Multi-Parameter Databases Remote-access and Automatic Layout and Conjunct Analysis by means of QT Cross-platform Application Framework

Wei Wu

Weiwu Info, Portugal

Abstract

Data acquirement and signal processing at Multi-parameter records such as PhysioNet database records require accessing records remotely, laying out uncertain number of parameters, and doing conjunct analysis. This paper proposes developing software by QT graphical C++ toolkit can meet the requirements described above in less developing time and higher efficiency. Practical works at a software tool development are introduced to show advantages of QT toolkit at software tool development, particularly the advantages of SIGNAL/SLOT and Layout-manage mechanisms that are fundamental for implement of conjunct analysisms of the abstract before continuing with the next heading.

1. Introduction

Some databases such as PhysioNet databases (WFDB) include the record with uncertain number of parameters, and are published through web. There are some tools, for example, Physionet ATM [1], allows browsing WFDB records online. But for scientific research purpose, downloading database records at local disk is necessary for effective signal processing.

This paper introduces a signal conjunct analysis tool which allows user to access remote WFDB database from a local desktop and to quickly browse the records to find an interesting one, then download it to local disk for further study. Different from many WFDB analysis toolkits, proposed tool holds the multi-parameter at an in-memory matrix in which, all signals are displayed at a multi-frame layout and processed jointly. The advantage of the conjunct analysis at a multi-parameter record is it may discover hidden relationship among multi-parameters.

2. Fundamental of QT

QT is a cross-platform application framework [2] that is widely used for developing application software with a graphical user interface (GUI), and also used for developing non-GUI programs such as command-line tools and consoles for servers. It is currently produced by Nokia's Qt Development Frameworks division, which came into being after Nokia's acquisition of Norwegian company, Trolltech, the original producer of Qt.

There are many powerful functions QT included and among those, the most interesting ones for our application mentioned above, are their layouts-manage mechanism and SIGNAL/SLOT communication mechanism.

2.1. Layout-manage mechanism

In QT, an object, such as a displayed signal, is defined as a widget. QT gives every widget that is placed on a form an appropriate size and position by layout-manage mechanism. Qt provides several classes that lay out widgets on a form: QHBoxLayout, QVBoxLayout, QGridLayout, and QStackedLayout. These classes can implement the signals automatic layout by assigning each signal a frame at a multi-frame layout, and ensure that frames adapt automatically to different fonts, languages, and platforms. If the user changes the system's font settings, the application's frames will respond immediately, resizing themselves if necessary.

2.2. SIGNAL/SLOT mechanism

One of Qt's major achievements has been the extension of C++ with signal/slot mechanism. In the traditional programming language, for example, Java GUI libraries, connecting a user action, such as clicking a button, to a corresponding method involves writing an event listener class; with QT's SIGNAL/SLOT mechanism, only one line of code is required to achieve the same thing. QT's SIGNAL/SLOT mechanism enables the application programmer to bind objects together without the objects knowing anything about each other. SLOTs are almost identical to ordinary C++ member functions. The difference is that a SLOT can also be connected to a SIGNAL, in which case it is automatically called each time the SIGNAL is emitted.

The connect() statement looks like this:

```
connect(sender, SIGNAL(signal), receiver, SLOT(slot));
```

It is very flexible: One SIGNAL can be connected to many SLOTs; many SIGNALs can be connected to the same SLOT and even more, a SIGNAL can be connected to another SIGNAL.

The advantages of QT illustrated above well meet the requirements of WFDB database record analysis. This paper proposes developing a software tool, GreatMiner [3], by QT graphical C++ toolkit, to access remote database, to lay out signals from multi-parameter record of the database and to perform conjunct analysis at acquired multi-parameter records.

3. WFDB database remote

All WFDB Multi-Parameter databases are listed at http://www.physionet.org/physiobank/database/, a portal-page where each database is linked to a record-page through hyperlink. The databases are not well cataloged and are difficult to identify by software so that, two algorithms are developed to search databases at the portal-page and to access records of the searched databases. The first algorithm is used to scan the portal-page and to analyze the portal-page by key words and hyperlinks, and to collect possible database names into a database-list. The second algorithm is used to confirm availability of listed database in the database-list by tracing its hyperlink; if the hyperlink leads to a valid records-list, the availability of the database is confirmed; otherwise, the name of traced database will be removed from the database-list. After that, clicking a database from the database-list and selecting a relative record from the records-list will trigger software to read the selected record from the remote database. In GreatMiner, the multiple signals included in a record are extracted according to WFDB file formats [4], and saved at an in-memory matrix, SIGNAL[N].

4. Automatic layout

We have developed a PLOTTER class to plot a signal at a figure. To plot multiple signals, some codes look like below are employed:

```
int col = ceil(N/2.0);
if (col>1){//N>=4, plot multi frame in two col
   for (int i=0;i<col;i++){
      Plotter::Plotter *plotter;
      plotter->setCurveData(0,signal[i]);
      // layout and show
     ui->workspaceLayout-> addWidget(plotter,i,0);
      plotter->show();
   }
   for (int i=col;i<N;i++){
      Plotter::Plotter *plotter;
      plotter->setCurveData(0,signal[i]);
     ui->workspaceLayout->addWidget(plotter,i-col,1);
      plotter->show();
   }
}
```

With the codes given above, we can always reflect the signals at matrix SIGNAL[N] to a multi-frames layout. The QT layout mechanism will ensure that frames adapt automatically to different fonts, languages, and platforms. If the user changes the size of the layout, remove a signal from the layout or append new signal to the layout, the frames inside the layout will respond immediately, resizing or relaying themselves if necessary.

5. SIGNAL/SLOT communication

In GreatMiner, a Four-Frame-Layout (Figure 1) is used to extract hidden information from a signal by wavelet transform:

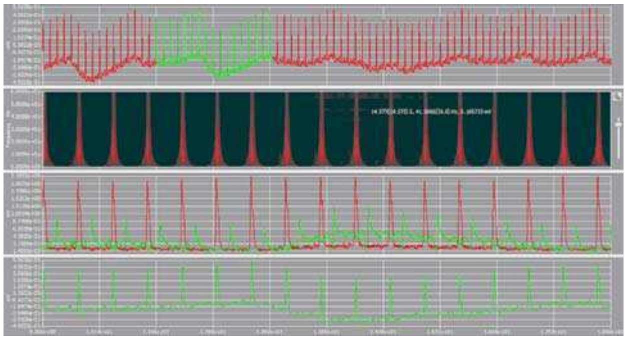

Figure 1. Four-Frame-Layout for time-frequency domain analysis.

In this Four-Frame-Layout, from top down, the first frame displays original signal, and the second frame shows the wavelet transform coefficient map of the signal, the third frame displays the frequency variation in time (the frequency is correlated to the mouse location at the second frame, the wavelet coefficient map) and the forth frame displays a zoomed portion of the original signal (marked green in the first frame). At the second frame, a SIGNAL is defined for moving mouse at the wavelet coefficient map and a SIGNAL is defined for dragging a rectangle at the map to zoom the image inside the rectangle. Two SLOTs are constructed: one is connected to the mouse moving SIGNAL, for reading mouse location at the second frame and calculate the frequency corresponding to the mouse location, and display the frequency variation at the third frame. Another SLOT is connected to the zooming SIGNAL and adjusts the image size at the second frame and the curve size at the third and the fourth frames. Based on the SIGNAL/PLOT configuration described above, moving the mouse at the wavelet coefficient map will emit a SIGNAL that triggers the relative PLOT to display at the third frame the variation of the frequency indicated by the mouse at the second frame; dragging a rectangle at the coefficient map will emit a SINGLA that triggers the relative PLOT to zoom the image in the second frame and the curves in the third and fourth frames synchronously.

6. Conjunct analysis

The conjunct analysis in GreatMiner is defined as: The analysis at one or multiple signals according to the operation result of another signal. Conjunct analysis can be essential for some applications that treat multiple data from the same source; for example, the application at WFDB record that includes the multiple signals generated by the identical person and should have some correlations among the signals.

The conjunct analysis in GreatMiner is implemented based on SIGNAL/SLOT mechanism and layout-manage mechanism, by enable Bond Signal or by functionality.

6.1. Conjunct analysis by bonding signals

6.1.1. Zoom signals

Go to options menu and select Bond Signals. Open a WFDB record, mgh003.txt and all six signals inside are assigned to a multi-frame layout by QT’s layout-manage mechanism (Figure 2). Zoom the signal CVP_mgh003.txt by dragging a rectangle at the signal to include a cycle of the signal, a moving mouse SIGNAL will be emitted that triggers relative SLOT to zoom all signals in other frames synchronously at the same time scale (Figure 3). This is particularly useful for users who want to investigate the detail of one signal meanwhile being aware of knowledge of other signals at the same time scale.

6.1.2. Noise reduction

Choose signal CVP_mgh003.txt from the multi-frame layout illustrated in Figure 2 and perform De-noise. Drag stick of the horizontal slider to modify de-noise depth until satisfied de-noise result achieved. This will emit a de-noise SIGNAL to trigger relative SLOT which de-noises all signals at other frames at the same de-noise depth (Figure 4).

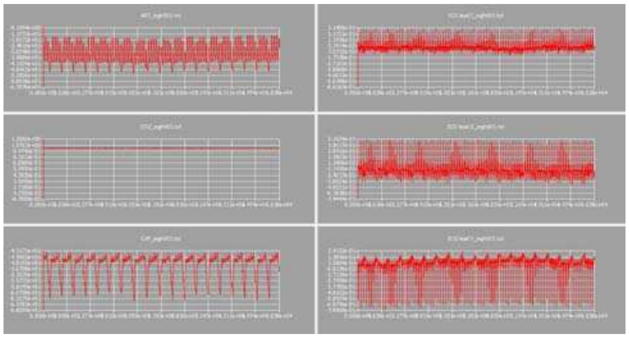

Figure 2. Original signals from a WFDB record.

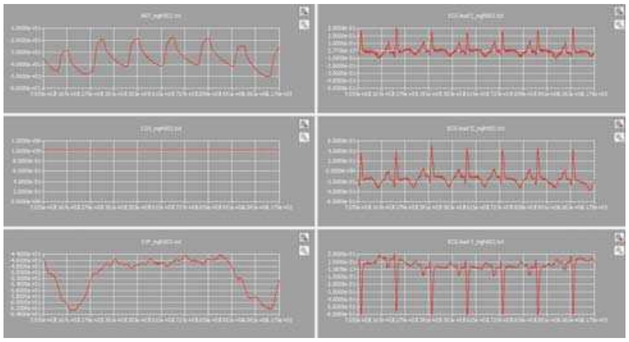

Figure 3. Zooming at one signal will zoom all other signals equally**.**

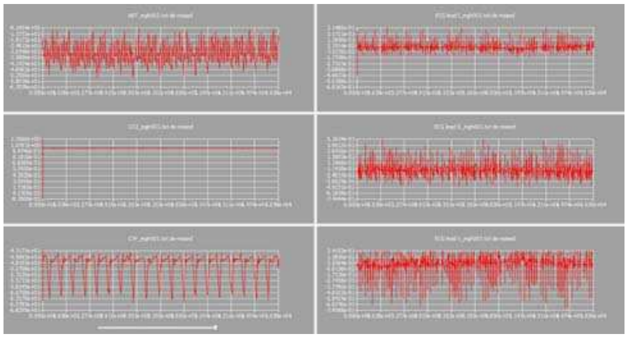

Figure 4. Conjunct signal De-noise.

6.2. Conjunct analysis by functionality

6.2.1. Signal decompose

GreatMiner gives a flexible way for signal decomposing. Because the wavelet transform decomposes a signal into different levels that are remained at the wavelet transform coefficient matrix, one row for one level of decompose result; the signal decompose results can be investigated by scanning the coefficient matrix. For example, select a signal, ECG lead II_mgh003.txt from the multi-frame layout displayed at Figure 2 and choose Wavelet Transform; the signal is wavelet transformed and the transform coefficient matrix is displayed at the second frame of Figure 5, in format of a matrix map image. Moving mouse at the wavelet coefficient map, the wavelet transform coefficient at a row (frequency) indicated by the cursor in the second frame will be displayed at the third frame of Figure 5. Moving mouse at the second frame and observing decompose result at the third frame until achieve satisfied decompose result (decompose at 0.045Hz for this example), right click mouse to remember the decompose level that will emit a SIGNAL to trigger a relative SLOT decomposing all signals at 0.045Hz (equal the 90th level of a 100 level decompose in this example), and the results are displayed at a multi-frame layout illustrated in Figure 6.

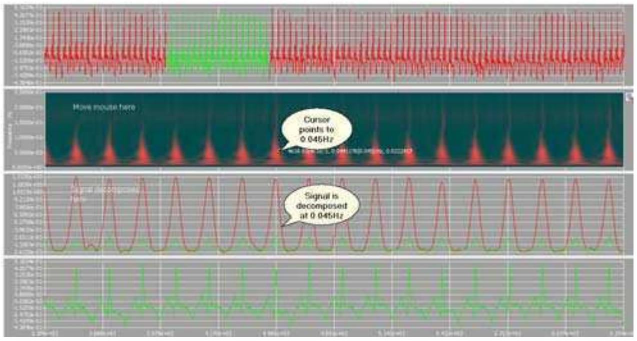

Figure 5. Moving mouse and observing decompose result until achieve satisfied decompose.

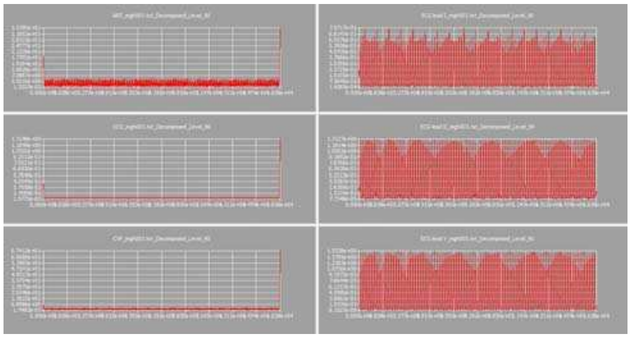

Figure 6. All signals are decomposed at the same decompose level.

6.2.2. Cross-correlation analysis

It is intuition to consider having a cross-correlation investigation at WFDB multi-parameter record to discover correlation among the parameters. In GreatMiner, this is implemented by selecting one signal as a main signal and all others as target signals and performing cross-correlation between the main signal and the target signals. For example, choose a signal, ECG lead II_mgh003.txt at the multi-frame layout illustrated in Figure 2 as a main signal, choosing Cross-Correlation from a right-clicking menu will emit a cross-correlation SIGNAL to trigger a relative SLOT, performing cross-correlation between the main signal and the others, the results are displayed together with relative target signals (Figure 7).

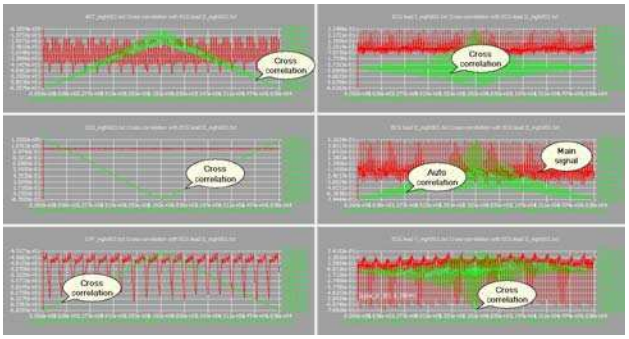

Figure 7. Correlation between main signal and other signals

7. Conclusion

Multi-parameter records such as PhysioNet databases records raise new requirements for data acquirement and signal processing. The requirements include accessing records remotely, laying out uncertain number of parameters, and doing conjunct analysis. This paper proposes developing software by QT graphical C++ toolkit can meet requirements described above and realize other functions in less developing time and higher efficiency. Practical works at a software tool development are introduced to show advantages of QT toolkit at software tool development, particularly the advantages of SIGNAL/SLOT and Layout-manage mechanisms that are fundamental for implement of conjunct analysis.

Due to tight schedule at project development, we have no time to deploy GreatMiner at other platform rather than Windows to confirm QT's cross-platform character until the publish of this paper. The interested reader can refer available reference [5]

References

[1] PhysioNet. An Introduction to the PhysioBank Archives, http://www.physionet.org/physiobank/physiobank-intro.shtml.

[2] Jasmin Blanchette, C++ GUI Programming with Qt 4, Second Edition. Prentice Hall, February 04, 2008.

[3] WEIWU INFO. GreatMiner user manual, version 1.0. www.greatminer.com 2011.

[4] George B. Moody, WFDB Applications Guide, Harvard-MIT Division of Health Sciences and Technology, Tenth Edition, 12 March 2011.

[5] Nokia. Qt developer guide. Nokiaa Co. 2008

Address for correspondence.

Wei Wu.
AV DA BOAVISTA, 991-1 ES TR, Porto, Portugal
w.wu@ieee.org