THE CHARACTERIZATION OF VECTORCARDIOGRAMS FOR PATTERN RECOGNITION

by

John Stuart Halliday

B. A. St. John's College, Cambridge
(1971)

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June, 1973

Signature of the Author John S. Halliday
Department of Electrical Engineering, May 3, 1973

Certified by
Thesis Supervisor

Accepted by
Chairman, Departmental Committee on Graduate Students

THE CHARACTERIZATION OF VECTORCARDIOGRAMS FOR PATTERN RECOGNITION

by

John S. Halliday

Submitted to the Department of Electrical Engineering on May 11, 1973 in partial fulfillment of the requirements for the degree of Master of Science.

ABSTRACT

A model of the Vectorcardiogram (VCG) is presented in which the three time-varying components are expressed as a function of 24 constant parameters. For a given VCG, these parameters characterize the waveform, and may form the basis of a diagnosis scheme using statistical Pattern Recognition techniques.

The model was developed by expanding the VCG components in terms of the dominant eigenfunctions of the covariance matrix. This set of eigenfunctions forms a known time-varying matrix which operates on the pattern vector to reconstruct the original waveforms with good accuracy. The essentially planar locus of the heart vector is exploited to allow a preliminary data reduction from three to two dimensions. This step is not essential, but it eases the computation requirements for calculation of the eigenfunctions, and the resulting two components are relatively independent of the physical orientation of the heart.

The characterizing parameters are readily estimated for a given VCG with only modest computation. It is suggested that the model might find application in the filtering of noisy VCG data.

Thesis Supervisor: Sanjoy K. Mitter, Ph.D.

Title: Professor of Electrical Engineering

ACKNOWLEDGMENTS

I would like to thank Professor Sanjoy K. Mitter for introducing me to this problem, and giving me the benefit of his constructive advice and comments during the course of the investigation.

I am also indebted to the Control and Flight Dynamics Division of the Charles Stark Draper Laboratory for the opportunity to participate in the Vectorcardiogram Classification project, and in particular to Dr. M. Edward Womble, with whom it was a great pleasure to work. Our many discussions helped me in the ordering of priorities, and put the work into perspective.

Colonel M. Lancaster and Major F. Yanowitz of the USAF School of Aerospace Medicine provided Vectorcardiogram records, and gave advice on medical matters.

My special thanks go to the Harkness Fellowships of the Commonwealth Fund of New York for giving me the opportunity to live and study in this country.

Finally I wish to thank Beverly Belfer and Margaret Erickson for their patience and skill in typing the final manuscript.

This report was prepared under DSR Project 52-51400 sponsored by the USAF School of Aerospace Medicine through contract F41609-73-C-0001.

The publication of this report does not constitute approval of the USAF School of Aerospace Medicine or the Charles Stark Draper Laboratory of the findings or the conclusions contained herein. It is published only for the exchange and stimulation of ideas.

TABLE OF CONTENTS

LIST OF FIGURES

6.9 Overall Reconstruction from Pattern Vectors: Typical Normal Data

6.10 Overall Reconstruction from Pattern Vectors: A Worst Case (Abnormal Data)

6.11 Histograms of Pattern Vector Coefficients: Angle of Rotation

6.12 Histograms of Pattern Vector Coefficients: z Component of Rotation Axis

6.13 Histograms of Pattern Vector Coefficients: Waveform Period

6.14 Histograms of Pattern Vector Coefficients: First Expansion Coefficient

6.15 Histograms of Pattern Vector Coefficients: Second Expansion Coefficient

8.1 Rotation Axis

1. Introduction

1.1 Electrical Activity of the Heart and the Vectorcardiogram

The mechanical events of the cardiac cycle are initiated and synchronized by electrical events. This relationship is exploited by physicians as an aid in determining the physiological condition of the heart. The mechanisms of electrical conduction within the heart are still incompletely understood, but the heart e.m.f.'s set up electrical fields on the surface of the body, where they are readily measured. A considerable body of empirical knowledge now relates the physical activity of the heart with potential differences measured by strategically placed sensor leads on the torso. This has enabled certain cardiac abnormalities to be routinely identified by recording these electrical signals, which are known as electrocardiograms (E.C.G.'s).

An idealized plot of the potential difference between two electrodes for the duration of one complete heart cycle is shown in Fig. 1.1.

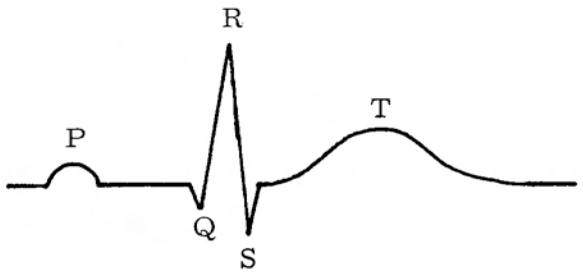


Fig. 1.1 Nomenclature of the Electrocardiogram

The cycle is initiated by the spontaneous excitation of muscle fibers in the Sinoatrial node of the heart, which is known as the "pacemaker." The electrical impulse so generated travels through the heart causing contraction of the muscles, first in the atria and then in the right and left ventricles. The P wave corresponds to the excitation of the atria. The QRS complex is related to the excitation of the ventricles. The right ventricle contracts slightly before the left ventricle, and this imbalance leads to the complex waveform observed on the body surface. The T wave is associated with recovery (depolarization) of the ventricles; recovery of the atria occurs during the QRS portion of the waveform, but the associated electrical signal is too small to be identified.

Biophysical models of the heart and Factor Analysis of surface potentials have shown that the electrical effects on the surface of the body can be accounted for by about ten electrical sources within the heart, and in fact 90 to 95 percent of the surface field could have been generated by a single equivalent electrical dipole free to rotate in three dimensions (1, 2, 3). Under suitable homogeneity assumptions, the components of this equivalent dipole can be estimated by measuring three potential differences in perpendicular directions on the surface of the body. The Frank orthagonal lead system is most commonly used, and the recording of the three components of the equivalent electrical dipole (known as the Heart Vector) is called the Vectorcardiogram (VCG). The simplicity and reproducibility of this lead system has led to its increasing acceptance in recent years over the previous clinical method which required twelve standard leads.

The three components of the heart vector, as measured by the orthogonal lead system are similar to those shown in Fig. 2.2.

For a fuller discussion of the mechanisms of electrocardiogram generation, reference should be made to two recent reviews (4, 5) or one of the standard texts (6, 7).

1.2 Automation of Diagnostic Procedures

The routine analysis of electrocardiograms is costly in terms of a trained physician's time. Since the late 1950's there have been many attempts to automate the classification of electrocardiograms into either normal or various abnormal categories. In most of these techniques, those criteria have been abstracted from the waveforms which physicians have empirically correlated with important aspects of the heart's functioning. These criteria are usually simple measurements of voltages or time intervals. A computerized binary decision-tree process determines the pathological category to which a particular waveform belongs. The accuracy of some of these schemes now approaches that of the average electrocardiographer. Ideally an automated system is evaluated using data from patients whose clinical category has been subsequently confirmed by postmortem. Practical disadvantages to this approach have meant that most schemes have been compared with the judgment of a panel of electrocardiographers. However, the imprecise nature of present clinical diagnosis techniques has been illustrated by a study in which 125 ECG's were analyzed twice by nine experienced interpreters. The results were consistent on the

average in only 73 percent of the cases (8). This is both an argument for automatic classification of electrocardiograms, and a warning that diagnostic rules must be verified as thoroughly as possible before they are incorporated into a practical system.

Although interest in rhythm analysis, or the cycle to cycle variation in the electrocardiogram, is increasing, most attention to date has been focussed on morphological analysis of a single cycle, usually a "typical" or average cycle taken from a recording. One of the most well-known of the current computer programs for classifying Vectorcardiograms is the one developed by Smith et al (9). This carries out both rhythm and waveform analysis.

Recently there have been systematic attempts, using statistical techniques, to select those features of the VCG which are most efficient for classification. Some of the most extensive investigations have been carried out by Pipberger et al, who now have the facilities of a considerable bank of data (10). Pattern recognition techniques can be used to formulate rules for classifying VCG's, based on the analysis of features selected from VCG's whose clinical categories are known.

Any scheme for classifying Vectorcardiograms, whether automated or manual, can be viewed as a three-step process:

1. Preprocessing and filtering of data
2. Feature selection
3. Formulation of decision rule

This report is concerned primarily with step 2, the selection of important features from the VCG recording. The aim is to characterize

the VCG by a small number of parameters whose statistics can be used as a basis for step 3. This set of parameters will be called the Pattern Vector. An attempt will be made to select features which provide as complete as possible a representation of the VCG. A major criterion for evaluating the characterization will be how well the original waveforms can be reconstructed from the pattern vector. Thus, if the reconstructed waveforms approximate well the original waveforms, all the diagnostic information should be preserved in the pattern vector. This is in contrast to schemes which use some of the clinical criteria such as height, width and area of the QRS complex etc.; since the original waveforms can not be reconstructed from these parameters it is quite possible that important information is being inadvertently neglected. There have been some efforts to provide a complete characterization of the QRS complex alone, using power series (11, 12). A complete representation of a cardiac cycle has not been achieved, however, except in some idealized form. (For instance, Cox et al have developed a data reduction scheme which they term AZTEC (13). This caricatures the ECG as a sequence of flat lines and sloping segments, at a rate of approximately 25 elements per second.

Our approach, therefore, is to produce a mathematical model of a complete cycle of the VCG, and to use the parameters of the model as coefficients in the pattern vector.

1.3 Preview of Following Chapters

Chapters 2, 3 and 4 describe essentially a process of step by step data reduction. Chapter 2 discusses how the three components of an average cycle of the VCG are estimated from a noisy recording on analogue tape of some sixty heartbeats, then how the waveforms are standardized for the subsequent modelling steps.

Chapter 3 describes a rotation of the co-ordinate system that reduces the dimension of the VCG data from three to two, with only small loss of accuracy. This is based on the observation that the locus of the heart vector is essentially planar. Chapter 4 shows how, using the discrete Karhunen-Loève expansion, the resulting two components can be expressed as a series in terms of some time-varying deterministic basis functions, which are estimated from the statistics of known data. This leads to a model of the vectorcardiogram as a linear time-varying system with some twenty-three known parameters. These parameters are estimated for a particular VCG, and become coefficients of the pattern vector. Appropriate mathematical results for these two sections are gathered into chapter 8, so as not to interrupt the flow of the text. Chapter 8 includes a derivation of the result that the discrete Fourier transform and the Karhunen-Loève expansion are equivalent for periodic stochastic processes which are wide-sense stationary.

Chapter 5 discusses some general questions relating to pattern recognition techniques for electrocardiogram analysis. The following

chapter records results which evaluate the validity of the model using 147 sets of VCG data, both normal and abnormal. Chapter 7 discusses the usefulness of the characterization and possible directions for further investigation.

An Appendix gives details of Fortran routines for finding part of the eigensystem of a large matrix. This computation is necessary for finding the basis functions of chapter 4.

2. Data Processing and Smoothing

2.1 Introduction

The preliminary processing of the raw VCG data is not a primary concern of this report, and so will be discussed only briefly for completeness. The data was obtained from the U.S.A.F. School of Aerospace Medicine where it was gathered during the course of routine checkups on flight personnel. Three hundred sets of VCG data were made available for the project. Of these, one hundred had been classified by the School electrocardiographers as Normal, one hundred as Questionable, and the remainder as unclassified for subsequent evaluation of the Pattern Recognition techniques to be based on this modelling work.

Fig. 2.1 shows several cycles of typical data. The waveforms contain a substantial amount of power line (60Hz) noise, as well as a drifting d.c. offset. The following two sections describe how a clean average cycle of the VCG is prepared and standardized.

2.2 Extraction of a Typical Waveform

Digital rather than analog methods were used to handle the data, and the original waveforms were sampled every four milliseconds. A simple averaging technique was used to produce a typical cycle from a record of some sixty heart beats. The peak of the R wave served as a reference point to identify a heart beat. Its position was found by searching for a maximum of the magnitude of the heart vector. More sophisticated methods have been commonly used. These identify the

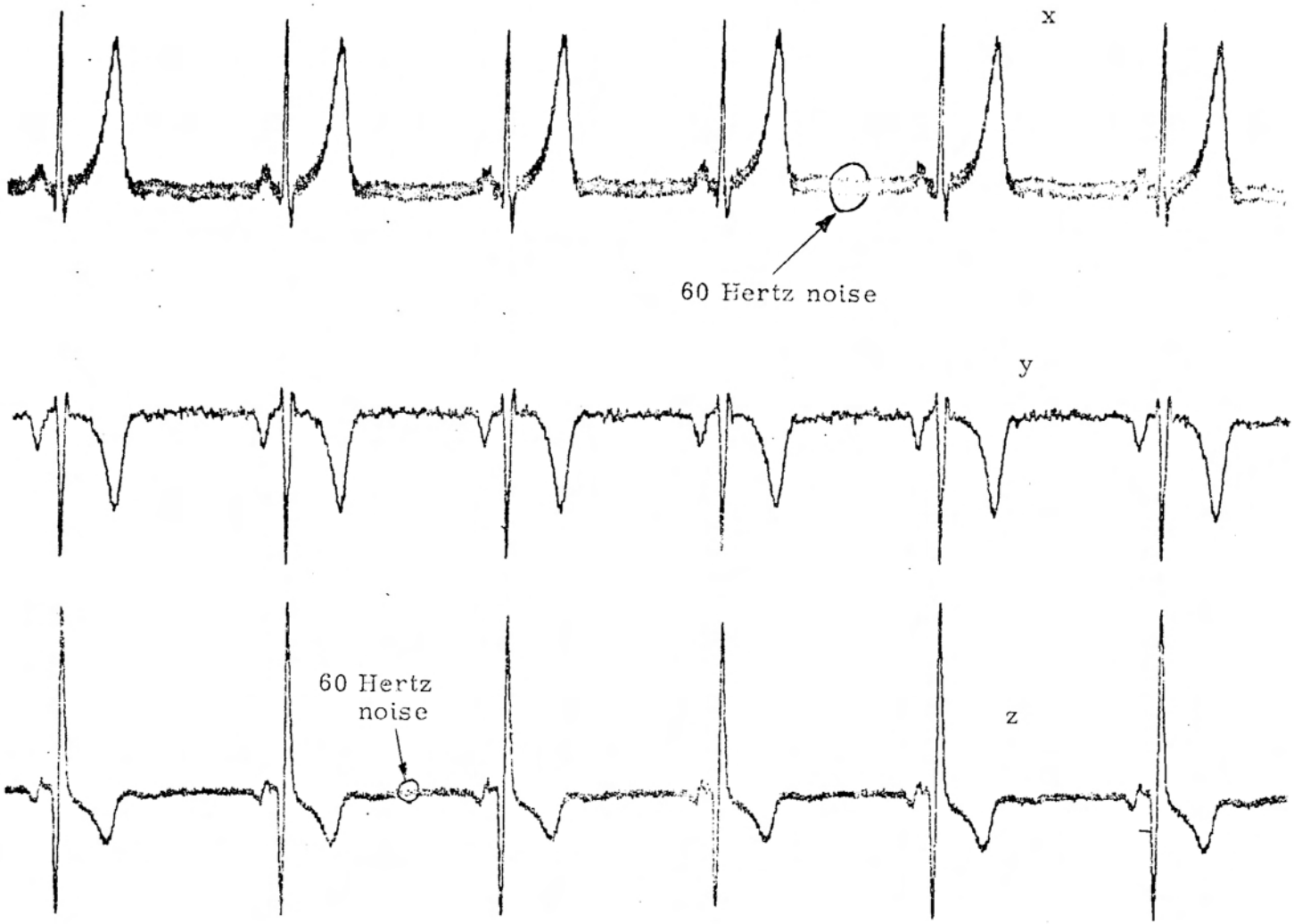


Figure 2.1 Typical VCG Recording

QRS complex by examining some suitably defined derivative function. However, for the present application, the search for a peak seemed to be adequate.

The averaged waveform still contains powerline noise and d.c. drift. The latter could be eliminated quite easily by subtraction, but the 60Hz powerline ripple required more thought; a low-pass filter or a band-pass filter might excessively distort the desired signal. In the end, a non-linear identification scheme was used to determine the amplitude, phase and frequency (relative to the sampling frequency) of the powerline noise, and also the d.c. drift. The total distorting noise was then eliminated by subtraction.

The noise was modelled as

$$e(nT) = a_0 + a_1 nT + a_2 \sin \omega nT + a_3 \cos \omega nT$$

where ω = ripple frequency

T = sampling period.

The unknown parameters a_0, a_1, a_2, a_3 and the deviation of the product ωT from its nominal value, were estimated from that part of the waveform which contains essentially no signal -- the "dead time" between the T wave and the next P wave. An example of an averaged waveform after subtraction of the estimated noise is shown in Fig. 2.2.

2.3 Standardization of Records

Each component of the average cycle is now represented by about 250 to 300 sample points, corresponding to a period of about one second.

The next question is how to take care of the variation in period between different VCGs. Time normalization appears attractive, but such a distortion of the time axis may conceal phase information. Instead, the record length was standardized at 300 samples in order to preserve time-interval information. The R peak, being fairly narrow and well-defined, was used as a datum for time, and its position was fixed at the 99th sample, or about one third of the way along the time axis. Since most of the VCG periods contain less than 300 samples, the beginning and end of the records were extended as necessary with zeros; the few records which exhibited a prolonged dead-time at the beginning and end were truncated. The actual period length was stored for incorporation into the pattern vector. The record in Fig. 2.2 has been standardized in this manner.

Normalization of the magnitude of the waveforms was rejected on the grounds that absolute voltage levels might prove to be an important criterion.

Along with the value of the period, the three components x, y and z, each described by 300 samples, represent the VCG information which will be modelled as described in the next two sections.

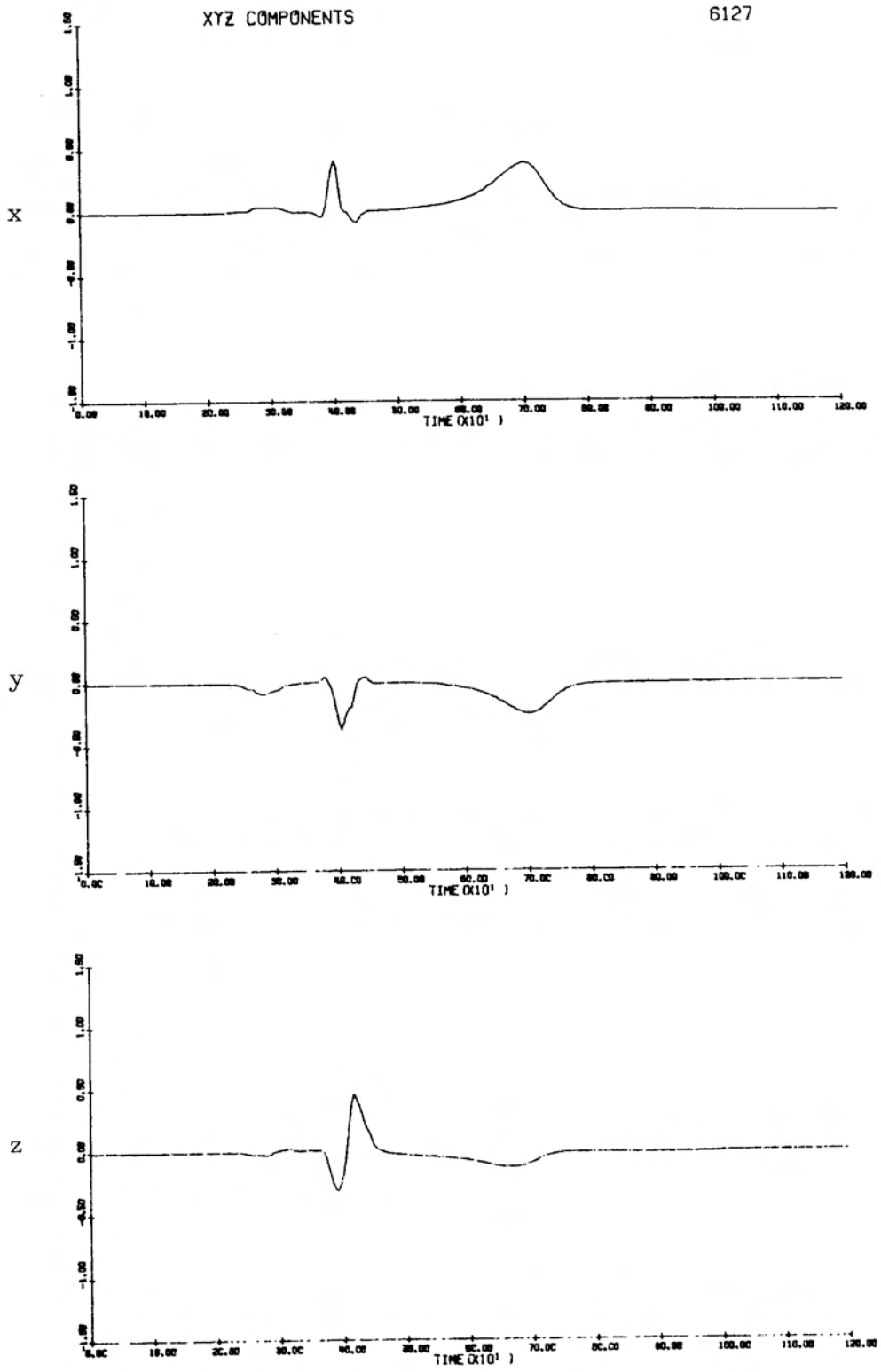


Figure 2.2 VCG Cycle, Averaged and Filtered

3. Choosing an Optimum Co-ordinate System

3.1 Introduction

The Vectorcardiogram is usually represented in terms of the time history of the three components, x, y and z, as measured by the Frank Orthagonal lead system. It could also be expressed by the time histories of the instantaneous magnitude (or length) of the heart vector and two angles. In either case there are three parameters for every instant of time.

However, it has been found that the remote field of the heart behaves as if generated by a dipole whose axis lies in a plane formed by the heart axis and the line perpendicular to the septum at its midpoint (4). Thus the heart vector sweeps in a plane defined approximately by the right shoulder, left waist, and the small of the back. The approximately planar locus of the tip of the dipole vector is illustrated in Fig. 3.1. There are substantial variations from patient to patient in the orientation of the plane, but the out of plane component of the heart vector is almost always relatively small in both normal and abnormal Vectocardiograms. This suggests that if the heart vector is expressed in terms of a new cartesian co-ordinate system, two of whose axes lie in the plane, then the VCG can be represented approximately by the two planar components, and the third component can be neglected. This reduction of the state of the VCG amounts to finding a rotation matrix $\underline{R}$ which expresses the original three components x(t), y(t) and z(t) in terms of three new components u(t), v(t) and w(t) of which one, say v(t), is approximately zero for all time. Thus

$$\begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix} = \underline{R} \begin{bmatrix} u(t) \\ v(t) \\ w(t) \end{bmatrix} \simeq \underline{R} \begin{bmatrix} u(t) \\ 0 \\ w(t) \end{bmatrix}$$

If the transformation information in $\underline{R}$ is stored, then the data required to represent the heart vector is reduced by one third. The transformation is illustrated diagramatically in Fig. 3.1 which shows the locus of the tip of the heart vector. The u and w axes define the Principal Plane of the VCG.

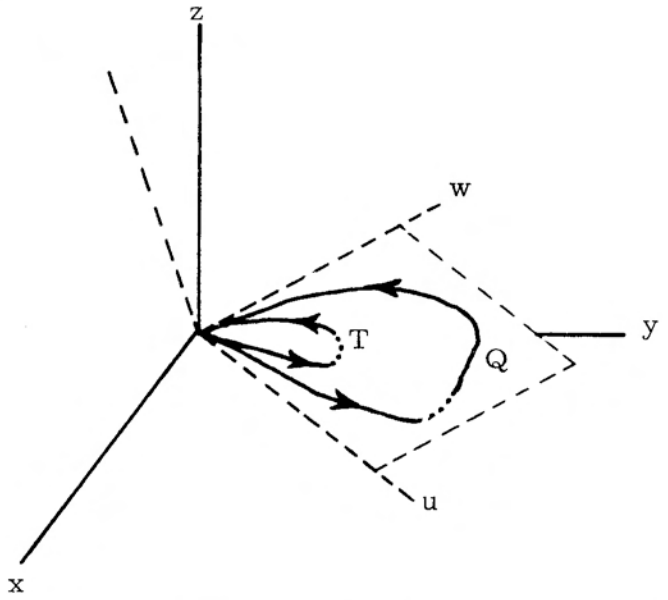


Fig. 3.1 The Principal Plane of the Vectocardiogram

3.2 Intrinsic Components; the Principal Plane

Young and Huggins (14) showed how Factor Analysis could be used to define a new co-ordinate system so that the direction containing maximum VCG energy is one of the axes, that containing minimum energy is the second axis, and the third axis is perpendicular to the other two. Details are given in section 8.21. If the original components are given by $\underline{X}(n) = [x(n)\ \ y(n)\ \ z(n)]$, $n = 1, N$ (we assume now that the waveforms

are sampled), the new axes are given by the normalized eigenvectors of the matrix

$$\underline{S} = \sum_{n=1}^{N} \underline{X}(n)\, \underline{X}'(n)$$

and the energy contributions are given by the corresponding eigenvalues. The new components, u(t), v(t) and w(t) are called Intrinsic Components.

We shall call the new axes Principal Axes. Young and Huggins argued that they define a more natural co-ordinate system, because the intrinsic components do not depend on the physical orientation of the heart, whereas the original components do. Fig. 3.2 shows the intrinsic components for a normal patient. For most normal VCG's that have been analyzed, the proportion of VCG energy in the component perpendicular to the principal plane has been less than three percent Fig. 3.3 shows the QRS and T loops formed by plotting the two components in the principal plane.

3.3 Representation of the VCG in the Principal Plane

There is no reason to suppose that the orientation of the two major principal axes in the principal plane has any physical significance other than that one direction contains as much as possible of the VCG energy. It would be convenient to choose the axis orientation so that corresponding features in plots of different normal VCGs are represented in some consistent manner that will enhance their similarities. Fig. 3.3 shows how the QRS complex exhibits considerable angular spread, whereas the angular spread during the T wave is very small. Accordingly, the T peak is used as a datum for angle in the principal

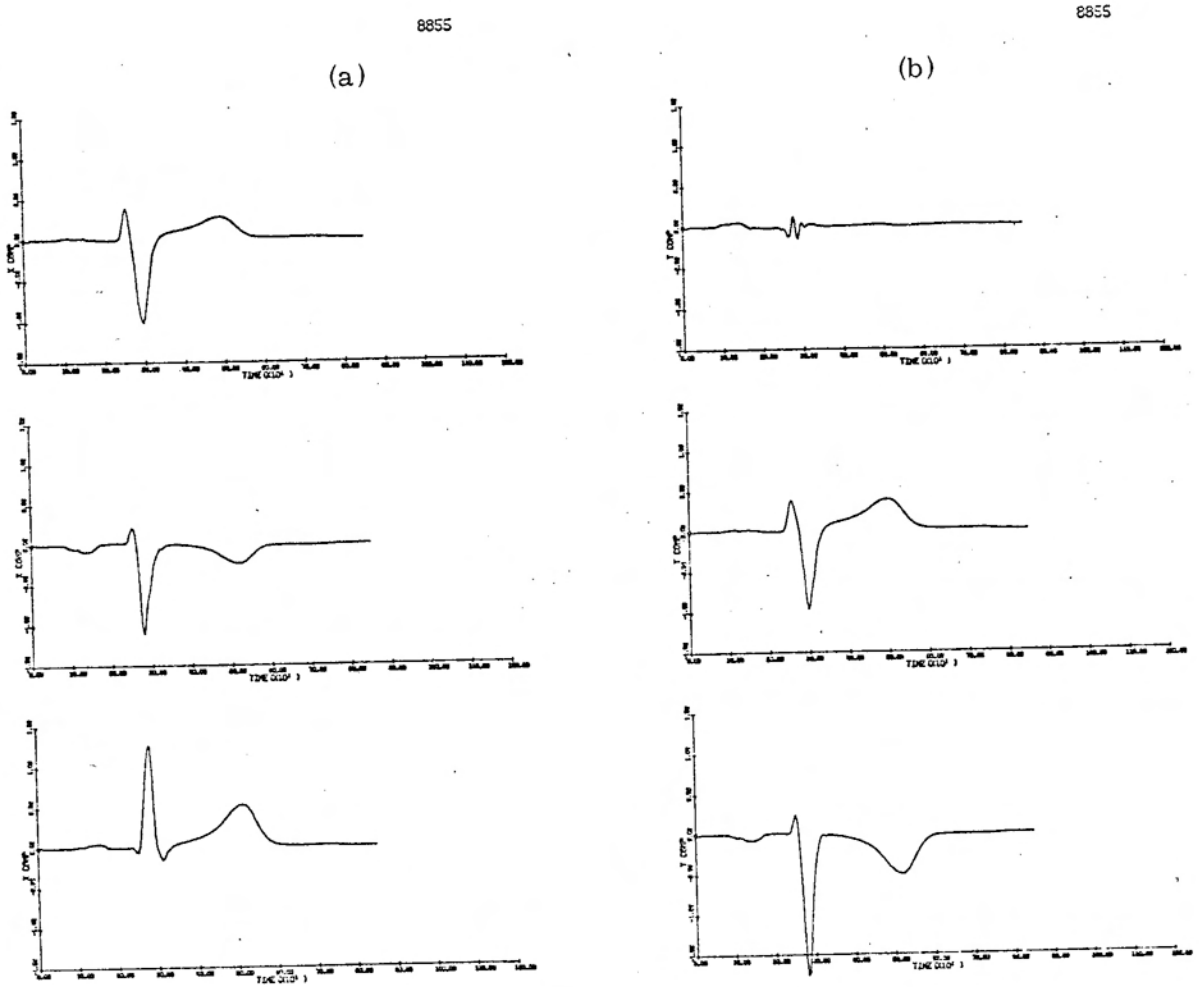


Figure 3.2 Intrinsic Components of the VCG
(a) xyz Components
(b) Intrinsic Components

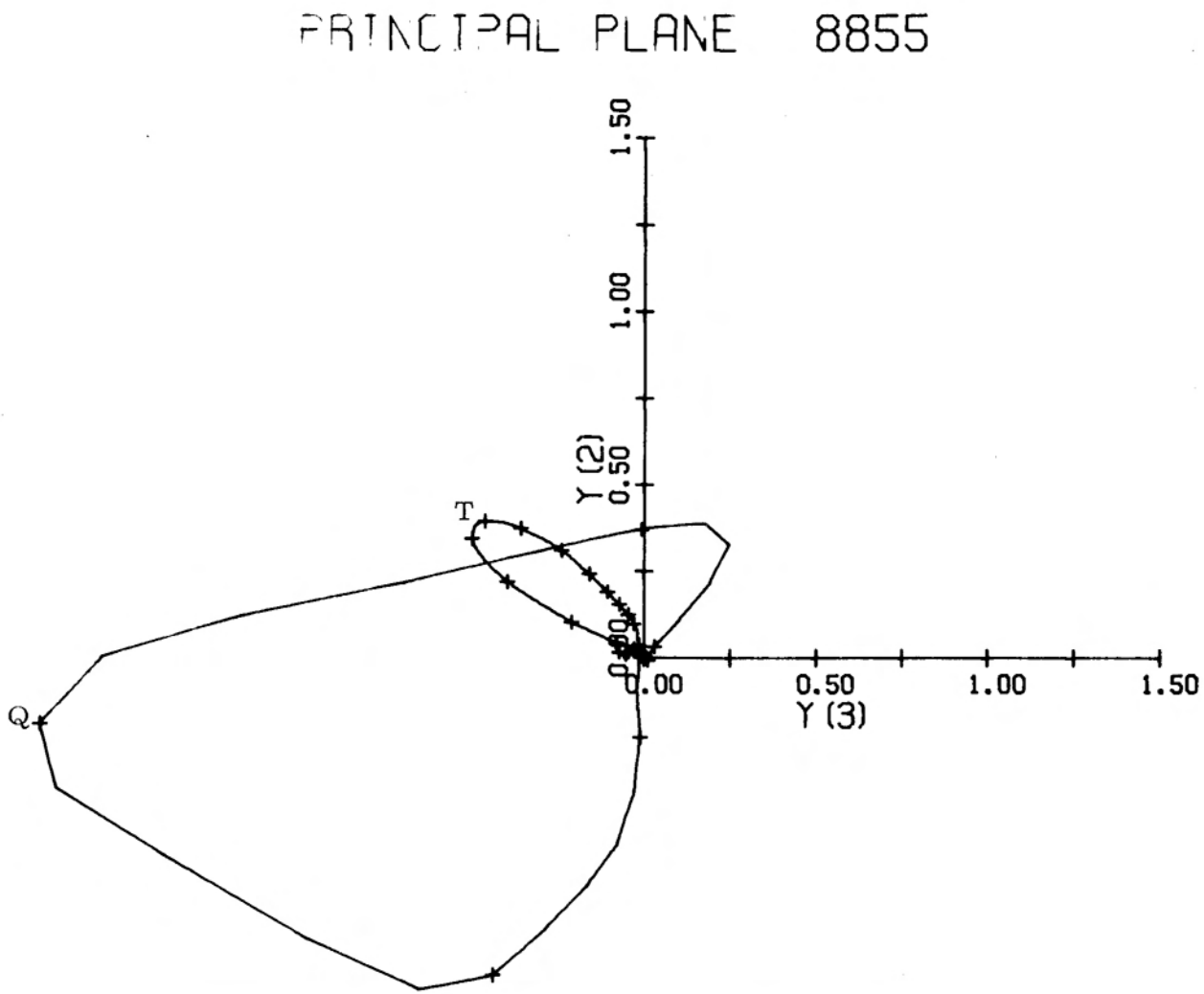


Figure 3.3 Locus of VCG in the Principal Plane

plane, and the w axis is made perpendicular to the u axis in such a way that the QRS complex rotates in a counter-clockwise direction about the v axis (which remains normal to the principal plane, although its direction may be reversed). Plots of the resulting three orthogonal components are shown in Figure 3.4 and a principal plane locus plot is shown in Figure 3.5.

Figure 3.6 shows the magnitude and angle corresponding to Figure 3.5. Originally it was intended that these two quantities should be used rather than the u and w components. However, when the magnitude of the vector is small, it is difficult to determine the angle, whereas its value at these instants may be important in determining how many complete revolutions have occurred. Also the steep slope of the angle during the QRS complex makes approximation by series expansion difficult. A further important consideration is that the vector Karhunen-Loève expansion (see section 8.1.3) can take advantage of the similarity between the shapes of the u and w waveforms, whereas any correlation between the magnitude and angle is difficult to see. Accordingly, the VCG is represented by the two perpendicular components u and w, and details of the transformation are stored in $\underline{R}$. It is shown in section 8.2.2 that this rotation matrix can be alternatively represented by a unit vector defining the axis of rotation, and a corresponding angle of rotation about that axis. Because the unit vector has a magnitude of one, the four numbers defining $\underline{R}$ can be reduced to three by multiplying each of the components of the unit vector by the angle.

The three original components of the VCG have now been represented approximately by two components U(t) and w(t), and a rotation matrix specified by three parameters.

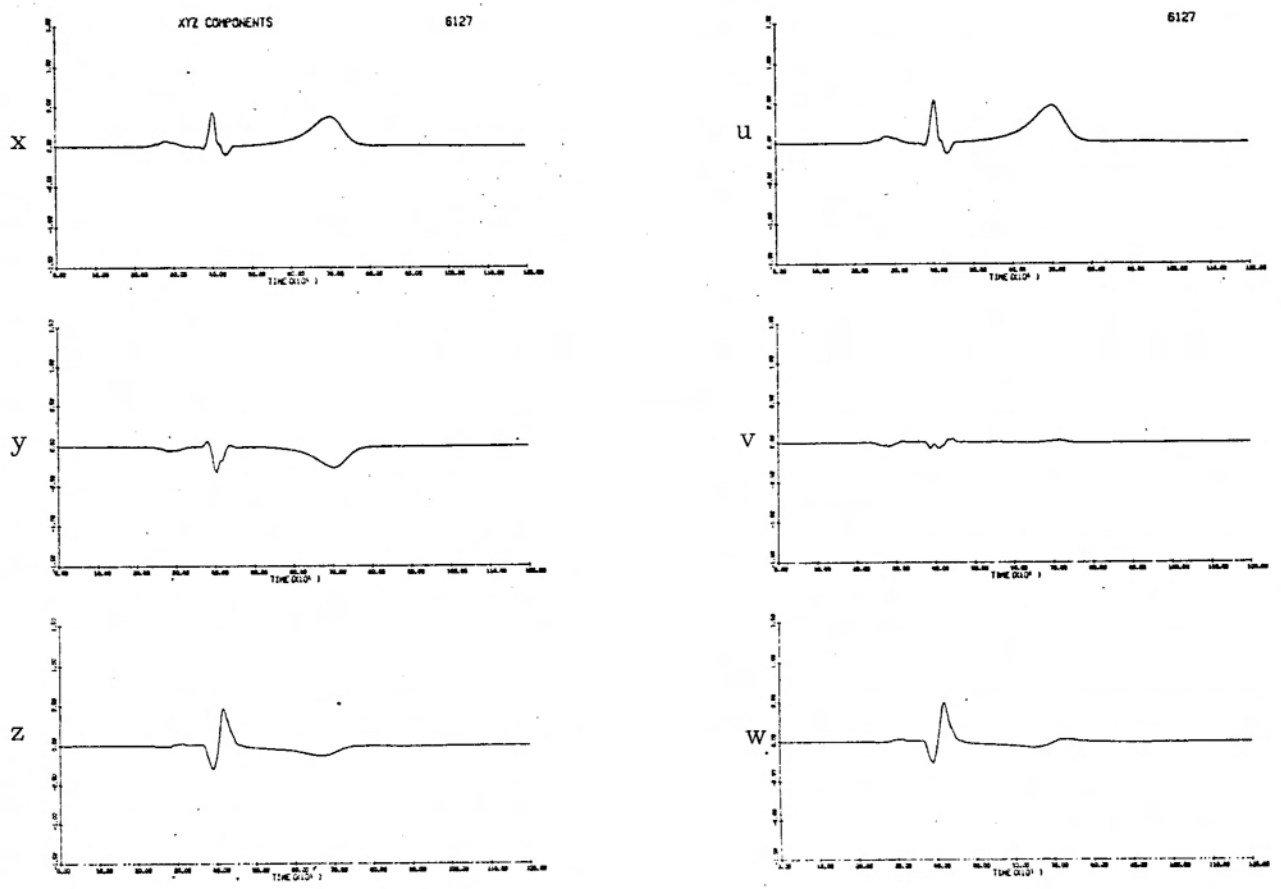


Figure 3.4 Standardized UVW Components
(a) xyz Components
(b) uvw Components

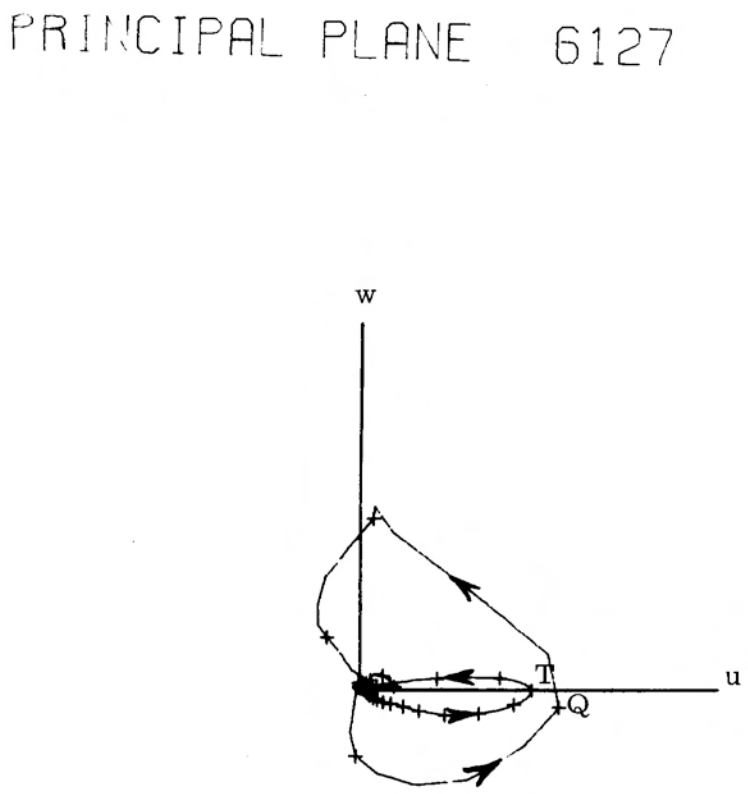


Figure 3.5 Locus of VCG in the Principal Plane, with Standardized Axes

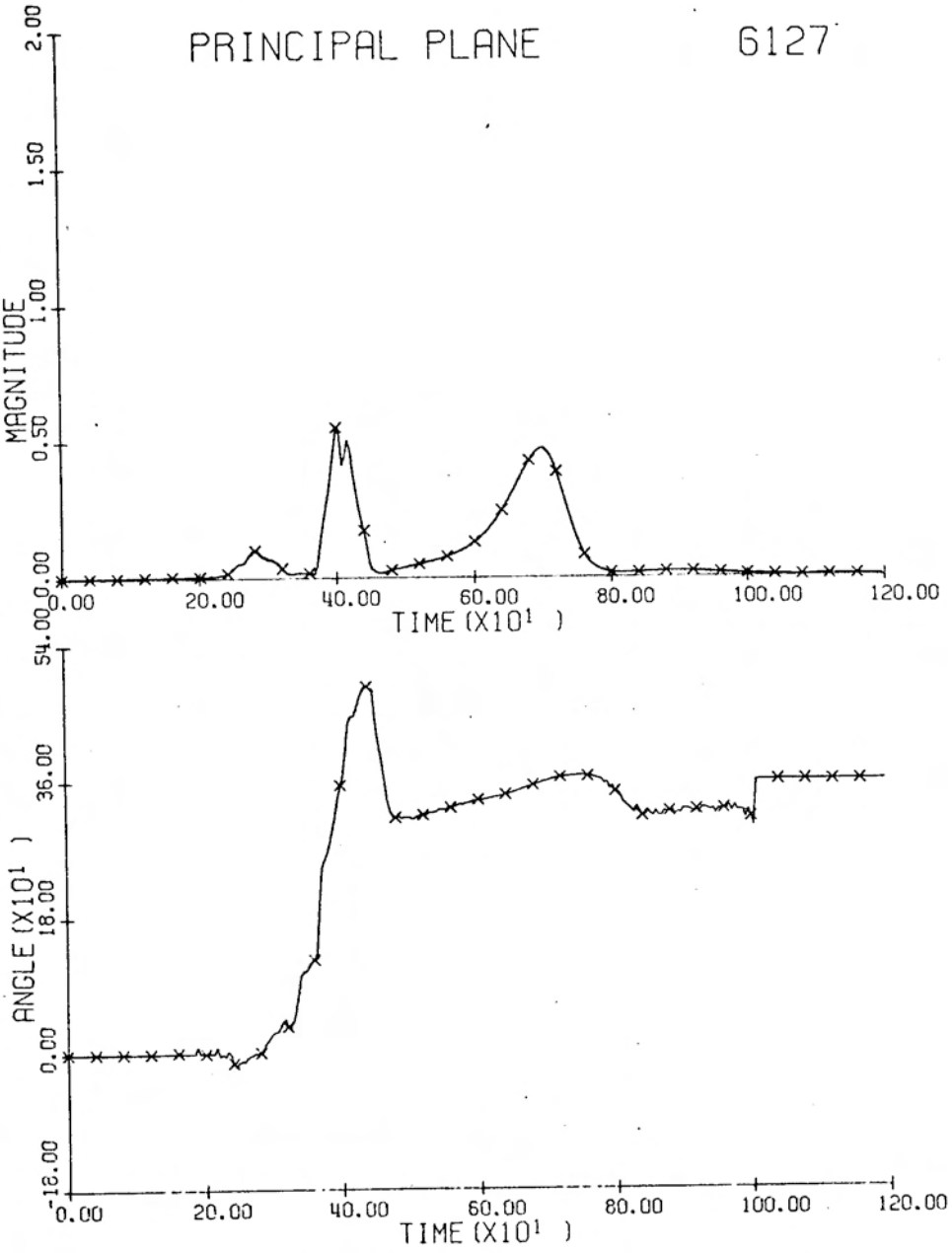


Figure 3.6 Magnitude and Angle in the Principal Plane

3.4. Factor Analysis and Multi-lead Electrocardiograms

The factor analysis of section 8.21 discussed here for the three dimensional case can be generalized to the situation where any number of surface potential measurements are made. The general problem can be stated as finding a set of vectors $\underline{\Phi}_i$ and corresponding time functions $y_i(n)$ such that the original data $\underline{X}(n) = [x_1(n)\ x_2(n). \ . \ .x_m(n)]$ can be expressed as

$$\underline{X}(n) = \begin{bmatrix} x_1(n) \\ x_2(n) \\ \cdot \\ \cdot \\ x_m(n) \end{bmatrix} = [\underline{\Phi}_1\ \underline{\Phi}_2 \cdot \quad \cdot\ \underline{\Phi}_m] \begin{bmatrix} y_1(n) \\ y_2(n) \\ \cdot \\ \cdot \\ y_m(n) \end{bmatrix}$$

The vectors $\underline{\Phi}_i$ are ordered according to the energy contributions of the corresponding functions $y_i(n)$. Together they span the space swept out by the components of $\underline{X}(n)$. The direction $\underline{\Phi}_1$ is chosen so that $y_1(t)$ accounts for as much as possible of the energy in the original components; $\underline{\Phi}_2$ is chosen to account for as much as possible of the energy in the remaining subspace, and so on. As before the $\underline{\Phi}_i$ are the eigenvectors of the matrix

$$\underline{S} = \sum_{n=1}^{N} \underline{X}(n)\underline{X}'(n)$$

We might approximate X(n) by the first r functions ($r \leq m$)

$$\underline{X}(n) = [\underline{\Phi}_1\ \underline{\Phi}_2 \cdot \cdot\ \underline{\Phi}_r] \begin{bmatrix} y_1(n) \\ y_2(n) \\ \cdot \\ \cdot \\ y_r(n) \end{bmatrix}$$

Studies (1, 2, 3) have indicated that about 90 percent of surface potentials can be accounted for with two functions, and 99 percent with eight functions.

This method would allow data from a 12 lead ECG system to be approximated by a number of its intrinsic components.

4. Feature Extraction

4.1 Introduction

In this section we discuss the development of a mathematical model of the vector cardiogram waveforms. First it is shown how a single waveform can be expressed approximately by a truncated series expansion. The expansion is then generalized to the case in which the waveform has more than one component. This results in the components of the VCG being expressed as a pattern vector multiplied by a matrix of known time-varying coefficients.

The final form of the model depends on whether the three components of the VCG are first reduced to two by neglecting the component normal to the Principal Plane.

4.2 Characterization in Terms of Orthonormal Functions

An electrocardiogram waveform can be considered to be a realization of a discrete stochastic process. The stochastic process can be a member of one of several classes corresponding to Normal and Abnormal hearts. To simplify the discussion, assume for the present that we are dealing with a single class of waveforms from Normal hearts. The set of time samples of a particular waveform can also be thought of as a point in N-dimensional space. Thus an electrocardiogram waveform is an N-dimensional random variable.

An efficient way of representing a continuous waveform is to expand it as a series of time-varying orthogonal functions. One such

representation is the familiar Fourier Series,

$$f(t) = a_0 + a_1 \cos\omega t + a_2 \cos 2\omega t + \ldots.$$

$$+ b_1 \sin\omega t + b_2 \sin 2\omega t + \ldots.$$

The signal f(t) can be reconstructed from the set of coefficients $(a_0, a_1, a_2, \ldots., b_1, b_2, \ldots.)$, which can thus be thought of as a Pattern Vector for the signal. There are in general an infinite number of Fourier coefficients, but for physical bandlimited signals, the coefficients become very small after a certain number of terms. Thus a Fourier Series can usually be truncated after a finite number of terms with only small error.

The equivalent representation for a sampled waveform is the Discrete Fourier Transform (DFT). If N samples define the original waveform (assumed real), then N coefficients of the DFT specify the waveform exactly. An approximate representation can be formed by neglecting the smaller coefficients.

The Fourier Series is just a special case of a class of expansions in terms of orthonormal functions. Cady et al expressed the electrocardiogram in this way (11), but Young and Huggins went one step further and used orthonormal complex exponentials (3). However, the most general form of an orthonormal series is the Karhunen-Loève Expansion (15). Over the set of all orthonormal expansions, this is optimum in the sense that for a given number of terms in the truncated series, the mean square error is minimized.

4.3 The Karhunen-Loève Expansion

The basis functions for the Karhunen-Loève Expansion are analogous to the sine and cosine terms in the Fourier Series, but instead of being commonly tabulated functions, they are derived from the statistics of the data. Section 8 gives a derivation of the expansion. For a sampled waveform, the basis functions are the eigenvectors of the covariance matrix. For a continuous waveform they are eigenfunctions of an integral equation (16), and may in general be difficult to derive. The corresponding eigenvalues measure the expected energy along each eigenvector component. Thus the discretized waveform can be expressed as

$$x(n) = \sum_{i=1}^{N} \alpha_i \Phi_i(n)$$

where

$$\alpha_i = \sum_{n=1}^{N} x(n) \Phi_i(n)$$

If the terms are ordered so that the associated eigenvalues decrease in magnitude with i, the expansion can be truncated after M terms, and

$$x(n) = \sum_{i=1}^{M} \alpha_i \Phi_i(n) + \epsilon(n)$$

This is more conveniently written in matrix form by defining

$$\underline{X} = [x(1)\, x(2) \ldots x(N)]'$$

$$\underline{\Phi}_i = [\Phi_i(1)\, \Phi_i(2) \ldots \Phi_i(N)]'$$

Then

$$\underline{X} = \sum_{i=1}^{M} \alpha_i \underline{\Phi}_i + \underline{e}$$

or

$$\underline{X} = [\underline{\Phi}_1 \ \underline{\Phi}_2 \cdots \underline{\Phi}_M] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdot \\ \cdot \\ \alpha_M \end{bmatrix} + \underline{e}$$

The basis functions $\underline{\Phi}_i$ span that subspace of R^N which contains on average the greatest proportion of the energy in $\underline{X}$. A useful interpretation is that if $\underline{X}$ if normally distributed (i.e., specified completely by its second order statistics), the functions $\underline{\Phi}_i$ are unit vectors along the axis of the covariance ellipsoid. An expansion to N terms is just a rotation of co-ordinates in R^N.

It is most convenient to expand the deviation of $\underline{X}$ about its expected value, so now for simplicity consider $\underline{X}$ to be zero mean. The $\underline{\Phi}_i$ are then eigenvectors of $\underline{R}$

$$\text{i.e. } \underline{R}\ \underline{\Phi}_i = \lambda_i \underline{\Phi}_i$$

where

$$\lambda_i = E\{ \alpha_i^2 \}$$

Some properties of the expansion are summarized below.

1. Over the set of all orthonormal functions, for a given number of terms in the expansion, the eigenvectors of $\underline{R}$:
   (a) Minimize the mean square error
   (b) Minimize the population entropy, defined by

$$h = -E\{ \ln p(\underline{\alpha}) \}$$

(c) Maximize the distance between independent samples from a single distribution, as defined by the scatter measure:

$$\bar{d}_{\alpha}^{\,2} = E\{ ||\alpha_i - \alpha_j|| \}^2$$

2. The coefficients α_i are uncorrelated
3. For a wide-sense stationary periodic process, the Karhunen-Loève expansion reduces to the D. F. T.

Property 3 shows that the D. F. T. is an efficient representation only if the electrocardiogram is stationary in a statistical sense, which it clearly is not, even if it is normalized to zero mean.

The foregoing discussion of the Karhunen-Loève expansion applied to a scalar stochastic process can be extended to a vector process. Here we consider an expansion with vector basis functions and scalar coefficients.

$$\underline{x}(n) = \sum_{i=1}^{M} \alpha_i \, \underline{\Phi}_i(n) + \underline{\epsilon}(n)$$

Section 8.13 shows that the results of the scalar expansion carry over directly if equation 4.1 is written (for a 2 component expansion):

$$\begin{bmatrix} \underline{X}_1 \\ \underline{X}_2 \end{bmatrix} = \sum_{i=1}^{M} \alpha_i \begin{bmatrix} \underline{\Phi}_i^1 \\ \underline{\Phi}_i^2 \end{bmatrix} + \begin{bmatrix} \underline{e}_1 \\ \underline{e}_2 \end{bmatrix}$$

We would expect that for a given number of terms the vector expansion would provide a more efficient representation than the scalar expansion carried out separately for two components, because it takes into account correlation between the two waveforms.

So far it has been assumed that the waveform is a random function from a single stochastic process. However, Chien and Fu (17) show that the Karhunen-Loève expansion is still optimum for random functions which are realizations of more than one stochastic process, provided that the covariance matrix is defined appropriately.

Suppose there are k stochastic processes, each with probability of occurence Pi. Then the covariance matrix is defined by:

$$\underline{K} = \sum_{i=1}^{k} p_i \, E \{\underline{X} \, \underline{X}'\}$$

4.4 The Karhunen-Loève Expansion: Computation of Basis Functions

The main problem with the Karhunen-Loève Expansion is the computation of the dominant eigenvalues and eigenvectors of a matrix which may have order 300. However, the matrix is symmetric and the eigenvalues are real. Furthermore it can be reduced by an orthogonal transformation to a symmetric tridiagonal matrix. Thus if the order of the covariance matrix is N, the eigenvalue computation only has to deal with (2N-1) numbers instead of the original N(N + 1)/2. The Appendix describes an efficient Fortran routine which carries out the computations for finding the dominant eigenvalues and eigenvectors.

The covariance matrix was estimated from the data in the training set (currently containing 93 Normal and 46 Abnormal VCG's). Basis functions were calculated for the u and w components of the rotated data (see section 3) to give a total of twenty coefficients in the truncated

expansion. In order to ease computation requirements, only that portion of the waveform lying between the 51st and 200th samples was expanded; outside of this range the deviation from the mean value was very small for all the VCG's that have been examined.

Expanding u and w separately meant ten basis functions and coefficients for each component. For the two component vector expansion, twenty eigenvectors were calculated from the combined covariance matrix. A comparison of computation times and core requirements is shown below. (These figures include input/output operations.)

	u and w expanded independently	u and w expanded as vector
	10 eigenvectors of 150 x 150 matric, twice	20 eigenvectors of 300 x 300 matrix
Time	3 · 62 minutes	9 . 61 minutes
Core	224K	352K

The vector expansion is computationally the most demanding, but since the basis computation needs to be carried out once only, this is not particularly significant.

Figure 4.1 shows how the expected mean square error depends on the number of terms, for the vector expansion. It can be seen that ten terms account for all but about 3 percent of the energy, on average. Figure 4.2 shows the mean waveforms about which the u and w components are expanded. Figure 4.3 and 4.4 show some typical basis functions.

4.5 Complete Model of the Vectorcardiogram

Chapter three showed how the original x, y, z components of the Vectorcardiogram could be represented approximately by just two

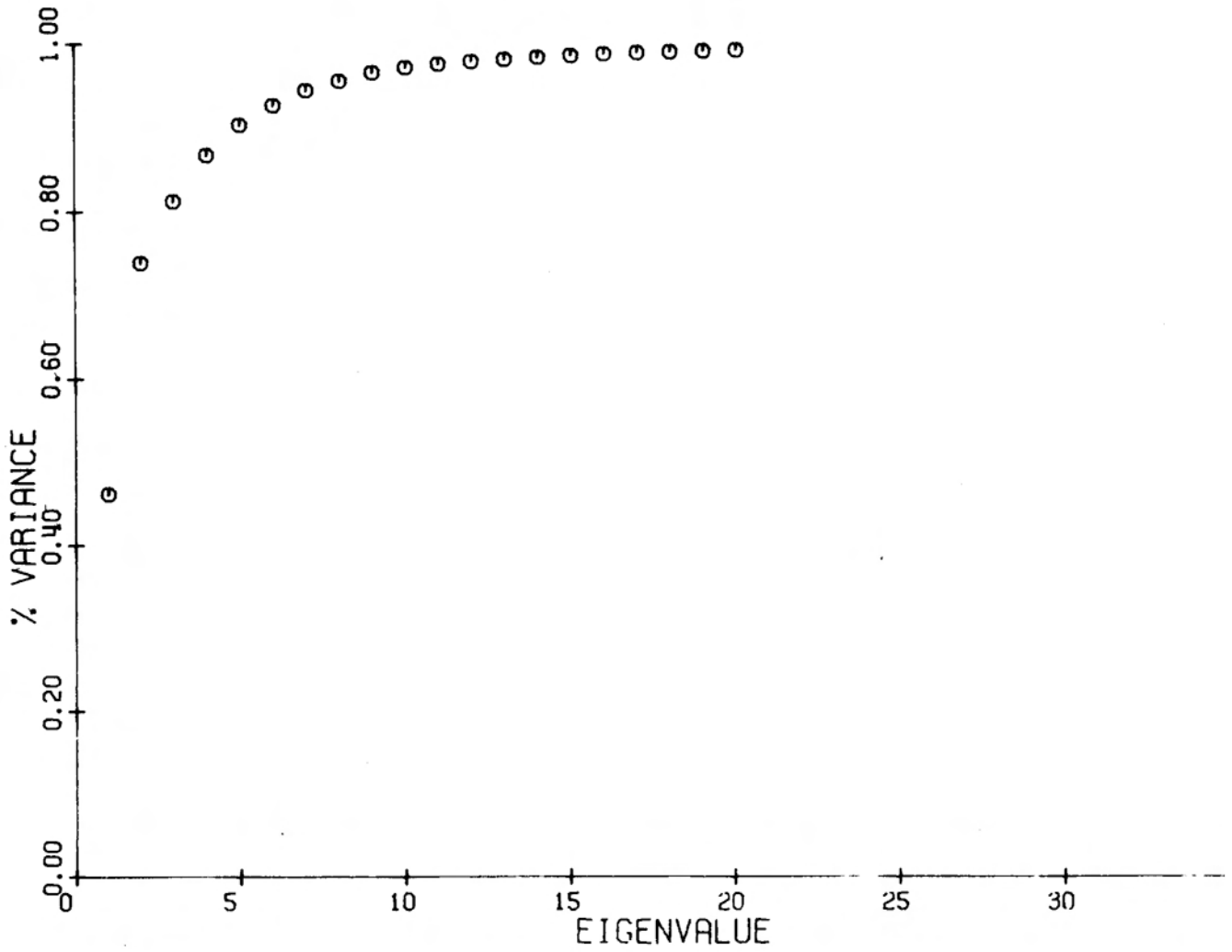


Figure 4.1 Karhunen-Loève Expansion: Distribution of Eigenvalues

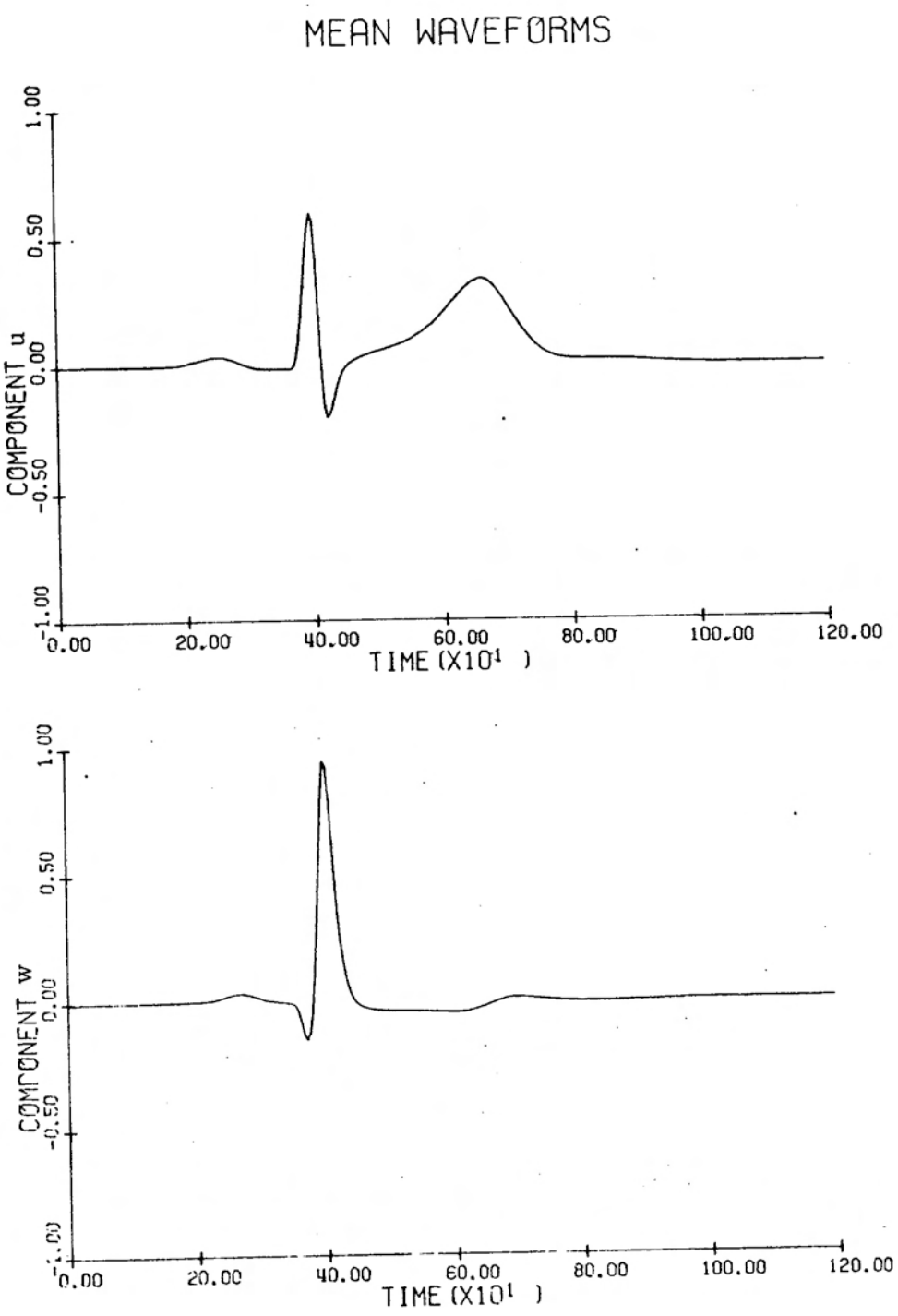


Figure 4.2 Karhunen-Loève Expansion; Mean Waveforms

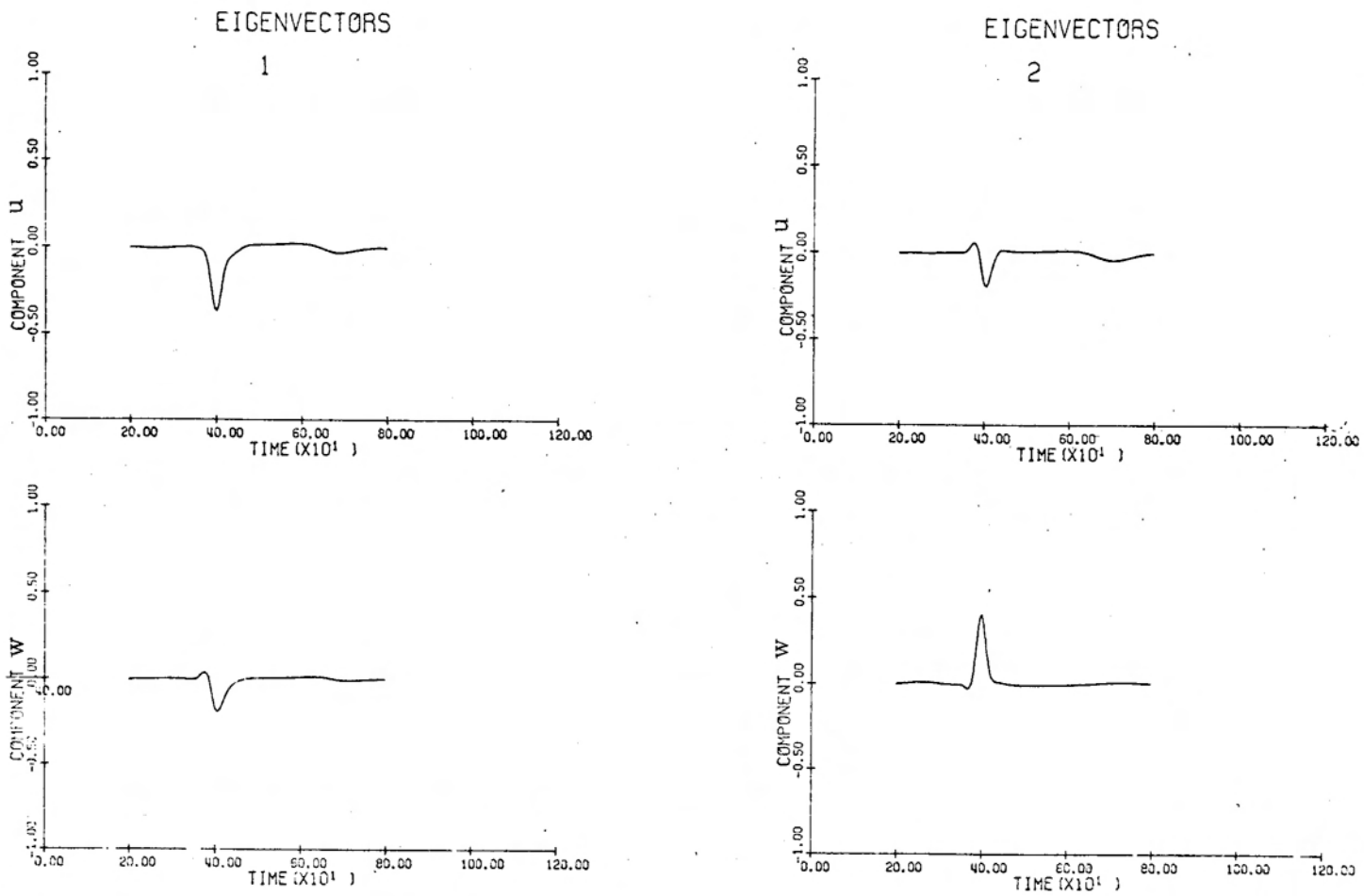


Figure 4.3 Karhunen-Loève Expansion: Eigenvectors 1 and 2

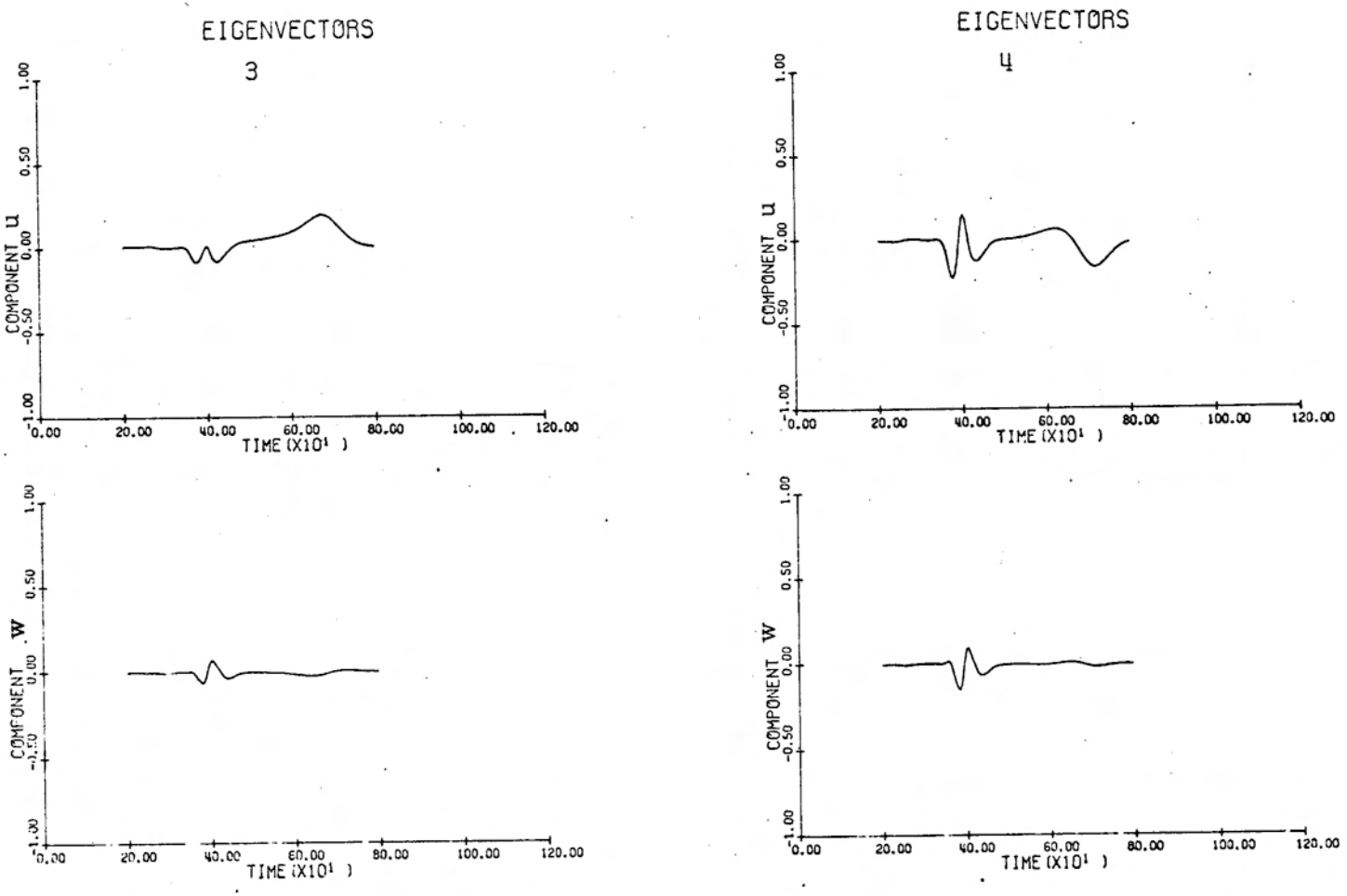


Figure 4.4 Karhunen-Loève Expansion: Eigenvectors 3 and 4

components u and w.

$$\begin{bmatrix} x(n) \\ y(n) \\ z(n) \end{bmatrix} = \underline{R} \begin{bmatrix} u(n) \\ v(n) \\ w(n) \end{bmatrix} = \underline{C} \begin{bmatrix} u(n) \\ w(n) \end{bmatrix} + \begin{bmatrix} 0 \\ e\ y(n) \\ 0 \end{bmatrix}$$

where $\underline{C}$ is a 3 x 2 matrix.

The preceding sections have shown how u and w can be expanded in terms of deterministic time-varying functions $\underline{\Phi}_i(n)$

$$\begin{bmatrix} u(n) \\ w(n) \end{bmatrix} = \sum_{i=1}^{M} \alpha_i \begin{bmatrix} \Phi_i^u(n) \\ \Phi_i^w(n) \end{bmatrix} + \begin{bmatrix} e_u(n) \\ e_w(n) \end{bmatrix}$$

or

$$\begin{bmatrix} u(n) \\ w(n) \end{bmatrix} = \underline{K}(n) \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ . \\ . \\ \alpha_M \end{bmatrix} + \begin{bmatrix} e_u(n) \\ e_w(n) \end{bmatrix}$$

where $\underline{K}(n)$ is a known, time varying 2 x M matrix.

Combining the two results, we obtain:

$$\begin{bmatrix} x(n) \\ y(n) \\ z(n) \end{bmatrix} = \underline{C}\underline{K}(n) \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ . \\ . \\ \alpha_M \end{bmatrix} + \begin{bmatrix} e_x(n) \\ e_y(n) \\ e_z(n) \end{bmatrix}$$

This equation models the VCG. The matrix $\underline{C}$ is specified by three independent parameters which define a rotation vector and the corresponding angle. These three numbers, the period, and the coefficients α_i characterize the VCG, and can be assembled in the pattern vector as

$$\underline{P} = [c_1\ c_2\ c_3\ \text{period}\ \alpha_1\ \alpha_2\ \ldots\ \alpha_M]'$$

Note that for pattern recognition purposes it is probably more meaningful to replace the three numbers c_1 c_2 and c_3 by the three components of the axis of rotation, and the angle of rotation. However, the above form is the most compact for storage.

4.6 Estimation of Model Parameters

For a particular VCG, the parameters of $\underline{C}$ can be estimated by a simple 3 x 3 eigenvector calculation, and the values of α_i are estimated by straightforward summations. Provided that the d.c. bias is removed first, the model parameters could be estimated from the original VCG recordings (time standardized) without the necessity of first eliminating the 60 H_z ripple. This is possible because the ripple is small, and (by inspection) it is uncorrelated with the K.L. basis functions.

The preliminary reduction of the three original components to two components was based on a physical observation. Moreover, it had intuitive appeal (the angle of rotation of co-ordinates may be an important criterion), and reduced the scale of the eigenvalue computation required for the Karhunen-Loève expansion. However, there is no reason why the rotation step should not be bypassed, and the original three components expanded as a series, to give a simpler model

$$\begin{bmatrix} x(n) \\ y(n) \\ z(n) \end{bmatrix} = \underline{A}(n) \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdot \\ \cdot \\ \alpha_M \end{bmatrix} + \begin{bmatrix} e_x(n) \\ e_x(n) \\ e_z(n) \end{bmatrix} \qquad (4.2)$$

This model has the advantage that all the observations are linear functions of the parameters α_i and

$$\underline{\alpha} = \sum_{n=1}^{N} \underline{A}'(n) \begin{bmatrix} x(n) \\ y(n) \\ z(n) \end{bmatrix}$$

Equation 4.2 suggests that standard linear filtering techniques (18) might be used to adaptively estimate the coefficients α_i directly from VCG recordings containing a train of heartbeats, without going through the process of filtering noise or producing an average cycle. The only difficulty is that the period length varies slightly from cycle to cycle and so each heartbeat would have to be detected and time-standardized relative to the R peak.

It may be possible to incorporate rhythm analysis into this scheme. A major change in waveform morphology would cause a sudden change in the filter residuals, and thus provide warning of an arrythmia. Cox et al (19), discussing rhythm analysis, suggest that "present rhythm classification potential appears strongly limited by the scope and quality of the extracted features." The completeness of the feature extraction in our case depends on the chosen dimension of the pattern vector, or the state of the model. Any deviation could be detected without a priori selection of criteria for abnormality. Methods which use standard clinical criteria, such as derivative thresholds in certain parts of the waveform appear very difficult to implement in practice.

5. The Pattern Recognition Problem: Discussion

5.1 Introduction

Once pattern vectors have been generated for the training data set of VCG's whose clinical categories are known, a decision rule can be formulated for classifying the data. This rule can then be used to classify new data. Three different approaches to the determination of this decision rule will be discussed briefly.

1. Determine from the training set a surface which partitions the pattern vector space into subspaces corresponding to different pathological categories.
2. Estimate the probability density functions of the data in each pathalogical category and use a decision rule based on determining for a given vector the most probably source category.
3. Group the training set data into clusters, hopefully corresponding to different pathalogical conditions. Assign an unclassified data point to that cluster which is closest, as determined by some suitable distance measure.

The question of which approach to use is influenced by the nature of the classification required (is a YES/NO type diagnosis appropriate?), and also by various factors which influence the scatter of the observations.

5.2 A Deterministic Scheme

The determination of a surface which separates data belonging to one or other of two categories, Normal or Abnormal, is in principle fairly straightforward. If the two sets are separable by a single hyperplane

or an ellipsoidal boundary, then finding a decision rule can be formulated as a linear-programming problem (20). If, as is more likely, such a simple separation of the sets is not possible, then an iteration of the same scheme can be used to find a sequence of several intersecting surfaces which divide the pattern vector space into Normal and Abnormal regions, whether or not the sets are connected. Extension to the case of diagnosis between different classes of abnormalities might be more difficult.

5.3 Probabilistic Schemes

As McFee and Baule have have pointed out (4), a fundamental problem in electrocardiogram diagnosis is cluster overlap. For example, there are too many normals that look as if they have Left Ventricular Hypertrophy, and too many subjects with LVH who look as if they were normal. McFee and Baule suggest that one solution to obtaining better diagnosis is to try to shrink the clusters, so that differences due to such purely physical sources of scatter as age, sex, body build and heart orientation are minimized. The change in VCG when the subject merely changes his position is quite striking. One approach to the problem may be to concentrate on choosing an optimum combination of leads. However, given the data at our disposal, it does seem that a probablistic approach to determination of a decision function may be more appropriate than a deterministic method.

Estimation of probability density functions can be approached either by assuming the form of the functions and estimating unknown para-

meters, or by assuming that the density functions can be expanded in terms of some known functions, and determining the unknown coefficients. For the second method we write:

The probability density function of the pattern vector $\underline{A}$ based on the condition that $\underline{A}$ is generated by a Normal heart;

$$p(\underline{A}/\text{Abnormal}) = B_0\Psi_0(\underline{A}) + B_1\Psi_1(\underline{A}) + \ldots.$$

The probability density function of the pattern vector $\underline{A}$ based on the condition that $\underline{A}$ is generated by an Abnormal heart;

$$p(\underline{A}/\text{Abnormal}) = C_0\Psi_0(\underline{A}) + C_1\Psi_1(\underline{A}) + \ldots.$$

where the Ψ_1 are known functions and B_i and C_i are to be determined from the training set data.

One choice for the Ψ_i is the gaussian density function. Training data for each category could be divided into several clusters, and the mean and variance of a gaussian density function estimated for each cluster. Specht suggests defining a gaussian density function about each point in the training set (21, 22), letting the covariance matrix be a unit matrix multiplied by some smoothing parameter. This smoothing parameter is chosen so that it optimizes the classification achieved using the resulting density functions.

A Bayes strategy can be used to determine the decision rule. This scheme compares the probabilities that a particular vector comes from the Normal set with the probability that it comes from the Abnormal set. The appropriate decision-threshold level depends on the a priori probabilities, and the weighting assigned to the probability of a false classification. It is determined by Bayes rule.

$$\frac{p(\text{Normal}/A)}{p(\text{Abnormal } \underline{A})} = \frac{p(A/\text{Normal})}{p(\underline{A}/\text{Abnormal})} \cdot \frac{p(\text{Normal})}{p(\text{Abnormal})}$$

For a simple classification scheme designed to screen out Normal data and select Questionable data for further diagnosis, it is more important to minimize the probability of a questionable VCG being classified Normal, than to minimize the probability of a Normal VCG being mis-classified. Unfortunately there is a trade-off between the safety factor that can be built into such a system, and the effectiveness with which Normal VCG's are identified.

5.4 Clusters and Distance Measures

Complicating the problem of making a correct diagnosis is the question of how far does the behavior of the VCG have to deviate from the normal shape for it to be classified as abnormal? Several workers have pointed out that rather than asking whether or not a patient has a particular abnormality, it is more appropriate to ask, "how much of this abnormality has he?" One approach to this problem of measuring the degree of an affliction might be to group different disorders into clusters in the pattern vector space, and by looking at appropriately defined distance measures, classify a VCG according to the cluster to which it most closely corresponds.

5.5 Clustering

The problems of cluster scatter and cluster overlap occur whichever of the three schemes are used. Non-linear mappings of the pattern

vectors may enhance cluster separability (15), but can not produce separation unless the original data was separable. It is now relevant to ask whether our method of feature selection leads to meaningful clustering at all. By ignoring explicit empirical diagnosis criteria we may have concealed some obvious diagnostic rules. For instance, suppose that the width of the QRS complex is a primary indicator of a certain heart abnormality; what form does this criterion take in the pattern vector space, when the pattern vector coefficients are derived from the Karhunen-Loève Expansion? A simple diagnostic rule may become considerably more complex when the data is transformed to another space. This scalar criterion may have become a complicated function of 15 of the pattern vector coefficients. Note, however, that if the model is good, then no diagnostic information has been lost. We can justify our representation on these grounds; at the possible expense of complicating the pattern recognition problem as far as some symptoms are concerned, we have retained additional information which may have been lost had features been selected on a heuristic basis. The technique of clustering, or unsupervised classification (15) could lead to finer discrimination between abnormalities than has been possible by visual scanning of many VCG waveforms.

5.6 Dimension of the Pattern Vector

In order to formulate a meaningful decision rule, the size of the training set should be large compared with the instrinsic dimensionality of the selecting features. This intrinsic dimensionality (15) is the

dimensionality of the data generating process, and it may be less than the dimension of the space chosen to represent the features. The question of what should be a minimum sample size is not easy in general. In our case we arranged for the coefficients of the pattern vector to be uncorrelated (apart from those representing the rotation matrix). If they were also normally distributed then they would be statistically independent. It has been suggested by workers who have assumed normally distributed features that the number of features should be no more than the square root of the number of training data sets available (13). We use this as a rule of thumb when deciding on the dimension of the pattern vector.

It is possible that the dimensionality of the pattern vector could be reduced by a non-linear mapping, because in using the Karhunen-Loève expansion, we restricted attention to linear mappings. Mucciardi and Gose investigated the problem of selecting subsets of Pattern recognition properties (23). Of the seven techniques which they considered, that based on performing a factor analysis and forming new properties corresponding to the eigenvectors ranked best (although they felt that computation requirements might be a practical disadvantage). Choosing properties which have the best mutual correlation was also shown to be important. These two findings are significant for our application, because the 20 pattern vector coefficients derived from the Karhunen-Loève expansion satisfy both these conditions.

6. Model Verification and Preliminary Analysis of Pattern Vectors

6.1 Introduction

This section presents some results which evaluate the validity of the model of the VCG. For this model the original three components were reduced to two by neglecting the energy in a direction perpendicular to the Principal Plane of the VCG, as described in section 3. These two components were then expanded using the vector form of the Karhunen-Loève Expansion.

The data used in the evaluation was a set of 93 Normal VCG's and 46 Abnormal VCG's. The data labelled Abnormal was in fact only "Questionable," because the cardiologist at the USAF School of Aerospace Medicine was primarily interested at this stage in a classification scheme which would duplicate his own preliminary screening procedure of separating obviously Normal VCG's from Questionable VCG's which warranted further analysis, even though they may turn out to be Normal. This is not important for the present aim of this study, which is representation of VCG's rather than classification. However, for convenience, the two classes of data were analyzed independently and labelled Normal and Abnormal, even though this distinction is not strictly accurate.

The basis functions for the expansion were estimated from the statistics of both the Normal and Abnormal data, the a priori probabilities being assigned to correspond to the number of data-sets available in each class.

Evaluation of the model is carried out primarily by analyzing the statistics of error criteria based on energy. For specific examples of the model performance, plots relating to two data-sets are displayed; one shows typical results obtained with a Normal VCG. The other is a selected "worst-case" (here, an abnormal VCG).

A preliminary analysis of the distributions of the pattern vector coefficients follows, in an attempt to determine whether any single coefficient leads to an obvious diagnosis criterion. However, detailed investigation of the classification problem is left for an extension of this work.

6.2 Validity of the Planar Assumption

Neglecting the VCG component perpendicular to the principal plane results in little loss of waveform energy. Figure 6.1 shows a histogram of the fractional energy in the non-planar component. For Normal data the mean is 1.25 percent, with a standard deviation of less than 1 percent. For Abnormal data the average energy is somewhat higher, 2.16 percent and the deviation is 2.13 percent, although for a particular data set the proportion goes as high as 11 percent. In Normal data the figure is never more than 6 percent.

The most important criterion, however, is how much detail is lost from the waveforms. This is rather subjective, and only extensive evaluations using many VCG's could tell whether the procedure is valid in general. Figure 6.2 shows for a typical Normal VCG a comparison of original xyz data with the same components reconsructed

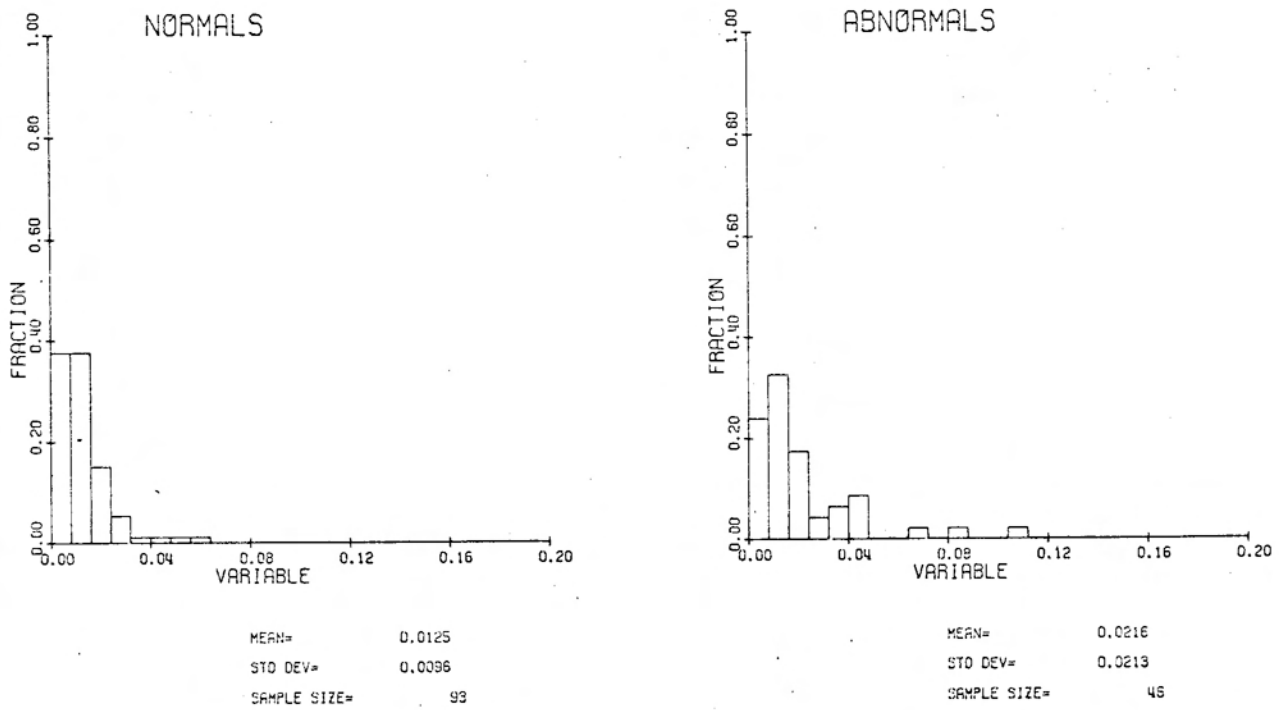


Figure 6.1 Histograms of Energy Fractions in the Component Normal to the Principal Plane

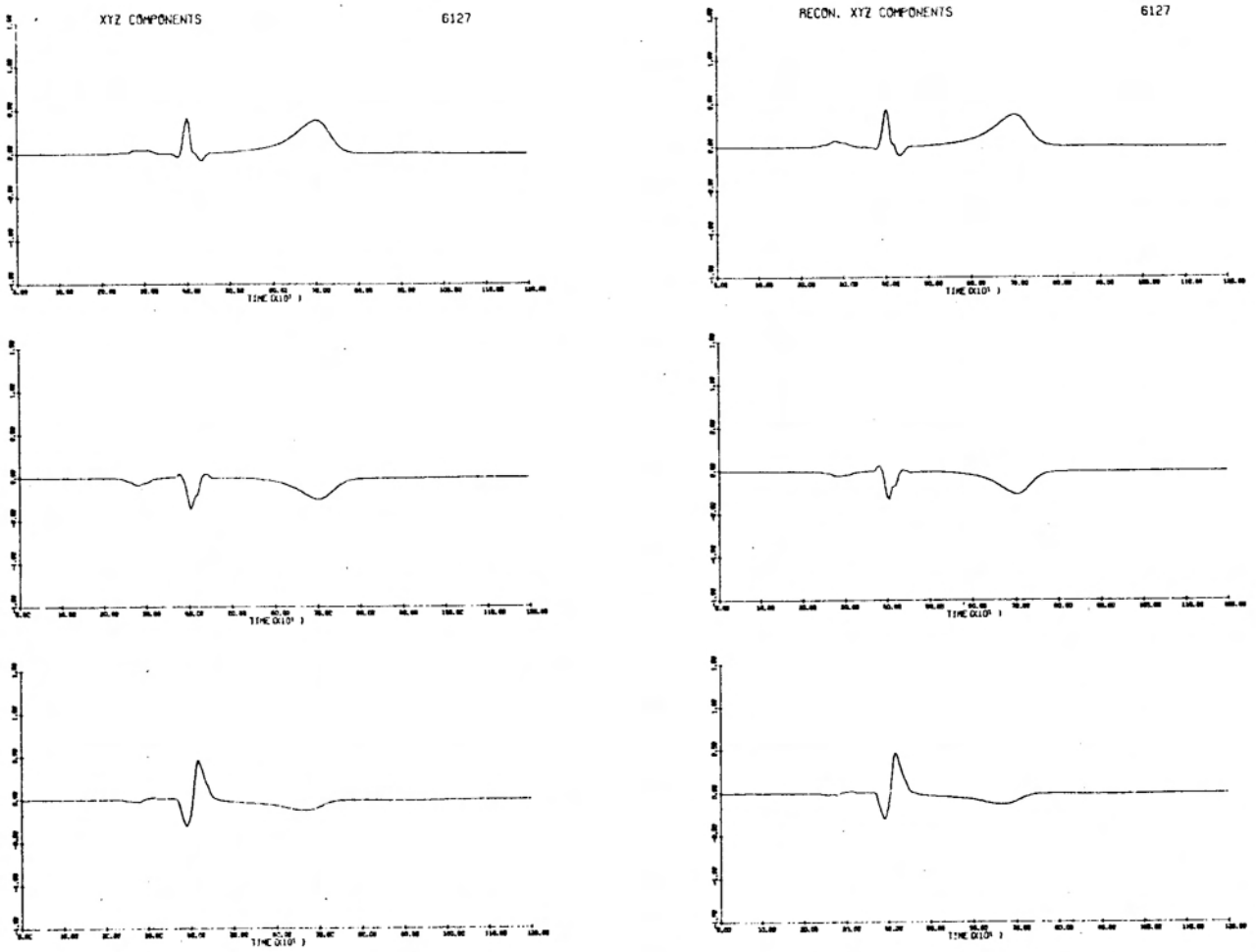
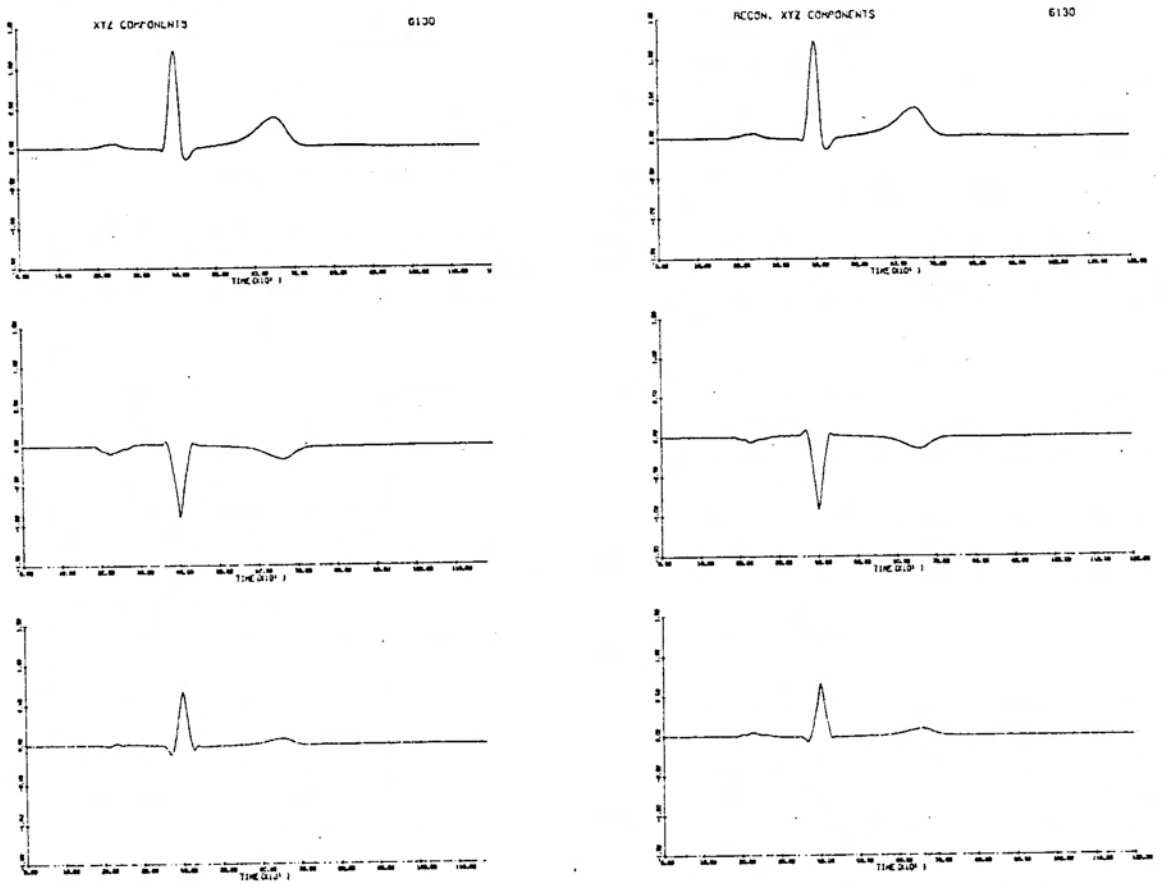


Figure 6.2 Effect of Neglecting Component v; Typical Data:
(a) xyz Components
(b) xyz Reconstructed from u and w



(a) (b)

Figure 6.3 Effect of Neglecting Component v; a Bad Case:
(a) xyz Components
(b) xyz Reconstructed from u and w

from the two planar components u and w. Figure 6.3 shows the same thing for a "worst-case." Here detail in the S wave has been lost, and the P wave amplitude has been reduced slightly.

6.3 Truncation of the Karhunen-Loève Expansion

Provided that the Karhunen-Loève basis functions are derived from sets of data which are representative of all types of VCG's to be encountered, the expansion should provide an approximation whose accuracy can be made as good as required by taking a sufficient number of terms. Figure 4.1 showed that on average, all but about 3 percent of energy can be accounted for by only 10 terms in the expansion, and 20 terms account for more than 99 percent of the total energy.

Figure 6.4 shows for Normal data how the error due to truncation of the expansion depends on the number of terms. The dashed line shows, for reference, the minimum error that would occur if the expansion produced a perfect fit to the waveform in the time range 200 mS to 800 mS. Note that the comparitively large residual error for patient 6306 is due almost entirely to the existence of a u wave, lying outside the expansion interval. If the approximating functions were extended to include the full time interval of 1200 mS, this error would be substantially reduced. Figure 6.5 shows the same result for Abnormal data. For both data sets, 20 terms, accounted on average for all but 0.5 percent of the total energy in the u and w components; the standard deviation was only 0.5 percent, and for Abnormal data, the worst case had only 1.4 percent error. Again,

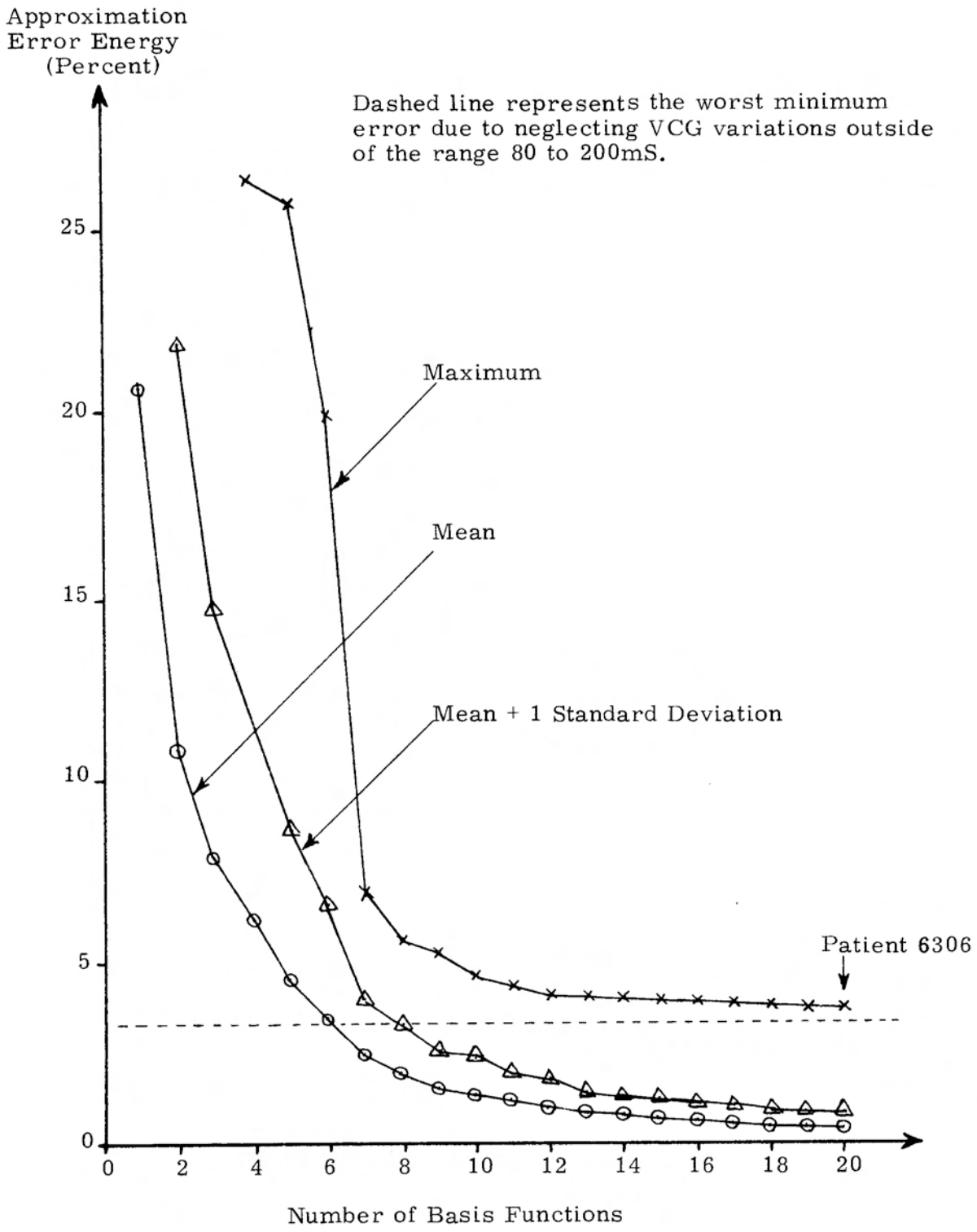


Figure 6.4 Karhunen-Loève Expansion: Mean and Maximum Error in Normal Data Vs. Number of Terms

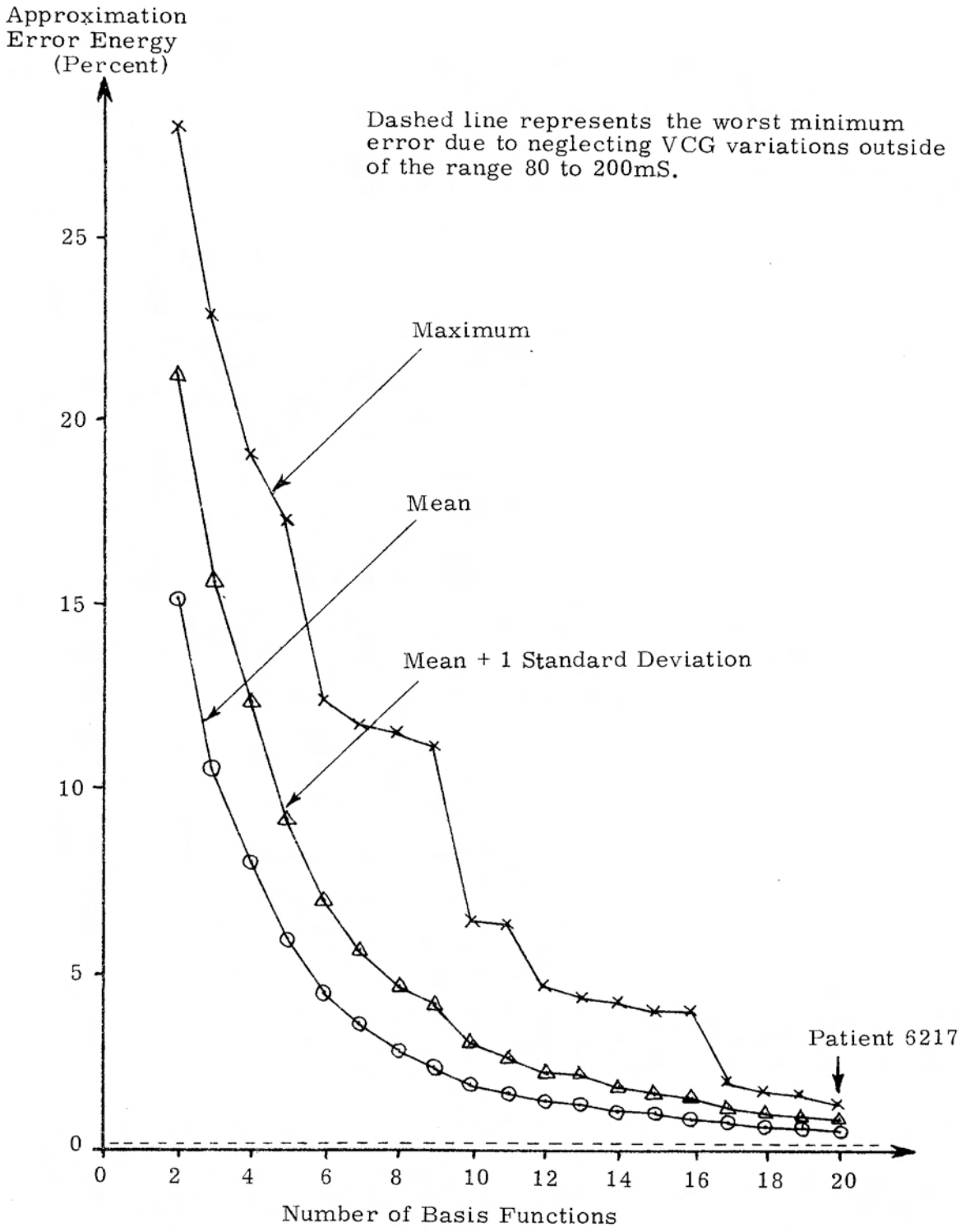


Figure 6.5 Karhunen-Loève Expansion: Mean and Maximum Error in Abnormal Data Vs. Number of Terms

the isolated larger error of Normal patient 6306 is due to the presence of the u wave outside the approximation interval.

Histograms of the expansion error with 20 terms are shown for Normal and Abnormal data in Figure 6.6. Apart from patient 6306 the data shows excellent reconstruction accuracy as measured by the energy criterion. Figure 6.7 shows a typical reconstruction of the u and w components. Figure 6.8 shows a selected worst-case. Here the reconstruction of the T wave is not very good. Features containing only little energy are not represented as well as those with large energy.

6.4 Original Components Reconstructed from Pattern Vectors

Figure 6.9 shows a typical Normal VCG reconstructed from a pattern vector of 24 terms, compared with the original data. The pattern vector includes 20 expansion coefficients together with three rotation parameters and the period. Subjectively the approximation appears to be very good on the whole. However, detailed comparisons by trained electrocardiographers will be necessary to determine whether significant information has been lost. An example of a "worst-case" reconstruction is shown in Figure 6.10.

6.5 Distribution of Pattern Vector Coefficients

Although a pattern recognition scheme must take advantage of the joint probability distributions of the pattern vectors, it is useful to

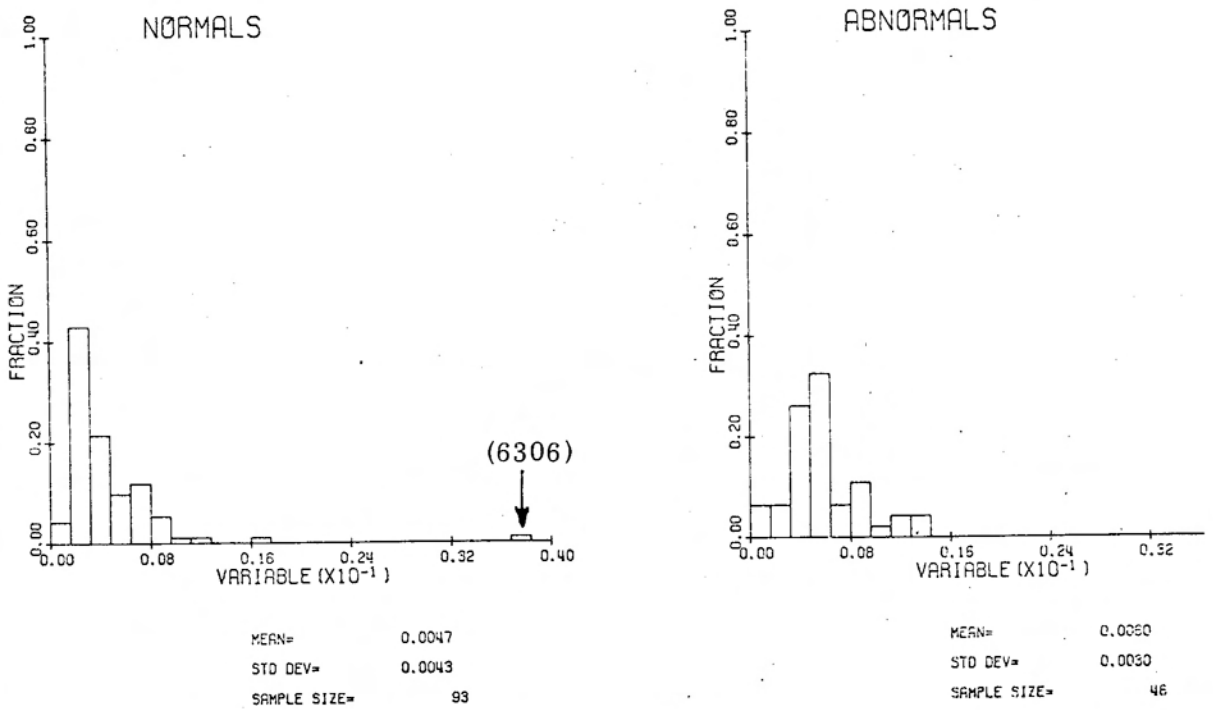


Figure 6.6 Karhunen-Loève Expansion: Histograms of Fractional Error Energy

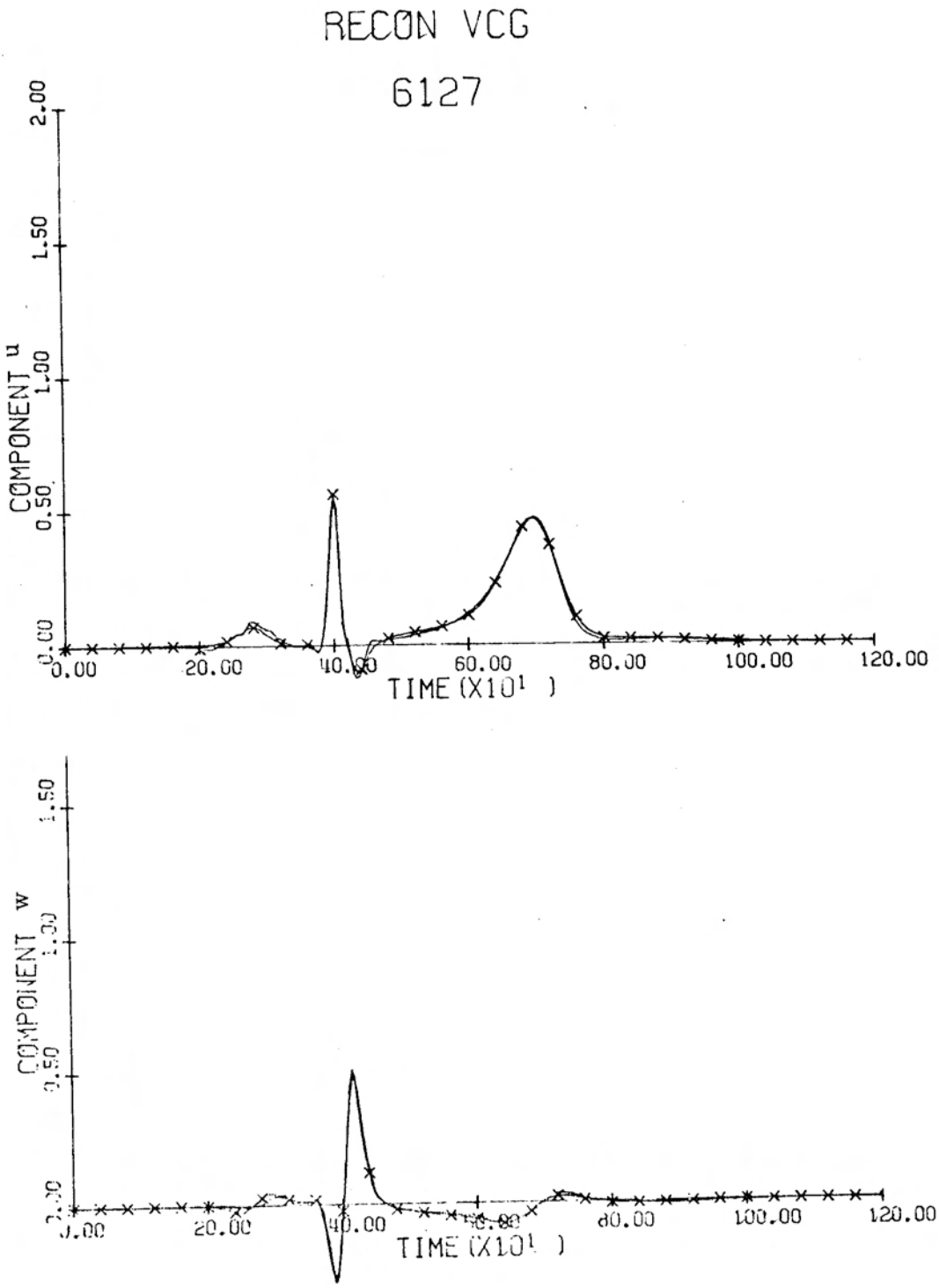


Figure 6.7 Approximation of u and w Components, Typical Normal Data

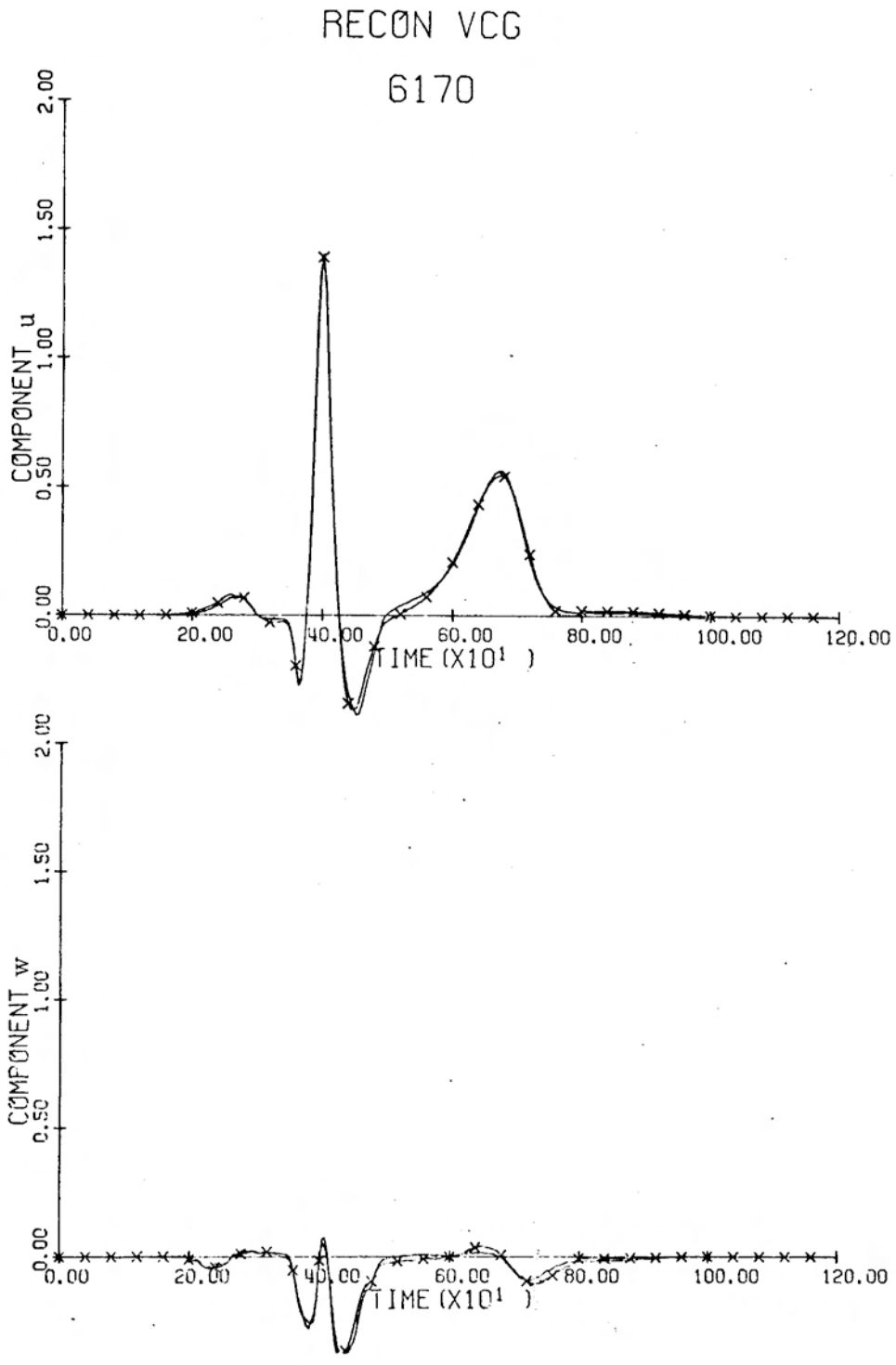


Figure 6.8 Approximation of u and w Components: An Abnormal VCG

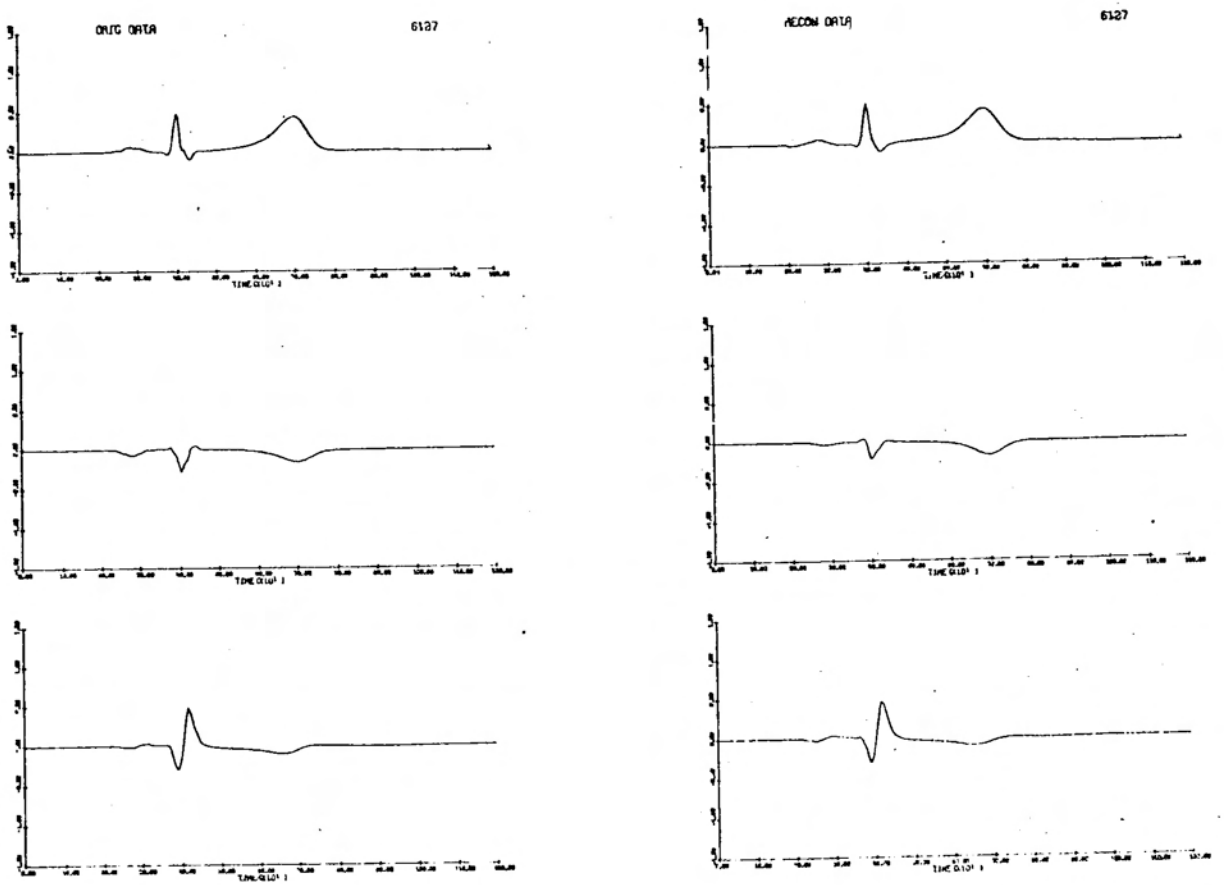


Figure 6.9 Overall Reconstruction from Pattern Vectors: Typical Normal Data

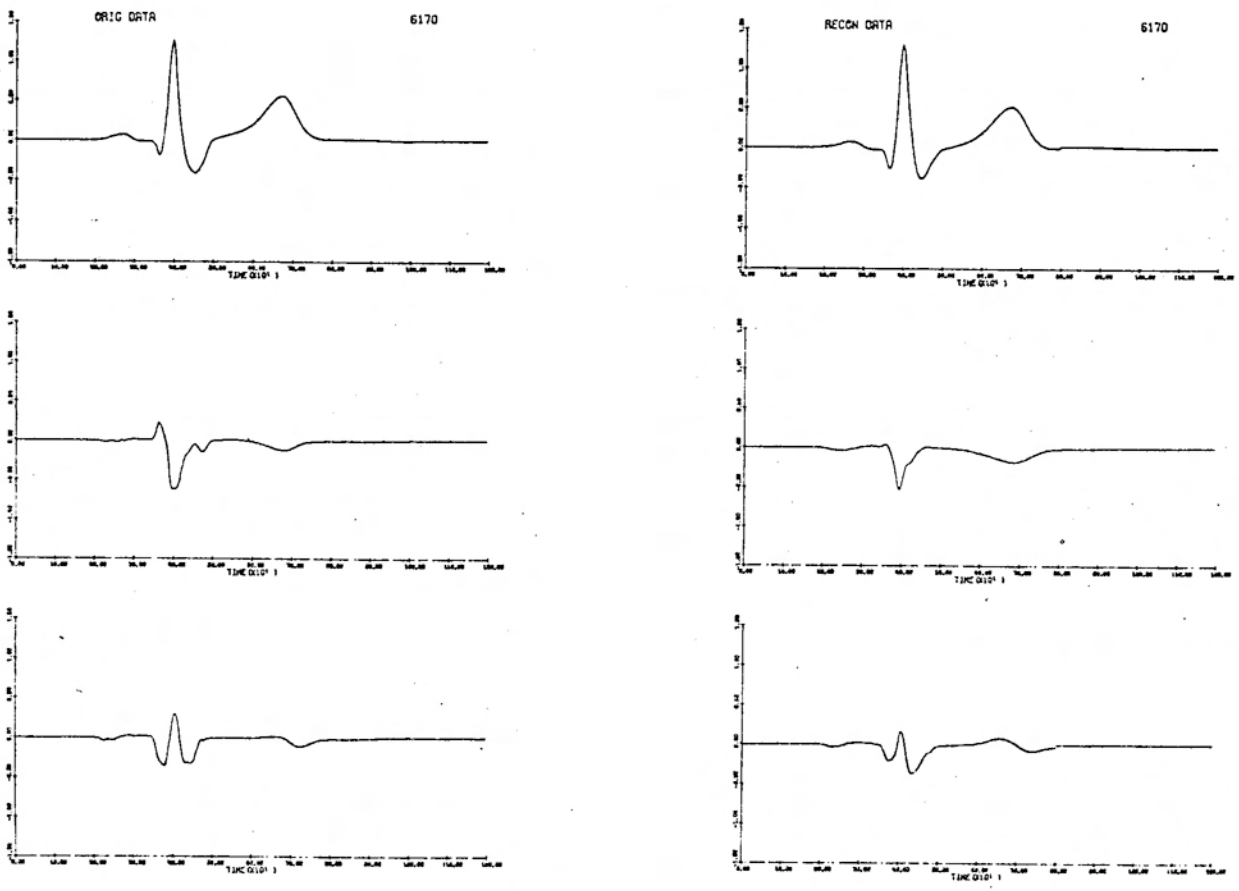


Figure 6.10 Overall Reconstruction from Pattern Vectors: A Worst Case (Abnormal Data)

plot histograms of the individual coefficients in order to determine whether any of them could be used as simple diagnostic criteria on their own. Distributions of some of the coefficients are shown plotted in Figures 6.11 to 6.15. The three rotation parameters were decomposed into an angle and three axis components. Many of these one dimensional distributions appear to be similar for both Normal and Abnormal data. The means of some of the parameters from Abnormal data do lie outside the range defined by the mean ± one standard deviation of the Normal data. However, the larger spread of the Abnormal data means that there is considerable overlap. The fact that the coefficients taken singly do not give very good separation is no immediate cause for alarm. Of their seven techniques for choosing subsets of pattern recognition properties, Mucciardi and Gose ranked last that technique which selected properties according to their expected probability of error when used separately (23). Recall also that some of the "Abnormal" data may in fact be Normal.

Two parameters which it may be worthwhile adding to the pattern vector are the out-of-plane energy, and the residual energy after approximation by the Karhunen-Loève expansion. Distributions of these quantities were shown in Figures 6.1 and 6.6.

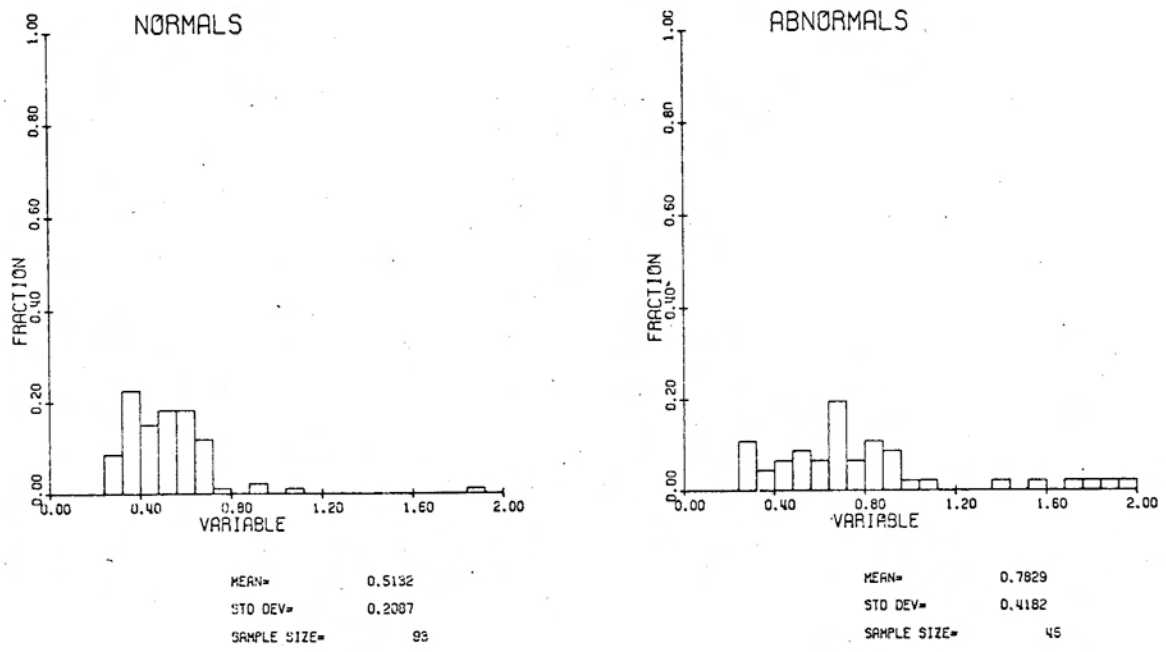


Figure 6.11 Histograms of Pattern Vector Coefficients: Angle of Rotation

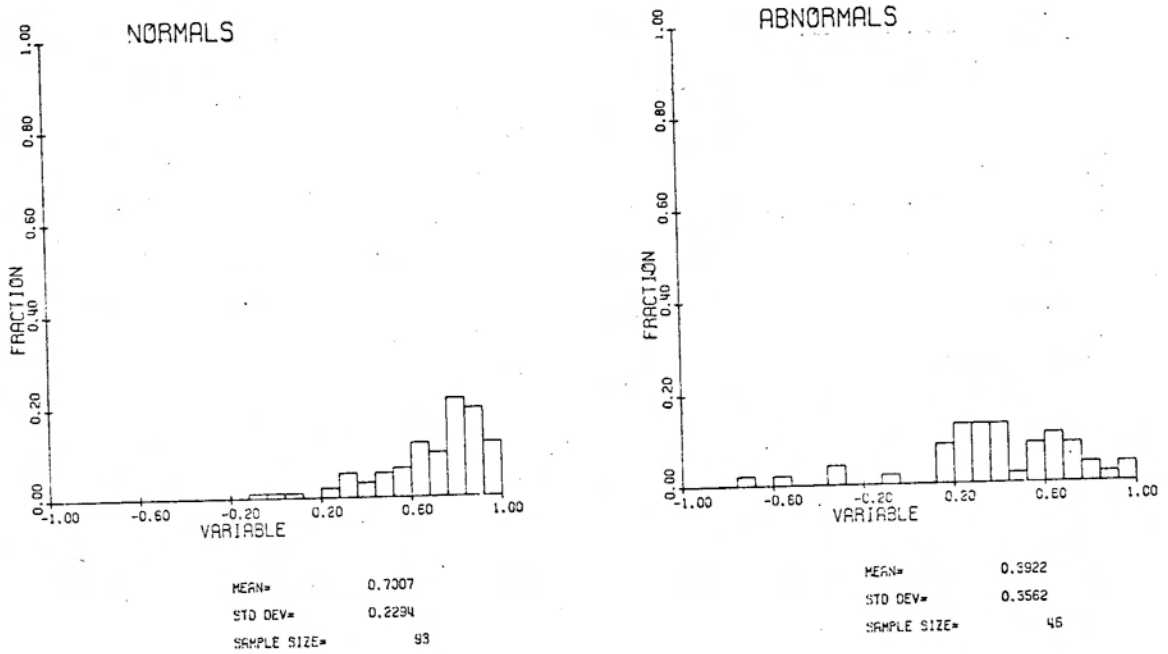


Figure 6.12 Histograms of Pattern Vector Coefficients: z Component of Rotation Axis

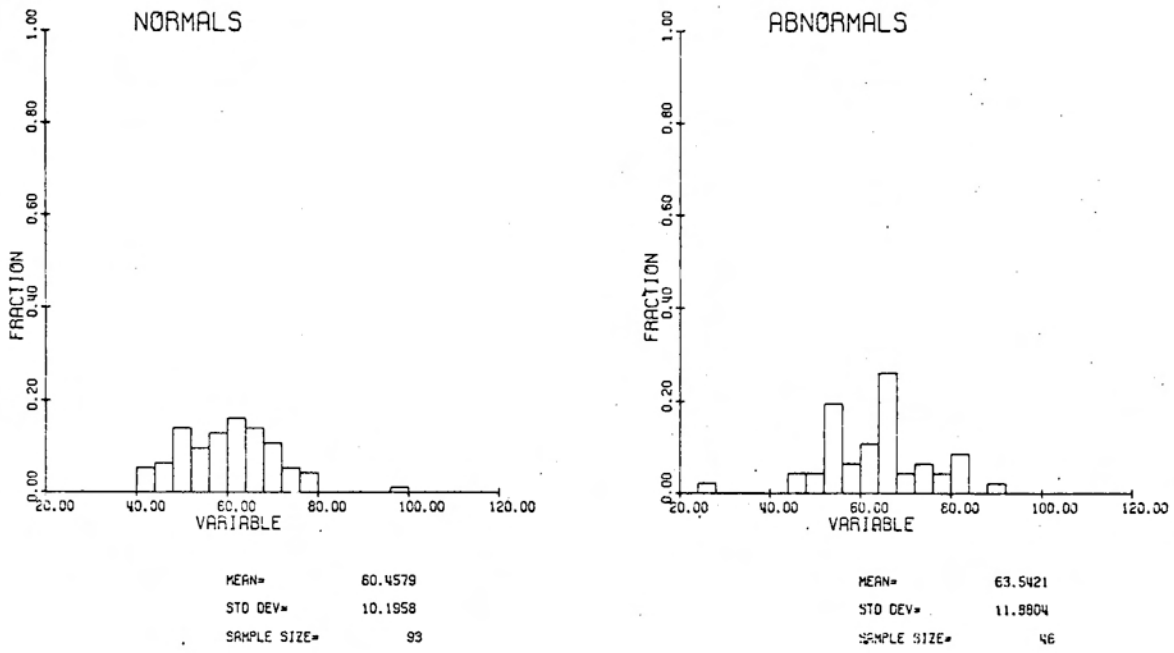


Figure 6.13 Histograms of Pattern Vector Coefficients: Waveform Period

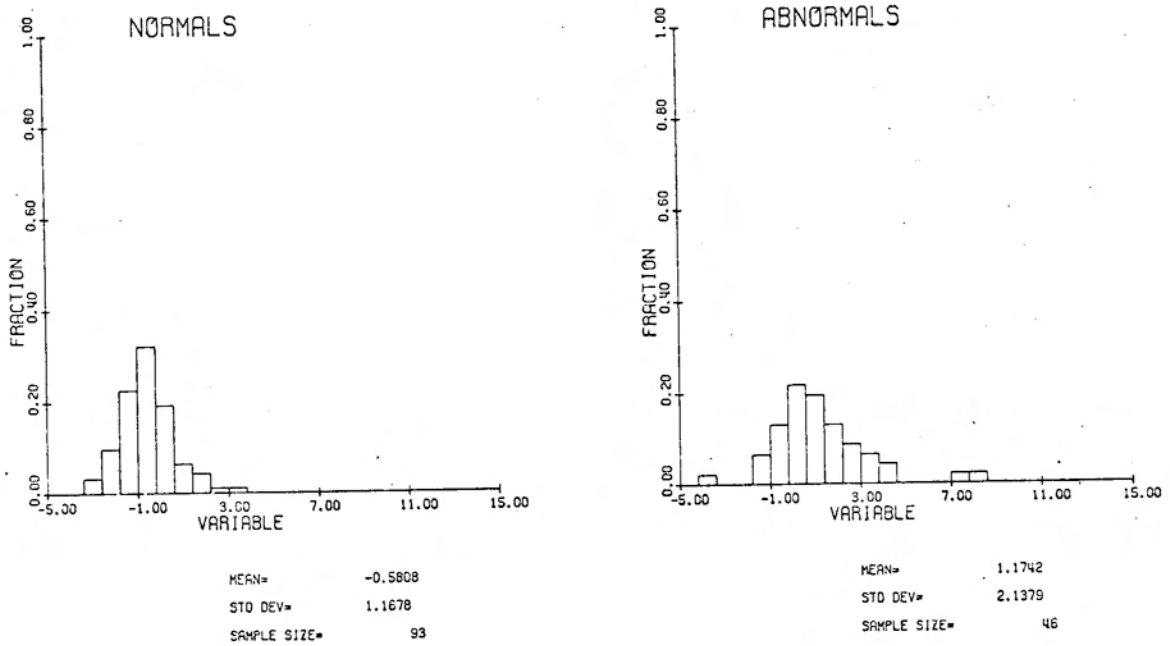


Figure 6.14 Histograms of Pattern Vector Coefficients: First Expansion Coefficient

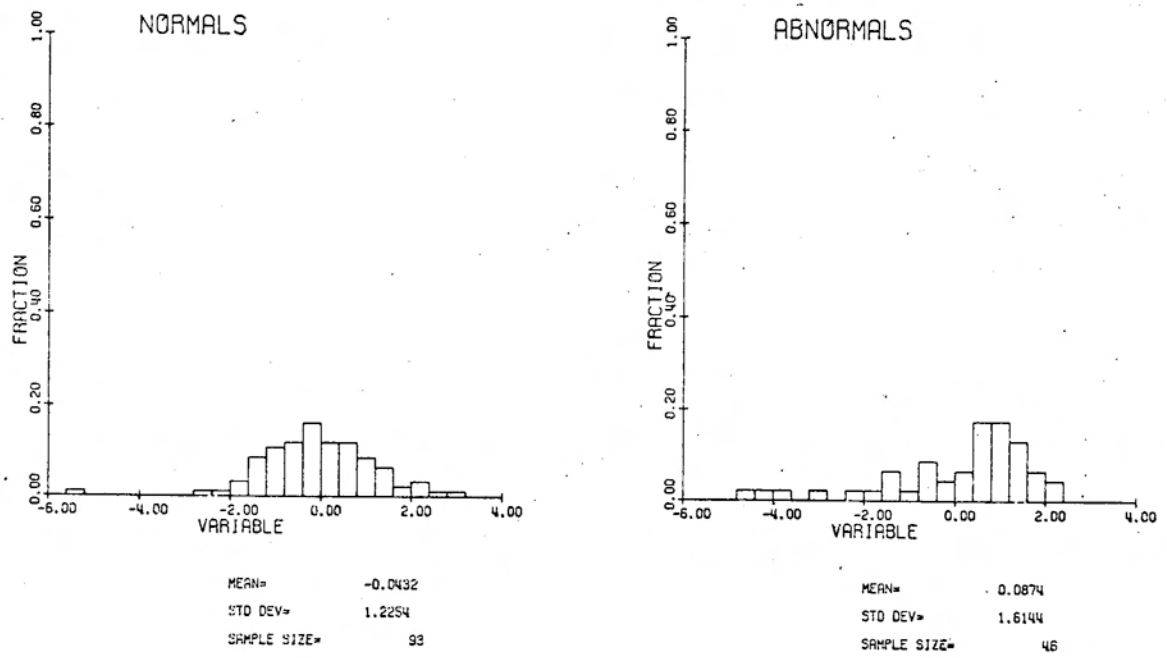


Figure 6.15 Histograms of Pattern Vector Coefficients: Second Expansion Coefficient

7. Conclusions and Recommendations

The vectorcardiogram has been modelled as a truncated series in terms of the dominant eigenfunctions of the covariance matrix. A vector version of the Karhunen-Loève Expansion was used to approximate the two orthogonal components of the heart vector which lie in the plane containing most of the energy of the VCG. For the 147 sets of VCG's available it has been verified that, as measured by an energy criterion, little waveform information is lost by neglecting the non-planar component. The three original waveforms can be reconstructed to good accuracy (again measured with respect to an energy criterion) from 24 parameters of the model.

This set of 24 parameters, the pattern vector, provides a compact means of storing VCG waveform morphologies; 24 numbers replace the 750 numbers required to store three components of a continuous waveform sampled at a 4mS rate.

The 24 pattern vector coefficients could form the basis of a scheme for classifying Normal and Abnormal hearts, based on statistical pattern recognition techniques. The development of a diagnostic scheme is suggested as a logical development of this work.

The model has been verified on a statistical basis, and many of the reconstructed waveforms have been compared by eye to the original components. Clearly, before there can be any confidence that a punched computer card could replace a magnetic tape as a patient's VCG record, some considerable evaluation work needs to be done, using a much larger data base, to ensure that small but diagnostically

important features are not lost as a result of the data reduction process. Further, it must be emphasized that a fairly restricted class of vectorcardiograms has been analyzed; all the data came from USAF flight personnel, who on the whole have a much sounder physique than the average patient in a hospital. Vectorcardiograms from patients with severe disorders, as well as recordings from children and exercise VCG's, may prove impossible to represent in this way. However, at the least, the results do appear to justify further investigation of the use of the model for the routine screening of VCG data.

This study has been concerned with the feasibility of the approach. For completeness it would be interesting to expand the three original components of the VCG directly, without going through the rotation step, as suggested in section 4.5. For a given number of terms, the representation accuracy should be substantially the same as for the original model. Further evaluation work on either model should use basis functions which extend over the full time range of the waveforms, so that the occasional u wave is not missed completely by the expansion.

As suggested in section 4.6, the modified form of the model lends itself to incorporation into a filter for raw VCG data. It would be interesting to consider using some of the techniques of modern estimation theory for filtering data, and for providing warning of arrhythmias.

(8) Mathematical Development

(8.1) The Discrete Karhunen-Loève Expansion

(8.1.1) Derivation

Let $x(n)$, $n = 1, 2, \ldots N$ be a discrete time random process. We would like to expand $x(n)$ in terms of some orthonormal functions $\Phi_i(n)$. N functions are sufficient to represent $x(n)$ exactly. Thus

$$x(n) = \sum_{i=1}^{N} \alpha_i \Phi_i(n) \tag{1}$$

Since the $\Phi_i(n)$ are orthogonal, we define

$$\sum_{n=1}^{N} \Phi_i(n) \Phi_j(n) = \delta_{ij} \tag{2}$$

Then

$$\alpha_i = \sum_{n=1}^{N} x(n) \Phi_i(n) \tag{3}$$

If the expansion is truncated after M terms:

$$x(n) = \sum_{i=1}^{M} \alpha_i \Phi_i(n) + \epsilon(n) \tag{4}$$

where

$$\epsilon(n) = \sum_{i=M+1}^{M} \alpha_i \Phi_i(n)$$

The expected value of the representation error $\epsilon(n)$ is zero if the α_i in the remainder are replaced by their expected values:

Let

$$b_i = E\{\alpha_i\} = \sum_{n=1}^{N} E\{x(n)\}\ \Phi_i(n)$$

Let

$$m(n) = E\{x(n)\} = \sum_{i=1}^{N} b_i \Phi_i(n)$$

Then

$$x(n) \simeq \sum_{i=1}^{M} \alpha_i \Phi_i(n) + \sum_{i=M+1}^{N} b_i \Phi_i(n)$$

$$= \sum_{i=1}^{M} \alpha_i \Phi_i(n) + \sum_{i=1}^{N} b_i \Phi_i(n) - \sum_{i=1}^{M} b_i \Phi_i(n)$$

$$= m(n) + \sum_{i=1}^{M} (\alpha_i - b_i)\Phi_i(n)$$

Or

$$x(n) \simeq m(n) + \sum_{i=1}^{M} \beta_i \Phi_i(n) \tag{6}$$

where

$$\beta_i = \sum_{n=1}^{N} [x(n) - m(n)]\Phi_i(n) \tag{7}$$

We are effectively expanding the deviation of x(n) from its expected value. We would like the coefficients β_i to be uncorrelated. Since β_i has zero mean:

$$E\{\beta_i \beta_j\} = \lambda_i \delta_{ij} \tag{8}$$

Substituting from (7) for β_i and β_j:

$$E\{\sum_{n=1}^{N} [x(n) - m(n)]\Phi_i(n) \sum_{u=1}^{N} [x(u) - m(u)]\Phi_j(u)\} = \lambda_i \delta_{ij}$$

Taking the expectation inside the summation, and defining the covariance of x(n):

$$\rho(n, u) = E\{[x(n) - m(n)] [x(u) - m(u)]\} \tag{9}$$

Then

$$\sum_{n=1}^{N} \Phi_i(n) \sum_{u=1} \rho(n, u) \Phi_{j(u)} = \lambda_i \delta_{ij} \tag{10}$$

Comparing (2) and (10), we see that

$$\sum_{u=1}^{N} \rho(n, u) \Phi_j(u) = \lambda_j \Phi_j(u) \tag{11}$$

If we define

$$\underline{X} = [x(1)\ x(2). \ . \ . \ x(N)]'$$

$$\underline{M} = [m(1)\ m(2). \ . \ . m(N)]'$$

$$\underline{\Phi}_j = [\Phi_j(1)\ \Phi_j(2). \ . \ . \Phi_j(N)]'$$

Then (11) can be written as a matrix equation:

$$\underline{R} \cdot \underline{\Phi}_j = \lambda_j \underline{\Phi}_j$$

Therefore $\underline{\Phi}_j$ is an eigenvector of $\underline{R}$ and λ_j is the corresponding eigenvalue.

The deviation from the mean, $\Delta\underline{X}$ can be written

$$\Delta\underline{X} = \underline{X} - \underline{M} = \sum_{i=1} \beta_i \underline{\Phi}_i$$

The mean energy in $\Delta\underline{X}$ is given by:

$$E\{\Delta\underline{X}'\Delta\underline{X}\} = E\{\sum_{j=1}^{N} \beta_j \underline{\Phi}_j' \sum_{i=1}^{N} \beta_i \underline{\Phi}_i\}$$

$$= \sum_{j=1}^{N} \sum_{i=1}^{N} \underline{\Phi}_j' \underline{\Phi}_i E\{\beta_i \beta_i\}$$

$$= \sum_{j=1}^{N} \lambda_j$$

Therefore the eigenvalues measure the expected energy along each orthogonal component. If we approximate $\underline{X}$ by M terms in the expansion, we should choose those terms which are associated with the M largest eigenvalues of the covariance matrix. If we define the matrix of eigenvectors

$$\underline{A} = [\underline{\Phi}_1 \ \Phi_2 \cdot \ \cdot \ \Phi_N]$$

Then the Karhunen-Loève Expansion can be represented as an orthogonal transformation,

$$\underline{X} = \underline{A}.\underline{\alpha} \qquad \text{where } \underline{\alpha} = (\alpha_1, \ \alpha_2 \ \cdot \ \cdot \ \alpha_N)'$$

It corresponds to rotating the co-ordinate axes to a system defined by the principal axes of the covariance ellipsoid of $\underline{X}$.

(8.1.2) Optimality

The assumption that the coefficients β_i are uncorrelated (equation 8) is not strictly necessary, but it facilitates the derivation.

Fukunaga (15) shows that in addition to giving uncorrelated coefficients, the choice of the eigenvectors of $\underline{R}$ as the basis functions is optimum with respect to a variety of criteria:

Over the set of all orthonormal basis functions, if the expansion is truncated after $M(<N)$ terms, then the eigenvectors of $\underline{R}$:

1. Minimize the mean square approximation error
2. Maximize the population entropy defined by
   $h = -E\{\ln p(\underline{\alpha})\}$
3. Are optimum with respect to a scatter criterion, the expected value of squared between-sample distances, $\overline{d}^2_{\alpha} = E\{||\alpha_i - \alpha_j||^2\}$ where α_i and α_j are mutually independent sample vectors taken from a single distribution.

Chien and Fu (17) show that the Karhunen-Loève Expansion generalizes to the case where $\underline{X}$ can be a sample from one of several different random processes. In this case, if p_i is the probability of occurence of one of k stochastic processes, then we define the covariance function

$$\underline{K} = \sum_{i=1}^{k} p_i E\{(\underline{X} - \underline{M})(\underline{X} - \underline{M})'\}$$

The expansion is still optimum with respect to the mean square approximation error, and a measure of the population entropy.

(8.1.3) Karhunen-Loève Expansion for a Vector Process

Consider the two component vector process

$$\underline{X}(n) = \begin{bmatrix} x_1(n) \\ x_2(n) \end{bmatrix}$$

Assume zero mean for simplicity.

X(n) can be expanded in terms of orthonormal basis functions $\underline{\Phi}_i(n)$ as a series with scalar coefficients

$$\underline{X}(n) = \sum_{n=1}^{N} \alpha_i \, \underline{\Phi}_i(n)$$

where

$$\sum_{n=1}^{N} \underline{\Phi}_i'(n) \, \underline{\Phi}_j(n) = \delta_{ij} \text{ and } \underline{\Phi}_i(n) = \begin{bmatrix} \phi_{i1}(n) \\ \phi_{i2}(n) \end{bmatrix}$$

$$\alpha_i = \sum_{n=1}^{N} \underline{\Phi}_i'(n) \, \underline{X}(n)$$

Following a derivation similar to the scalar case, we obtain

$$\lambda_i \, \underline{\Phi}_i(n) = \sum_{n=1}^{N} \underline{R}(n, u) \, \underline{\Phi}_i(u)$$

where

$$\underline{R}(n, u) = \begin{bmatrix} R_{11}(n, u) & R_{12}(n, u) \\ R_{21}(n, u) & R_{22}(n, u) \end{bmatrix} = E\{\underline{X}(n) \, \underline{X}'(u)\}$$

Then

$$\lambda_i \begin{bmatrix} \phi_{i1}(n) \\ \phi_{i2}(n) \end{bmatrix} = \sum_{u=1}^{N} \begin{bmatrix} R_{11}(n, u) & R_{12}(n, u) \\ R_{21}(n, u) & R_{22}(n, u) \end{bmatrix} \begin{bmatrix} \phi_{i1}(u) \\ \phi_{i2}(u) \end{bmatrix}$$

If we now define $\underline{K}_{ij} = \begin{bmatrix} R_{ij}(1,1) & R_{ij}(1,2) & \cdots & R_{ij}(1,N) \\ R_{ij}(2,1) & & & \\ R_{ij}(N,1) & & & R_{ij}(N,N) \end{bmatrix}$

and

$$\underline{\Phi}_{ij} = [\phi_{ij}(1) \quad \phi_{ij}(2) \quad \cdots \quad \phi_{ij}(N)]'$$

Then

$$\lambda_i \begin{bmatrix} \underline{\Phi}_{i1} \\ \underline{\Phi}_{i2} \end{bmatrix} = \begin{bmatrix} \underline{K}_{11} & \underline{K}_{12} \\ \underline{K}_{21} & \underline{K}_{22} \end{bmatrix} \begin{bmatrix} \underline{\Phi}_{i1} \\ \underline{\Phi}_{i2} \end{bmatrix}$$

Therefore the vector expansion is equivalent to forming the vector

$$\underline{Y} = [x_1(1) \; x_1(2) \; \cdots \; x_1(N) \vdots x_2(1) \; x_2(2) \; \cdots \; x_2(N)]'$$

and carrying out the conventional expansion as before. Obviously the derivation extends to a process which has more than two components.

(8.1.4) Stationary Periodic Processes: Relationship to the Discrete Fourier Transform

For a wide sense stationary process, the covariance matrix becomes (in the scalar case):

$$\underline{R} = \begin{bmatrix} 1 & \rho(1) & \rho(2) & \cdots & \rho(N-1) \\ \rho(1) & 1 & \rho(1) & & \rho(N-2) \\ \rho(2) & \rho(1) & 1 & & \rho(N-3) \\ \vdots & & & & \\ \rho(N-1) & \rho(N-2) & & & 1 \end{bmatrix}$$

For a Periodic Process, $\underline{X}(n) = \underline{X}(N+n)$, and

$$\underline{R} = \begin{bmatrix} 1 & \rho(1) & \rho(2) & \cdots & \rho(N-1) \\ \rho(N-1) & 1 & \rho(1) & & \rho(N-2) \\ \vdots & & & & \\ \rho(1) & \rho(2) & & & 1 \end{bmatrix} \qquad (1)$$

The determinant $|\underline{R} - \lambda \underline{I}|$ is a circulant (24)

And

$$|\underline{R} - \lambda \underline{I}| = \prod_{k=1}^{N} \left[-\lambda_k + \sum_{i=1}^{N} \rho(i)\, e^{j \frac{2\Pi k}{N}} \right]$$

So that

$$\lambda_k = \sum_{i=1}^{N} \rho(i)\, e^{j \frac{2\Pi ki}{N}}$$

Since $\rho(i)$ is an even function, λ_k is real and we can write:

$$\lambda_k = \sum_{i=1}^{N} \rho(i)\, e^{- \frac{2\Pi ki}{N}} \qquad (2)$$

Therefore λ_k is the DFT of $\rho(i)$

Suppose that $N = 2n$. Then λ_N and $\lambda_{N/2}$ are distinct and the remaining roots occur in pairs with $\lambda_k = \lambda_{N-k}$

The eigenvectors are

$$\underline{\Phi}_k = \frac{1}{\sqrt{N}} \left[\cos \frac{2\Pi k}{N} \quad \cos \frac{4\Pi k}{N} \quad - - - \quad \cos 2\Pi k \right]'$$

or

$$\Phi_k(n) = \frac{1}{\sqrt{N}} \cos \frac{2\Pi nk}{N}$$

and

$$\Phi_{N-k}(n) = \frac{1}{\sqrt{N}} \sin \frac{2\Pi nk}{N}$$

The expansion is:

$$x(n) = \frac{1}{N} \sum_{k=1}^{N} U(k) \cos \frac{2\Pi nk}{N} +$$

$$\frac{1}{N} \sum_{k=1}^{N} V(k) \sin \frac{2\Pi nk}{N} \qquad (3)$$

where

$$U(k) = \sum_{n=1}^{N} x(n) \cos \frac{2\Pi nk}{N} \qquad (4)$$

$$V(k) = \sum_{n=1}^{N} x(n) \sin \frac{2\Pi nk}{N} \qquad (5)$$

Now consider the DFT

$$\text{Let } F(k) \longleftrightarrow x(n)$$

Then

$$F(k) = \sum_{n=1}^{N} x(n)\, e^{\frac{-j2\Pi kn}{N}}$$

$$x(n) = \frac{1}{N} \sum_{k=1}^{N} F(k)\, e^{\frac{j2\Pi kn}{N}}$$

If x(n) is real, then Re F(k) is symmetric and Im F(k) is antisymmetric

$\therefore$ If $$F(k) = F_R(k) - j F_I(k)$$

Then

$$x(n) = \frac{1}{N} \sum_{k=1}^{N} F_R(k) \cos \frac{2\Pi kn}{N} + \frac{1}{N} \sum_{k=1}^{N} F_I(k) \sin \frac{2\Pi kn}{N} \quad (6)$$

where

$$F_R(k) = \sum_{n=1}^{N} x(n) \cos \frac{2\Pi kn}{N} \quad (7)$$

$$F_I(k) = \sum_{n=1}^{N} x(n) \sin \frac{2\Pi kn}{N} \quad (8)$$

Hence comparing (3) to (5) and (6) to (8), we can identify

$$U(k) \equiv F_R(k), \quad V(k) \equiv F_I(k)$$

Note that there are only N (not 2N) independent coefficients, because

$$U(k) = U(n-k), \quad V(k) = -V(N-k)$$

Therefore for a periodic, stationary process, the Karhunen-Loève expansion reduces to the Discrete Fourier Transform. The eigenvalues are the DFT of the covariance function.

As the sampling rate becomes large, the eigenvalue distribution approaches the power spectral density function of the waveform.

(8.2) Defining the Principal Directions of the Vectorcardiogram

(8.2.1) Derivation

Let the instantaneous vector be represented by

$$\underline{x}(n) = \begin{bmatrix} x_1(n) \\ x_2(n) \\ x_3(n) \end{bmatrix} \qquad n = 1, 2 \ldots N$$

Define

$$\langle x_i \, x_j \rangle = \sum_{n=1}^{N} x_i(n) \, x_j(n)$$

And

$$\underline{S}_x = \begin{bmatrix} \langle x_1^2 \rangle & \langle x_1 x_2 \rangle & \langle x_1 x_3 \rangle \\ \langle x_2 x_1 \rangle & \langle x_2^2 \rangle & \langle x_2 x_3 \rangle \\ \langle x_3 x_1 \rangle & \langle x_3 x_2 \rangle & \langle x_3^2 \rangle \end{bmatrix}$$

$$= \sum_{n=1}^{N} \underline{X}(n)\, \underline{X}'(n)$$

Define the new co-ordinate system by an orthogonal transformation matrix $\underline{A}$, and let the components of the heart vector in this new co-ordinate system be

$$\underline{Y}(n) = \begin{bmatrix} y_1(n) \\ y_2(n) \\ y_3(n) \end{bmatrix}$$

Then

$$\underline{Y}(n) = \underline{A}\, \underline{X}(n)$$

Now we note that the diagonal elements of $\underline{S}_x$ give the energy of the VCG along the three axis directions, and further, that the total energy is given by the sum of the diagonal terms, i.e. total energy in VCG = tr $(\underline{S}_x)$. The trace of a matrix is invariant under an orthogonal transformation. Therefore if we define

$$\underline{S}_y \quad \begin{bmatrix} \langle y_1^2 \rangle & \langle y_1 y_2 \rangle & \langle y_1 y_3 \rangle \\ \langle y_2 y_1 \rangle & \langle y_2^2 \rangle & \langle y_2 y_3 \rangle \\ \langle y_3 y_1 \rangle & \langle y_3 y_2 \rangle & \langle y_3^2\ 1 \rangle \end{bmatrix} = \sum_{n=1}^{N} \underline{Y}(n)\, \underline{Y}'(n)$$

Then Total Energy = $\mathrm{tr}(\underline{S}_x) = \mathrm{tr}(\underline{S}_y)$ We define the Principal Directions so that the off-diagonal terms of $\underline{S}_y$ are zero

$$\underline{S}_y = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \quad \underline{\Delta} \ \underline{\lambda}$$

$$\therefore \underline{\lambda} = \sum_{n=1}^{N} \underline{y}(n)\, \underline{y}'(n) = \sum_{n=1}^{N} \underline{A}\, \underline{X}(n)\, \underline{X}'(n)\, \underline{A}'$$

$$= \underline{A}\, \underline{S}_x\, \underline{A}'$$

$$\therefore \underline{A}\, \underline{S}_x = \underline{\lambda}\, \underline{A}$$

Or since $\underline{S}_x$ is symmetric,

$$\underline{S}_x \cdot \underline{A}' = \underline{\lambda} \cdot \underline{A}'$$

Therefore $\underline{A}$ is the transpose of the matrix of Right-Hand Eigenvectors of $\underline{S}_x$.

Each eigenvalue measures the total energy of the VCG along the direction defined by the corresponding eigenvector. If, as is indeed the case, one of the eigenvalues is very small compared to the other two, then the energy contributed by the corresponding component of $\underline{y}(n)$ is very small. ie. if $\lambda_1 \ll \lambda_2, \lambda_3$ then the VCG is defined to a good approximation by the two components $y_2(n)$ and $y_3(n)$.

(8.2.2) Representation of a Rotation Matrix

A convenient way to represent a rotation matrix is to find the axis about which the dotation occurs, and the angle of rotation about that axis.

Let $\underline{U}$ be the axis of rotation (unit vector)

θ be the angle of rotation

$\underline{R}$ be the rotation matrix

Let $\underline{Y} = \underline{R} . \underline{X}$ define a rotation

1. Examine the plane defined by the rotation of the vector $\underline{X}$ in Fig. 8.1

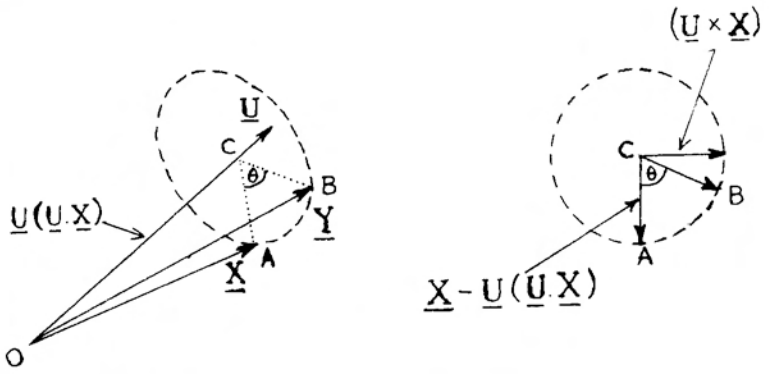


Figure 8.1 Rotation Axis

It is convenient to express the rotated vector Y in terms of an orthogonal co-ordinate system defined by the rotation axis $\underline{U}$, and two perpendicular vectors in the plane, for which we will choose CA and a line perpendicular to CA.

Now

$$\underline{Y} = \underline{OC} + \underline{CB}$$

And

$$\underline{OC} = \underline{U}(\underline{U} . \underline{X})$$

$\underline{CA} = \underline{X} - \underline{U}(\underline{U} . \underline{X})$, and $(\underline{U} \times \underline{X})$ is perpendicular to $\underline{CA}$ and has the same length

$$\therefore \underline{CB} = [\underline{X} - \underline{U}(\underline{U} . \underline{X}] \cos \theta + (\underline{U} \times \underline{X}) \sin \theta$$

$$\therefore \underline{Y} = \underline{R} . \underline{X} = \underline{U}(\underline{U} . \underline{V}) + \cos \theta [\underline{X} - \underline{U}(\underline{U} . \underline{X})] \cos \theta + (\underline{U} \times \underline{X}) \sin \theta$$

$$\therefore \underline{R} = \cos \theta \underline{I} + (1 - \cos \theta) \underline{U}\underline{U}' + \sin \theta (\underline{U} \times) \qquad (1)$$

To find $\underline{U}$ and θ in terms of $\underline{R}$ we make use of the fact that the first two terms in (1) are symmetric, and the last is skew-symmetric. Note that $(\underline{U} \times)$ can be written

$$\begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}$$

2. Find the trace of both sides of (1)

$$\text{tr } \underline{R} = 3 \cos \theta + (1 - \cos \theta) [u_1^2 + u_2^2 + u_3^2]$$

Therefore, since $\underline{U}$ is a unit vector: -

$$\cos \theta = \frac{1}{2} (\text{tr } \underline{R} - 1) \qquad (2)$$

3. Let

$$\underline{R} = \underline{R}_S + \underline{R}_A$$

where

$$\underline{R}_S = \frac{1}{2}\left[\underline{R} + \underline{R}'\right]$$

$$\underline{R}_A = \frac{1}{2}\left[\underline{R} - \underline{R}'\right]$$

Find the skew symmetric part of both sides of (1)

$$\underline{R}_A = 0 + 0 + \sin\theta\,[\underline{U}\times]$$

$$= \sin\theta \begin{bmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{bmatrix}$$

$$\therefore\ u_1 \sin\theta = -R_{A23}$$

$$u_2 \sin\theta = R_{A13}$$

$$u_3 \sin\theta = -R_{A12}$$

$$\therefore\ \sin\theta = \sqrt{R_{A23}^2 + R_{A13}^2 + R_{A12}^2} \qquad 0 \le \theta \le 180^\circ \qquad (3)$$

And

$$\underline{U} = \frac{1}{\sin\theta} \begin{bmatrix} -R_{A23} \\ R_{A13} \\ -R_{A12} \end{bmatrix} \qquad (4)$$

9. Appendix

Calculation of Part of the Eigensystem of a Large Matrix

If only a small number, say less than 25 percent, of the eigenvalues of a large matrix are required, it is very inefficient to use one of the standard algorithms which compute all of the eigenvalues. However, Wilkinson and Reinsch (25) give procedures, originally programmed in Algol, which find for a symmetric matrix all of the eigenvalues which lie within a given range. The eigenvalue computation is carried out after reducing the original matrix to tridiagonal form by an orthogonal transformation. Eigenvectors are calculated for the tridiagonal matrix by inverse iteration, and then back-transformed to eigenvectors of the original symmetric matrix.

The appropriate procedures have been translated into Fortran, and correspond to the Algol subroutines as follows:

Algol routine	Fortran routine
Tred 3	STRI1
Tristurm	TRIST
Sturmcnt	STM
Trbak 3	SBK

The algorithms are quite complicated, and reference should be made to (25) for details.

The Fortran subroutines are given at the end of this Appendix. A brief explanation of each is given below.

STRI1

Reduces a real symmetric N x N matrix to tridiagonal form by an orthogonal transformation. The lower triangle of the original matrix

is stored by rows in a one dimensional N(N + 1)/2 array in order to minimize storage requirement. The procedure overwrites these elements with details of the transformation which are required by the back-transformation SBK. The diagonal and subdiagonal of the reduced matrix are stored in 1 x N arrays.

TRIST

Calculates for a symmetric tridiagonal matrix the eigenvalues which lie between a given upper bound and a lower bound, and finds the corresponding normalized eigenvectors. If any of the roots are pathologically close, the corresponding eigenvectors are orthogonalized. If the tridiagonal matrix can be split into several submatrices, eigenvectors corresponding to each submatrix are found independently. Thus eigenvectors from different submatrices are exactly orthogonal.

TRIST uses function subroutine STM.

STM

Calculates the number of eigenvalues less than a given lower bound in a submatrix of a symmetric tridiagonal matrix. STM is used by subroutine TRIST to isolate the eigenvalues that lie between two bounding values.

STM can also be used to calculate by iteration the lower bound required by TRIST if a specific number of the largest eigenvalues is wanted. Note that a convenient upper bound for TRIST is the sum of the squares of the diagonal terms (of either the original or the reduced matrix).

SBK

Back transformation of eigenvectors of the symmetric tridiagonal matrix to eigenvectors of the original symmetric matrix. Uses the orthogonal transformation information produced by STRI1.

BASIS

Main subroutine for finding a given number of basis functions from a covariance matrix. Calculates and punches eigenvalues and eigenvectors. Uses subroutines STRI1, TRIST, STM and SBK.

```
      SUBROUTINE STRI1 (NORDER,A,DIAG,SUBDIA)                          00005190
C                                                                      00005200
      DOUBLE PRECISION DIAG,SUBDIA,RELTOL,F,G,H                        00005210
      DOUBLE PRECISION DSIGN,DSQRT                                     00005220
      INTEGER NDIM,NORDER,I,II,NP1,IM1,K,J,JP1                         00005230
      DIMENSION A(1),DIAG(NORDER),SUBDIA(NORDER)                       00005240
      DATA RELTOL/2.44D-63/                                            00005250
C                                                                      00005251
C TRANSFORMS A SYMMETRIC MATRIX A TO TRIDIAGONAL FORM.                 00005252
C TRANSFORMATION INFO IS STORED IN A ON OUTPUT.                        00005253
C A(I,J) IS STORED AS A(I*(I-1)/2+J).                                  00005254
C                                                                      00005260
C SAVE DIAGONAL ELEMENTS.                                              00005270
      DO 10 I=1,NORDER                                                 00005280
   10 DIAG(I)=A(I*(I+1)/2)                                             00005290
C                                                                      00005300
C PERFORM REDUCTIONS IN REVERSE ORDER OF SUBSCRIPTS.                   00005310
      NP1=NORDER+1                                                     00005320
      DO 120 II=1,NORDER                                               00005330
      I=NP1-II                                                         00005340
      KI=I*(I-1)/2                                                     00005341
      H=0.0D0                                                          00005350
C                                                                      00005360
      IM1=I-1                                                          00005370
      IF(IM1.LE.0) GO TO 30                                            00005380
      DO 20 K=1,IM1                                                    00005390
   20 H=H+A(KI+K)**2                                                   00005400
C                                                                      00005410
C IF H IS TOO SMALL, UNDERFLOWS MAY JEOPARDIZE ORTHOGONALITY,          00005420
C SO SKIP TRANSFORMATION.                                              00005430
   30 IF(H.GT.RELTOL) GO TO 40                                         00005440
      SUBDIA(I)=0.0D0                                                  00005450
      GO TO 110                                                        00005460
C                                                                      00005470
   40 F=A(KI+IM1)                                                      00005480
      G=-DSIGN(DSQRT(H),F)                                             00005490
      SUBDIA(I)=G                                                      00005500
```

```
      H=H-F*G                                                          00005510
      A(KI+IM1)=F-G                                                    00005520
C                                                                      00005530
      F=0.0D0                                                          00005540
      DO 80 J=1,IM1                                                    00005550
      KJ=J*(J-1)/2                                                     00005551
C                                                                      00005560
C FORM ELEMENT OF A*U                                                  00005570
      G=0.0D0                                                          00005580
      DO 50 K=1,J                                                      00005590
   50 G=G+A(KJ+K)*A(KI+K)                                              00005600
C                                                                      00005610
      JP1=J+1                                                          00005620
      IF(IM1.LT.JP1) GO TO 70                                          00005630
      DO 60 K=JP1,IM1                                                  00005640
      KK=K*(K-1)/2                                                     00005641
   60 G=G+A(KK+J)*A(KI+K)                                              00005650
C                                                                      00005660
C FORM ELEMENT OF P                                                    00005670
   70 G=G/H                                                            00005680
      SUBDIA(J)=G                                                      00005690
      F=F+G*A(KI+J)                                                    00005700
   80 CONTINUE                                                         00005710
C                                                                      00005720
      H=F/(2.0D0*H)                                                    00005730
C                                                                      00005740
C FORM REDUCED A                                                       00005750
      DO 100 J=1,IM1                                                   00005760
      F=A(KI+J)                                                        00005770
      G=SUBDIA(J)-H*F                                                  00005780
      SUBDIA(J)=G                                                      00005790
C                                                                      00005800
      KJ=J*(J-1)/2                                                     00005801
      DO 90 K=1,J                                                      00005810
   90 A(KJ+K)=A(KJ+K)-F*SUBDIA(K)-G*A(KI+K)                            00005820
C                                                                      00005830
  100 CONTINUE                                                         00005840
```

```
C                                                                       00005850
  110 H=DIAG(I)                                                         00005860
      ID=I*(I+1)/2                                                      00005861
      DIAG(I)=A(ID)                                                     00005870
      A(ID)=H                                                           00005880
  120 CONTINUE                                                          00005890
C                                                                       00005900
      RETURN                                                            00005910
      END                                                               00005920
```

```
      SUBROUTINE TRIST(N,LB,UB,C,B,BETA,M,ROOT,VEC,CNT,*)                00002110
                                                                         00002120
C                                                                        00002130
      INTEGER I,P,Q,R,M1,M2,N,M,CNT(M),FAIL,STM,J,K,S,ITS,GRP,NUM,       00002140
     1 INT(300),MDUM,KDUM,IDM1,IDM2,PDUM,QDUM,PDM2,L1,L2                 00002150
      REAL*8 C(N),B(N),VEC(N,M),BETA(N),LB,UB,DABS,DSQRT,ROOT(M),MA,     00002160
     1 X(300),WU(300),Z(300),D(300),E(300),F(300),Y(300),                00002170
     1 EPS1,EPS2,EPS3,EPS4,X1,X0,XU,U,V,BI,NORM                          00002180
      EXTERNAL STM                                                       00002190
C                                                                        00002200
C C CONTAINS DIAGONAL TERMS, B THE SUB-DIAGONAL TERMS, BETA THE SQUARES  00002210
C OF THE SUB-DIAG TERMS OF THE SYMMETRIC TRIDIAG MATRIX, ORDER N         00002220
C WE REQUIRE THE EIGENVALUES LESS THAN UB AND GREATER THAN LB.           00002230
C EIGENVALUES IN ROOT( ), EIGENVECTORS IN VEC( , ),STORED COLUMNWISE.    00002240
C PROCEDURE FAILS IF M ON ENTRY IS LESS THAN THE NUMBER OF               00002250
C EIGENVALUES IN THE RANGE. A CALL WITH M=-1 DETERMINES THIS NUMBER.     00002260
C ON EXIT M GIVES THE NUMBER OF EIGENVALUES ACTUALLY FOUND.              00002270
C NUMBER OF ITERATIONS PER VECTOR IS STORED IN CNT( ). PROCEDURE         00002280
C FAILS IF 5 ITERATIONS IS NOT ENOUGH. THEN CNT=6.                       00002290
C MA IS THE RELATIVE MACHINE PRECISION. EPS1 IS THE ERROR TO BE          00002300
C TOLERATED IN THE SMALLEST EIGENVALUE.                                  00002310
C PROCEDURE REPLACES NEGLIGIBLE ELEMENTS OF B BY ZERO.                   00002314
C                                                                        00002315
C DIMENSIONS ALLOW FOR MATRICES UP TO ORDER 300.                         00002320
C                                                                        00002330
C THIS SUBROUTINE USES FUNCTION SUBPROGRAM STM                           00002340
      MA=2.22D-16                                                        00002350
  200 EPS1=(DABS(LB)+DABS(UB))*MA                                        00002360
C                                                                        00002370
C LOOK FOR SMALL SUB-DIAGONAL ENTRIES                                    00002380
C                                                                        00002390
  201 DO 202 I=2,N                                                       00002400
      IF(DABS(B(I))-MA*DABS(C(I-1))) 203,203,202                         00002410
  203 B(I)=0.                                                            00002420
      BETA(I)=0.                                                         00002430
  202 CONTINUE                                                           00002440
C                                                                        00002450
C CALCULATE THE NUMBER OF EIGENVALUES REQUIRED. FAIL IF MORE THAN M (IN)
```

```
C                                                                       00002460
  204 R=M                                                               00002470
      M=STM(1,N,C,B,BETA,UB,N)-STM(1,N,C,B,BETA,LB,N)                   00002480
      IF(M-R) 205,205,206                                               00002490
  206 RETURN 1                                                          00002500
C                                                                       00002510
C LOOK FOR INDEPENDANT SUB-MATRICES                                     00002520
C                                                                       00002530
  205 Q=0                                                               00002540
      R=1                                                               00002550
  250 P=Q+1                                                             00002560
      NDUM=N-1                                                          00002570
      IF(NDUM-P) 209,208,208                                            00002580
  208 DO 207 Q=P,NDUM                                                   00002590
      IF(DABS(B(Q+1))) 207,300,207                                      00002600
  207 CONTINUE                                                          00002610
C                                                                       00002620
  209 Q=N                                                               00002630
C                                                                       00002640
C LOOK FOR ISOLATED ROOT ON DIAGONAL                                    00002650
C                                                                       00002660
  300 IF(P-Q) 409,310,409                                               00002670
C                                                                       00002680
C ISOLATED ROOT                                                         00002690
C                                                                       00002700
  310 IF(LB-C(P)) 311,311,312                                           00002710
  311 IF(C(P)-UB) 313,312,312                                           00002720
  313 DO 314 I=1,N                                                      00002730
  314 VEC(I,R)=0.                                                       00002740
      ROOT(R)=C(P)                                                      00002750
      VEC(P,R)=1.                                                       00002760
      CNT(R)=0                                                          00002770
      R=R+1                                                             00002780
  312 GO TO 1200                                                        00002790
C                                                                       00002800
C NUMBER OF ROOTS IN MATRIX P TO Q                                      00002810
C                                                                       00002820
```

```
  409 M1=STM(P,Q,C,B,BETA,LB,N)+1                                          00002830
      M2=STM(P,Q,C,B,BETA,UB,N)                                            00002840
      IF(M1-M2) 415,415,1200                                               00002850
  415 CONTINUE                                                             00002860
C                                                                          00002870
C FIND ROOTS BY BISECTION                                                  00002880
C                                                                          00002890
      X0=UB                                                                00002900
      DO 416 I=M1,M2                                                       00002910
      X(I)=UB                                                              00002920
  416 WU(I)=LB                                                             00002930
C                                                                          00002940
C LOOP FOR KTH EIGENVALUE                                                  00002950
C                                                                          00002960
      MDUM=M2-M1+1                                                         00002970
      DO 400 KDUM=1,MDUM                                                   00002980
      K=M2-KDUM+1                                                          00002990
      XU=LB                                                                00003000
      IDM2=K-M1+1                                                          00003010
      DO 420 IDM1=1,IDM2                                                   00003020
      I=K-IDM1+1                                                           00003030
      IF(XU-WU(I)) 419,420,420                                             00003040
  419 XU=WU(I)                                                             00003050
      GO TO 402                                                            00003060
  420 CONTINUE                                                             00003070
  402 IF(X0-X(K)) 421,421,422                                              00003080
  422 X0=X(K)                                                              00003090
  421 IF(X0-XU-2.0*MA*(DABS(XU)+DABS(X0))-EPS1) 423,423,424                 00003100
  424 X1=(XU+X0)*0.5                                                       00003110
      S=STM(P,Q,C,B,BETA,X1,N)                                             00003120
      IF(S-K) 425,426,426                                                  00003130
  425 IF(S-M1) 427,428,428                                                 00003140
  427 XU=X1                                                                00003150
      WU(M1)=X1                                                            00003160
      GO TO 421                                                            00003170
  428 XU=X1                                                                00003180
      WU(S+1)=X1                                                           00003190
```

```
      IF(X(S)-X1)  429,429,430                                          00003200
  430 X(S)=X1                                                           00003210
  429 GO TO 421                                                         00003220
  426 X0=X1                                                             00003230
      GO TO 421                                                         00003240
  423 X(K)=(X0+XU)*0.5                                                  00003250
  400 CONTINUE                                                          00003260
C                                                                       00003270
C FIND VECTORS BY INVERSE ITERATION                                     00003280
C                                                                       00003290
      NORM=DABS(C(P))                                                   00003300
      PDUM=P+1                                                          00003310
      DO 500 I=PDUM,Q                                                   00003320
  500 NORM=NORM+DABS(C(I))+DABS(B(I))                                   00003330
C                                                                       00003340
C EPS2 IS THE CRITERION FOR GROUPING                                    00003350
C EPS3 REPLACES ZERO PIVOTS & EQUAL ROOTS ARE MODIFIED BY EPS3          00003360
C EPS4 IS TAKEN VERY SMALL TO AVOID OVERFLOW                            00003370
C                                                                       00003380
      EPS2=NORM*0.001                                                   00003390
      EPS3=MA*NORM                                                      00003400
      EPS4=EPS3*(Q-P+1)                                                 00003410
      GRP=0                                                             00003420
      S=P                                                               00003430
      DO 1150 K=M1,M2                                                   00003440
      ITS=1                                                             00003450
      ROOT(R)=X(K)                                                      00003460
      X1=X(K)                                                           00003470
C                                                                       00003480
C LOOK FOR CLOSE OR COINCIDENT ROOTS                                    00003490
C                                                                       00003500
      IF(K-M1) 501,502,501                                              00003510
  501 IF(X1-X0-EPS2)  503,504,504                                       00003520
  503 GRP=GRP+1                                                         00003530
      GO TO 505                                                         00003540
  504 GRP=0                                                             00003550
  505 IF(X1-X0) 506,506,502                                             00003560
```

```
  506 X1=X0+EPS3                                                        00003570
C                                                                       00003580
  502 U=EPS4/SQRT(Q-P+1.)                                               00003590
      DO 507 I=P,Q                                                      00003600
  507 Z(I)=U                                                            00003610
C                                                                       00003620
C ELIMINATION WITH INTERCHANGES                                         00003630
C                                                                       00003640
      U=C(P)-X1                                                         00003650
      V=B(P+1)                                                          00003660
      DO 708 I=PDUM,Q                                                   00003670
      BI=B(I)                                                           00003680
      IF(DABS(BI)-DABS(U)) 709,710,710                                  00003690
  710 INT(I)=1                                                          00003700
      Y(I)=U/BI                                                         00003710
      XU=Y(I)                                                           00003720
      D(I-1)=BI                                                         00003730
      E(I-1)=C(I)-X1                                                    00003740
      IF(I-Q) 711,712,711                                               00003750
  711 F(I-1)=B(I+1)                                                     00003760
      GO TO 713                                                         00003770
  712 F(I-1)=0.0                                                        00003780
  713 U=V-XU*E(I-1)                                                     00003790
      V=-XU*F(I-1)                                                      00003800
      GO TO 708                                                         00003810
  709 INT(I)=0                                                          00003820
      Y(I)=BI/U                                                         00003830
      XU=Y(I)                                                           00003840
      D(I-1)=U                                                          00003850
      E(I-1)=V                                                          00003860
      F(I-1)=0.0                                                        00003870
      U=C(I)-X1-XU*V                                                    00003880
      IF(I-Q) 714,708,714                                               00003890
  714 V=B(I+1)                                                          00003900
  708 CONTINUE                                                          00003910
      IF(U) 715,716,715                                                 00003920
  715 D(Q)=U                                                            00003930
```

```
      GO TO 717                                                          00003940
  716 D(Q)=EPS3                                                          00003950
  717 E(Q)=0.0                                                           00003960
      F(Q)=0.0                                                           00003970
C                                                                        00003980
C BACK-SUBSTITUTION                                                      00003990
C                                                                        00004000
  850 IDM1=Q-P+1                                                         00004010
      DO 800 IDM2=1,IDM1                                                 00004020
      I=Q-IDM2+1                                                         00004030
      Z(I)=(Z(I)-U*E(I)-V*F(I))/D(I)                                     00004040
      V=U                                                                00004050
      U=Z(I)                                                             00004060
  800 CONTINUE                                                           00004070
C                                                                        00004080
C ORTHOGONALISE WITH RESPECT TO PREVIOUS MEMBERS OF GROUP                00004090
C                                                                        00004100
      L1=R-GRP                                                           00004110
      L2=R-1                                                             00004120
      IF(L2-L1) 904,950,950                                              00004130
  950 DO 900 J=L1,L2                                                     00004140
      XU=0.0                                                             00004150
      DO 901 I=P,Q                                                       00004160
  901 XU=XU+Z(I)*VEC(I,J)                                                00004170
      DO 902 I=P,Q                                                       00004180
  902 Z(I)=Z(I)-XU*VEC(I,J)                                              00004190
  900 CONTINUE                                                           00004200
  904 NORM=0.                                                            00004210
      DO 903 I=P,Q                                                       00004220
  903 NORM=NORM+DABS(Z(I))                                               00004230
C                                                                        00004240
C FORWARD SUBSTITUTION                                                   00004250
C                                                                        00004260
      IF(NORM-1.0) 1001,1002,1002                                        00004270
 1001 IF(ITS-5) 1003,1004,1003                                           00004280
 1004 CNT(R)=6                                                           00004290
      RETURN 1                                                           00004300
```

```
 1003 IF(NORM)  1005,1006,1005                                          00004310
 1006 Z(S)=EPS4                                                         00004320
      IF(S-Q)  1007,1008,1007                                           00004330
 1007 S=S+1                                                             00004340
      GO TO 1010                                                        00004350
 1008 S=P                                                               00004360
      GO TO 1010                                                        00004370
 1005 XU=EPS4/NORM                                                      00004380
      DO 1011 I=P,Q                                                     00004390
      Z(I)=Z(I)*XU                                                      00004400
 1011 CONTINUE                                                          00004410
 1010 DO 1012 I=PDUM,Q                                                  00004420
      IF(INT(I)-1)  1015,1016,1015                                      00004430
 1016 U=Z(I-1)                                                          00004440
      Z(I-1)=Z(I)                                                       00004450
      Z(I)=U-Y(I)*Z(I)                                                  00004460
      GO TO 1012                                                        00004470
 1015 Z(I)=Z(I)-Y(I)*Z(I-1)                                             00004480
 1012 CONTINUE                                                          00004490
      ITS=ITS+1                                                         00004500
      GO TO 850                                                         00004510
C                                                                       00004520
C NORMALISE SO THAT SUM OF SQUARES IS 1,& EXPAND TO FULL ORDER          00004530
C                                                                       00004540
 1002 U=0.0                                                             00004550
      DO 1101 I=P,Q                                                     00004560
 1101 U=U+Z(I)**2                                                       00004570
      XU=1.0/DSQRT(U)                                                   00004580
      DO 1102 I=P,Q                                                     00004590
 1102 VEC(I,R)=Z(I)*XU                                                  00004600
      PDM2=P-1                                                          00004610
```

```
      IF(PDM2-1) 1111,1110,1110                                  00004620
1110  DO 1103 IDM1=1,PDM2                                        00004630
      I=P-IDM1                                                   00004640
1103  VEC(I,R)=0.0                                               00004650
1111  QDUM=Q+1                                                   00004660
      IF(N-QDUM) 1112,1113,1113                                  00004670
1113  DO 1104 I=QDUM,N                                           00004680
1104  VEC(I,R)=0.0                                               00004690
1112  CNT(R)=ITS                                                 00004700
      R=R+1                                                      00004710
      X0=X1                                                      00004720
1150  CONTINUE                                                   00004730
1200  IF(Q.LT.N) GO TO 250                                       00004740
      RETURN                                                     00004750
      END                                                        00004760
```

```
      INTEGER FUNCTION STM(P,Q,D,E,F,LAMBDA,N)                          00004770
      INTEGER P,Q,CNT,STM                                               00004780
      REAL*8 D(N),E(N),F(N),LAMBDA,MA,X                                 00004790
C                                                                       00004800
C STM GIVES THE NUMBER OF EIGENVALUES LESS THAN LAMBDA IN THE SUB-MATRIX00004810
C OF THE SYMM TRIDIAG MATRIX DEFINED BY D,E,F.                          00004820
C D IS THE DIAGONAL, E IS THE SUBDIAGONAL.                              00004822
C F CONTAINS THE SQUARES OF THE TERMS IN E.                             00004824
C                                                                       00004830
      MA=2.22D-16                                                       00004840
      CNT=0                                                             00004850
      X=1.0                                                             00004860
      DO 105 I=P,Q                                                      00004870
      IF(X) 101,102,101                                                 00004880
  102 X=D(I)-LAMBDA-DABS(E(I)/MA)                                       00004890
      GO TO 103                                                         00004900
  101 X=D(I)-LAMBDA-F(I)/X                                              00004910
  103 IF(X) 104,105,105                                                 00004920
  104 CNT=CNT+1                                                         00004930
  105 CONTINUE                                                          00004940
      STM=CNT                                                           00004950
      RETURN                                                            00004960
      END                                                               00004970
```

```
      SUBROUTINE SBK(N,A,E,M,Z)                                          00004980
      INTEGER N,M,I,J,K,L                                                00004990
      REAL*8 E(N),Z(N,M),H,S                                             00005000
      REAL*4 A(1)                                                        00005010
C                                                                        00005011
C TRANSFORMS THE EIGENVECTORS IN Z BACK TO THE ORIGINAL COORD SYSTEM,     00005012
C USING THE TRANSFORMATION INFO (A) PRODUCED BY STRI1. E CONTAINS THE    00005013
C SUBDIAGONAL OF THE TRIDIAGONAL MATRIX PRODUCED BY STRI1                00005014
C                                                                        00005020
      DO 2 I=2,N                                                         00005030
      KI=I*(I-1)/2                                                       00005031
      IF(E(I)) 4,2,4                                                     00005040
    4 L=I-1                                                              00005050
      H=E(I)*A(KI+L)                                                     00005060
      DO 3 J=1,M                                                         00005070
      S=0.                                                               00005080
      DO 5 K=1,L                                                         00005090
    5 S=S+A(KI+K)*Z(K,J)                                                 00005100
      S=S/H                                                              00005110
      DO 6 K=1,L                                                         00005120
    6 Z(K,J)=Z(K,J)+S*A(KI+K)                                            00005130
    3 CONTINUE                                                           00005140
    2 CONTINUE                                                           00005150
C                                                                        00005160
      RETURN                                                             00005170
      END                                                                00005180
```

```
      SUBROUTINE BASIS(N,LL,R,ROOT,VEC,*)                                00000010
      REAL*8 ROOT(LL),VEC(N,LL),CABS,LB,UB,OUB,OLB                       00000020
      REAL*8 DIAG(300),SUB(300),SUBDIA(300),BETA(300)                    00000030
      REAL*4 R(1)                                                        00000040
      INTEGER STM,CNT(20)                                                00000050
      EXTERNAL STM                                                       00000060
C                                                                        00000070
C CALCULATES AND PUNCHES THE KARHUNEN BASIS. WRITES EIGENVALUE FRACTIONS 00000080
C CALLS SUBROUTINES STRI1,SBK.                                           00000090
C EIGENVALUES IN ROOT(I) I=1,LL INCREASING IN MAGNITUDE.                 00000100
C EIGENVECTORS IN VEC(J,I) I=1,M J=1,N CORRESPONDING TO THE EIGENVALUES. 00000110
C DIMENSIONS ALLOW FOR ORDER UP TO 300, AND 20 EIGENVALUES.              00000120
C R IS STORED AS A LIEAR ARRAY. MATRIX(I,J) STORED AS R(I*(I-1)/2+J)     00000130
C                                                                        00000140
      CALL STRI1(N,R,DIAG,SUBDIA)                                        00000150
      WRITE(6,56)                                                        00000160
   56 FORMAT('  DIAGONAL         SUBDIAGONAL')                          00000170
      DO 6 I=1,N                                                         00000180
      WRITE(6,55) DIAG(I),SUBDIA(I)                                      00000190
   55 FORMAT(2E15.6)                                                     00000200
      SUB(I)=SUBDIA(I)                                                   00000210
    6 BETA(I)=SUBDIA(I)**2                                               00000220
C                                                                        00000230
C COMPUTE ENERGY AND BOUNDS FOR EIGENVALUES                              00000240
C                                                                        00000250
      ENERGY =0.                                                         00000260
      DO 5 I=1,N                                                         00000270
    5 ENERGY=ENERGY+DIAG(I)                                              00000280
C                                                                        00000290
C FIND LOWER BOUND                                                       00000300
C                                                                        00000310
      LB=ENERGY*1.0D-8                                                   00000320
      OLB=0.                                                             00000330
      UB=ENERGY*1.5                                                      00000340
      OUB=ENERGY*1.5                                                     00000350
      YF=0.5                                                             00000360
      I=0                                                                00000370
```

```
      WRITE(6,50) ENERGY                                                00000380
   50 FORMAT('  ENERGY=',E15.6)                                         00000390
   44 M=N-STM(1,N,DIAG,SUBDIA,BETA,LB,N)                               00000400
  200 WRITE(6,51) LB,M,I                                               00000410
   51 FORMAT ('  LB=',E15.6,'  NUMBER OF EIGENVALUES=',I4,' I=',I4)     00000420
      I=I+1                                                            00000430
      IF(I-500) 52,53,53                                               00000440
   53 WRITE(6,54)                                                      00000450
   54 FORMAT('  ROUTINE DID NOT FIND  A LOWER BOUND AFTER 500 ITS.')    00000460
      RETURN 1                                                         00000490
   52 IF(M-LL) 41,42,43                                                00000500
   41 OUB=LB                                                           00000510
C DECREASE LB                                                          00000520
      LB=(1.-YF)*LB+YF*OLB                                             00000530
      GO TO 44                                                         00000540
   43 OLB=LB                                                           00000550
C INCREASE LB                                                          00000560
      LB=YF*OUB+(1.-YF)*LB                                             00000570
      GO TO 44                                                         00000580
   42 CONTINUE                                                         00000590
C                                                                      00000600
C COMPUTE THE NUMBER OF EIGENVALUES BETWEEN LB AND UB                  00000610
C                                                                      00000620
      IF (M-30) 7,7,8                                                  00000630
    8 WRITE(6,9) LB,UB,M                                               00000640
    9 FORMAT(' REQUIRED NUMBER OF EIGENVALUES IS MORE THAN 30'//       00000650
     1 ' LB=',E15.6/' UB=',E15.6/' M=',I3)                             00000660
      CALL EXIT                                                        00000670
C                                                                      00000680
C COMPUTE THE M MOST SIGNIFICANT EIGENVALUES AND VECTORS OF R          00000690
C                                                                      00000700
    7 CALL TRIST(N,LB,UB,DIAG,SUBDIA,BETA,M,ROOT,VEC,CNT,&12)          00000710
      GO TO 11                                                         00000720
   12 WRITE(6,13) M                                                    00000730
   13 FORMAT(' CALC UNSUCCESSFUL. M= ',I2//' ITS. PER VECTOR')         00000740
      WRITE(6,14) (CNT(I),I=1,M)                                       00000750
   14 FORMAT(I2)                                                       00000760
```

```
      CALL EXIT                                                          00000770
   11 CONTINUE                                                           00000780
C TRANSFORM BACK TO ORIGINAL CO-ORDINATE SYSTEM                          00000790
C M IS THE NUMBER OF ROOTS                                               00000800
C ROOT CONTAINS THE EIGENVALUES                                          00000810
C VEC CONTAINS THE EIGENVECTORS                                          00000820
C CNT IS THE NUMBER OF ITERATIONS PER VECTOR                             00000830
C                                                                        00000840
      CALL SBK(N,R,SUB,M,VEC)                                            00000850
C                                                                        00000860
   20 DO 17 I=1,M                                                        00000870
      ROOT(I)=ROOT(I)/ENERGY                                             00000880
      WRITE(6,15) ROOT(I),CNT(I)                                         00000890
   15 FORMAT(//' EIGENVALUE FRACTION= ',E15.6,' ITS PER VECTOR= ',I2)    00000900
   16 FORMAT(E15.6)                                                      00000910
   17 CONTINUE                                                           00000920
C                                                                        00000930
C PUNCH OUT BASIS INFORMATION                                            00000940
      WRITE(7,400) N,M                                                   00000950
  400 FORMAT(2I4)                                                        00000960
      DO 401 K=1,M                                                       00000970
      I=M-K+1                                                            00000980
  401 WRITE(7,402) ROOT(I)                                               00000990
  402 FORMAT(E13.6)                                                      00001000
      DO 403 K=1,M                                                       00001010
      I=M-K+1                                                            00001020
  403 WRITE(7,404) (VEC(J,I),J=1,N)                                      00001030
  404 FORMAT(5(E13.6,1X))                                                00001040
      RETURN                                                             00001050
      END                                                                00001060
```

REFERENCES

1. Scher, A. M., A. S. Young, and W. M. Meredith, "Factor Analysis of the Electrocardiogram; A Test of Electrocardiographic Theory; Normal Leads," Circ. Res., Vol. 8, p. 519, 1960.

2. Noran, L. G., N. C. Flowers, and D. A. Brody, "Principal Factor Waveforms of the Thoracic QRS Complex," Circ. Res., Vol. 15, pp. 131-145, August, 1964.

3. Young, T. Y., and W. H. Huggins, "On the Representation of Electrocardiograms," IEEE Trans. Bio-Med. Electron., Vol. BME-10, pp. 86-95, July, 1963.

4. R. McFee, and G. M. Baule, "Research in Electrocardiography and Magnetocardiography," Proc. IEEE, Vol. 60, No. 3, pp. 290-321, March, 1972.

5. Plonsey, R., "The Biophysical Basis for Electrocardiography," Crit. Rev. Bioeng., Vol. 1, pp. 1-48, October, 1971.

6. Burton, A. C., Physiology and Biophysics of the Circulation, Yearbook Medical Publishers, Chicago, Illinois, 1965.

7. Goldman, M. J., Principles of Clinical Electrocardiography, Lange Medical Publications, Los Altos, California, 1970.

8. Yamada, K. et al, "Observer Variation in Electrocardiographic Interpretation," Ann. Rep. Res. Inst. Environ. Med. (Nagoya University, Nagoya, Japan), Vol. 16, pp. 27-41, 1968.

9. Smith, R. E., and C. M. Hyde, "A Computer System for Electrocardiographic Analysis," in Proc. 3rd Ann. Rocky Mountain Bio-Engineering Symp. (University of Colorado, Boulder, May, 1966).

10. Pipberger, H.V., "Computer Analysis of the Electrocardiogram," in Computers in Biomedical Research, Vol. 1, R. W. Stacy and B. D. Waxman, Eds., Academic Press, New York, 1965.

11. Cady, L. A., M. A. Woodbury, L. J. Tick, and M. M. Gertler, "A Method for Electrocardiogram Wave-pattern Estimation," Circ. Res., Vol. 9, pp. 1078-1082, September, 1961.

12. Karlsson, S., "Representation of Electrocardiogram Records by Karhunen-Loève Expansions," in Dig. 7th Int. Conf. Med. and Biol. Engineering (Stockholm, Sweden), p. 105, 1967.

13. Cox, J. R., H. A. Fozzard, F. M. Nolle, G. C. Oliver, "Some Data Transformations Useful in Electrocardiography," in Computers in Biomedical Research, Vol. 3, R. W. Stacy and B. D. Waxman, Eds., Academic Press, New York, 1961.

14. Young, T. Y., and W. H. Huggins, "The Intrinsic Component Theory of Electrocardiography," IRE Trans. Bio-Med. Electron., Vol. BME-9, pp. 214-221, October, 1962.

15. Fukunaga, F., Introduction to Statistical Pattern Recognition, Academic Press, New York, 1972.

16. Van Trees, H., Detection, Estimation and Modulation Theory, Part I, Wiley, New York, 1968.

17. Chien, Y. T., and K. S. Fu, "On the Generalized Karhunen-Loève Expansion," IEEE Trans. Inf. Theory, Vol. IT-13, No. 3, pp. 518-520, July, 1967.

18. Schweppe, F. C., Uncertain Dynamic Systems, Prentice-Hall, Englewood Cliffs, New Jersey, 1973.

19. Cox, J. R., F. M. Nolbe, and R. M. Arthur, "Digital Analysis of the Electroencephalogram, the Blood Pressure Wave and the Electrocardiogram," Proc. IEEE, Vol. 16, No. 10, pp. 1137-1164, October, 1972.

20. Mangasarian, O. L., "Multisurface Pattern Recognition," IEEE Trans. Info. Theory, Vol. IT-14, pp. 801-807, November, 1968.

21. Specht, D. F., "Vectorcardiographic Diagnosis Using the Polynomial Discriminant Method of Pattern Recognition," IEEE Trans. Bio-Med. Electron., Vol. BME-14, No. 2, pp. 90-95, April, 1967.

22. Specht, D. F., "Generation of Polynomial Discriminant Functions for Pattern Recognition," IEEE Trans. Elect. Comput., Vol. EC-16, No. 3, pp. 308-319, June, 1967.

23. Mucciardi, A. N., and E. E. Gose, "A Comparison of Seven Techniques for Choosing Subsets of Pattern Recognition Properties," IEEE Trans. Comput., Vol. C-20, No. 9, pp. 1023-1031, September, 1971.

24. Jenkins, G. M., and D. G. Watts, Spectral Analysis and its Applications, Holden-Day, San Francisco, 1969.

25. Wilkinson, J. H., and C. Reinsch, Handbook for Automatic Computation, Vol. II, Linear Algebra, Springer Verlag, New York, 1971.