A Signals and Systems and Object Oriented Programming Approach to Development of ECG Analysis Software

SN Mohammad, AK Sureka, DR Ely, JM Jenkins

University of Michigan, Ann Arbor, MI, USA

Abstract

A Visual Software Development Framework (VSDF) has been created to facilitate the software coding process of biological signal analysis. A special purpose graphical user interface (GUI) was developed using VSDF for automating the analysis and classification of waveforms in electrocardiographic (ECG) data. Graphing capabilities were included in the program to allow the user to plot the signal as well as trigger location. Object Oriented Programming (OOP) techniques were used in the Java programming language to calculate correlation coefficients between a template QRS complex and detected QRS complexes throughout an ECG recording. The OOP techniques employed in the development of the software package allow visualization of the entire analysis process. The ECG signal, read from a file, is stored into a simple array data structure and is passed to each of the modules. The signals and systems approach allows incorporation of trigger modules, data conversion modules, and numerical analysis modules directly into the software package, providing ease of software design.

1. Introduction

Classic signal processing and analysis of electrocardiographic (ECG) data employs pattern recognition of abnormal waveforms for classification of abnormal beats and arrhythmias. A variety of algorithms have been employed in the past three decades for morphological analysis of the QRS. These include feature extraction of signal parameters such as timing, amplitude, polarity of waveform, and area of difference. Novel methods have been proposed such as signature analysis of the QRS complex [1], polynomial coefficient techniques [2], normalized area of difference (NAD) [3] and amplitude distribution analysis (ADA) [4] as well as frequency domain methods [4]. A signal processing technique employed in our research laboratory (Medical Computing Laboratory, University of Michigan, Ann Arbor) is correlation waveform analysis (CWA), a statistical means for point-by-point comparison of digital data with a normal template [4]. We have used this technique on surface leads acquired from coronary care monitors [5] and intracardiac leads acquired in the clinical electrophysiology laboratory [6]. Software developed in research laboratories such as ours is typically platform dependent, not necessarily user friendly, not easily portable, and must typically be rewritten for a variety of systems.

The software design technique presented in this paper is a modular, hierarchical scheme that can be used to develop generic software packages for biological signal analysis.

Others have addressed this problem for a variety of applications. NEMESIS [7] is an object oriented system, with a highly modular structure, and network friendly interface for use in cardiologic telemedicine. CAVIAR (Comparative Analysis of Vectorcardiograms and their Interpretation with Auto-reference to the patient) is an open and modular database management system that supports the acquisition and management of serial ECGs [8]. OODBMS (Object-Oriented Data Base Management System) applies similar techniques to management of temporal clinical data [9]. Twin Expert Systems for Complex Analysis of 12-lead ECGs reports an object oriented system that mimics the cardiologists' decision-making methods[10].

We have designed an object oriented system development technique for automating the analysis and classification of waveforms in signals such as electrocardiograms, blood pressure data, and electroencephalograms. The modular techniques employed to develop the software package allow vivid visualization of the entire analysis process. The levels of abstraction allow the person developing triggering systems to work independently of the person developing systems which analyze and classify the triggered data. The ECG waveform is stored into a simple array data structure and is passed to the analysis module.

The signals and systems approach allows one to incorporate several different types of triggers and numerical analysis modules directly into the software package. The ECG module currently developed for our software model allows the user to graphically pick a template QRS complex (by selecting forward and aft ranges from point of trigger detection) and generate correlation coefficients with all other trigger-detected beats in the ECG signal. The user can set a threshold for the correlation coefficient to classify individual beats as normal or abnormal.

File conversion utilities were also built into the program to allow reading (and conversion) of CODAS (DATAQ Instruments Inc, Akron OH) and our custom formatted SIG files (Medical Computing Laboratory, Ann Arbor MI) directly, as well as signal files previously converted into an ASCII (TXT) file.

2. Materials and methods

2.1. Materials

VSDF software was developed, tested, and deployed on standard Pentium computers (Intel Corp, Santa Clara CA). Since the purpose of this project was to demonstrate software engineering methods as opposed to developing a specific high-fidelity software package, specifications for computing equipment were kept minimally sufficient to run the operating system and compilers.

Java 1.3.1, an OOP programming language (Sun Microsystems Inc, Santa Clara CA) was used to develop the software. Using Java ensured portability of the software system, since code written in Java can be executed on any platform equipped with the Java Runtime Environment.

Java's Swing graphics library was used to develop the graphical user interface. Additionally, a freely-available plotting library for Java was used to provide charting capabilities to the system.

Before developing the modules of the software, the interfaces, protocols, and data structures were well defined. Each module was then developed as an independent identity.

2.2. Methods

The data preparation module consists of all operations involved with acquiring data, formatting, and storage. The end-result of this module is to obtain data from a wide variety of sources (analog/digital recordings, digital files, live acquisition directly from a physiological measurement, etc.), transform them into a data structure defining the notion of a *signal* in the computer memory.

The standardized format of all signal data subject to processing and analysis, within a well-defined structure, affords accessibility by all algorithms/processing modules. (See Figure 1.)

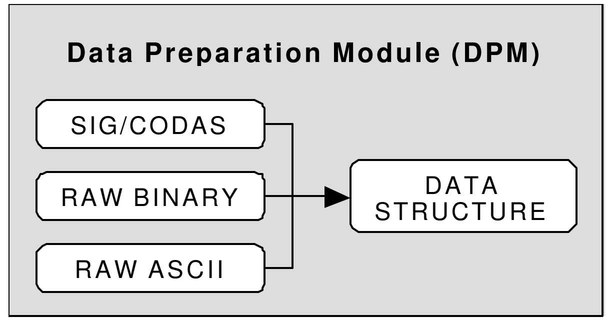

Figure 1. Block diagram of data preparation. Data available in a multitude of formats are transformed into a standardized format for entry into a database with a well-defined structure. These data are accessible to all analysis modules.

After data have been prepared and stored in a standard structure in computer memory, it is ready to be passed to the algorithms-and-processing module (APM). The APM module for our ECG software required triggering the ECG signal for QRS detection and correlating waveforms against a known *normal* template. After each QRS of the signal has been correlated with the template, the correlation values are sent to the data output module. (See Figure 2.)

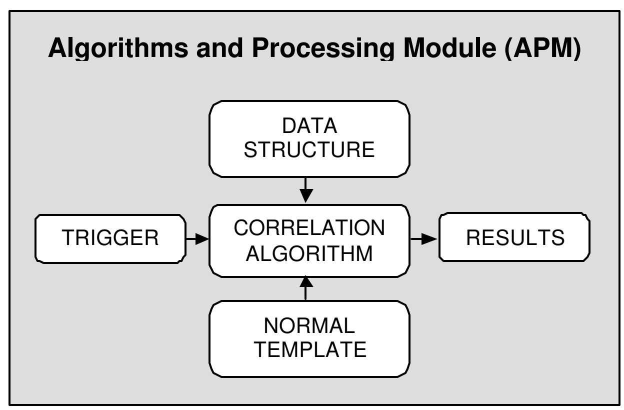

Figure 2. Block diagram of the algorithms and processing module. Biological or physiological signals are passed into the algorithm block. The processed data are delivered as results.

The data output module receives results from the algorithms-and-processing module and delivers them in a format acceptable to the user. Custom outputs, such as Excel worksheets (Microsoft Corp, Redmond WA), as well as other file formats are programmed into this module. Our model for analysis of electrocardiographic and blood pressure data generated Excel data tables and also plotted them directly in the GUI using a plotting library. (See figure 3.)

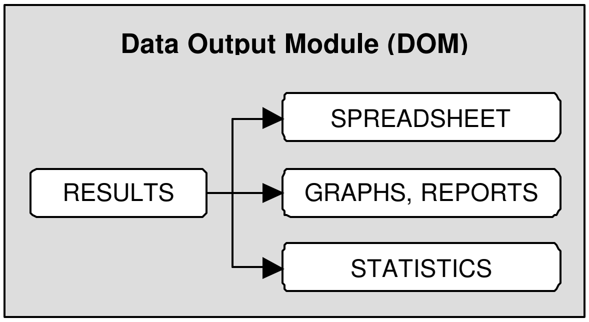

Figure 3. Block diagram of output of the system. Spreadsheet, graphical, tabular, textual, or statistical representations are presently included in VSDF, but additional output modules can easily be incorporated into the system.

2.3. Standard specifications

2.3.1. Data structures

Standard data structures used to store the signal in computer memory were defined. In Java OOP, the notion of a generic signal was represented as an index array of integers (analog data digitized using 12-bit linear quantization, resultant digital data ranging from -2,048 to +2,047).

2.3.2. Interfaces and documentation

Before developing each of the modules, standard interfaces were defined to facilitate data communication. In order to have a well-defined interface for a module, each sub-module needed to be thoroughly defined. For example, the trigger within APM has an interface defined by the following inputs: signal data; integer threshold; integer hold time (samples, corresponding to milliseconds); and the following output: an index array of integers (each item representing time of trigger assertion).

In order to ensure hassle-free integration, detailed data flow diagrams were created and contained not only high-level description of the data, but also type and storage information. By defining each sub-module interface and utilizing detailed visual data flow diagrams, each data movement within the module was tested for consistency. Using a hierarchical approach, and accounting for data flow, eased the integration process and produced the well-defined interface required of each module.

The interfaces and data-flow diagrams for each module and sub-module were formally compiled into a specifications document. Documentation is a key component of VSDF and allows the developed modules to be part of the organization's body of knowledge. As other software packages are developed in the lab, completed and documented modules are available for immediate use to developers and do not need to be recreated. This makes the software design process more efficient by reducing redundancy and produces quicker turn-around time, allowing the developers to focus their efforts on new tasks adding value to their final products.

2.3.3. Integration

Integration of the component modules was straightforward and the addition of future modules is uncomplicated due to the existence in VSDF of well-defined interfaces, protocols, and data structures. (See Figure 4.)

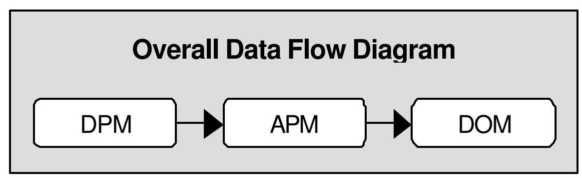

Figure 4. Module integration process visualized by the data flow diagram.

3. VSDF implementation

An exemplary account of the employment of VSDF was utilized in the simultaneous analysis of intracardiac electrogram (EGM) and arterial blood pressure (BP) signals to detect lethal tachycardias, done as a research project in the Medical Computing Laboratory. The ECG software processed the EGM and BP data concurrently to detect hemodynamic change, and analyzed the signals for rhythm irregularities such as stable versus unstable ventricular tachycardias (VTs). VSDF was applied to rhythm and blood pressure measurements to demonstrate ease of use and improved performance metrics.

3.1. Materials

Electocardiographic and blood pressure signals from canines were acquired in an animal electrophysiology laboratory for subsequent signal processing. Data were recorded on a four-channel FM tape recorder (Hewlett

Packard 3968 Instrumentation Recorder, San Diego, CA) (3 ¾ ips), digitized at 1000 Hz via a 12-bit A/D converter, on CODAS (DATAQ Instruments, Inc., Akron, OH) a real-time data acquisition system, and converted into a custom format (SIG). Once the signals were digitized they were displayed in a Windows utility program (Microsoft Corp, Redmond WA), AAELVIEW (Ann Arbor Electrogram Libraries, Ann Arbor MI) and converted into text format to be correlated by ECG software. The data were then plotted using Excel so that results could be viewed. Features implemented in the program for concurrent signal detection were: two-channel signal acquisition, ECG triggering, signal analysis algorithms and analytic output. Correlation waveform analysis (CWA) was employed for the detection of abnormal beats. Amplitude variation was used as a measure of normalcy for the blood pressure waveforms.

Results of the classification of benign versus malignant VT using ECG software produced results identical to earlier processing of the same data using our conventional means.

4. Conclusions

The purpose of this project was to demonstrate that a modular, hierarchical scheme could be used to develop software packages for signal analysis. Functions of classic ECG analysis software were reviewed and broken down into independent modules. Our specific model consisted of three modules: data preparation, algorithm-and-processing, and data output.

In research laboratories, student-based research or project-based research is often driven by targets and deadlines which preclude an orderly and systematic development of robust software. Signal analysis software is typically *ad hoc* and platform dependent. The lack of portability and flexibility results in replication of effort many times over. We propose a methodical, structured approach for the creation of programs designed for the academic laboratory. VSDF is intended to provide a framework for the construction of easy to use, portable software suitable for multi-site utilization.

Acknowledgements

This work was supported in part by National Science Foundation grant BES9825125, Guidant Corporation, and an equipment grant from Intel Corporation.

References

[1] Greenhut SE, Steinhaus BM, Murphy AJ. Comparison of a new template matching algorithm, correlation, and Area of Difference methods for detection of ventricular tachycardia. *Computers in Cardiology* 1992. Durham, NC USA: IEEE Computer Society Press , 1992:371-4.

[2] Shiu J, Parthasarathi AA, Jenkins JM, DiCarlo LA. Comparison of polynomial coefficients for pattern recognition of sinus rhythm (SR) and ventricular tachycardia (VT). *Computers in Cardiology* 1996. Indianapolis, IN USA: IEEE Computer Society Press, 1996:65-8.

[3] Throne RD, Jenkins JM, DiCarlo LA. A comparison of four new time-domain techniques for discriminating monomorphic ventricular tachycardia from sinus rhythm using ventricular waveform morphology. *IEEE Trans on Biomed Eng* 1991;38(6):561-70.

[4] Lin D, DiCarlo LA, and Jenkins JM. Identification of ventricular tachycardia using intracavitary ventricular electrograms: Analysis of time and frequency domain patterns. *PACE* 1988;11:1592-1606.

[5] Jenkins JM, Wu D, Arzbaecher RC. Computer diagnosis of supraventricular and ventricular arrhythmias: A new esophageal technique. *Circulation* 1979;60(5):977-87.

[6] Ann Arbor Electrogram Libraries, Ann Arbor MI.

[7] Magdalena JR, Soria E, Serrano A, Calpe J, Guerrero J, Martinez M. NEMESIS: A new telemedicine approach to cardiologic software. *Computers in Cardiology* 1999;26:37-40.

[8] Fayn J, Rubel P, Girard P, Ravel C, Forlini MC, Boissel JP. CAVIAR, a serial ECG processing and management system for pharmacological drug trials. *Computers in Cardiology* 1991. Venice Italy: IEEE Computer Society Press. 1992:301-4.

[9] Combi C, Pinciroli F, Pozzi G. Temporal clinical data modelling and implementation for PTCA patients in an OODBMS environment. *Computers in Cardiology* 1994. Bethesda MD USA: IEEE Computer Society Press. 1994: 505-8.

[10] Nieberl J, Khoor S, Khoor M, Kekes E, Szaboki F. Twin expert systems for the complex analysis of conventional 12 lead electrocardiograms. *Computers in Cardiology* 1993. London UK: IEEE Computer Society Press. 1993:511-3.

[11] Ely D, Mohammad S, Sureka A, Jenkins J. Detection of hemodynamic compromise by blood pressure analysis for an implantable defibrillator. 27th Annual International Society of Computerized Electrocardiology 2002. Doorwerth, Netherlands: (*Journal of Electrocardiology*, in Press.)

Address for correspondence.

Professor Janice M Jenkins
Dept. of Electrical Engineering and Computer Science,
and Dept. of Biomedical Engineering
Medical Computing Laboratory
EECS Bldg 1301 Beal Avenue
University of Michigan
Ann Arbor MI 48109-2122
jenkins@umich.edu