a 10<sup>th</sup> order 150Hz low pass filter, which suppresses some high frequency murmurs. This is followed by slicing a running average of the signal energy, with a threshold based on a longer-time running average. After this slicing, he applies several heuristics to eliminate split sounds and center the beat detection. To distinguish S1 from S2, he assumes that  $T_{\text{systole}} < T_{\text{diastole}}$  and  $\Delta T_{\text{systole}} < \Delta T_{\text{diastole}}$ . Beat edges are then adjusted using a couple of threshold heuristics. He reports 96% correct detection, over 645 heartbeats. The main process described here is similar to the peak segmenter

described in chapter 5.

Carvalho [27] presents a system which detects the S1 sound primarily from the QRS of the associated ECG, and then uses a mixture of energy measurement and heuristics to classify the energy peaks in order to locate the S2 sounds. One of the most important rules for detecting the S2 peak is the fact that  $T_{\text{systole}}$  remains fairly constant. The system performed well (>96%) on a small sample of 105 heartbeats from group of patients with various heart diseases and prosthetics, and handles irregular heartbeats well.

Kurnaz [9] uses multi-level wavelet decomposition to segment the heart sounds. As a first stage, the heart sound is segmented by isolating the 6<sup>th</sup> level wavelet component, and locating potential S1 and S2 sounds. He classifies the S1 and S2 sounds using the same rules as Liang [6]. Post-processing to extract murmur features uses the 2<sup>nd</sup> level wavelet to classify the components between S1's. This paper is written as a disclosure statement, and without explicit results.

Haghighi-Mood [10] presents a system which creates an AR model of the signal as the first step, to determine the frequency characteristics of the particular patient. He then uses a sliding window and repeats the AR modeling, extracting a measure of “beat energy” in a narrow frequency band, which is subsequently sliced to provide S1/S2 detection. Murmurs are rejected by only detecting peaks which have their energy compressed into a short time span. Although the method was tested, the results are deferred and not stated.

Ricke [11] built a system to detect respiration in children, indirectly, through changes in the heart rate. (For example, the S2 is often split during inhalation). They incorporated segmentation in their design, describing a system which extracts features from the

incoming sound (frequency bins, running-average Shannon energy) and passes these on to a hidden Markov model, which determines the location of S1 and S2. The process described is not a stand-alone segmentation process, but is part of a larger system to detect respiration and apnea. There is no attempt to build a segmenter which yields synchronous information, simply to sense the S2 split of inhalation. Although the system is reported as 98% accurate, there is no statement of the size experimental dataset.

Gupta [12] built a simple effective system by simply filtering the energy of the PCG signal to between 10 and 20Hz, detecting the peaks and post-processing with some simple heuristics: peaks must be separated by at least 80ms, peaks that are too narrow are rejected. Many murmurs are suppressed this way. The resulting peaks were clustered and classified, assuming the  $T_{systole} < T_{diastole}$ . He reported ~86% correct segmentation in the presence of murmurs, rising to 99% for normal heart sounds, based on an experiment of 340 heartbeats.

Myint [17] et al. built an entire stethoscope, which calculates the rate, the segmentation and detects some murmurs. The segmentation is a simple normalize, peak detect and classification, assuming  $T_{systole} < T_{diastole}$ . This work is simply a disclosure of early work, with no results reported.

Jimenez [28] tests the improvement in segmentation “quality” achieved by preprocessing the signal with a wavelet decomposition *before* running a segmenter based on peak detecting the Hilbert magnitude. To measure the improvement, he compares the variation in S1/S2 location against the ECG; all the reported results are based on simulated fetal PCG signals (with maternal PCG sounds as interference). The segmenter is simply a peak

detector, similar to the system described in chapter 5.

Rajan [13] also uses a wavelet decomposition, taking advantage of the fact that the Morlet kernel resembles (and correlates well) with both the S1 and S2 sounds. This leads to clear peaks in the decomposition for the S1 and S2, which are then detected against a rolling average. He reports 90-91% correct detection of S1 and S2 in a database of 534 heartbeats.

In summary, many of the segmenters described are simply threshold detectors and sorting logic, taking advantage of the fact that most PCG signals have a signal to noise ratio which allows envelope detection, and of the fact that usually  $T_{systole} < T_{diastole}$ . The searched literature did not cover a segmentation system that handles:

- segmentation of signals where  $T_{systole} > T_{diastole}$
- segmentation in the presence of large interfering signals
- synchronous segmentation

The *sliding window autocorrelation* segmentation algorithm (chapter 6) addresses these aspects.

### **2.4.3 Peak Detection**

The existing literature provides a range of solutions, but comparing the alternatives is difficult because of the lack of a common base of PCG samples. The *peak detection* segmenter described in chapter 5 is a composite of some of the concepts described in 2.4.2, included here in order to provide a benchmark against which to compare the *sliding window autocorrelation* algorithm (chapter 6). It is a simple but effective segmentation technique that uses the envelope of the PCG. This method is described fully in chapter 5.

The results reported in chapter 5 for the *peak detection* segmenter are much lower than that reported in the literature, most likely because the database of PCGs used in this thesis include some extremely difficult samples. Further, the scoring used in this thesis requires that the heartbeats be located in a synchronous fashion, in such a way that multiple heartbeats can be overlaid for analysis.

#### **2.4.4 Sliding Window Autocorrelation**

Simple *peak detection* can be expected to suffer from several problems. There are the limitations listed at the end of 2.4.2 above. Also, in many PCG samples, the S1 and S2 amplitudes and periods may not be enough to determine which sound is which; indeed, there are PCGs where the S1 or S2 are not even obvious (see the examples in 2.4.1).

By taking advantage of the self-similarity of the beats, one can build a measure of the period of the beats. Then, using a time-limited sliding window, one can identify the time region where one beat starts to differ from the next, and thereby estimate the beat boundary. This technique is described in chapter 6.

## Chapter 3

# Experimental Setup

### 3.1 Heart Sounds

The bulk of this thesis deals with signal analysis and not with the acquisition of clinical signals. To simplify this work, PCG samples were taken from several training CDROMs [29], [30], [31] that are designed for training physicians and nurses. The waveforms were digitized at various rates, from 8000Hz to 22050Hz, and with 12 bit or 16 bit linear quantizing; all were filtered and subsampled to 4000Hz and requantized at 16 linear bits/sample (mono). Six additional sound files were acquired from the Children's Hospital of Eastern Ontario (located in Ottawa, Canada), using a Littman electronic stethoscope. This device samples at 4000Hz, and uses 12 bit linear coding.

Together the dataset comprises 206 separate sound files, containing a total of 2709 heartbeats. The heartbeat periods range from 0.25 s to 1.88 s. The number of heartbeats in a sound sample ranges from 3 to 76, with the mean count of 13 and a median of 11 beats.

The samples cover a range of physiological conditions listed in Table 3.1.

<i>Murmur type</i>	(1)	(2)	(3)	(4)	<i>Murmur type</i>	(1)	(2)	(3)	(4)
aortic regurgitation	3	13			mitral regurgitation ruptured		1		
aortic valve stenosis	7		9		Noonans syndrome	1			
arteriovenous fistulae	1		1		patent ductus arteriosus	5	2	4	2
arteriovenous block	5				pericardial knock		1		
atrial septal defect	5	1	3		premature infant				1
bicuspid block		1			pulmonary regurgitation	4	1		
Blalock-Taussig shunt		1			pulmonary valve stenosis	4	2	4	1
coarctation of the aorta	1				pulmonary artery branch stenosis		1	2	
congenital aortic stenosis		4			pulmonary hypertension		2		
Ebstein murmur		2	1		rheumatic mitral regurgitation		1		
Eisenmenger murmur			1		stab wound		1		
Gallavardin murmur		4			Still's murmur			4	
gallop				1	tetralogy of fallot	7		4	
hypertrophic cardiomyopathy			5		transposition	2			
innocent murmurs	7		3		tricuspid atresia	1		1	
mitral regurgitation	5	4	1	1	tricuspid regurgitation		2	1	
mitral stenosis	6	6	2		tricuspid stenosis		2		
mitral valve prolapse	7	7	2		venous septal defect	5	4	8	
normal		1	2		splits			2	

Table 3.1: Database of heart sounds, broken down by murmur type

The numbers following the classification indicate the number of samples available for each source:

- (1) EarsOn [29], 76 files
- (2) HeartSounds [31], 64 files
- (3) PA [30], 60 files
- (4) CHEO-Nov04, 6 files

The signals in the collection are not amplitude normalized; when required, a normalization first step is mentioned explicitly.

Some parametric information was extracted from each sound file (PCG):

- the longest and shortest inter beat time

- the number of complete heartbeats
- whether the S1 and S2 sounds are distinct, and whether one dominates (in energy)
- whether interfering artifacts exist: breathing noises, talking, motion noises
- complexity

The observed heart rate was measured manually by simply noting the time from peak to peak of the waveform, using a graphical view of the sound file; it was necessary to record the minimum and maximum periods of the heartbeats, as some sound files exhibit a 2:1 range in the period of the heartbeat.

Most heartbeats contain distinct pulses of energy at S1 and S2, and often one sound will be louder than the other. A naive segmentation algorithm may be fooled into swapping the S1 and S2 sounds by incorrectly assuming that one is louder than the other, or that the diastolic time is always longer than the systolic time. This database contains

35% of the files have S2 louder than S1

21% of the files have S1 louder than S2

22% of the files have S1 and S2 comparable in size; neither dominates

16% of the files have heart sounds with loud systolic murmurs, such that the S1 and S2 sounds are bridged by continuous energy

The “complexity” of the heart sound is scored 1, 2 or 3, depending on a combined visual and aural judgment of the graph of the heart sound. Simple heart sounds, with quiet diastolic activity and a clear demarcation of the beat, are scored as 1 (Figure 3.1). Heart

sounds which are masked by significant murmur noises, so that the edges of the heartbeats is indistinct, are scored as 2 (Figure 3.2). Heart sounds which are buried beneath breathing sounds, are scored 3 (Figure 3.4). Extremely irregular beat sequences are also scored 3 (Figure 3.3).

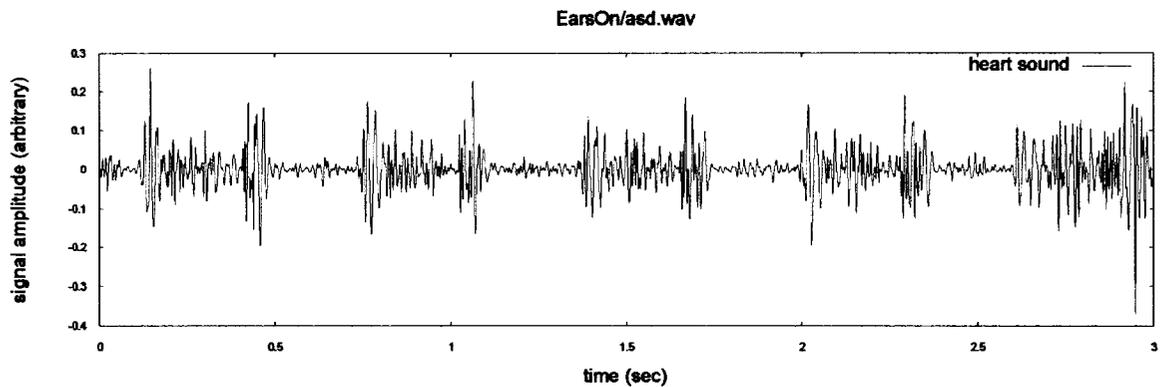


Figure 3.1: Typical waveform of complexity "1"; clear S1 and S2 sounds

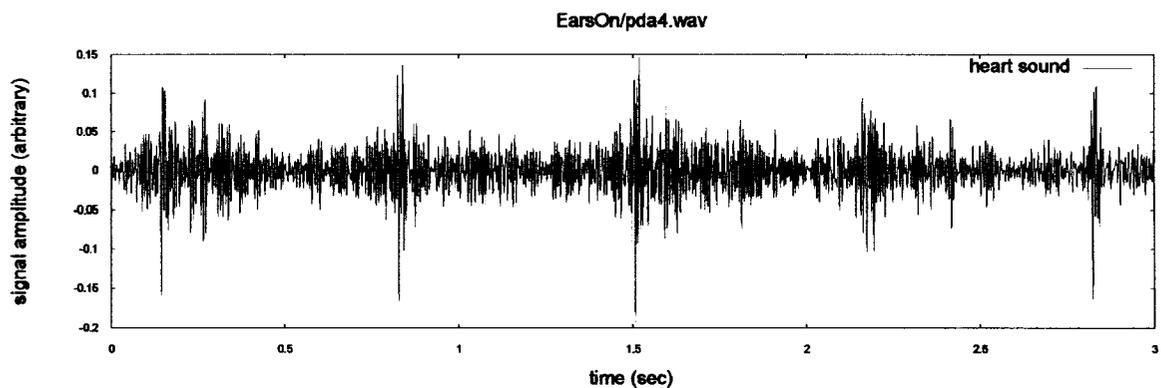


Figure 3.2: Typical sound of complexity "2"; S1 and S2 not obvious, but no large interference

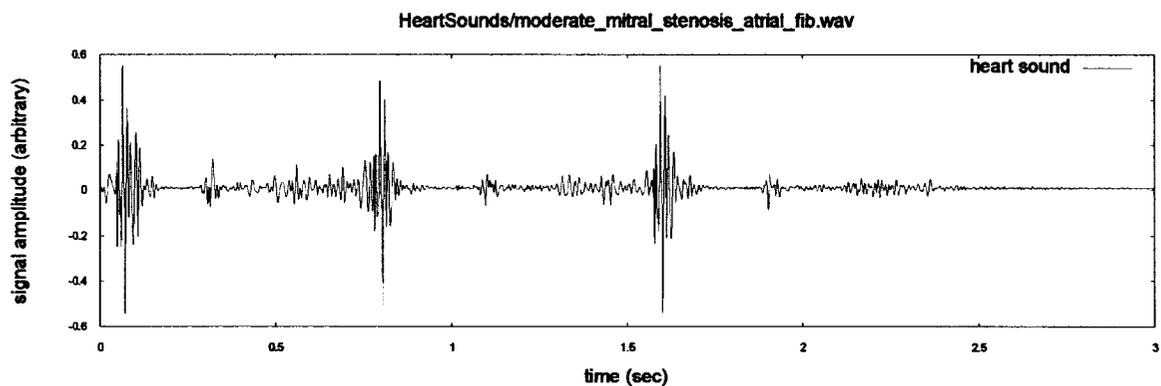


Figure 3.3: Example of complexity "3", note the irregular beats

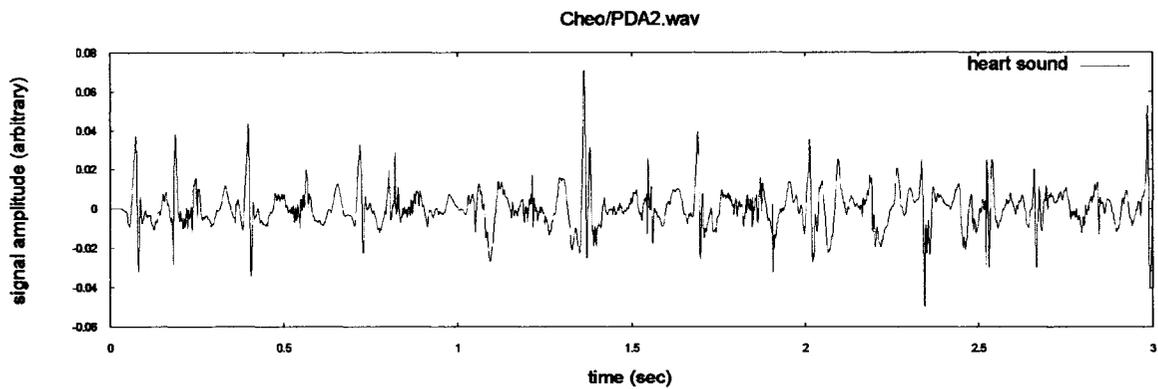


Figure 3.4: Example of complexity "3"; heartbeats is buried under breathing noises

<b>Complexity</b>	<b>Number of files</b>	<b>Number of heartbeats</b>
1-simple	166	1948
2-moderate	30	589
3-complex	10	172

Table 3.2: Database of heart sounds, broken down complexity

This database is highly skewed toward samples from patients with diseases of the heart. As a result, the success rate of any processing system can be expected to be lower than that of an experiment run with mostly healthy patients.

This should not, however, preclude using this database as a design qualification tool. The heart sounds of diseased patients still contain enough of the basic S1-S2 pattern to test and stress an analysis system or sub-system.

The database built here is significantly larger than most databases used by other researchers. Table 3.3 enumerates the sizes of test data used in some recent papers.

Reference	Number of heart sound files	Number of heartbeats
Liang [6]	37	515
Liang [26]	77	1165
Saha [8]	145	645
Kurnaz [9]	28	336
Haghighi-Mood [10]	30	960
Gupta [12]	-	340
Rajan [13]	42	534
Carvalho [27]	-	106
This work	206	2709

*Table 3.3: Comparison of the size of experimental databases*

It is appropriate to use this database to compare the performance of two systems. The database include sounds from all the major heart diseases, and is large enough to stress a design for comparative purposes.

The lack of non-diseased samples would suggest that this database should not be used to estimate the absolute performance of any system against the general population.

### **3.2 Experimental Setup**

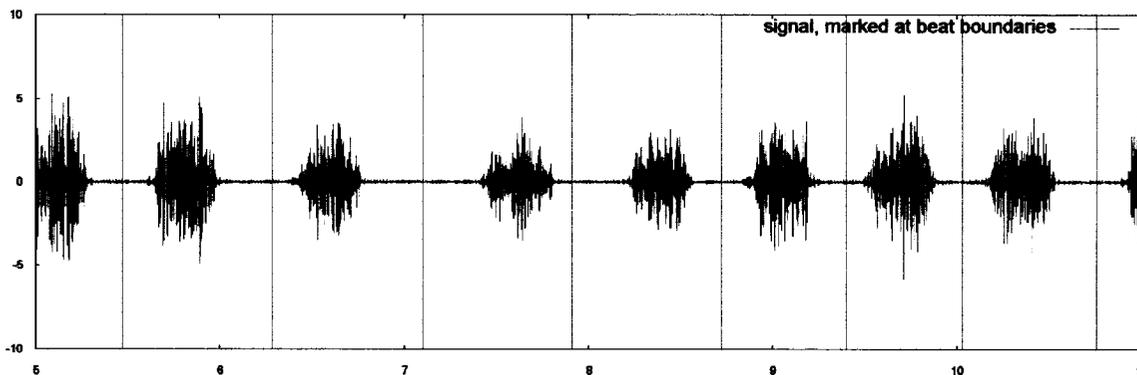
The experimental processing was performed on an Intel Pentium IV running at 1800MHz, with 1.5GB of RAM, using the Linux 2.6.15 operating system. The experimental environment was GNU/Octave 2.1.73 unless otherwise specified.

Experiments consist of Matlab/Octave programs written to process the sound waveforms. If non trivial output is generated, it is written to ASCII files and imported into OoCalc (an Excel-like spreadsheet program).

Because of the off-line processing, non-causal filtering is used in several places; in each case, the benefits of using it is described.

The output of the *rate estimator* (chapter 4) is tested against the max and min beat periods recorded in the ensemble database.

The output of the segmenters (chapters 5 & 6) is a set of “marker” files, which are displayed in the wave-file viewer WaveSurfer. The author then scored the results by visually checking the wave files for reasonable placement of the markers. A marker is scored as “failed” if it is placed within the systolic time, or placed at a significantly different place in diastole than the other markers in the same file. For example, in Figure 3.5, the markers at 7.1 s and 7.9 s are scored as failing, because they do not align with the other markers in a way that would allow the beats to be superimposed (for feature extraction).



*Figure 3.5: Typical segmented sound file  
the markers at 7.1 and 7.9 s are scored as "failed"; see the text*

### 3.3 Processing Techniques

#### 3.3.1 Similarity Correlation

The signals recorded in a PCG often vary in magnitude, as is typical of many biological processes and transducers. Traditionally, signal processing might include an *automatic gain control* (AGC) function, with design decisions involving attack and release timings, gain ranges and decay envelopes.

However, chapters 4 and 6 focus on self similarity and autocorrelation. A possible alternative to AGC is to modify the autocorrelation function to compensate for the signal magnitude.

An analogy to this idea can be presented by reviewing a simple concept in 2D vector geometry. To test if vectors **A** and **B** are parallel, one can take the dot product of the vectors, and divide by both vector magnitudes.

$$m = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\| \|\vec{B}\|} = \cos(\text{angle between } \vec{A} \text{ and } \vec{B}) \quad 3.1$$

$m$  ranges from -1 to +1, indicating the “similarity” of **A** and **B**, ignoring the magnitudes of each.

One can extend this concept to a sample of sound, simply changing the dimensions from 2 up to, in our case, the template length (typically 2000 to 4000 samples).

The process described in chapter 6 of this thesis is a sweeping autocorrelation, which would normally be<sup>1</sup>

---

<sup>1</sup> We are only concerned with the right hand side of the autocorrelation, so the expressions here involve  $t+m$  instead of the usual  $t-m$ .

$$X_t = \sum_{m=0}^{Length-1} w_m w_{t+m} \quad 3.2$$

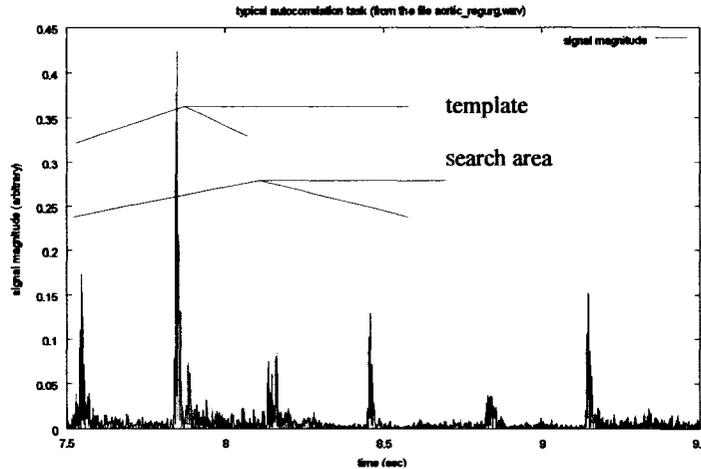


Figure 3.6: Example of the autocorrelation process

Figure 3.6 shows how a heartbeats sequence may vary substantially in magnitude. It also shows the requirement to correlate a shorter segment with a longer one. Before the summation, the shorter segment is zero-extended to match the longer.

$$\begin{aligned} w_m^{template} &= w_m \text{ for } m < TemplateLength \\ &= 0 \text{ for } TemplateLength \leq m < SearchLength \end{aligned}$$

3.3

$$X_t = \sum_{m=0}^{SearchLength-1} w_m^{template} w_{t+m}, \text{ for } t \geq 0$$

To normalize the autocorrelation, normalize the dot product by dividing the autocorrelation by the magnitude of each segment. The magnitude of the shorter segment is well defined, and is simply

$$M_{template} = \sqrt{\sum_{m=0}^{SearchLength-1} (w_m^{template})^2} = \sqrt{\sum_{m=0}^{TemplateLength-1} w_m^2}$$

However, the magnitude of the longer segment is more difficult to evaluate. Since the shorter segment is effectively “sweeping across” the longer segment, the magnitude should simply be the energy of the matched samples, the others being ineffective, and zero.

The “magnitude” of the longer segment is therefore a function of time, and can be calculated by masking the  $w^2$  vector with a rectangle mask that is the same length as the shorter segment.

In Matlab/Octave code, this can be realized with a few lines:

```
function c = xc(a,b)          # just a wrapper to return the +ve terms
    c = xcorr(a,b);
    c = c(ceil(length(c)/2):end);
end

function r = xcorr_similarity(a,b)
    c = xc(a,b);              # standard x-correlation
    ap = xc((a.*a),ones(size(b))); # mask the energy sums
    r = c./abs(sqrt(ap))/abs(sqrt(b*b')); # b*b' is a dot product
end % function
```

where  $a$  is the longer segment, and  $b$  is the shorter segment. In the first line, simply calculate the standard correlation of two waveforms (keeping only the values for  $t \geq 0$ ). The second line correlates the energy of the longer segment( $a$ ) with a shorter rectangle window, yielding the “moving” summation of energy within  $a$ ; this summation is synchronous with  $b$ , and we can now divide the dot product  $c$  by the two magnitudes, yielding a measure of similarity.

Figure 3.7 shows a short section of the waveform “aortic\_regurg.wav”, cut to show a “template” section and a “search” section. The windowing rectangle is also shown.

A standard cross-correlation of the first two waveforms is shown, and is characterized by a large peak at 0 (complete signal match at  $t=0$ , the signal with itself), a large peak at 0.3 s,

the time from  $S_1$  to  $S_2$  ( $t_{\text{systole}}$ ). The next peak is at 0.6 s, smaller because the 2<sup>nd</sup> waveform is smaller in magnitude.

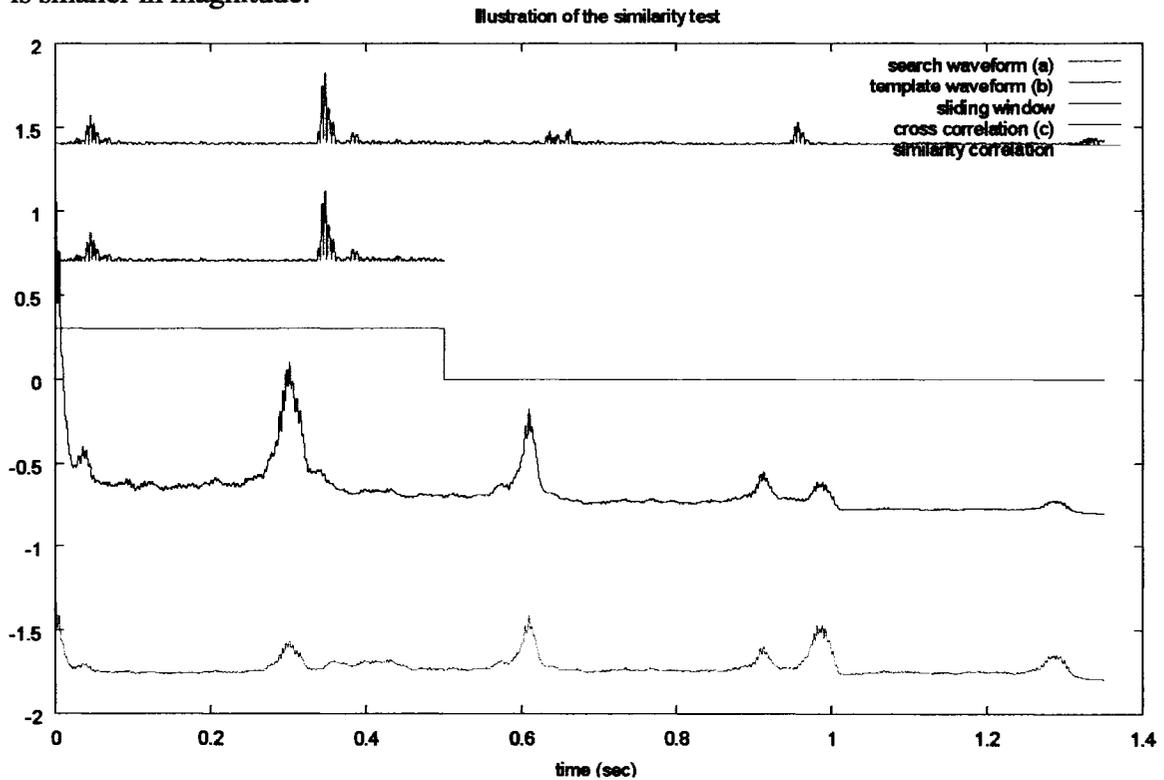


Figure 3.7: Breakdown of the similarity calculation

The 5<sup>th</sup> line in figure 3.7 is the similarity cross-correlation is the same signal, divided by the two magnitudes (described above). Note the peak at 0.6 s is larger than the one at 0.3 s, indicating that the data from 0.6 s to 1.1 s is a better match to the data at 0 s to 0.5 s (the template waveform) than is the data from 0.3 to 0.8 s. Note also the large peak at 1.0 s, because of the small energy in the  $b$  waveform from 1.0 s to the end; this term goes into the denominator of the similarity fraction, boosting it upward.

To summarize, the similarity correlation can be evaluated as

$$X_t = \frac{\vec{A}_m * \vec{B}_m}{\|\vec{A}_m\| \|\vec{B}_m\|_{m=t+templateLength}}, \text{ easily coded in either Matlab or Octave; it invokes}$$

two cross-correlations, which are efficiently implemented via FFT methods.

For signals of varying magnitude, this “similarity” cross-correlation clearly yields a better measure of similarity than does a standard autocorrelation.

### 3.3.2 Matching Correlation Curves to Gaussian Template

Many of the correlation curves extracted from the heart data have significant peaks that take the shape of a bell curve. One of the problems posed indirectly in this thesis is to determine the best measure to use in to create these correlation curves: the signal magnitude, the Hilbert magnitude or the signal energy.

A possible figure of merit might be to compare the various autocorrelation curves by examining the amplitude and width of the peak that corresponds to the main interbeat time (the beat to beat similarity). By modeling the autocorrelation curve with a synthetic curve of only a few dimensions, one can extract the important parameters.

By observation, a possible model function might be

$$g(t) = a + bt + m e^{-\frac{(t-\mu)^2}{s^2}} \quad 3.4$$

The first and second terms establish a match to the baseline, and the third term is a Gaussian function to match the bell shape. The  $m$  parameter establishes the height of the bell, rising up out of the noise floor. The  $\mu$  factor is in the time domain, establishing the center of the bell shape. Referring to Figure 3.8,  $\mu$  is the beat-to-beat period, the time offset at which the waveform becomes self similar. The  $s$  factor is the width of the bell, the range

over which the processed waveform approaches self similarity; in some sense, this reflects an uncertainty, but more so, it reflects the spreading affect of any preprocessing applied to  $f(x)$ .

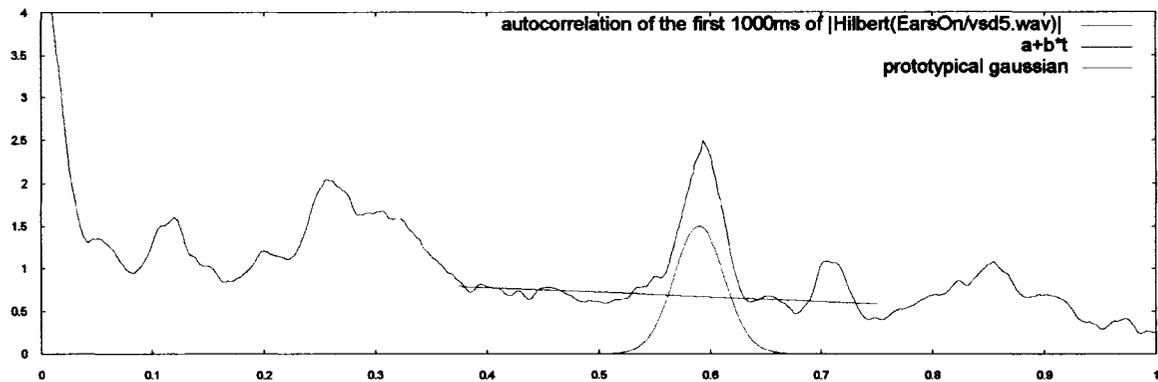


Figure 3.8: A typical autocorrelation and the components of a curve match

The next few pages deal with finding the best  $g(x)$  to model an arbitrary dataset  $f(x)$ .

Traditionally the “best” match is the one that minimizes the sum of the errors-squared.

That is, the one with the smallest

$$E = \sum_{t \in \text{sample range}} [f(t) - g(t)]^2 \quad 3.5$$

Considering the number of variables in this function,  $g(t)$  might be expressed as

$$g(t, a, b, m, \mu, s) = a + bt + m e^{\frac{-(t-\mu)^2}{s^2}} \quad 3.6$$

So the problem can be restated as finding the smallest  $E$  such that

$$\begin{aligned}
E(a, b, m, \mu, s) &= \sum_{t \in \text{dataset}} (f(t) - g(t, a, b, m, \mu, s))^2 \\
&= \sum_{t=0}^{N-1} (f(t) - a - bt - me^{-\frac{(t-\mu)^2}{s^2}})^2 \\
&= \sum_{t=0}^{N-1} f^2(t) + a^2 + b^2 t^2 + m^2 e^{-2\frac{(t-\mu)^2}{s^2}} \\
&\quad - 2af(t) - 2btf(t) + 2abt + (-2mf(t) + 2am + 2btm) e^{-\frac{(t-\mu)^2}{s^2}}
\end{aligned} \tag{3.7}$$

A classic least squares approach to this could be to set the 5 first derivatives to 0

$$\frac{dE}{da} = \frac{dE}{db} = \frac{dE}{dm} = \frac{dE}{d\mu} = \frac{dE}{ds} = 0 \tag{3.8}$$

which resolves to these 5 equations:

$$\begin{aligned}
0 &= \frac{dE}{da} = \frac{d}{da} \left[ \sum_{t=0}^{N-1} (f(t) - g(t))^2 \right] \\
&= \sum_{t=0}^{N-1} (-2f(t) + 2a + 2bt + 2ame^{-\frac{(t-\mu)^2}{s^2}}) \\
&= -2 \sum_{t=0}^{N-1} f(t) + 2a \sum_{t=0}^{N-1} 1 + 2b \sum_{t=0}^{N-1} t + 2m \sum_{t=0}^{N-1} e^{-\frac{(t-\mu)^2}{s^2}}
\end{aligned} \tag{3.9}$$

$$\begin{aligned}
0 &= \frac{dE}{db} = \sum_{t=0}^{N-1} (-2tf(t) + 2at + 2bt^2 + 2tme^{-\frac{(t-\mu)^2}{s^2}}) \\
&= -2 \sum_{t=0}^{N-1} tf(t) + 2a \sum_{t=0}^{N-1} t + 2b \sum_{t=0}^{N-1} t^2 + 2m \sum_{t=0}^{N-1} te^{-\frac{(t-\mu)^2}{s^2}}
\end{aligned} \tag{3.10}$$

$$\begin{aligned}
0 &= \frac{dE}{dm} = \sum_{t=0}^{N-1} (-2f(t)e^{-\frac{(t-\mu)^2}{s^2}} + 2ae^{-\frac{(t-\mu)^2}{s^2}} + 2bte^{-\frac{(t-\mu)^2}{s^2}} + 2me^{-2\frac{(t-\mu)^2}{s^2}}) \\
&= -2 \sum_{t=0}^{N-1} f(t)e^{-\frac{(t-\mu)^2}{s^2}} + 2a \sum_{t=0}^{N-1} e^{-\frac{(t-\mu)^2}{s^2}} + 2b \sum_{t=0}^{N-1} te^{-\frac{(t-\mu)^2}{s^2}} + 2m \sum_{t=0}^{N-1} e^{-2\frac{(t-\mu)^2}{s^2}}
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
0 &= \frac{dE}{d\mu} \\
&= \sum_{t=0}^{N-1} \left( (-2mf(t) + 2am + 2btm) \frac{2}{s^2} (t-\mu) e^{-\frac{(t-\mu)^2}{s^2}} + m^2 (t-\mu) \frac{4}{s^2} e^{-2\frac{(t-\mu)^2}{s^2}} \right) \\
&= -4m \sum_{t=0}^{N-1} (t-\mu) f(t) e^{-\frac{(t-\mu)^2}{s^2}} + 4am \sum_{t=0}^{N-1} (t-\mu) e^{-\frac{(t-\mu)^2}{s^2}} \\
&\quad + 4bm \sum_{t=0}^{N-1} t(t-\mu) e^{-\frac{(t-\mu)^2}{s^2}} + 4m^2 \sum_{t=0}^{N-1} (t-\mu) e^{-2\frac{(t-\mu)^2}{s^2}} \quad (\text{if } s^2 \neq 0)
\end{aligned} \tag{3.12}$$

$$\begin{aligned}
0 &= \frac{dE}{ds} \\
&= \sum_{t=0}^{N-1} \left( (-2mf(t) + 2am + 2btm) \frac{2}{s^3} (t-\mu)^2 e^{-\frac{(t-\mu)^2}{s^2}} + m^2 \frac{4}{s^3} (t-\mu)^2 e^{-2\frac{(t-\mu)^2}{s^2}} \right) \\
&= -4m \sum_{t=0}^{N-1} (t-\mu)^2 f(t) e^{-\frac{(t-\mu)^2}{s^2}} + 4am \sum_{t=0}^{N-1} (t-\mu)^2 e^{-\frac{(t-\mu)^2}{s^2}} \\
&\quad + 4bm \sum_{t=0}^{N-1} t(t-\mu)^2 e^{-\frac{(t-\mu)^2}{s^2}} + 4m^2 \sum_{t=0}^{N-1} (t-\mu)^2 e^{-2\frac{(t-\mu)^2}{s^2}} \quad (\text{if } s^3 \neq 0)
\end{aligned} \tag{3.13}$$

Applying some notation will simplify these 5 equations. Let

$$\begin{aligned}
T_0 &= \sum_{t=0}^{N-1} 1, & T_1 &= \sum_{t=0}^{N-1} t, & T_2 &= \sum_{t=0}^{N-1} t^2, \\
F_0 &= \sum_{t=0}^{N-1} f(t), & F_1 &= \sum_{t=0}^{N-1} tf(t), & F_2 &= \sum_{t=0}^{N-1} tf^2(t) \\
A &= e^{-\frac{(t-\mu)^2}{s^2}} \\
A_0 &= \sum_{t=0}^{N-1} e^{-\frac{(t-\mu)^2}{s^2}}, & A_1 &= \sum_{t=0}^{N-1} te^{-\frac{(t-\mu)^2}{s^2}}, & A_{F0} &= \sum_{t=0}^{N-1} f(t) e^{-\frac{(t-\mu)^2}{s^2}}, \\
A_{F1} &= \sum_{t=0}^{N-1} tf(t) e^{-\frac{(t-\mu)^2}{s^2}} \\
A_{20} &= \sum_{t=0}^{N-1} e^{-2\frac{(t-\mu)^2}{s^2}}, & A_{21} &= \sum_{t=0}^{N-1} te^{-2\frac{(t-\mu)^2}{s^2}}, & A_{2F0} &= \sum_{t=0}^{N-1} f(t) e^{-2\frac{(t-\mu)^2}{s^2}}, \\
A_{2F1} &= \sum_{t=0}^{N-1} tf(t) e^{-2\frac{(t-\mu)^2}{s^2}}
\end{aligned} \tag{3.14}$$

The 5 equalities can then be written

$$F_0 = aT_0 + bT_1 + mA_0 \quad 3.15$$

$$F_1 = aT_1 + bT_2 + mA_1 \quad 3.16$$

$$A_{F0} = aA_0 + bA_1 + mA_{20} \quad 3.17$$

$$(A_{F1} - \mu A_{F0}) = a(A_1 - \mu A_0) + b(A_2 - \mu A_1) + m(A_{21} - \mu A_{20}) \quad 3.18$$

$$(A_{F2} - 2\mu A_{F1} + \mu^2 A_{F0}) = a(A_2 - 2\mu A_1 + \mu^2 A_0) + b(A_3 - 2\mu A_2 + \mu^2 A_1) + m(A_{2F2} - 2\mu A_{2F1} + \mu^2 A_{2F0}) \quad 3.19$$

NOTE: equalities 3.18 and 3.19 assume  $s \neq 0$ , and  $m \neq 0$ .

Each of the summations ( $F$ ,  $T$ ,  $A$ ) can be determined from the source data and  $\mu$  and  $s$ .

The 5 equations do not immediately present themselves for a linear solution, because the summation terms functions of  $\mu$  and  $s$ . However, if one can estimate  $\mu$  and  $s$ , the first 3 equalities form a simple 3x3 set of linear equations, simply solved by Gaussian elimination. This leads to the possibility of a 2 phase search for an optimum fit.

For each  $\mu$  and  $s$ , one can quickly run the first three equations and determine  $a$ ,  $b$  and  $m$ . A non-linear search technique can then be applied to  $\mu$  and  $s$  to search for an optimum.

A reasonable first choice of  $\mu$  is simply the value of  $t$  which has the largest  $f(t)$  value.

The algorithm is:

1. select  $\mu$  by simply locating the largest  $f(t)$
2. set  $s=50$  (an arbitrary but effective value)

3. evaluate the summations
4. determine  $a$ ,  $b$  and  $m$  by solving the three first equations
5. substitute all 5 unknowns ( $a$ ,  $b$ ,  $m$ ,  $\mu$  and  $s$ ) back and evaluate E over the range
6. modify  $\mu$  and  $s$  in a Hooke and Jeeves [32] manner, repeating back to step 3 until the Hooke and Jeeves step size is smaller than a time sample

Experiments with this algorithm show a tendency to lock onto local minima easily if  $\mu$  is not close to the bell shape of the largest peak. Despite this tendency, this algorithm is quite useful and fast.

After running the curve match, the 5 parameters reflect some aspect of the autocorrelation. The  $a$  and  $b$  parameters simply establish a baseline above which the detection peak occurs. The  $m$  parameter is a measure of “signal to noise”; that is, how distinct the peak is above the local noise floor. Since the scale of the autocorrelation is not consistent, this experiment reports a figure of merit called  $m'$  which is  $20\log_{10}(m/(a+b*\mu))$ .

$$m' = 20\log_{10} \frac{m}{(a+b\mu)} \quad 3.20$$

The  $\mu'$  parameter is in the time domain and indicates the most likely period of the signal; the time shift where the signal is most similar to itself. The  $s$  parameter indicates the width of the bell, and indicates the “confidence” that  $\mu$  is the true period.

### **3.2.1.1 Experiment and Results**

This algorithm is applied to the first pair of heartbeats from “pda5.wav”, one of the

---

1 The  $\mu$  number reported in these experiments is in terms of “sample index” from the experimental database. Since the database is sampled at 4000Hz, these  $\mu$  can be converted to time by dividing by 4000.

samples from the experimental database. The signal is then preprocessed using four different signal filters:

- 1.the raw signal (unmodified):  $w(t)$
- 2.the energy of the signal:  $w(t)*w(t)$
- 3.the “Hilbert energy”:  $|hilbert(w(t))|$
- 4.the magnitude of the signal:  $|w(t)|$

The graphs and processing are based on the samples 500 to 5000 of the autocorrelation curve, to eliminate the dominant peaks at  $t=0$ . All the values of  $t$  (time) and  $\mu$  (time offset) are in units of “data samples”; the heart sounds are all quantized at 4000Hz, so a  $\mu$  of 3000 would correspond to a time offset of 0.75 s.

In many cases, the autocorrelation of the raw signal yields a peak that quite definite in the time domain; that is, the  $s$  is quite small, implying a precise knowledge in the time domain of the beat-to-beat period (based on self-similarity). The  $\mu$  is often incorrect, usually by a small amount (2-3%). Unfortunately, if the heartbeats are not exactly the same shape, this peak does not appear. By observation, this method of constructing the autocorrelation is deprecated. Figure 3.9 illustrates the narrow peak of raw correlation.

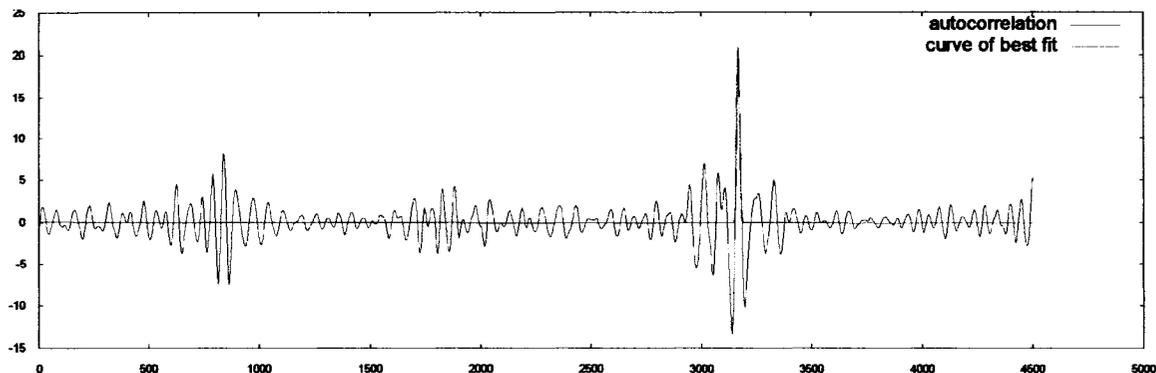


Figure 3.9: The autocorrelation of the raw signal, with curve match

$$m'=46\text{dB}; \mu=3164 (0.79 \text{ s}); s=9$$

The energy of the signal forms a consistently good autocorrelation shown in Figure 3.10. The values of  $\mu$  align with visual estimates of the signal period. The  $s$  parameter is often in the 50-80 range, indicating a narrow bell, and some confidence in the determination of  $\mu$ . The  $m'$  parameter is often in the 20-30dB range, making it easy to recognize the peak.

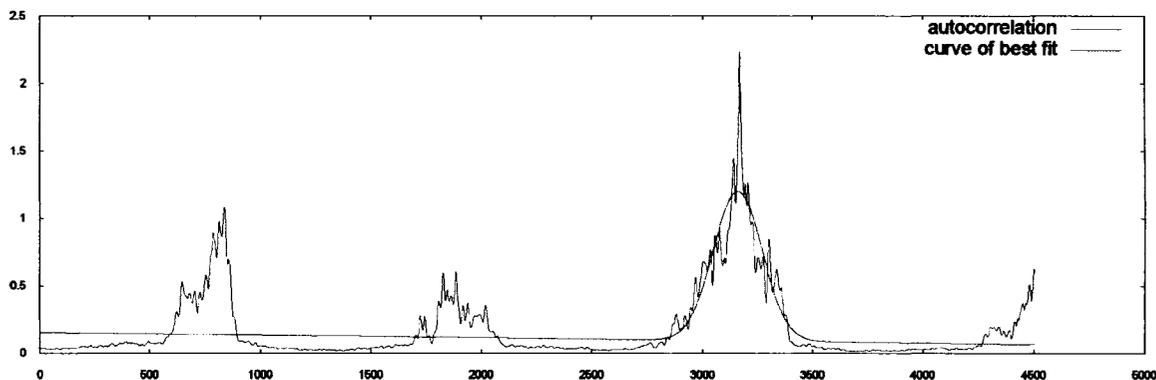


Figure 3.10: Autocorrelation of  $x^2$

$$m'=21 \text{ db}; \mu=3156; s=159$$

The Hilbert magnitude method forms a visually similar autocorrelation (Figure 3.11). The values of  $\mu$  usually match the values determined by the energy method. But the values of  $s$  are almost always larger than the corresponding  $s$  from the energy method, and the  $m'$  parameter is often significantly less, sometimes in the 4-6dB range. This makes it harder to

find the peaks.

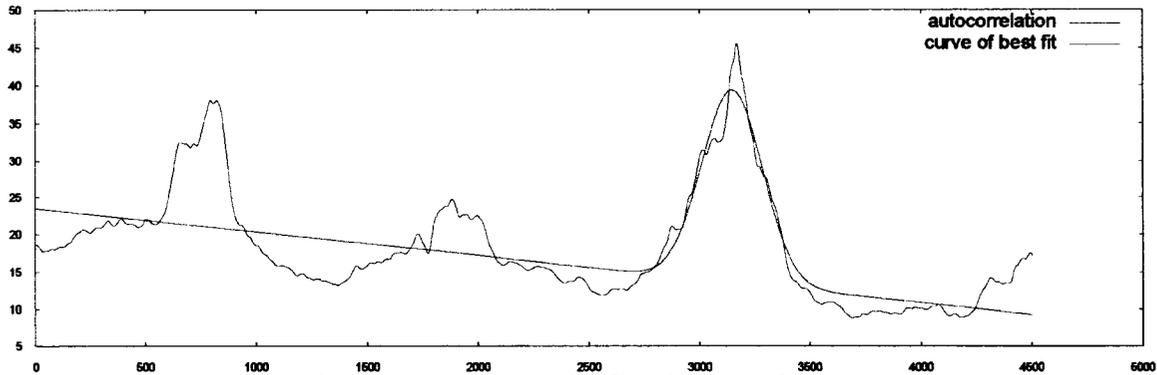


Figure 3.11: Autocorrelation of  $|\text{Hilbert}(w(x))|$ , with curve match

$$m'=5.6\text{dB}, \mu=3143, s=195$$

Finally, the curve for magnitude (Figure 3.12) is very similar to the curves formed by the Hilbert magnitude. Again,  $m'$  is smaller (the peak is less distinct from the noise) and the  $s$  is larger (the bell is wider) than the equivalent values from the energy autocorrelation.

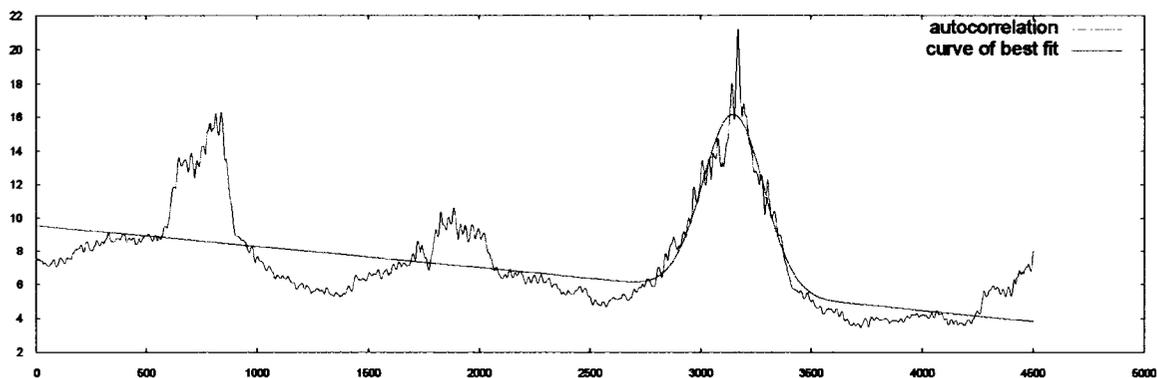


Figure 3.12: Autocorrelation of  $|x|$ , with curve match

$$m'=5.7\text{dB}, \mu=3145, s=195$$

For the test wave file “pda5.wav”, the curve matcher was run against a sliding window of length = 2.4 times the estimated heart sound period. Table 3.4 illustrates the outputs of the

algorithm. This is a fairly high fidelity recording of Patent Ductus Arteriosis, with very little interfering noise. (Figure 3.13) The table seems shows the most consistent beat period when the “raw” autocorrelation is used. Note that the other three forms still give an accurate measure of the heart period  $\mu$ . The  $s$  factor (the width of the Gaussian bell) for the energy correlations is smaller than the Hilbert-magnitude and magnitude versions, because the  $x^2$  effect tends to make the autocorrelation peaks taller and thinner.

On a high quality sample of heart sounds, all four algorithms perform adequately.

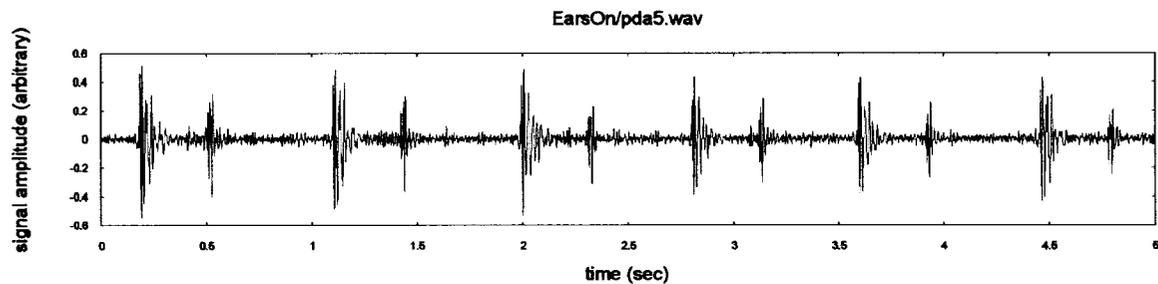


Figure 3.13: Clean heart sound of patent ductus artereosis (pda5.wav, first 5 s)

Span starting at	Raw autocorrelation			Energy autocorrelation			Hilbert magnitude autocorrelation			Magnitude correlation		
	snr	$\mu$	s	snr	$\mu$	s	snr	$\mu$	s	snr	$\mu$	s
0	47.5	3663	10	25.5	3634	173	10.1	3638	223	10.1	3638	224
0.91	48.3	3575	11	18.8	3490	264	5	3470	350	5.1	3469	353
1.81	48.7	3235	10	22.4	3207	134	7.8	3207	182	7.8	3207	178
2.72	49.3	3453	9	17	3356	286	5	3332	294	5.1	3335	295
3.63	48.9	3567	10	23.7	3542	171	6.3	3513	232	6.3	3512	232
4.53	48.3	3508	10	23.3	3511	138	5.8	3507	203	5.9	3505	203
5.44	46.9	3351	10	21	3345	105	3.2	3349	178	3.4	3351	177
6.35	46.5	3530	9	22.1	3539	114	4.4	3536	184	4.6	3538	183
7.25	47.9	3689	9	23.3	3708	161	5.3	3686	194	5.2	3686	195
8.16	45.5	3664	11	22.5	3659	162	4.3	3667	203	4.3	3664	203
9.07	48.5	3369	11	22.3	3369	152	3.4	3382	226	3.5	3382	225
9.97	48.9	3191	10	22	3181	112	4.8	3191	161	4.9	3189	158
10.88	48.3	3372	10	21.9	3385	124	5.1	3375	163	5.2	3379	159

Table 3.4: Curve matching output (sound file: pda5.wav)

The test waveform “pda.wav” (Figure 3.14) is also Patent Ductus Arteriosus, but with a

noisy background and a constant sibilant murmur. Table 3.5 is the output of the curve matcher at various spans across the file. The correlation of the raw signal shows significant variations, consistent with the assumption of section 6.2 of the thesis (see Figure 6.5). Even the energy correlation version suffers from an occasional jitter (at time=7.5 s).

Span starting at time (s)	Raw autocorrelation			Energy autocorrelation			Hilbert magnitude autocorrelation			Magnitude correlation		
	snr	$\mu$	s	snr	$\mu$	s	snr	$\mu$	s	snr	$\mu$	s
0	43.4	2838	9	25.5	2512	372	8.1	2503	561	8.1	2502	562
0.63	-0.8	1139	1769	26.1	2489	422	9.8	2518	603	9.8	2518	603
1.25	2.4	3373	1695	26.8	2305	498	8.8	2389	565	8.8	2389	565
1.88	123.2	4166	59	20	2269	348	3.8	2261	540	3.7	2258	540
2.5	38.9	1624	9	21.8	2519	556	4.3	2524	660	4.3	2525	664
3.13	76.3	4139	65	20	2457	519	6.1	2448	759	6	2448	759
3.75	-6.2	2376	39	16.4	2654	501	2.9	2754	788	2.9	2755	788
4.38	116.6	4124	45	23.8	2611	396	9.3	2600	630	9.3	2599	630
5	14	3140	30	23.2	2602	392	8.8	2546	606	8.7	2545	606
5.63	0	2936	1700	20.2	2675	367	6.1	2630	563	6.1	2631	566
6.26	110	4138	50	23.9	2734	342	9.3	2676	561	9.9	2733	579
6.88	42.9	3109	9	21.2	2779	613	8.5	2713	824	9.5	2735	850
7.51	85.6	4084	39	6.8	3076	28	10.3	2826	1039	10.9	2834	1048
8.13	46.2	2627	9	19.4	2620	137	5.1	2622	751	4.9	2617	749
8.76	44.6	2933	9	21.2	2902	541	4.6	2957	745	3.6	2932	713
9.38	41.9	2894	9	17.8	2594	534	3	2756	668	3	2755	668
10.01	-0.3	848	2685	17.9	2392	45	2.2	2717	601	2.2	2716	599
10.63	0	1384	1700	20	2864	494	3.2	2897	697	2.5	2873	669
11.26	53.5	4029	34	17.7	2744	422	4.1	2793	786	2.8	2738	739
11.88	129.5	4158	57	18.7	2549	32	5.7	2720	878	4.4	2704	813

Table 3.5: Curve matching output (sound file: pda.wav)

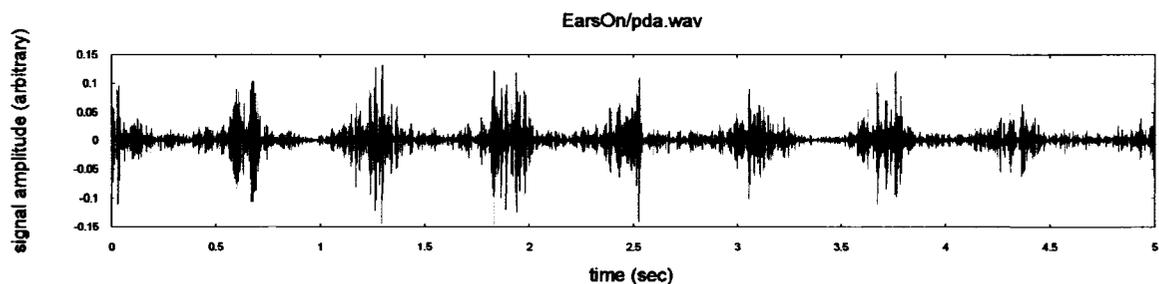


Figure 3.14: Waveform of patent ductus arteriosus (pda.wav, first 5 s)

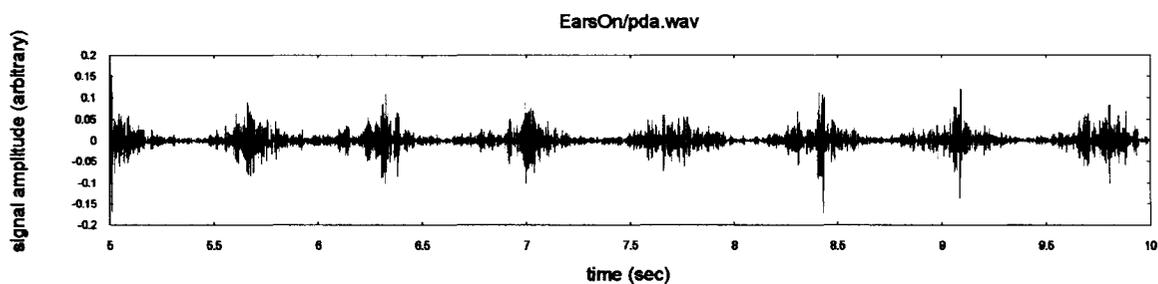


Figure 3.15: Waveform of patent ductus arteriosus (*pda.wav*, second 5 s)

The Hilbert magnitude and magnitude methods yields a more steady estimate than the energy version because the  $x^2$  effect of the latter tends to exaggerate the importance of subharmonic peaks in the autocorrelation, distracting from the peak of the heart rate.

Overall, one can conclude that the Hilbert magnitude and magnitude methods show evidence of being useful in algorithms that use autocorrelations to estimate signal periods. This is consistent with the results reported in chapter 4 of this thesis. This is also the justification for using signal magnitude in the processing described in chapter 6.

### 3.2.1.2 Supplement

After performing the above analysis, it was noted that the autocorrelation waveforms all tend to have a peak at the beat period, even without filtering or curve matching.

For the *pda5.wav* waveform, the peaks are listed in Table 3.6

Span starting at time (s)	Raw autocorrelation	Energy autocorrelation	Hilbert magnitude autocorrelation	Magnitude correlation
0	3665	3667	3667	3667
0.91	3577	3578	3574	3577
1.81	3237	3239	3236	3238
2.72	3455	3454	3454	3454
3.63	3568	3569	3565	3568
4.53	3511	3511	3513	3512
5.44	3352	3351	3350	3350
6.35	3531	3531	3534	3532
7.25	3690	3690	3690	3690
8.16	3666	3667	3669	3667
9.07	3372	3373	3373	3374
9.97	3192	3190	3191	3191
10.88	3373	3371	3371	3371

*Table 3.6: Peaks of autocorrelation (sound file: pda5.wav)*

Note the clean and consistent detection of period.

For the pda.wav waveform, the peaks are listed in Table 3.7.

Again, the autocorrelation of the raw signal shows variations in peak location that suggest that it is an inappropriate measure of heartbeat period. The energy autocorrelation peaks also vary, but not as much. The Hilbert magnitude and magnitude autocorrelation peaks give the most useful measure of heartbeat period. The measurements presented in Table 3.7, are not identical to those presented in Table 3.5, but are certainly within a range that makes this a useful technique for estimating the heart rate (chapter 4).

Span starting at time (s)	Raw autocorrelation	Energy autocorrelation	Hilbert magnitude autocorrelation	Magnitude correlation
0	2502	2658	2652	2658
0.63	2313	2381	2380	2381
1.25	2289	2289	2289	2289
1.88	2291	2356	2315	2186
2.5	2432	2830	2469	2627
3.13	2524	2422	2417	2423
3.75	2568	2579	2575	2579
4.38	2723	2611	2667	2662
5	2876	2611	2616	2611
5.63	2655	2700	2707	2701
6.26	2504	2700	2756	2746
6.88	2676	2658	2672	2673
7.51	3149	3082	2607	2610
8.13	2628	2628	2631	2627
8.76	2875	2874	2877	2875
9.38	2658	2594	2590	2593
10.01	2386	2385	2738	2738
10.63	2628	2891	2888	2879
11.26	3137	2777	2779	2777
11.88	2467	2556	2554	2557

*Table 3.7: Peaks of autocorrelation (sound file: pda.wav)*

In a practical system, the “peak” of the autocorrelation is much simpler to calculate than the curve matching process described here. Testing shows that simply taking the peak of the magnitude autocorrelation gives a useful measure of heart rate.

## Chapter 4

# Heart Rate Estimation

The human heart can beat at a range of rates, depending on age, health, stress, exercise and other factors. The sample data used in this thesis contains samples that range from 3bps (180bpm, or about 330ms per beat) down to 32bpm (or about 1.9 s per beat), a 6:1 range.

In the algorithms described in the chapters 5 and 6 there are several parameters which are proportional to the beat time. This chapter describes a standalone *rate estimator*.

The heart waveform exhibits significant self similarity, beat to beat, and one can capitalize on this to detect the beat rate.

To detect this self similarity, the envelope of the signal waveform can be autocorrelated, and one would expect a peak of correlation at or around the S1-to-S1 period, the beat-to-beat period.

For example, a 3 second sample of a PCG is shown in Figure 4.1, with its autocorrelation (only the second half of the correlation is shown). Note the significant peak at 800ms, indicating the S1-to-S1 period.

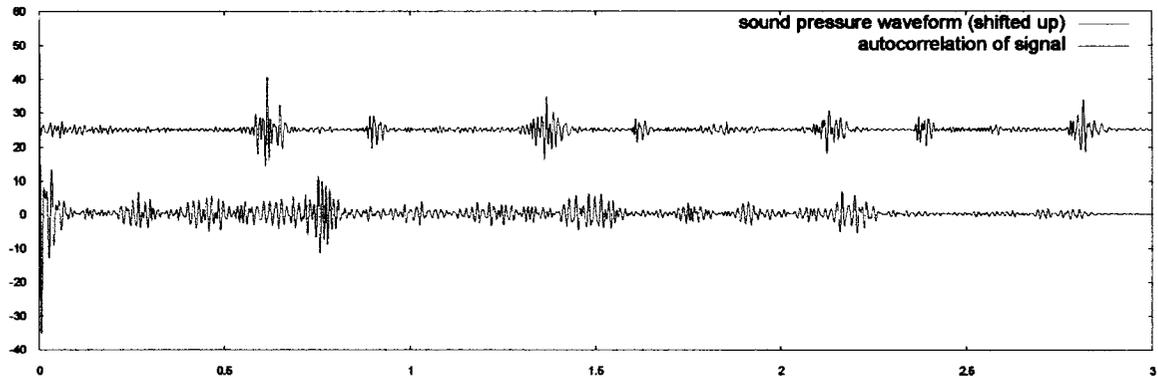


Figure 4.1: Typical heart sound and its autocorrelation

Figure 4.1 illustrates two features:

- the autocorrelation clearly shows the period of the heart signal, but does not indicate the beat boundaries
- the shape of the heart signal shows enough variation beat-to-beat that a simple correlation may be inadequate.

By observation, the time domain signal does not autocorrelate well. The signal is bipolar and not consistent in the details from beat to beat. This is reasonable, based on the physiology, considering that the acoustic path probably varies from beat to beat. A sensible way to estimate the beat period is by examining the autocorrelation of the signal energy, or Hilbert magnitude.

$$R_e(j) = \sum_{m=0}^{N-1} w_m^2 \times w_{m+j}^2$$

$$R_h(j) = \sum_{m=0}^{N-1} |\hat{w}_m| \times |\hat{w}_{m+j}|$$

4.1

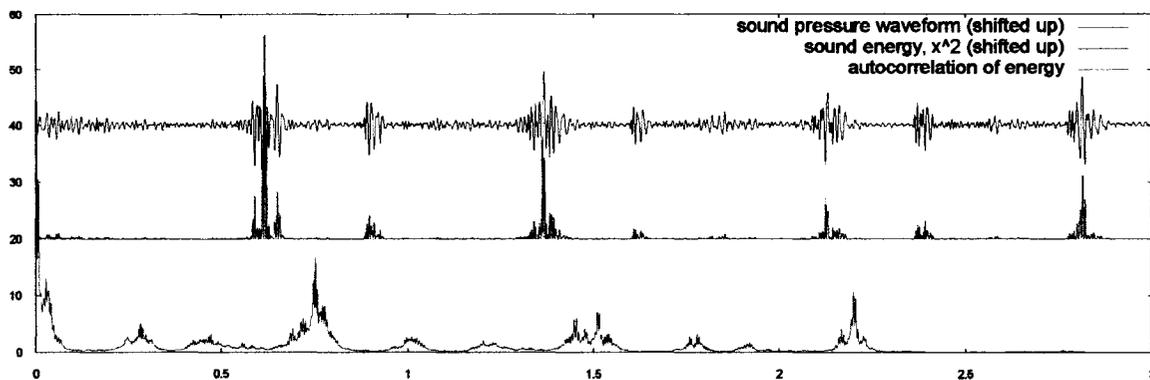


Figure 4.2: Autocorrelation of the signal energy,  $Re(j)$

The magnitude of the Hilbert transform was chosen because it operates like a frequency dependent low pass filter, filling in the troughs of the signal while keeping the high frequency content (edges).

This formulation results in a sharp waveform as shown in Figure 4.2. The period, which is the target measurement, is an obvious peak at 800ms. It is not the highest point on the curve; since this is an autocorrelation, the absolute peak is at  $\Delta time=0$ .

For the purpose of preprocessing, one only needs a rough estimate of the heart rate, which is usually the time of the first large peak of the autocorrelation. To extract the first and largest, the autocorrelation is multiplied by a window designed to isolate the most likely candidate, taking into account the physiological properties of the heart.

A trivial window function suppresses the autocorrelation peak at  $\Delta time=0$  and the accompanying skirts out to  $\Delta time=0.1$  s. When applied against the experimental database, 160 (77.7%) out of the 206 sound files gave a correct rate estimate (between the min. and max. heart rates of the sound file).

A better taper will emphasize the components in the range that are sensible for a human

heartbeats: between 300ms and 1.5ms. A two sided taper window is used because there are cases in the corpus where the autocorrelation have peaks much below the “period” or much above the “period”. At the same time, the taper can eliminate the natural peak which occurs in autocorrelations at  $\Delta time=0$

This tapered window used gives good results. A shaped exponential window of

$$\begin{aligned} T_{corner} &= 750 \text{ ms} \\ f_{window}(t) &= t/T_{corner} \times e^{-t/T_{corner}} \end{aligned} \quad 4.2$$

used over the autocorrelation yielded the correct average beat period in 187 out of the 206 sound files in the sample corpus (90.8%).

Note: Other window functions that were tested performed poorer. For instance, Figure 4.3 shows a window function of:

$$\begin{aligned} hf_{window} &= t & t < 0.2s \\ &= 0.2 & 0.2s \leq t < 1.0s \\ &= 0.25 - t/20 & 1.0s \leq t < 5.0s \\ &= 0 & otherwise \end{aligned} \quad 4.3$$

yielded a correct average beat period in 180 out of 206 waveforms (87%).

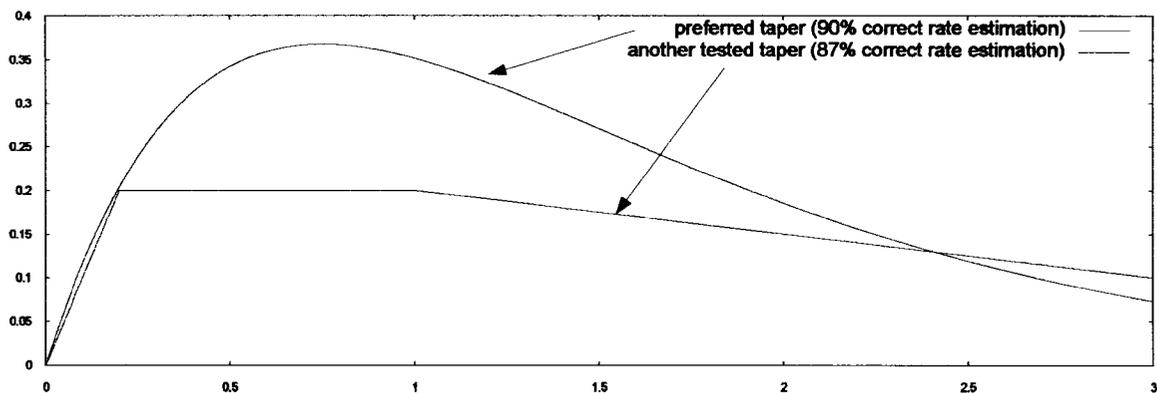


Figure 4.3: Windows used on the autocorrelation to isolate the most likely rate peaks

To summarize, the elements are in place to design a heart rate estimation algorithm: autocorrelation, using either energy or Hilbert magnitude, followed by a tapered window function.

#### 4.1 Implementation

This section discloses the algorithm used in the experiments.

Given a heart sound file of samples  $w'_m$ , the signal first has its DC energy removed,

$$w_m = w'_m - \text{mean}(w') = w'_m - \sum (w'_m)/N \quad 4.4$$

The energy is

$$E_m = w_m^2 \quad 4.5$$

and its autocorrelation is

$$R_t = \sum_{m=-N-1}^{N-1} E_m E_{m-t} \quad 4.6$$

In searching for “similarity” peaks, we are only interested in the positive  $t$  side of  $R_t$ . The autocorrelation is shaped by the window function specified in Eq 4.7

$$R'_t = R_t \frac{t}{0.75} e^{\frac{-t}{0.75}} \quad t > 0 \quad 4.7$$

Finally, the maximum peak of  $R'_t$  is selected, and this value of  $t$  is taken as a measure of the heart rate of the sound file.

Because of the nature of autocorrelation, this method tends to yield more of a statistical mode than a mean, and this fact is borne out in the results. Heart sounds which have a

significant variation in beat period demonstrate a spreading of the first peak of the autocorrelation. This does not interfere with the performance of the algorithm. The statistical mode is a useful figure for the algorithms described in chapters 5 and 6.

## 4.2 Experimental Results

The corpus of experimental data was processed through the algorithm described above. To be considered a success, the estimator must yield a heartbeat period that is inside the window:

$$0.95 t_{min} < t_{estimate} < 1.05 t_{max} \quad 4.8$$

where  $t_{min}$  is the shortest observed beat-to-beat time, and  $t_{max}$  is the longest.

The basic rate estimator produces a useful rate estimate in 90.8% of the sound files. If only the “complexity type 1” (simple) sounds are considered, it performed better, returning a meaningful rate estimate in 94.6% of the files. (see Table 4.1)

The bulk of the failures of the basic estimator were due to incorrectly estimating the rate of pediatric heart sounds (with heartbeat periods  $< 0.5$  s). The window function (Eq. 4.7) gives emphasis to peaks between 0.5 and 1.5 s, and for samples from pediatric patients, the algorithm picks up the 2<sup>nd</sup> or 3<sup>rd</sup> peak, incorrectly estimating the heart rate.

The rate estimator was recoded with an alternate window function, with a faster time constant:

$$R'_t = R_t \frac{t}{0.5} e^{\frac{-t}{0.5}} \quad t > 0 \quad 4.9$$

While this modification improved the performance of the algorithm against pediatric

sounds, it failed on others, giving similar performance to the original design. (see Table 4.1)

<i>algorithm</i>	<i>all samples 206 files</i>	<i>only simple sounds (type 1) 166 files</i>	<i>only moderate sounds (type 2) 30 files</i>	<i>only complex sounds (type 3) 10 files</i>
basic energy $x^2$	187 = 90.8%	157 = 94.6%	27 = 90.0%	3 = 30%
basic energy, faster taper ( $T_{\text{corner}} = 0.5$ s)	180 = 87.4%	146 = 88.0%	29 = 96.7%	5 = 50%
Hilbert magnitude	192 = 93.2%	160 = 96.4%	27 = 90.0%	5 = 50%
magnitude $ x $	188 = 91.3%	157 = 94.6%	28 = 93.3%	3 = 30%

Table 4.1: Performance of the rate estimator against the experimental data

For completeness, the rate estimator was rebuilt to use the Hilbert magnitude, instead of the energy ( $w^2$ ).

$$R_t^H = \sum_{m=-N-1}^{N-1} |\text{Hilbert}(w_m)| \times |\text{Hilbert}(w_{m-t})| \quad 4.10$$

$$R_t^{H'} = R_t^H \frac{t}{0.75} e^{\frac{-t}{0.75}} \quad t > 0 \quad 4.11$$

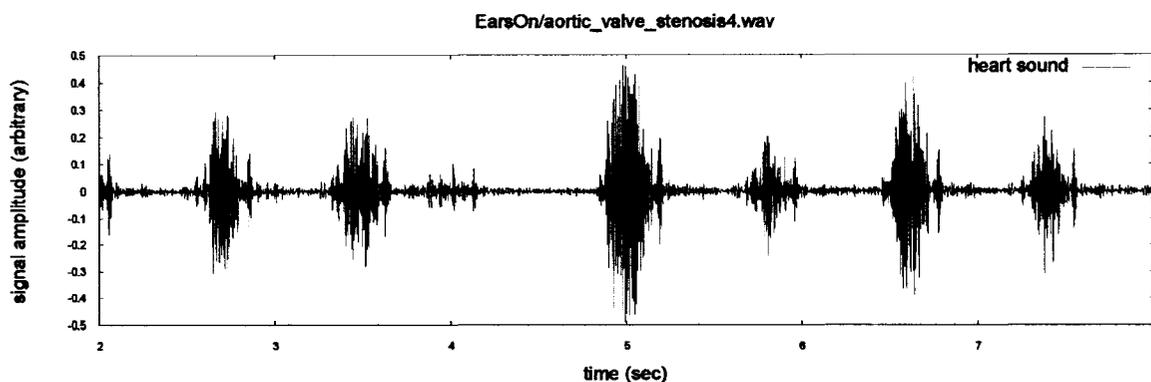
and again using the signal magnitude.

The performance difference of the latter two modifications is small. The Hilbert magnitude version scored slightly higher than the energy version, for the reason explained below, and discussed further in 3.3.2 above. The energy and Hilbert versions are used in the experiments reported in chapters 5 and 6.

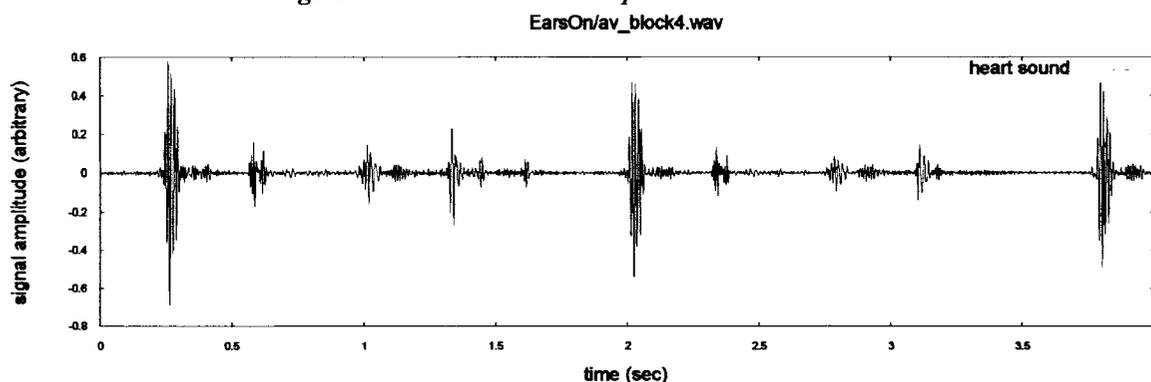
The sound files which failed to score correctly were sounds

- with dominant similarity in alternate heartbeats; see Figures 4.4 & 4.5; the estimate of beat period is  $\sim 2$  times too long
- with similar sized S1 and S2, and significant rate variations ( $>\pm 20\%$ ); the autocorrelation will have a large peak at  $\Delta t = T_{\text{systole}}$  which can distract from the various smaller peaks that are due to full heartbeats; the estimate of beat period is  $\sim 2$  time too short; fortunately, this is fairly rare, only occurring in 2 of the files in the database
- with a rapid heartbeats,  $< 0.5$  seconds, so that the peak was attenuated by the 0.75 s time constant of the window function; these are almost always pediatric cases, and if the system is aware, *a priori*, that the heart sound is pediatric, a shorter time constant could be used

To evaluate the effect of the taper length, the database was divided in two, separating those files that had average heart rates faster than 0.5 s. There are 192 sound files in the slow group, and 14 files in the fast heartbeat group. When the longer time constant taper is used only on heartbeats that are over 0.5 s average period, it scored correctly 181 out of 192 sound files (94.3%), whereas the faster taper only scored 169 against same 192. When the shorter time constant taper is used only on heartbeats that are faster than 0.5 s, it scored correctly 11 out of 14 sound files (78.6%), whereas the longer taper only scored 6 out of the same 14 sound files.



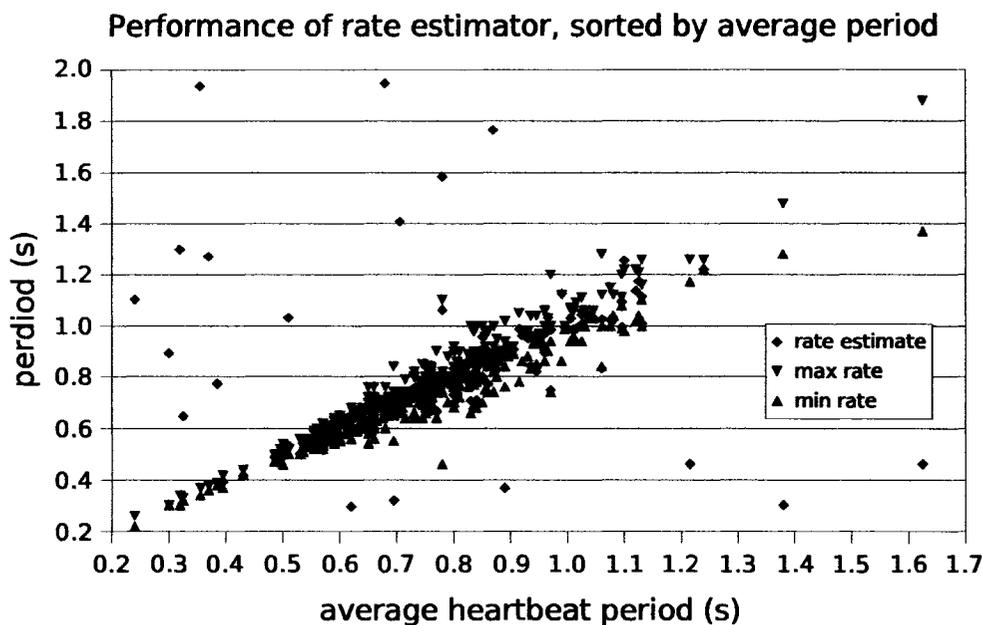
*Figure 4.4: Heart sound with emphasis on alternate beats*



*Figure 4.5: Heart sound with alternate sounds emphasized*

These results are remarkably accurate, and in several cases, caused a review of the manual method, because the computer estimate pointed out several errors due to keyboard errors during the manual scan.

The Hilbert magnitude version outperforms the energy version of the rate estimator primarily because several files in the experimental data include several files with a modulation that affects every second beat. After autocorrelation, the peaks that correspond to twice the beat period are over-emphasized by the  $x^2$  effect of the energy calculation. For example, if beats 1, 3 and 5 are 50% larger than beats 2 and 4, then the energy of the odd beats will be 2.25 times larger, with a corresponding larger peak in the autocorrelation. Further work is required to determine which version is better for general work.



*Figure 4.6: Scatter graph of the performance of the rate estimation algorithm*

Figure 4.6 shows the performance of the main rate estimation algorithm, with the diamonds plotting the output of the algorithm. The outliers are clearly shown, and are much more frequent at the extremities of the working range. This suggests that the algorithm may benefit from external information about the heart rate, such as might come from the age of the patient (pediatric patients tend to have faster heart rates).

The proposed *rate estimation* algorithm performs well overall. It is used as the first stage for the segmentation systems proposed in the next two chapters.

## Chapter 5

# Peak Detecting Segmentation

### 5.1 Introduction

The PCG pressure signal is bipolar, and any peak detector will need to rectify it somehow.

One could consider using several techniques:

- signal magnitude ( $|x|$ )
- the magnitude of the Hilbert transform ( $|\text{hilbert}(x)|$ )
- the energy ( $x^2$ )

Based on early analysis of experimental data (qualifying the shape of the autocorrelation curves, using the of 3.3.2 above), the energy version was chosen to be used for the rest of this section. It accentuates the height of the peaks, and leads to better detection in the absence of interference.

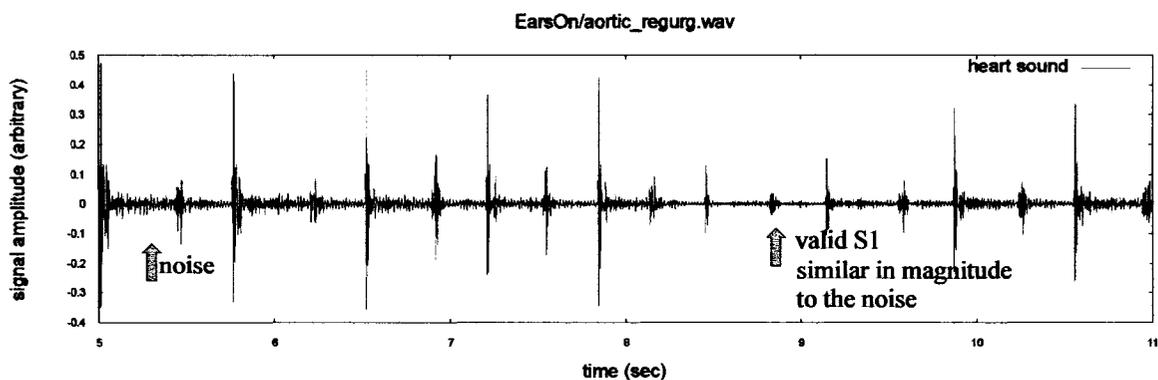


Figure 5.1: Typical heart sound showing variation in magnitude over time

It is common for the amplitude to vary in magnitude by a factor of 3 or more, even over a short period. Any fixed magnitude threshold that extracts unique peaks may fail to function properly as the signal fades; in Figure 5.1, a threshold set above the noise at time 5.2 s (greater than 0.03 magnitude) will fail to detect the S1 sound at time 8.8 s.

## 5.2 Block diagram

This algorithm is built around a simple threshold detector, where the threshold varies in time, tracking the signal magnitude. Figure 5.2 illustrates the concept.

To create the threshold level, the energy of the signal ( $x^2$ ) is passed through a simple third order low pass Butterworth filter. The sound waveform has frequencies from 30 to 250 Hz [18]. A low corner frequency (about 2Hz) is chosen to span the variations that occur during a beat cycle (~50ms), but to relax between beats (~200ms).

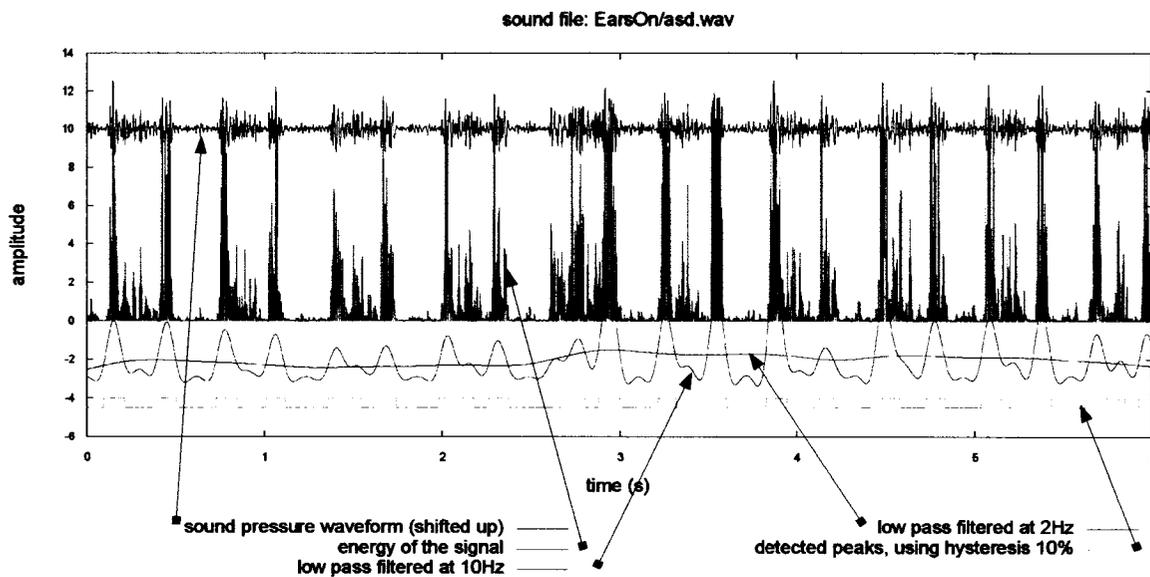
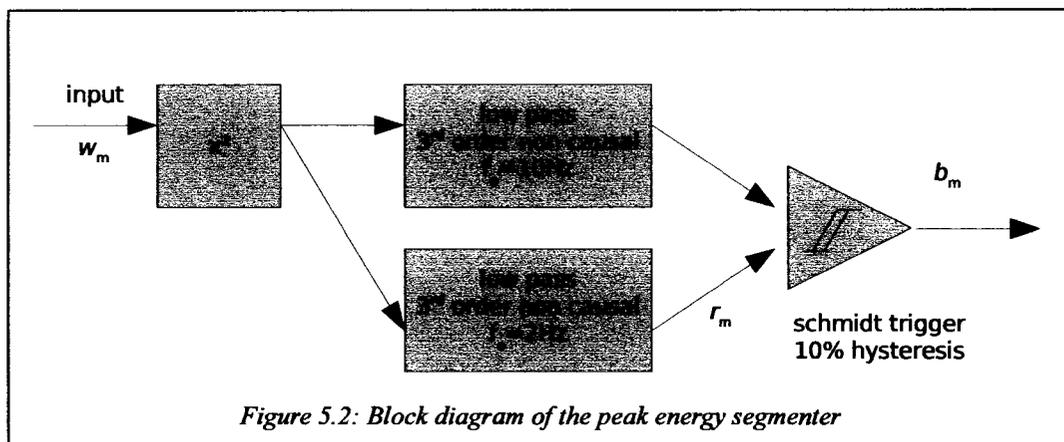
$$r_m = H(w_m^2) \quad 5.1$$

Comparing the signal energy against this windowed average gives a first reasonable beat detector.

$$b_m = w_m^2 > r_m \quad 5.2$$

where  $r_m$  are the windowed local averages and  $w_m$  are the data from the PCG.

To avoid multiple detections from the various spikes of the S1 and S2 sounds, the  $w_m^2$  signal path is also filtered, at a higher frequency, enough to let the features of the sound envelope pass through to the threshold detector. As is typical in practical systems, the threshold detector implements some hysteresis, in this case  $\pm 10\%$ .



When the heartbeat contains two dominant peaks, as in Figure 5.3, the simple slicing detector is inadequate. The simple “slicing” detector will pick up both S1 and S2. A post-processing step is required to take every 2<sup>nd</sup> edge. A common technique to separate the beats is to simply assume that  $T_{\text{systole}} < T_{\text{diastole}}$ . The post-processor must be able to deal with waveforms that only yield one edge per heartbeats, as in Figure 5.4. Even this will fail

when the slicer fails to pick up both S1 and S2, as shown in Figure 5.5.

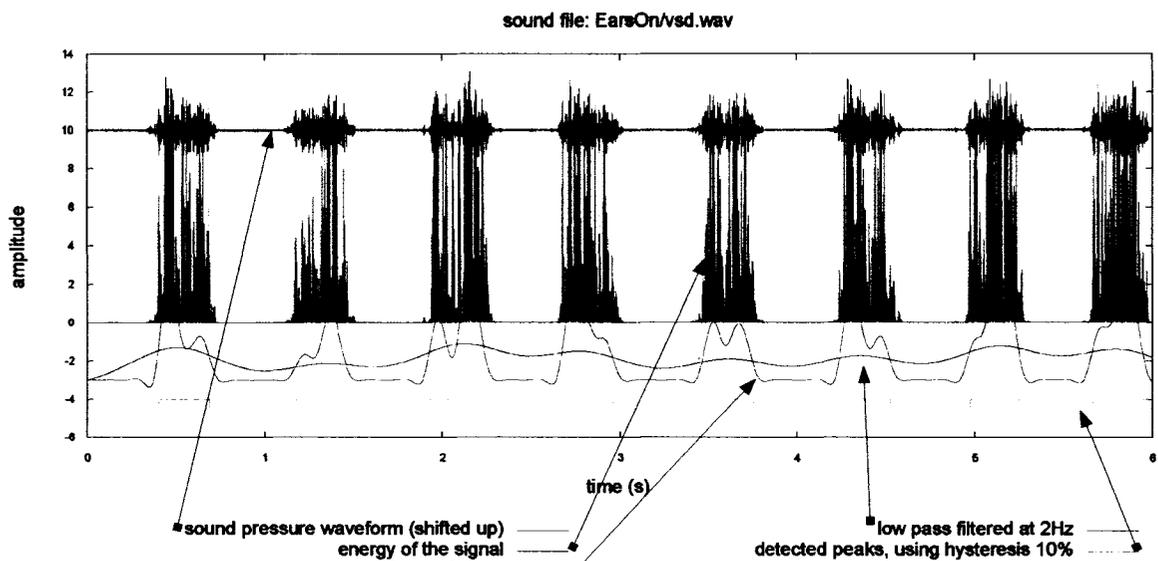


Figure 5.4: Peak energy detector on heart sound with holosystolic murmur

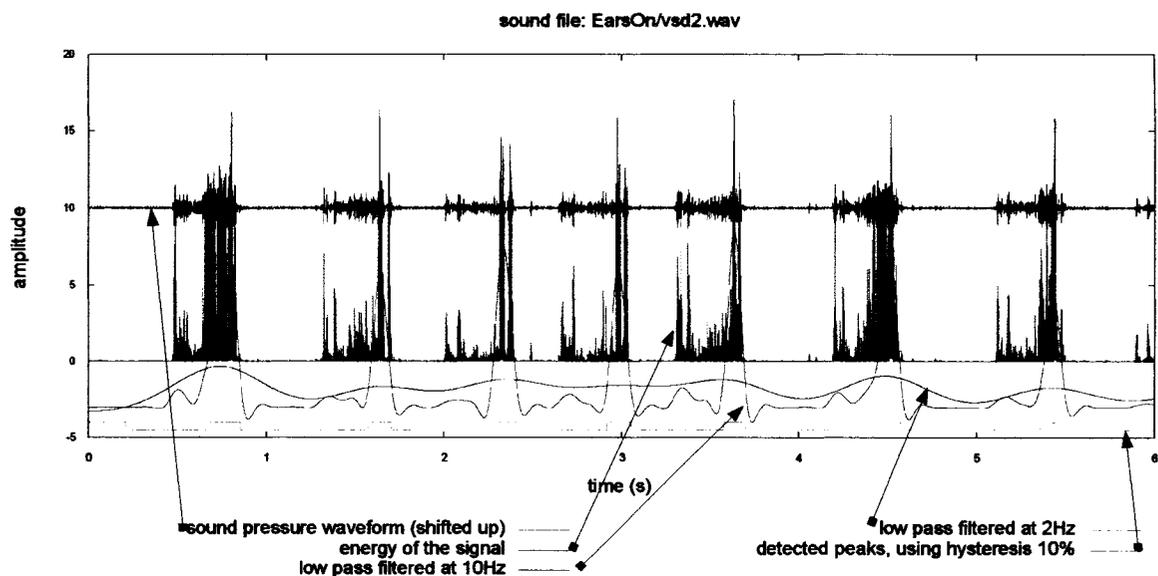


Figure 5.5: Peak energy detector operating on a dominant S2

By observation, it seems possible to improve the beat detection by adding heuristics to the

post-processing stage, including:

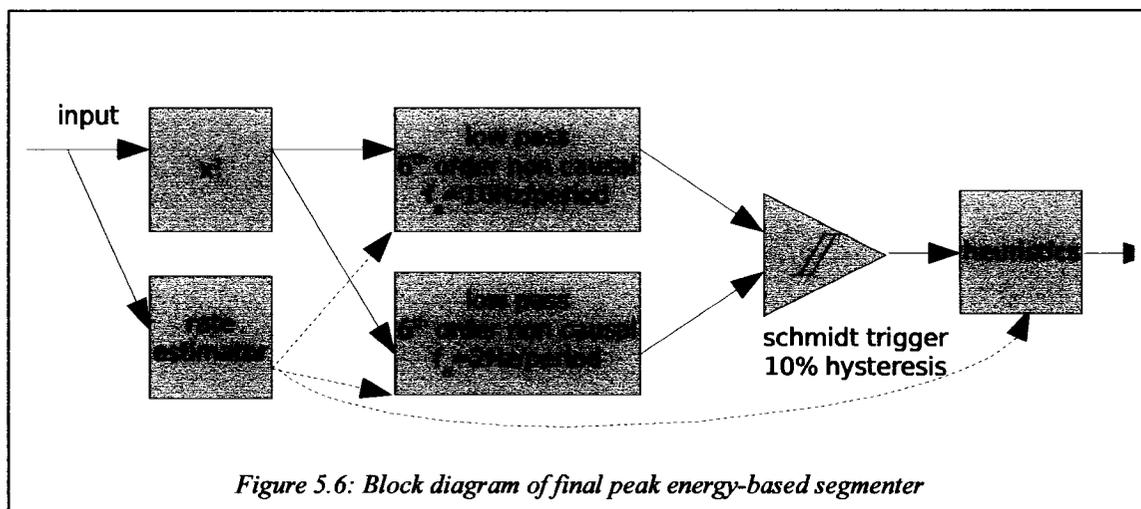
- i) ignoring peaks which occur too close together
- ii) opening a “candidate” window only after a time has passed (approx  $0.7 * T_{\text{beat\_rate}}$ )  
where the rate is estimated by the process described in chapter 4
- iii) mark the beat boundary as beginning  $T_{\text{beat\_rate}}/12$  before the first edge of the beat

These rules are coded into a Matlab/Octave program, and run against the corpus of heart sounds. The results are presented in section 5.4.

### 5.3 Implementation

The long term target of this segmentation is to time-align the heartbeats. However the low pass filters used in the processing above introduce phase delays. Since all this processing is off-line, one can take advantage of non-causal filters, and in particular, the `filtfilt()` function of Matlab/Octave. This allows one to build zero-delay filters and keep the time alignment. In brief, the 3 pole Butterworth low pass is replaced by a 6 pole Butterworth non-causal filter, 3 poles outside the unit circle and 3 poles inside.

Further, a fixed-frequency filter seems too constricting, considering that the sample data includes heartbeats that range from 0.3 s each, up to 1.8 s each. To track the signal, the filter banks in Figure 5.2 are modified to be proportional to the beat rate. The lower filter is set to 2Hz/period, and the faster filter is set to 10Hz/period, where the period can be determined by running the “rate estimator” described in chapter 4.



The algorithm of Figure 5.6 was created in Matlab/Octave as described in chapter 3, and run against the experimental database.

## 5.4 Results & Discussion

The energy detector does a moderately good job at detecting the presence of heart sounds. These techniques are currently used in Holter counters and other diagnostic equipment to detect missing beats and profile the heart rate.

This techniques does *not* determine the edge of the heartbeats boundary explicitly, which is the target of the *segmentation* challenge. The approximation used in these heuristics is that the heartbeat begins at a point in time 10% of the period *before* the S1 sound. Additionally, the identification of the S1 sound is based simply on the gap that usually exists during the diastolic time.

For completeness, this algorithm was run twice: once using Figure 5.6, and again using the Hilbert magnitude version of the rate estimator, and replacing the  $x^2$  block with a

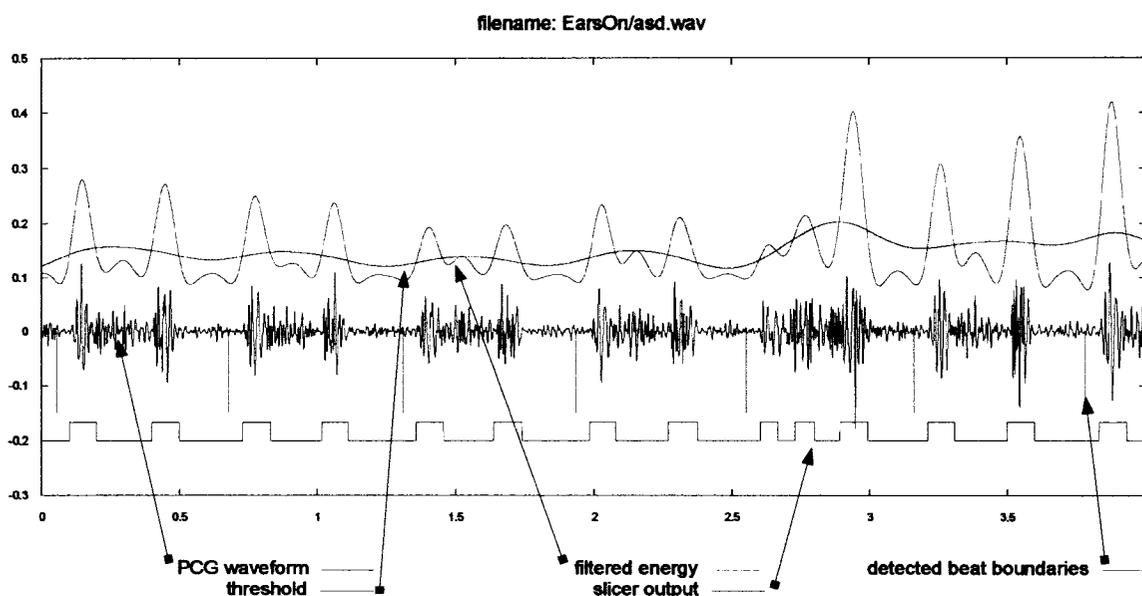
$|\text{Hilbert}(x)|$  block.

The algorithm was run against the experimental database, which contains 2709 heartbeats within heart sounds of varying complexity. The segmentation results were scored manually by the author, using a graphical display of the segmentation marks superimposed on the waveform. An individual heartbeat was marked incorrect if the corresponding segmentation mark was missing, located between S1 and S2, or located inconsistently with the majority of segmentation marks in the file.

<i>algorithm</i>	<i>all samples 2709 beats</i>	<i>only simple sounds (type 1) 1948 beats</i>	<i>only moderate sounds (type 2) 589 beats</i>	<i>only complex sounds (type 3) 172 beats</i>
Peak energy segmenter	1742 = 64.3%	1278 = 65.6%	446 = 75.7%	18 = 10.5%
Peak energy segmenter, using Hilbert magnitude	1845 = 68%	1387 = 71.2%	421 = 71.5%	37/172 = 21.5%

*Table 5.1: Performance of the peak energy segmenter*

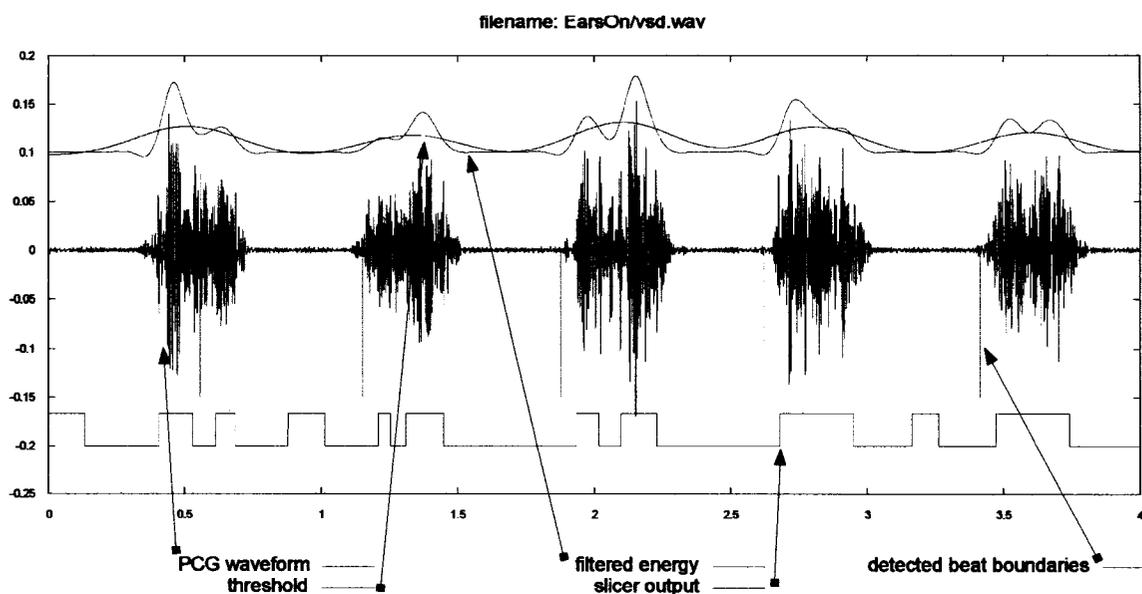
Before proceeding to the next section, here are some sample waveforms with comments.



*Figure 5.7: Peak energy segmenter operating on a sample with distinct S1 and S2*

In Figure 5.7 the S1 and S2 components are clearly visible, and the energy envelope gives a clean outline of the signal. The slicer detects both S1 and S2, but the heuristic rule eliminates the gap between them, and the blue vertical line correctly outlines the separation between beats.

Note that the energy envelope *does* dip down below zero, because of the effect of the 6 pole Butterworth low pass.



*Figure 5.8: Peak energy segmenter operating on a sound with holosystolic murmur*

The sample in Figure 5.8 demonstrates good performance with heart sounds that have merged S1/S2 sounds. The figure also demonstrates the synchronization problem at the left hand side, where the simple heuristics lock into a mid-point of the first beat, and then recovers as the rules are enforced. Edge effects are common in this simple approach.

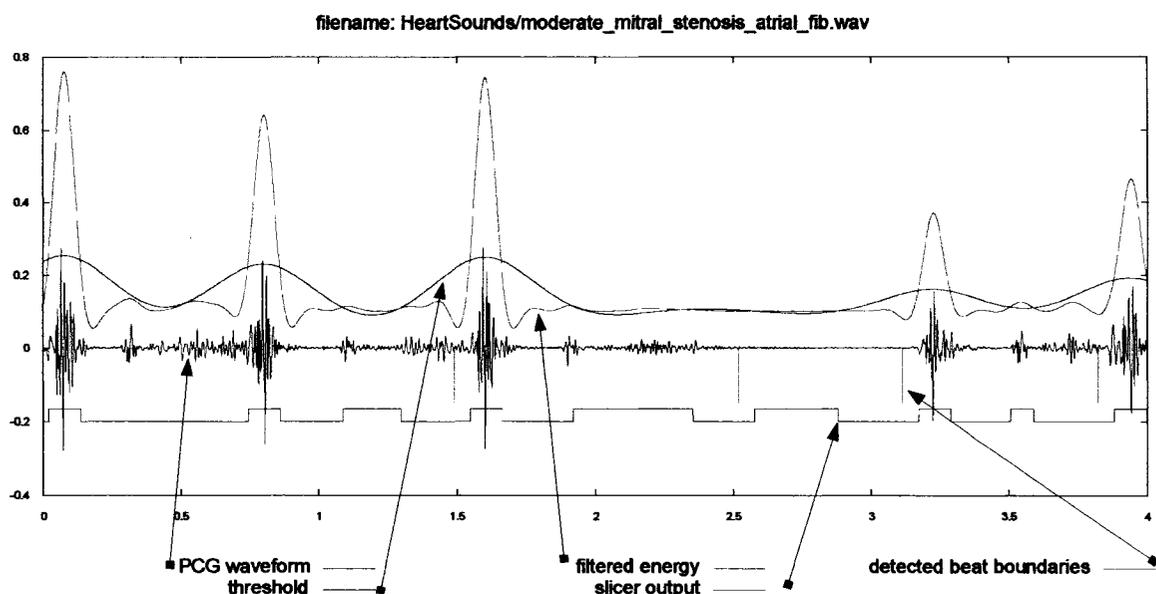


Figure 5.9: Peak energy segmenter with sample that includes a missing beat

Figure 5.9 illustrates a weakness in the simple algorithm, in that it searches for a beat in the interbeat gap at 2.5 s.. The adaptive threshold dips down into the noise and the algorithm incorrectly locks into a bit of noise, reporting a beat at time 2.52 s.

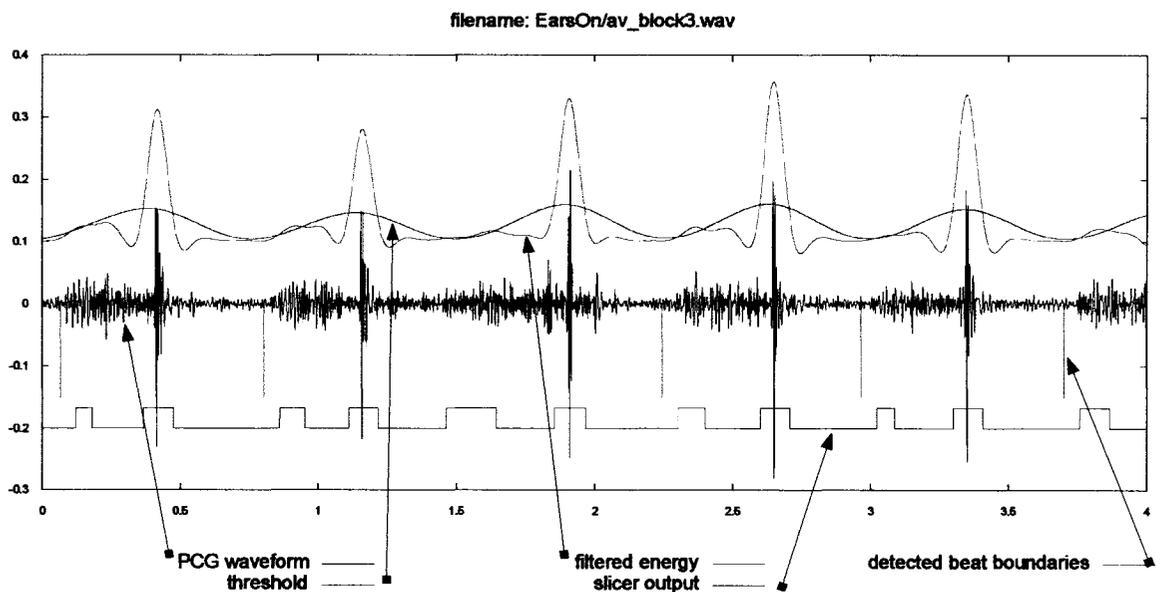


Figure 5.10: Peak energy segmenter with inconsistent presystolic murmur

Figure 5.10 illustrates a subtle problem with this energy detector. Although it marks the boundaries of the heartbeats approximately correctly, note the small burst of energy at time 1.4 seconds. This pre-S1 murmur is common in patients with atrial septal defect and ventricular septal defect. The variation in the energy of this sound causes the beat boundary to be moved to the left, just 0.1 second.

The intention of this segmentation algorithm is to pass information on to further processing stages, where the signal is folded over itself and the heartbeats are superimposed. In this case, the first, second, fourth and fifth beat superimpose nicely, but the third beat is shifted, by this variation in the small murmur. For the purposes of this thesis, the segmentation of the 3<sup>rd</sup> heartbeat is marked as incorrect.

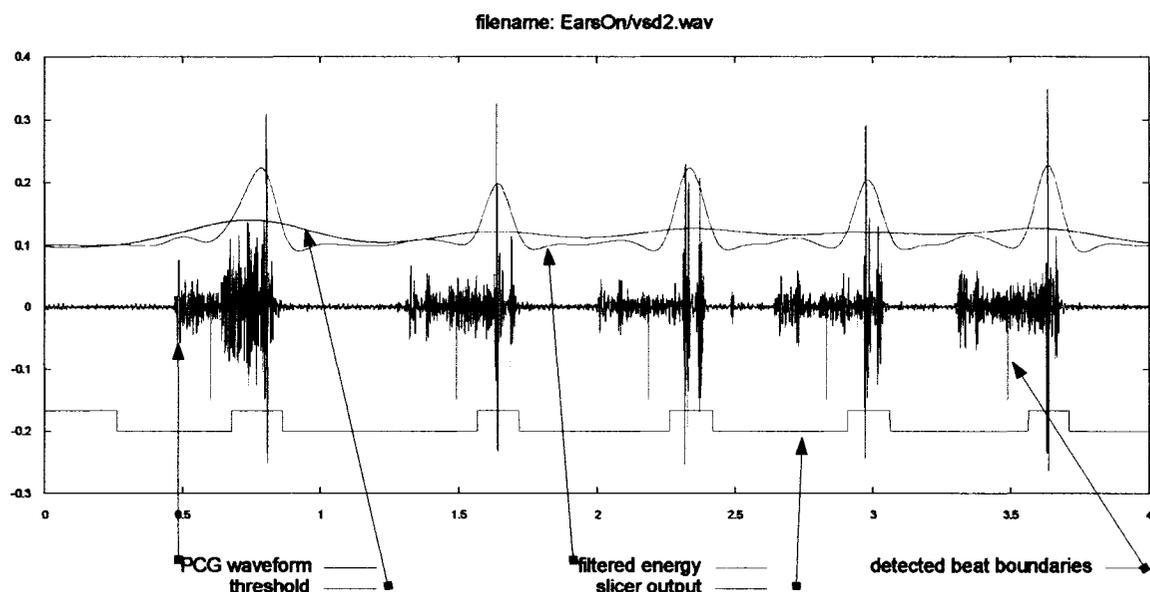


Figure 5.11: Peak energy segmenter with a strong S2 waveform

In Figure 5.11 the waveform has a strong S2 component, which dominates the threshold setting, and masks the S1. The slicer and heuristics yield the correct period, but the phase

is incorrect.

As stated above, this simple energy based beat detector finds the correct heartbeats boundary in 69% of all the beats in the experimental collection. If we restrict ourselves to only the complexity 1 and 2 waveforms, this raises the success rate to 73%. A lot of these errors are simple phase shifts, as illustrated above. One can imagine more sophisticated heuristics to deal with the phase shift, so the experiment was scored again, forgiving simple phase shifts as shown above, anticipating a better heuristic post-processing block. This increased the score for this energy-envelope detector to 85%, which may be considered an upper bound for this kind of detector (with this dataset).

In summary, the *peak energy* segmentation technique is useful for beat counting and detecting missed beats, but not particularly effective in aligning the heartbeats.

## Chapter 6

### Sliding Window Autocorrelating Segmentation

Another possible method for segmentation is to take advantage of the self-similarity of the sounds of the heartbeats.

In order to segment a PCG recording without external synchronization information, one can take advantage of several characteristics of heart sounds:

1. the waveform is periodic, known to lie between 40 and 200 beats per minute (300 to 1500ms period)
2. the waveform has significant beat-to-beat similarity, both in energy, spectrum and waveform

The waveform is periodic (almost cyclostationary, changing slowly with time), with a working range of 40 to 200 beats per minute, except in pathological cases [18]. Further, any sequence of beats will have interbeat times that lie within a working range of 2:1. This allows a search scheme with a limited range, based on information from immediately preceding beats. This is the basis of the rate estimator described in Chapter 4.

In order to align the murmurs with the correct heartbeats, one also needs to determine the start and end of the heartbeats, in a sense, the “phase”. This algorithm attempts to do this by taking advantage of a third fact:

3. the time sequence of events within a heartbeats varies little, compared to the variation

of the inter-beat time; in other words

$$\Delta T_{systole} \ll \Delta T_{beat} \text{ , and } \Delta T_{systole} \ll \Delta T_{diastole} \quad 6.1$$

The next section partially justifies these three assumptions.

### 6.1 Physiology of Heartbeat Timing

Based on the physiology of the heart, the systolic time (and wave shape) is determined by the propagation speed of a nerve impulse through the heart, starting at the SA node, through the delay structures of the Bundle of His and the Perkinje fibers and finally through the ventricular mass. This time is dominated by the propagation velocity through this tissue, and is influenced by temperature, electrolyte and hormone levels, and perfusion, and it is fairly consistent on a short term basis; this thesis assumes that it varies little over the duration of the sample period.

On the other hand, the total cardiac cycle time (and implicitly, the diastolic time) is determined by the repolarization process in the SA node (the pacemaker cells), and its autonomic depolarization. The cells in the SA node are peculiar in that they leak sodium ions into the cell, and oscillate, triggering the heartbeats. The time is first order affected by electrolyte concentrations ( $K^+$ ,  $Na^+$ ,  $Ca^{++}$ ) [33], enzyme concentrations (epinephrine/adrenaline [18]), acetylcholine, glucose levels, temperature [18] and indirectly affected by thoracic pressure<sup>1</sup>. If one examines the curve of intracellular potential with respect to time (Fig 6.1), one can see that the cycle time is sensitive to the slope of the depolarization (phase 4); a small change in slope will cause a large change in

---

<sup>1</sup> There are dozens of factors currently being studied, including methadone, fatty acids, congenital gene deformity and sildenafil (Viagra). This thesis is only concerned with factors that may change quickly, in times on the same scale as the period of the heartbeat.

the delay between beats. From Keating: “relatively small changes in ion current during this phase can have a major impact on action potential duration” [34] Alternately, a change in the trigger potential (shown at -40mV) will have an equally dramatic effect. In fact, the body takes advantage of these characteristics [18]; small quantities of hormones (epinephrine and acetylcholine, for example) can modify the heart rate. Note that neither of these modifiers have a significant affect on the propagation time of the action potential, and therefore little effect on  $T_{systole}$ .

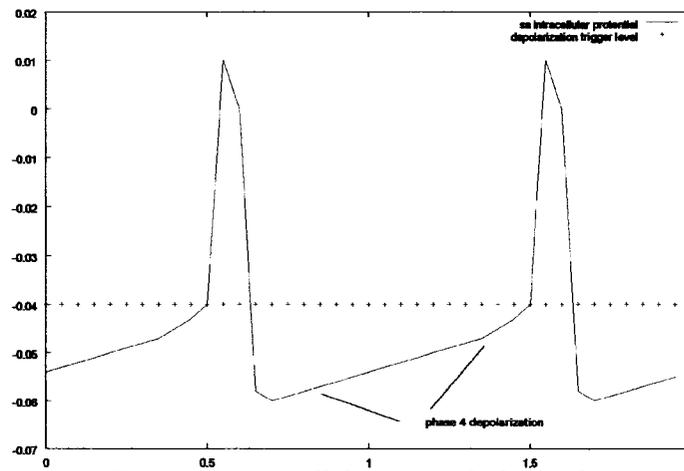


Figure 6.1: intracellular potential of SA node

One can assume then, that

$$\Delta T_{(systole)} \ll \Delta T_{(diastole)} \text{ or } \Delta(T_{S2(n)} - T_{S1(n)}) \ll \Delta(T_{S1(n+1)} - T_{S2(n)}) \quad 6.2$$

There are a few cases which violate the above assumption:

Ectopic beats are a rare condition where a heartbeats is initiated by a cell cluster other than the normal SA node. Normally, the assumption that  $\Delta t_{systole} < \Delta T_{diastole}$  will continue to hold true. However, extreme cases of ectopic beat activity may cause beats which are so close

together that the normal repolarization process has not finished, This will cause the next depolarization phase to be incomplete, causing a modified systole.

Similarly, any fibrillation will cause an irregular depolarization cycle, also increasing the systolic timing variations. Of course, a patient with fibrillation has a serious real time problem, and is not appropriate for the murmur analysis discussed here.

There is a specific disease, AV block, where the propagation through the bundle of His is limited; this case may cause significant systolic timing variations.

Together, these three cases represent a small percentage of patients, and in the first two cases, patients who have urgent medical needs that the stethoscope can easily detect. The rest of his thesis ignores the problems which might arise from these cases.

To investigate the  $\Delta T_{\text{systole}} < \Delta T_{\text{diastole}}$  assumption, the sample corpus was processed using the method of chapter 5, and manually processed to remove extra transitions. The S1-to-S2 time ( $T_{\text{systole}}$ ) and the S2-to-S1 time ( $T_{\text{diastole}}$ ) was averaged. In a large majority of cases, the S1-to-S2 times had a standard deviation of 25ms or less, and the S2-to-S1 timing had a standard deviation of 40ms or less.

This is consistent with the assumption that  $\Delta T_{\text{systole}} < \Delta T_{\text{diastole}}$

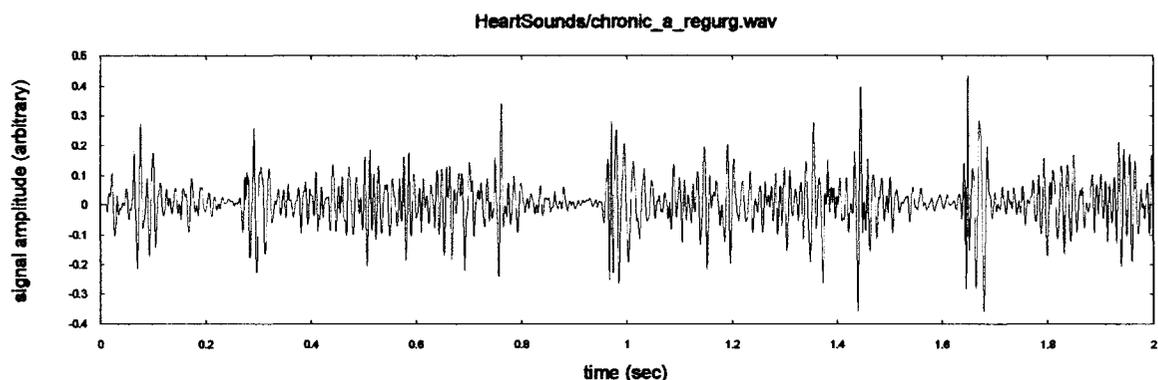
## 6.2 Justification for Energy/Envelope Detection

In signal processing work, signal self-similarity is usually detected by autocorrelation.

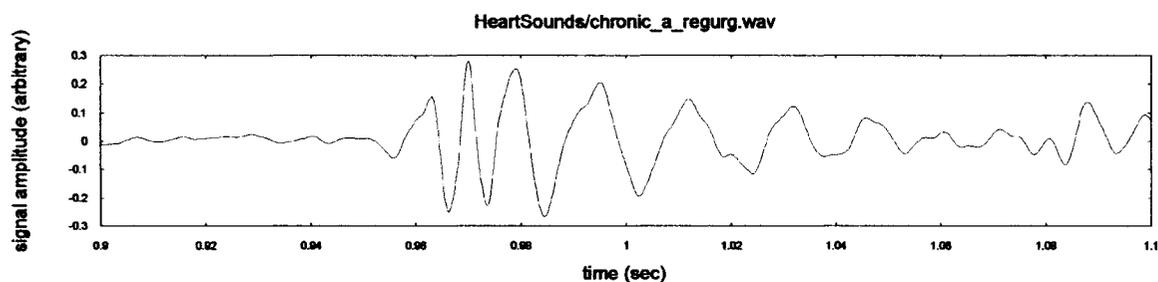
If the effective filter formed by the acoustic path through the body was stationary, then the resulting waveforms could be modeled as the filtered response to a string of impulses (one per heartbeat). Each heartbeats waveform would be identical in shape with respect to time

(cyclostationary) and should autocorrelate well.

In fact, the heart sounds that are detected at the surface of the skin have traveled a heterogeneous acoustic path, and have been subject to delay distortion, echo and attenuation. The torso is in constant motion, and the sounds themselves are percussive. So the signal source is not stationary, and the exact path is not stationary, and it appears that the shape of each heartbeats is not the same. Figures 6.2, 6.3, & 6.4 show a typical pair of heartbeats. Note the details of the waveforms, that while they are similar in shape, they are not similar enough to correlate; Figure 6.5 shows the autocorrelation of the first few seconds of the waveform.



*Figure 6.2: Two complete beats of a patient with chronic atrial regurgitation*



*Figure 6.3: Single heartbeat, details at time=0.9 seconds*

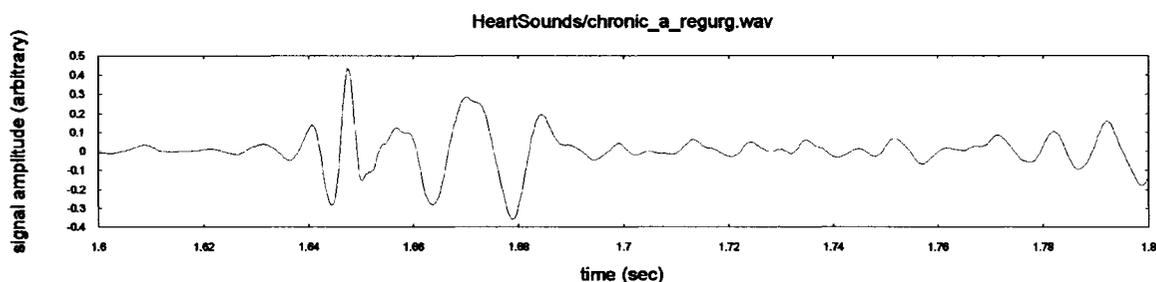


Figure 6.4: Single heartbeat, details at time=1.6 seconds

Note the difference with respect to Figure 6.3

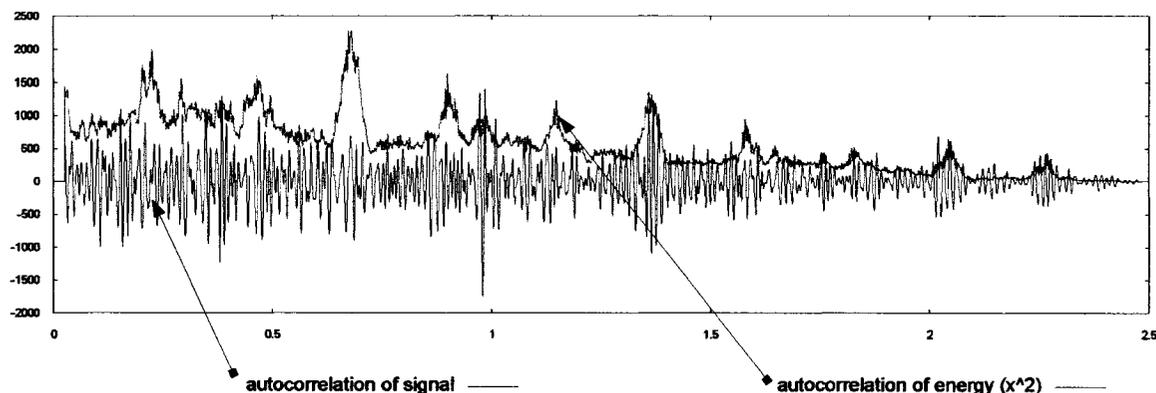


Figure 6.5: Autocorrelation of the waveform in Figure 6.1

The beat to beat variation in the detail reduces the effectiveness of direct correlation, and this was observed on the experimental data during early experiments.

A hypothesis is that the *energy* of the waveform (or perhaps, its *envelope*) provides a better vehicle for correlation, rather than the signal itself. Three possible approaches were considered: the signal energy ( $x^2$ ), the magnitude ( $|x|$ ) and the magnitude of the Hilbert transform. Figure 6.5 shows the autocorrelation of the energy of the signal shown in Figure 6.2. Based on early experimental feedback, this thesis examines only the magnitude version. See also 3.3.2 above for a discussion of magnitude vs. energy autocorrelations.

### 6.3 Waveform Correlation

To detect the heart cycles, one can take advantage of the similarity between cycles and the

fact that the similarity dissipates when the cycle boundary is exceeded. For example, in Figure 6.6, the rate estimation algorithm tells us that the period is about 800-1000ms. If one takes the cross-correlation of the signal (or rather, its magnitude) from 6.7 to 7.3 s against the signal from 7.5 to 8.1 s, the cross-correlation will be fairly significant. Sliding along the input stream, another cross-correlation taken from 7.0 to 7.6 s against the signal from 7.8 to 8.4 s will be weaker, because the S2-to-S1 timing is different.

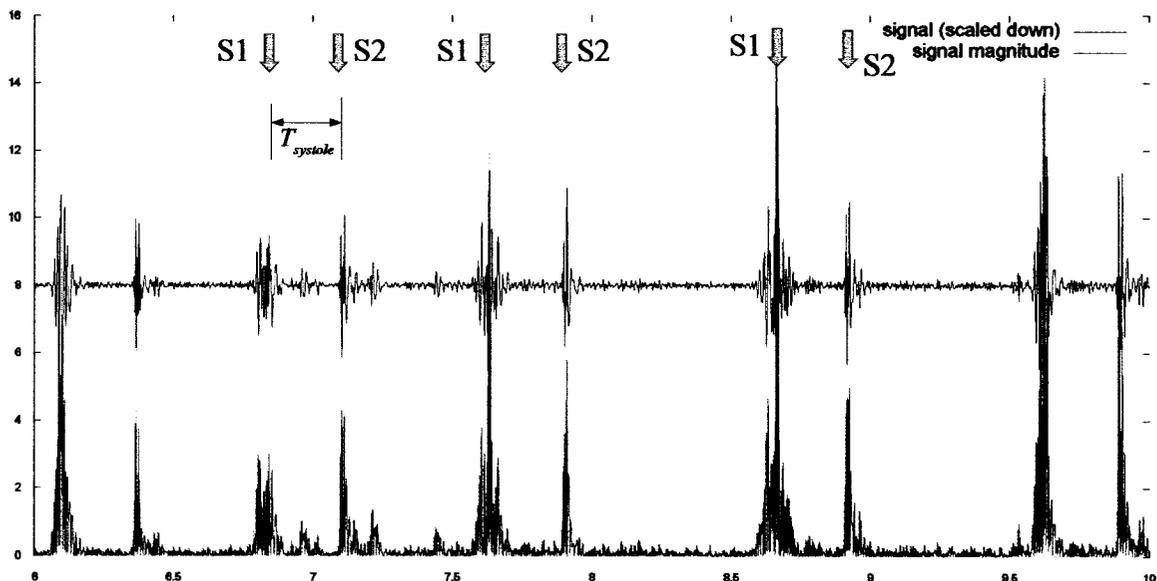


Figure 6.6: Heart sound, showing local variations in beat period

While the search window is in the zone between 6.5-7.1 s and 6.9-7.4 s, the correlation peak (the time shift which give the best match) will be a consistent 0.8 s. As the search window moves forward into the range of 7.0-7.6 up to 7.3-7.9 s, the correlation peak is uncertain. Then as the search window moves up past 7.4-8.0 s, the correlation peak will again stabilize at 1.0 seconds, the time to the next match.

This observation is used to build the *sliding window autocorrelation* algorithm.

## 6.4 Algorithm Summary

Here is an overview of the algorithm, given a quantized sequence of sample PCG data.

- phase 0
  - remove any DC component from the signal; normalize the signal to an RMS mean of 1.0, and then use the absolute value
- phase 1
  - estimate the heart rate, using the algorithm described in chapter 4 (*rate estimation*)
- phase 2
  - use 70% of the average beat length as a “template length”
  - use 1.8 times the average beat length as the “search length”
  - mark the entire sample waveform at steps of 40ms (the “search step”)
  - for each search step position, cross-correlate a short section of the waveform (the “template length”) with a longer section (the “search length”); the actual cross-correlation is the “similarity function” described in the 3.3.1.
  - save the result of each of these similarity tests for further processing
- phase 3
  - for each of the saved similarity tests, measure the peak, and record the time offset to next beat:  $T_{\text{onb}}$
  - scan these  $T_{\text{onb}}$  array, and take the end of the longest steady run as the the start of a

heartbeat (the start of systole); use the following beat-period as the heartbeats template

- phase 4
  - using the heartbeats template as a prototype, scan the entire heartbeats, left and right, searching for matches using the similarity test; each individual search is constrained to lie within 0.6 and 1.8 times the average period; take the peak of the similarity test as the start of the heartbeat

The output of phase 4 is a set of markers for the start-of-heart-beat.

In the next few subsections, the algorithm is described in greater detail. Design choices are explained in 6.5 below.

#### **6.4.1 Phase 0: Normalize**

The corpus of sound samples varies in amplitude. Each waveform file of samples  $w_{raw}$  is normalized to produce  $w_n$

$$w_{ac}(n) = w_{raw(n)} - \overline{w_{raw(n)}} = w_{raw(n)} - \frac{1}{N} \sum_{n=0}^{N-1} w_{raw}(n)$$

$$w_n = \frac{w_{ac}(n)}{\text{RMS}(w_{ac}(n))} = \frac{w_{ac}(n)}{\sqrt{\frac{1}{N} \sum_{n=0}^{N-1} w_{ac}^2(n)}} \quad 6.3$$

The rest of the algorithm uses  $|w_n|$

#### **6.4.2 Phase 1: Heart Rate Estimation**

This process is fully discussed in chapter 4.

### 6.4.3 Phase 2: Sliding Correlation

Once a beat period estimate is acquired, the magnitude waveform is processed with a sliding window autocorrelation. This results in a correlation per step time, where multiple consecutive correlations are often similar, reflecting the cyclostationary nature of the dataset.

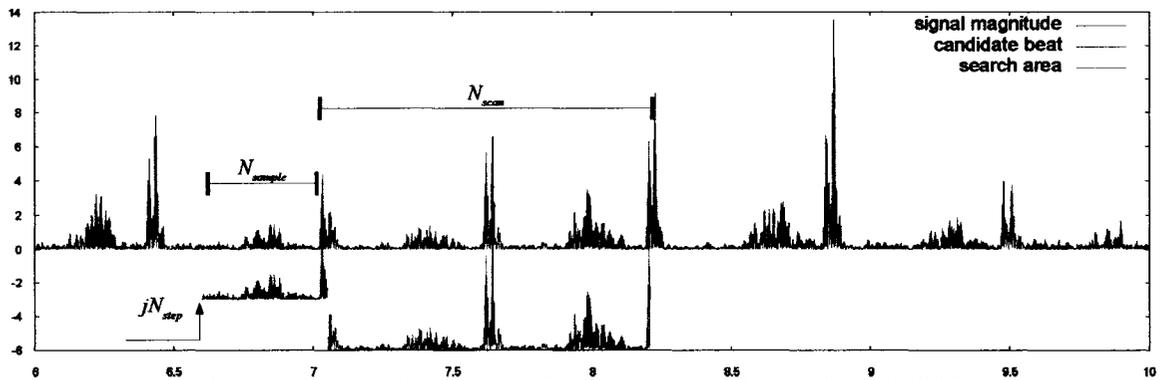


Figure 6.7: Heartbeat magnitude, showing a template and search field

For each step  $j$ , a short sample of the waveform is used as a “template”. The sample length  $N_{sample}$  is chosen to be 70% of the estimated heart sound period  $N_{period}$ , large enough to capture the bulk of the signal “energy” of each heart cycle, and small enough to allow several consecutive measurements within a heart cycle.

At each step point, the “template” waveform is similarity cross-correlated (see 3.3.1 above) with the original magnitude waveform, within a search distance of  $N_{scan} = 1.8$  times  $N_{period}$ . The result is a function of a new variable  $i$ , and is windowed with a taper function that emphasizes peaks near the average period. (see Figure 6.7)

These steps are repeated by incrementing  $j$ , moving the entire process ahead  $N_{step}$ , which is the time resolution of the system.  $j$  is typically chosen to be 20-50ms.

The collection of these cross-correlations is

$$X_{j,i} = \sum_{m=0}^{N_{scan}-1} W_1(|w_{jN_{step}+m}|) \circ W_2(|w_{jN_{step}+m+i}|) Q_m \quad \text{where } 0 \leq i < N_{scan} - N_{sample},$$

$$0 \leq i < N_{scan}$$

$$0 \leq jN_{step} < N - N_{scan}$$

$W_1()$  is a rect window of size  $N_{sample}$

$W_2()$  is a rect window of size  $N_{scan}$

$Q_m = m e^{-m/N_{period}}$ , a taper window of size  $N_{sample}$

$\circ$  is the similarity cross correlation

PA/stills.wav

6.4

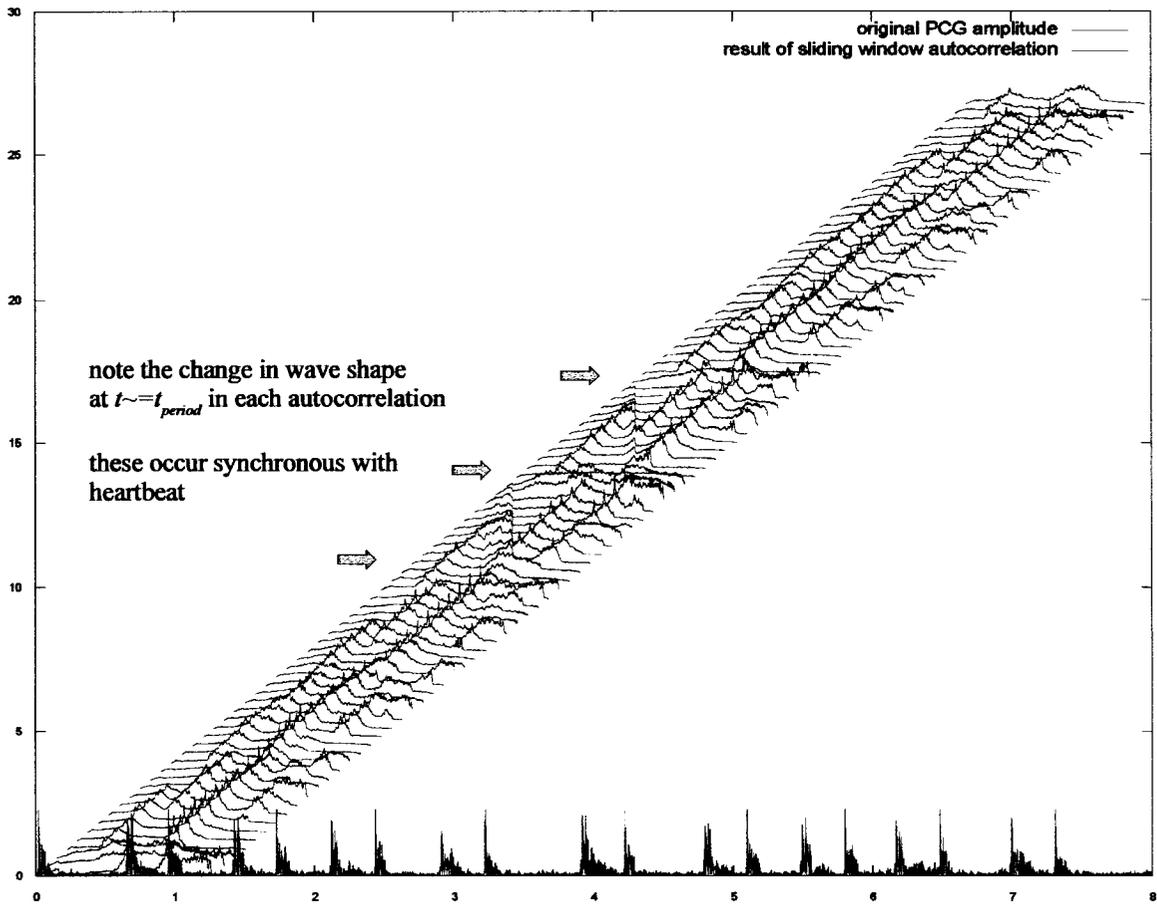


Figure 6.8: Sliding window correlations

Figure 6.8 shows the  $X_{j,i}$  correlations plotted in a step plot for visibility. Each brown line is a different  $j$ , and the horizontal sweep of each brown line is  $i$ . Note the change in shape of the brown line as the scan passes from one heartbeat to another.

### 6.4.4 Phase 3: Peak Detection

The autocorrelation waveforms indicate the time offset when the waveform becomes similar. The shapes are remarkably similar as long as the “template” sample spans data from a single heartbeat. This reflects the similarity of one heartbeats to the next.

As  $j$  increases, the “template” sample moves to span the end of one heartbeats, and the beginning of the next, the similarity between the correlation waveform fades or changes.

The peak of each  $X_{j,i}$  autocorrelation is taken as the most likely time offset to the next heartbeats<sup>1</sup>. The collection of these peaks forms a data sequence which characterizes the original heartbeat. Long runs of repeating data in this sequence indicate a zone of beat-to-beat similarity. The extremities of these runs can be taken at the border between beats.

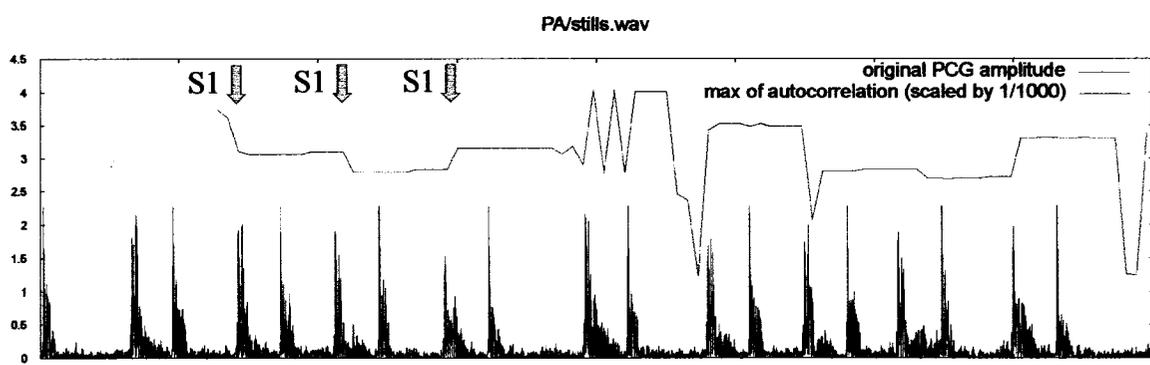


Figure 6.9: Sliding window autocorrelation maximums overlaid in the original signal magnitudes

In Figure 6.9, note how the graph of maximums stays flat during most of the heartbeat, and changes at systole. [The units of the green line in this figure is time-offset.]

<sup>1</sup> Note in the graph how the shape of each  $X_{j,i}$  is fairly constant as the sample zone passes through systole. The algorithm described here simply takes the peak of each autocorrelation  $X_{j,i}$ , extracting the time-offset to the next heartbeat. An alternate design is to test the similarity of each  $X_{j,i}$  in the  $j$  dimension. See the discussion in 6.5

In the actual experiment, the search for the runs within these data is slightly modified. The actual algorithm is as follows:

- the data sequence is scanned from left to right, searching for consecutive samples which lie within 0.2% of the first sample of the run; runs which exceed 80% of the length of the  $N_{beat}$  are ignored, as they may represent multiple beats
- the longest run is taken as the best run; in the case of multiple runs of the same length, the one with the smallest variation is taken as best
- starting at the end of the best run, the algorithm steps back  $N_{beat}/4$ ; the algorithm scans the next  $N_{beat}$  samples of the signal magnitude for the first sample that exceeds 5% of the maximum; from here, the algorithm steps back  $N_{beat}/7$
- this location is taken as the start of the “best” heartbeat

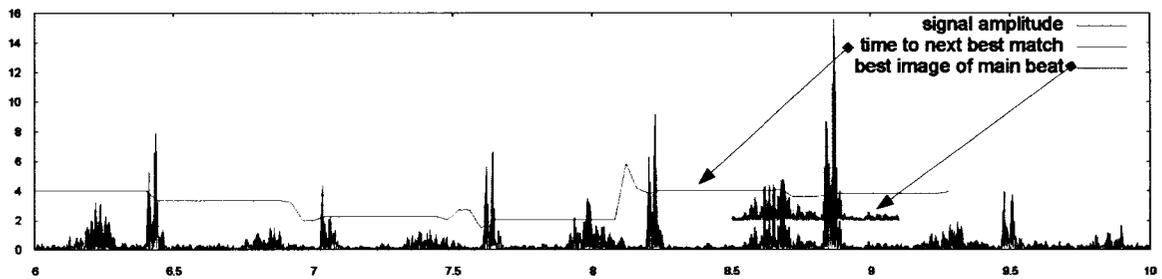


Figure 6.10: sliding window autocorrelation with the best beat identified

#### 6.4.5 Phase 4: Left and Right Beat Search

Having found a good sample heartbeats, the algorithm searches to the right, using the similarity cross-correlation test to find the best match between 60% and 180% of  $N_{beat}$ . The peak of the function is taken as the distance to the next heartbeats. As the test sweeps

repeatedly to the right, continue to test against the best heartbeat located in 6.4.4 above.

Having located the heartbeats to the right, sweep again to the left, testing the waveform from  $t-1.5N_{beat}$  to  $t$ . For each sweep step, use the similarity cross-correlation to compare the swept data to the best heartbeat located in 6.4.4 above.

Both of these sweeps terminate when there are no more data.

This is the algorithm that was coded in Matlab/Octave and run against the experimental database, as described in chapter 3.

## 6.5 Details and Justification

There are many aspects of this algorithm which may be varied, and some of these variations were explored during the development of the algorithm.

In the Phase 1 processing there are many degrees of freedom in the rate estimator, including the choice of signal manipulation:

- signal magnitude  $|x|$
- signal energy  $x^2$
- signal Hilbert magnitude  $|\text{Hilbert}(x)|$

This implementation uses the signal magnitude. During the pre-experimental phase of this thesis, all three forms of the signal were used in the autocorrelations, with the *magnitude* and *Hilbert magnitude* forms yielding better performance over the *energy* form. Only the first two of these are reported in this work.

As well, the time constants and shape of the window function may be varied. If there is

any external information available, it may be appropriate to bias the shape of window function and improve the rate estimation (for example, this PCG is from a child).

Some other variations on the design of the *rate estimator* are also mentioned in section 7.2.1 of this thesis as future work.

In Phase 2, the implementation described here runs a sliding autocorrelation of 70% of  $N_{period}$  against 180% of  $N_{period}$  to build the correlation curves shown in Figure 6.8. The former ratio was chosen to be large enough to capture the bulk of the signal energy of each heart cycle, and small enough to allow several consecutive measurements within a heart cycle. Testing of both 50% and 90% of  $N_{period}$  yielded substantially worse scores, and this variation was not pursued. The 180% of  $N_{period}$  factor is quite soft, and may be varied upward to 300% without significant change in the results. Any further increases in the scan range made very small differences, and only seemed to invite interference from subsequent beats; this was not investigated further. The lower limit is set by the range of beat variations expected in the input data.

In Phase 3, the peaks of the autocorrelations form lines that change at-or-near the beginning of systole. Testing with the experimental data, the transition lines were sometimes just before, and sometimes just after the visual appearance of S1 (Figure 6.9 & 6.10). However, the method of searching for the longest run also seems to search for the heartbeat with the best signal to noise ratio in the waveform, and a 2<sup>nd</sup> stage search (involving backing off and searching for a rise in signal magnitude) is reasonable.

This implementation simply takes the peak of each  $M_{j,i}$  as the parameter to examine to locate systole. The correlations are not filtered in delta-time, so that a small instantaneous

peak will be detected as the best time offset; perhaps a filter on the input (or output) of the autocorrelation may improve the stability and acuity of the detector.

In examining Figure 6.8, one can see the change of shape in the small autocorrelations as they sweep across the PCG. Our first approach to processing these samples was to perform a similarity test of  $M_{j,i}$  against  $M_{j+1,i}$ . This approach did indeed result in a measure of similarity, but we quickly noticed that one could estimate the similarity of these curves by simply taking noting the time-shift location of the largest peak. Because of its simplicity, the phase 3 process reported here simply uses the latter approach.

In Phase 4, this implementation of the sweeping beat detection simply compares the chosen template against the entire PCG. This is probably less than ideal, because heartbeats change slowly over time, with respiration, exhaustion, posture, etc. Perhaps a better choice would be to use the shape of beat  $n$  to search for beat  $n+1$ . Of course, one would need to qualify the use of beat  $n$  as a template, because it may introduce noise or phase slippage that may be inappropriate; some sort of averaging will be needed.

One of the targets of this work is to locate the beat boundaries so that the beats can be superimposed, to detect murmur similarities or perform feature extraction. The Phase 4 beat search may be modified so that the similarity correlation function output is used as a measure of confidence for each heartbeat; individual beats with a low confidence may be removed from any aggregation.

In summary, there are several degrees of freedom in the design of this algorithm, any of which may improve it. This thesis reports only a few of the possible combinations.

## 6.6 Experimental Results

The *sliding window autocorrelation* algorithm (SWA) is a significant improvement over the simple energy peak detector (Table 5.1).

<i>algorithm</i>	<i>all samples</i> 2709 beats	<i>only simple sounds (type 1)</i> 1948 beats	<i>only moderate sounds (type 2)</i> 589 beats	<i>only complex sounds (type 3)</i> 172 beats
sliding window autocorrelation segmenter	2253 = 83.2%	1731 = 88.9%	493 = 83.7%	29 = 16.9%
SWA, using Hilbert magnitude	2201 = 81.3%	1669 = 85.7%	474 = 80.5%	45 = 26.2%

Table 6.1: Performance of the sliding window autocorrelation segmenter

The experiments were run once using the algorithm described in 6.4, and then a second time, using the Hilbert magnitude for the Phase 1 rate estimation. The *sliding window autocorrelation* algorithm performed much better than the *peak energy* algorithm, properly segmenting 81-83% of all the heartbeats in the experimental database. Even through the Hilbert magnitude *rate estimator* gives a better rate estimate, once it is run through the SWA algorithm, the overall system performed slightly worse.

The *rate estimation* first stage has a limited success rate (91-93%), which puts an upper bound on the success opportunities for both the *peak energy* and *sliding window autocorrelation* algorithms.

The SWA algorithm recognizes heartbeats embedded in several kinds of noise. However, the success of this mechanism on any particular PCG recording depends on locating a single good beat to take as the search template (Phase 4 above); if this template beat has

significant constant-level noise, the effectiveness of beat detection is reduced.

Beats are also well detected in the presence of impulse noise, such as motion artifacts and room noises.

The SWA algorithm is a significant improvement over the *peak energy* algorithm, especially for diastolic murmurs, such as the murmurs of aortic regurgitation. Figures 6.11 and 6.12 shows a segment of a sample from the experimental database, marked according to the SWA method and peak-energy methods. The SWA algorithm picks up the small systolic sound, and associates it with the correct heart beat, whereas the peak-energy algorithm misses the systolic sound and only detects the large S2 (diastolic) sound.

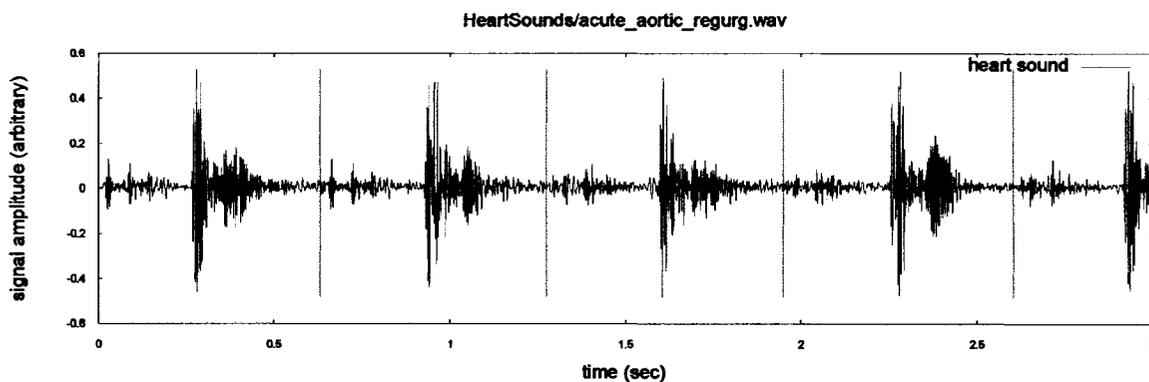


Figure 6.11: Diastolic murmur of aortic regurgitation, segmented and marked correctly by the SWA algorithm.

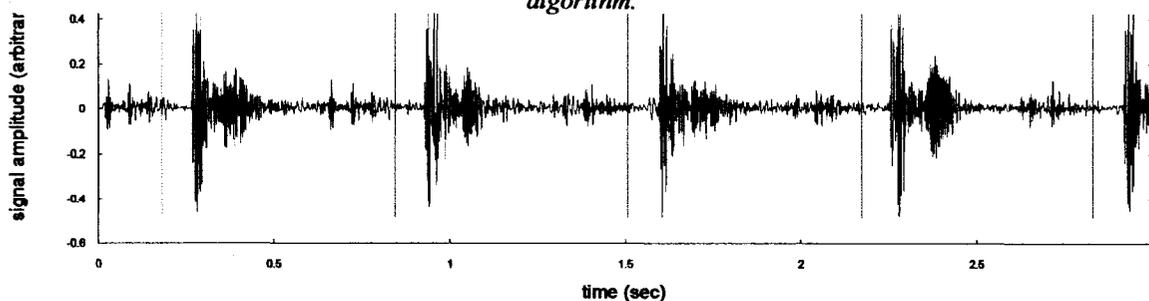


Figure 6.12: Diastolic murmur of aortic regurgitation, segmented and marked incorrectly by the peak energy algorithm (chapter 5)

The SWA algorithm also excels at determining heartbeat boundaries in large holodiastolic murmurs, such as those of aortic regurgitation. Figure 6.13 illustrates a properly segmented section of the sound of aortic regurgitation, despite the loud and inconsistent diastolic murmur.

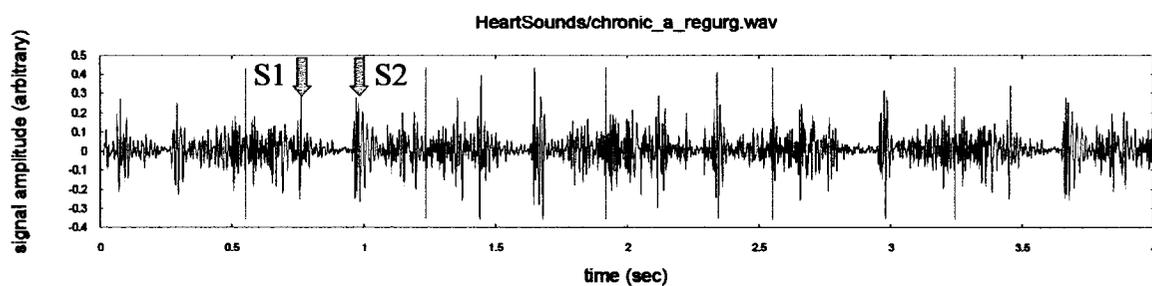


Figure 6.13: Diastolic murmur of aortic regurgitation, properly segmented

The SWA algorithm is also tolerant of many types of intermittent noise, since the autocorrelation alignment works on the whole of the template waveform. Figure 6.14 & 6.15 show how a moderate amount of noise (in this case, breathing) can shift the results of the *peak energy* segmenter, but has little effect on the SWA algorithm segmentation.

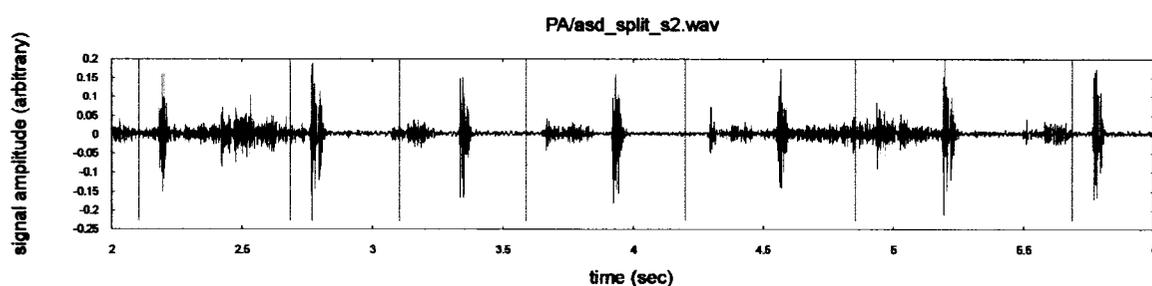


Figure 6.14: Heart sound including breathing noises, segmented incorrectly by the peak energy algorithm

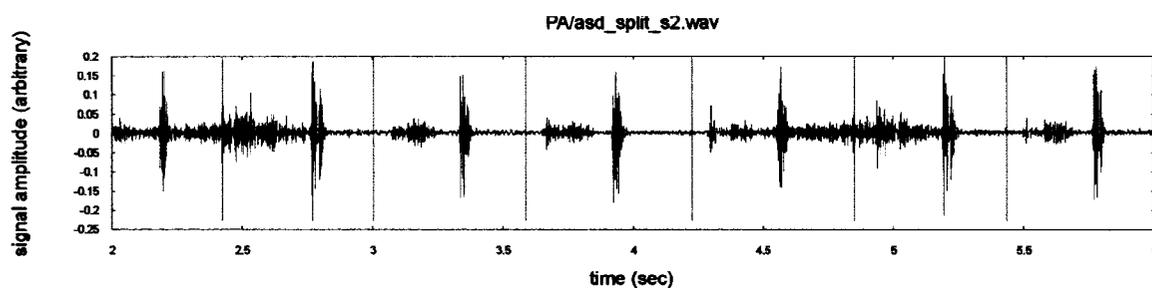


Figure 6.15: Heart sound including breathing noises, segmented correctly by the sliding window autocorrelation algorithm

The SWA algorithm fails for several reasons:

- the phase 0 rate estimate is incorrect
- samples where the period of the heart rate changed suddenly by  $>50\%$ ; this is common for missed beats and tachycardia
- samples where the period of the heart beat is unusually constant, varying by less than  $2\%$  over the entire sample

Above and beyond these explicable failures, there are still segmentation failures that arise due to the algorithm extracting the incorrect location for systole. Figure 6.17 shows a segment of a Still's murmur, marked incorrectly by the SWA algorithm; a slight modification<sup>1</sup> to the phase 3 process corrects the detection of S1, shown in Figure 6.16. This suggests that the SWA algorithm is sensitive to minor changes in its design, and further work needs to be done (see section 7.1).

<sup>1</sup> The detection of the “longest run” was changed from an  $80\%$  of  $N_{beat}$  limit to a  $100\%$  limit.

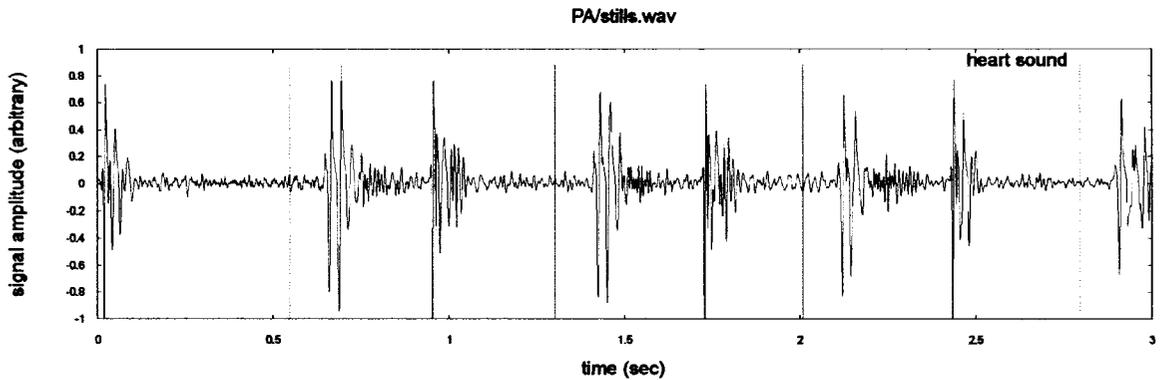


Figure 6.16: Stills murmur, marked correctly by a modified SWA algorithm

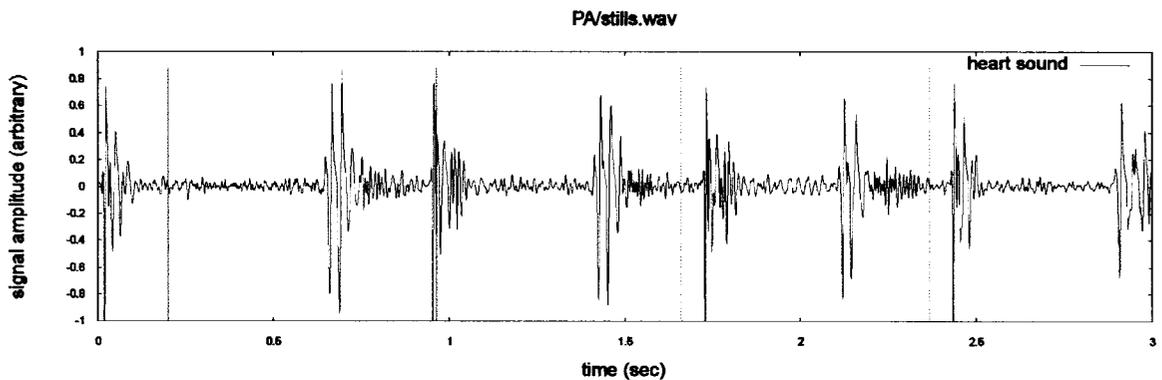


Figure 6.17: Stills murmur, marked incorrectly by the SWA segmentation algorithm

Another type of failure occurs when the beat sound includes a constant noise sound, with a very quiet and long diastolic time. The algorithm can sometimes lock into the quiet period and incorrectly match the beat sound to the quiet period; the “similarity” detector ignores large differences in amplitude (by design). This leads to incorrect beat detection. Figure 6.18 shows incorrect segmentation at times 6.2, 7.5 and 8.3 s. Note how the heart sound itself is very squared up, much like the sliding `rect()` window that is intrinsic to the similarity correlation.

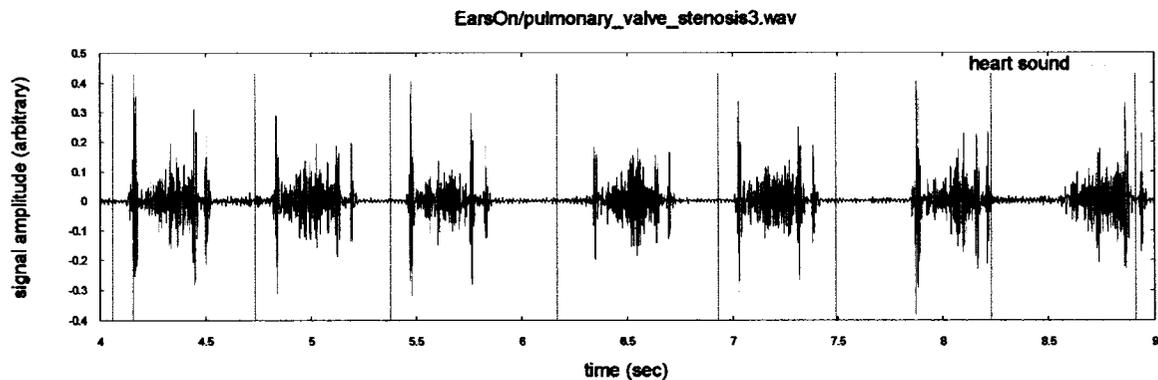


Figure 6.18: Pulmonary valve stenosis, marked incorrectly by the SWA algorithm

In summary, the factor which dominates the failure mechanisms is an incorrect rate estimate in the phase 1 processing; any improvement in that subsystem should improve the performance of the entire algorithm.

### 6.7 Factors Affecting Segmentation

This algorithm is based principally on a varying time between heartbeats, or

$$\Delta T_{diastole} \gg \Delta T_{systole} \quad 6.5$$

If the heartbeats are generated by a patient with an electronic pacemaker which is in “forced mode”, the  $T_{diastole}$  will not vary, and this algorithm will not function. For these cases, the *peak energy* algorithm of Chapter 5 may work, since it operates on a completely different principle.

As outlined in 6.1, there are several health conditions which may interfere with the operation of this algorithm, including

- ectopic beats
- tachycardia/fibrillation

- AV block

The algorithm also performs poorly in presence of a constant dominant interference. The Phase 3 beat detector needs two fairly clean beats in a row to compare, and failing that, will not lock into a pattern. Several of the incorrect segmentations in the experiment were due to a dominant and constant interference. This also makes the SWA algorithm inappropriate for extracting fetal heartbeats in the presence of the maternal heart sounds.

The length of the PCG recording has some effect on the performance. The PCG should be long enough to have at least two consecutive heartbeats for the phase 3 process to detect. In practice, a string of 4 complete heartbeats is sufficient, but more are required if there is noise in the recording. On the other hand, the PCG should not be so long that the shape of the heartbeat changes significantly over the course of the recording; the phase 4 beat detection is based on self similarity against the template heartbeat located in phase 3, so a significant change in the shape of the heartbeat will reduce accuracy of the alignment.

## **6.8 Conclusion**

The SWA segmentation system is a clear improvement over the *peak energy* segmenter. The design has several untested degrees of freedom, and should merge well with the work of other researchers.

## Chapter 7

# Conclusions

The *heart rate estimator* described here is simple to implement and performs well on most PCG signals.

The *sliding window autocorrelation segmentation* method introduces a new method for segmentation. It adapts well to both short recordings (as few as 4 heartbeats), low signal to noise recordings and a wide range of heart rates. It yields synchronous segmentation information to allow alignment of beats which may be required for an ongoing process. It compares very well against a simple *peak energy* detector, the basis for many segmentation systems reported in other works.

The SWA algorithm is capable of extracting segmentation from PCG data which are difficult for the *peak energy* algorithms, specifically the recordings of loud holodiastolic murmurs.

The *sliding-window autocorrelation* technique is a robust technique, based on physiological principals. It is practical to realize, with affordable access to high speed DSP technology. It is extendable, with several dimensions identified above. It should also work well as a complement to techniques proposed by other researchers.

A summary of this thesis was presented in [35]

## 7.1 Future Work

The results reported here are from a specific combination of design decisions: a single pass rate estimator, magnitude processing for the autocorrelations, a fixed template for the beat searching (chapters 4 and 6). Future work may investigate various degrees of freedom, some of them listed here.

### 7.1.1 Rate Estimation

The performance of the rate estimator is sufficient for the purpose of this thesis, yielding performance that is in the same range as previously reported estimators. It is open to several improvements. All of the analysis in this chapter was performed on the sound file as a whole. The failures which occur due to large beat period variations may be addressed by performing a second pass, running a sliding window across the sound file, covering a small region (about 4 beat periods); if there is a large specific peak at twice the 1<sup>st</sup>-pass period estimate, this may be a better overall rate estimate.

Similarly, a sliding window scan may improve the performance on files with alternate-beat-emphasis, because the beats are similar in shape, and the magnitude-compensating similarity function (3.3.1) may forgive the emphasis effect, enough to yield a better rate estimate.

There will always be a peak in the autocorrelation at  $t=T_{systole}$ ; in fact, the *sliding window autocorrelation algorithm* in chapter 6 capitalizes on that fact (see 6.1). If the heart rate is fairly steady, the largest peak of the autocorrelation is a single dominant peak that corresponds to the heart rate,  $T_{beat}$ . However, if the heart rate varies significantly, the peak may in fact be composed of several small peaks, one for each beat period. The peak at

$T_{systole}$  may be larger than any of the small ones at  $T_{beat}$ . Future work may examine the use of a smoothing filter to sum together several of these small peaks, in the name of building a peak at  $T_{beat}$  large enough to be detected.

### **7.1.2 Segmentation**

Many investigators in this field are exploring time-frequency decomposition of PCG's as a way to reduce the effect of noise on segmentation placement. Typically, the segmenter will use one dimension of the output of a decomposition process and feed it a threshold detector [9][7].

One may consider replacing the threshold detector with the beat detector described in phase 2, 3 & 4 of the SWA segmenter (section 6.4).

Restated, the block diagram shown in Figure 2.10 shows the *segmentation* process preceding the *time-frequency analysis* process. The work in this thesis was performed entirely in this configuration. One would expect that the order of these blocks could be reversed, with the *time-frequency analysis* block feeding sub-band streams to the SWA segmenter.

The advantages of the SWA algorithm should continue to stand:

- it can detect very small S1 signals, based solely on the similarity, and not on the energy/magnitude
- it can properly segment holodiastolic murmurs
- the segmentation markers are synchronous with the beat information

In this way, the work of other investigators is complementary to the work of this thesis, and there are opportunities to explore the use of the SWA algorithm within their work.

The database used in this thesis is dominated by sounds from patients with heart disease. Now that the SWA segmenter has been shown to outperform the *peak energy* segmenter with this data, it would be appropriate to measure its operation against heart sounds which include a balanced representation of normals.

This thesis only reports the operation of the SWA algorithm using the sound files in their original form. Although some of the sounds included incidental noise, there was no attempt to quantify the noise immunity of the algorithm. It would be appropriate to retest this algorithm with the addition of uncorrelated noise, or perhaps with samples of heart sounds taken from non-standard locations on the body.

## References

- [1] K. Isselbacher, *Principles of Internal Medicine*, McGraw-Hill Inc., 1994
- [2] M. Swartz, *Textbook of Physical Diagnosis, History and Examination*, W.B. Saunders Comp., 1994
- [3] I.R. Hanna, and M.E. Silverman, "A History of Cardiac Auscultation and Some of Its Contributors", in *American Journal of Cardiology*, Aug 2002, pp. 259-267
- [4] S. Mangione, and L.Z. Nieman, "Cardiac auscultation skills of internal medicine and family practice trainees; a comparison of diagnostic proficiency", in *Journal of the American Medical Association*, 1997, pp. 717-722
- [5] P.R.A. Gaskin, S. Owens, N. Talner, S.P. Sanders, and J.S. Li, "Clinical Auscultation Skills in Pediatric Residents", in *Pediatrics*, June 2000, pp. 1184-1187
- [6] H. Liang, S. Lukkarinin, and I. Hartimo, "Heart Sound Segmentation Algorithm Based on Heart Sound Envelopogram", in *Computers in Cardiology*, Sept 1997, pp. 105-108
- [7] S. Omran, and M. Tayel, "A Heart Sound Segmentation and Feature Extraction Algorithm Using Wavelets", in *Circuits and Systems, Proc. 48th Symposium 2003*
- [8] G. Saha, and P. Kumar, "An Efficient Heart Sound Segmentation Algorithm for Cardiac Diseases", in *Proc. IEEE India Conference*, Dec 2004, pp. 344-348
- [9] M.N. Kurnaz, and T. Olmez, "Determination of Features for Heart Sounds by Using Wavelet Transforms", in *15th IEEE Symp. on Computer-Based Medical Systems 2002*
- [10] A. Haghghi-Mood, and J.N. Torry, "A Sub-Band Energy Tracking Algorithm for Heart Sound Segmentation", in *Computers in Cardiology*, Sept 1995, pp. 501-504
- [11] A.D. Ricke, R.J. Povinelli, and M.T. Johnson, "Automatic Segmentation of Heart Sound Signals Using Hidden Markov Models", in *Computers in Cardiology*, Sept 2005, pp. 953-956
- [12] C.N. Gupta, R. Palaniappan, S. Rajan, S. Swarminathan, and S.M. Krishnan, "Segmentation and Classification of Heart Sounds", in *Canadian Conf. on Electrical and Computer Engineering*, May 2005, pp. 1678-1681
- [13] S. Rajan, E. Budd, M. Stevenson, and R. Doraiswami, "Unsupervised and Uncued Segmentation of the Fundamental Heart Sounds in Phonocardiograms Using a Time-Scale Representation", in *Engineering in Medicine and Biology Conf.*, 2006, pp. n/a
- [14] W.R. Thompson, C.S. Hayek, C. Tuchinda, J.K. Telford, and J.S. Lombardo, "Automated Cardiac Auscultation for Detection of Pathologic Heart Murmurs", in *Pediatric Cardiology*, 2001, pp. 373-379
- [15] T.R. Reed, N.E. Reed, and P. Fritzson, "Heart Sound Analysis for Symptom Detection and Computer-aided Diagnosis", in *Simulation Modelling Practice and Theory*, 2004, pp. 129-146

- [16] G. Fragasso, M. De Benedictis, M. Palloshi, A. Moltrasio, M. Cappelletti, A. Carlino, A. Marchisi, M. Pala, O. Alfieri, and A. Margonato, "Validation of Heart and Lung Teleauscultation on an Internet-Based System", in *American Journal of Cardiology*, Nov 2003, pp. 1138-1139
- [17] W.W. Myint, and B. Dillard, "An Electronic Stethoscope with Diagnostic Capability", in *Proc. 33rd SW Symposium on System Theory*, March 2001, pp. 133-137
- [18] E.E. Selkurt et al, *Physiology*, Little, Brown, 1971
- [19] R.M. Berne, and M.N. Levy, *Cardiovascular Physiology*, C.V. Mosby Comp., 1981
- [20] R. Rushmer, *Cardiovascular Dynamics*, W.B. Saunders Comp, 1976
- [21] N.J. Mehta, and I.A. Khan, "Third heart sound: genesis and clinical importance", in *International Journal of Cardiology*, 2004, pp. 183-186
- [22] L.B. Dahl, P. Hasvold, E. Arild, and T. Hasvold, "Heart Murmurs Recorded by a Sensor Based Electronic Stethoscope and Emailed for Remote Assessment", in *Archives of Disease in Childhood*, 2002, pp. 297-301
- [23] J.C. Wood, and D.T. Barry, "Time-frequency analysis of the first heart sound", in *Eng. in Med. and Bio. Magazine*, March 1995, pp. 144-151
- [24] Z. Zhidong, Z. Zhijin, and C. Yuquan, "Time frequency analysis of heart sound based on HHT", in *Int. Conf. on Communications, Circuits and Systems 2005*
- [25] H. Liang, S. Lukkarinen, and I. Hartimo, "A Boundary Modification Method for Heart Sound Segmentation Analysis", in *Computers in Cardiology*, Sept 1998, pp. 593-595
- [26] H. Liang, S. Lukkarinen, and I. Hartimo, "A Heart Sound Segmentation Algorithm Using Wavelet Decomposition and Reconstruction", in *Proc. of 19th IEEE Engineering in Medicine and Biology 1997*
- [27] P. Carvalho, P. Gil, J. Henriques, L. Eugenio, and M. Antunes, "Low Complexity Algorithm for Heart Sound Segmentation using Variable Fractal Dimension", in *IEEE International Workshop on Intelligent Signal Processing*, Sept 2005, pp. 194-199
- [28] A. Jimenez, M.R. Ortiz, M.A. Pena, S. Charleston, A.T. Aljama, and R. Gonzalez, "The Use of Wavelet Packets to Improve the Detection of Cardiac Sounds from the Fetal Phonocardiogram", in *Computers in Cardiology*, Sept 1999, pp. 463-466
- [29] D.L. Roy, and B. Hoyt "EarsOn!", 1997; ISBN: (n/a); an interactive CD-ROM for auscultation training
- [30] C.A. Altman, M.R. Nihill, and J.T. Bricker "Pediatric Cardiac Auscultation", 2001; ISBN: 0781723825; an interactive CD-ROM for auscultation training
- [31] J.M. Criley "The Physiological Origins of Heart Sounds and Murmurs", 1997; ISBN: 0781716209; an interactive CD-ROM for auscultation training
- [32] R. Hooke, and T. Jeeves, ""Direct Search" Solution of Numerical and Statistical

- Problems", in *Journal of the ACM*, 1961, pp. 212 - 229
- [33] J.M. Nerbonne, and R.S. Kass, "Molecular Physiology of Cardiac Repolarization", in *Physiological Reviews*, 2005, pp. 1205-1253
- [34] M.T. Keating, and M.C. Sanguinetti, "Molecular and Cellular Mechanisms of Cardiac Arrhythmias", in *Cell*, Feb 2001, pp. 569-580
- [35] Beirne, P.; Goubran, R., "A Comparison of Two Techniques for Unsupervised Segmentation of Heart Sounds" in *16th World Congress in Cardio-Thoracic Healthcare*, 2006