

AUTOMATED DETECTION AND RECOGNITION OF DIAGNOSTICALLY SIGNIFICANT ECG CHANGES

V. M. Nikitin*, V. V. Muromtsev, O. A. Efremova, E. A. Lipunova, L. A. Kamyshnikova

Belgorod State University, 85 Pobedy St, Belgorod, 308015, Russia.

Published online: 15 February 2017

ABSTRACT

The article considers the system of computer analysis of ECG and the software implementation of this system. The system provides for automatic recognition of singular points of ECG and automatic detection of ECG characteristic areas. In the design and implementation of the system there were widely used the finite automata. This allowed to visualize the logic of the system, to avoid design errors and to accelerate the process of creating the system. The analysis of test results developed by the team of the prototype of the computerized ECG analysis system with the advanced capabilities of automated search for characteristic sites allows us to conclude that the reliability of ECG interpretation by a specialist increases with a substantial reduction in the time required to perform this procedure. This positive effect is achieved through the implementation of the procedure for automatic recognition of special ECG points and the use of automated means of searching for ECG lines. The system increases the reliability of decoding ECG by a doctor-cardiologist for the purpose of diagnosis and significantly reduces the time to perform this procedure.

Keywords: ECG; ECG annotation; the state machine; state diagram; UML; LabVIEW.

INTRODUCTION

Electrocardiography is an inexpensive, but quite informative, method of cardiac research. The result of electrocardiography is an electrocardiogram (ECG) - a graphical representation of the potential difference that arises from the work of the heart [1].

Author Correspondence, e-mail: nikitin@bsu.edu.ru

doi: <http://dx.doi.org/10.4314/jfas.v9i1s.746>



In cardiological practice, computerized ECG analysis is widely used, the main purpose of which is to relieve the physician of much of the routine work. In this case, it is possible to automate recognition of special points and ECG areas. Also, the stage of further analysis of isolated ECG sites is amenable to automation. There are many algorithms for identifying special ECG points and ECG analysis techniques [2, 3].

At the same time, it is necessary to note the following: When developing the hardware-software complex for automated ECG processing, the team of authors proceeded from the assumption that the setting of the final diagnosis is the prerogative of the doctor. In this case, the implementation of the technology of computer analysis of ECG, will significantly speed up the process of diagnosing. But in this case too, the amount of information that needs to be analyzed by a doctor for diagnosis is still significant.

RESEARCH METHODOLOGY

To increase the responsiveness of the specialist's solution of diagnostic tasks, the author's team proposed a system that was implemented in the form of a hardware-software complex for computer analysis of ECG, which allows performing two types of functions:

1. Functions of annotating the ECG, the purpose of which is to perform automatic recognition of specific ECG points and assign to each such point some unique code.
2. Functions that realize the advanced capabilities of automated search for characteristic ECG sections (lines). Each line consists of an ECG section starting at an annotated point with the given code including this point and ending at the next annotated point with the given code, not including this point.

In the future, the above system will be supplemented with functions that implement various methods of ECG analysis. But, even in the existing form, the system significantly increases the productivity of the doctor. This is achieved due to the quality of the user interface of the doctor's workplace, the quality of annotation and the ability to search for arbitrary ECG lines. At present, the author's team has developed a prototype of the automated ECG processing system, which has the following functionality:

1. Input of ECG data from files in PhysioBank format [4].
2. View the ECG signal on the chart with the ability to select and view individual sections (windows) of the ECG.
3. Storage of the parameters of the selected windows and move around them.
4. Automatic annotation of ECG.
5. Moving on annotated points.

6. Moving along R deflection.
7. Selection of the initial code of the annotation and automatically search for ECG lines.
8. Selection of a line and moving through the selected ECG lines.

The idea of the system was considered in work [5]. Some design solutions were considered in work [6]. Let's consider in more detail some aspects of software implementation of the system.

The implementation of the system is performed in the LabVIEW environment [7]. This choice is due to the fact that LabVIEW has extensive signal processing capabilities and allows you to quickly create applications. When annotating the ECG, the source code of the library presented in [8] is used.

RESULTS AND DISCUSSION

The source data of the program is an array of real numbers D of the size $\text{len}D$. The array element $D[i]$ contains the digitized value of the ECG signal at the i -th time (ECG signal count). Abstract The ECG is stored as two AT and AS arrays of $\text{len}A$ size. The element $AT[j]$ contains the annotation code assigned to the j th special ECG point. The $AS[j]$ element contains the reference number of this singular point.

At present, the system of automated ECG processing proposed by the authors allows to recognize 44 special ECG points. With special points of the ECG, some chains of symbols are associated, revealing the semantic content of the annotation at this point. Each chain has a unique code, so you can assume that each symbol of the ECG is associated with a certain symbol of the annotation with this code. The annotations and the corresponding symbol strings are illustrated in Table 1 [5,6].

Table 1

Annotation codes

	Code	Annotation		Code	Annotation
1.	NotQRS	No QRS complex	23.	STCH	STchange
2.	N	Normal rhythm	24.	PACESP	Blocking
3.	LBBB	Left bunch brunch block	25.	T	T wave
4.	RBBB	Right bunch brunch block	26.	RTM	Rhythm change
5.	ABERR	Premature auricular extra systole	27.	LEARN	Learn

6.	PVC	Ventricular premature contraction	28.	FLWAV	Waves of ventricles flutter
7.	FUSION	The fusion of the ventricular and normal rhythm	29.	VFON	The start of fibrillary contraction
8.	NPC	Nodal escape	30.	VFOFF	The end of fibrillary contraction
9.	APC	Auricular extra systole	31.	AESC	Atrial escape beat
10.	SVPB	Premature orectopic supraventricular extra systole	32.	SVESC	Supraventricular escape beat
11.	VESC	Ventricular rhythm	33.	NAPC	no P wave
12.	NESC	Atrioventricular nodal rhythm	34.	PFUSE	Change into sinus rhythm
13.	PACE	Rhythm	35.	FLWAV	Waves of ventricles flutter
14.	Q	Q wave	36.	RONT	Premature ventricular beat from R to T type
15.	ARFCT	Isolated QRS complex	37.	(p	The start of P wave
16.	STCH	STchange	38.	p)	The end of P wave
17.	TCH	The change of deflection	39.	(t	The start of T wave
18.	SYSTOLE	Systole	40.	t)	The end of T wave
19.	DIASTOLE	Diastole	41.	ECT	Pacemaker
20.	MEASURE	Annotation setting	42.	R	R wave
21.	P	P wave	43.	S	S wave
22.	BBB	Left or right bundle of Atrioventricular bundle	44.	RONT	Premature ventricular beat from R to T type

In the development of the automated processing system, automatic programming was widely used [9]. This is due both to a number of advantages of this programming style, and to the fact that the ECG abstract is actually a language where $L = \%_0^*$, where $\%_0$ – is an alphabet consisting of 44 symbols with codes that can be assigned to specific ECG points when

performing automatic annotation. Language L belongs to the class of regular languages, and finite automata are suitable for formalizing the processing of chains of a regular language.

We illustrate the use of finite automata in the design of functions that realize the advanced capabilities of automated search for ECG lines. To store the ECG lines, we used the nested-type structure shown in Fig. ECG lines are stored in the LT array. The i -th element of the uLT array of size $\text{numL} + 1$ points to the beginning of the i -th line, where numL is the number of lines. Thus, the i -th ECG line consists of annotated points with the following codes: $\text{LT}[\text{uLT}[i]]$, ..., $\text{LT}[\text{uLT}[i + 1] - 1]$, and the size of the LT array is $\text{uLT}[\text{numL} + 1] - 1$.

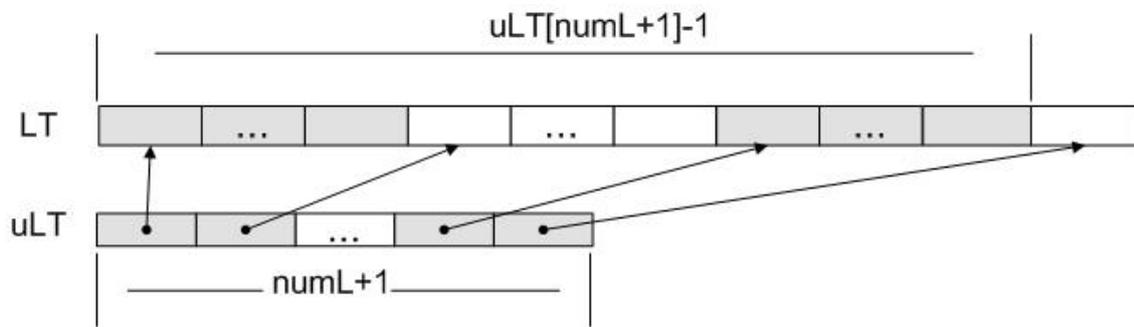


Fig.1. Data structures for storing ECG lines

Formation of this structure is performed by the finite automaton, represented in Fig. 2 in the form of a UML state diagram [10].

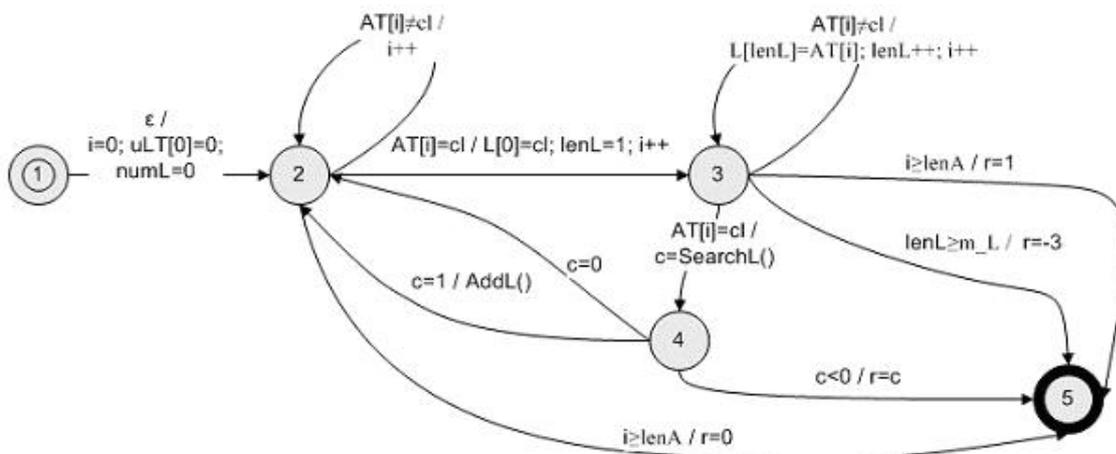


Fig.2. The state diagram of a finite automaton, forming a data structure for storing ECG lines

The annotation code of the starting point of the line is stored in the variable *cl*. The current line is stored in an array *L* of length *lenL*. Also in the automatic machine functions *SearchL ()* and *AddL ()* are used. The *SearchL ()* function searches for the *L* line in the *LT*, *uLT* structure. This function returns: 0 - the annotation line is not unique; 1 - the annotation line is unique and can be added to the structure of *LT*, *uLT*; -1 - the annotation line is unique, but to add it to the *LT*, *uLT* structure, there is no place in the *LT* array; -2 - the annotation line is unique, but to add it to the *LT*, *uLT* structure, there is no place in the *uLT* array. The *AddL ()* function adds the *L* line to the *LT*, *uLT* structure. The following situations are also recognized in the machine: 0 - the initial code of the annotation line was not found; 1 - there are no errors, the annotation lines are highlighted, the structure of *LT*, *uLT* is formed; -3 - for array *L*, a larger size is required.

On the basis of the developed automaton, a function is created. All functions are made in the form of *dll*. This uses the C language and the Microsoft Visual Studio environment. All possible situations of memory shortage correspond to unique codes returned by functions. Functions are called from the LabVIEW environment using the Call Library Function Node. After the function is completed, the returned code is parsed, and if there is an error related to a lack of memory, LabVIEW allocates more memory and the function is called again.

Note that the size of the data used is known at the stage of program execution or is well predictable. Therefore, the situation associated with a lack of memory is extremely rare. Using the same static structures allows you to increase the speed of data processing and abandon the mechanisms of Labview Memory Manager, which in turn allows you to create independent libraries from LabVIEW external functions.

After all ECG lines are found, the user is given the opportunity to select the line of interest. After this choice is made, a list of the initial counts of the selected line is formed. These operations are performed by the automaton illustrated in Fig. The number of the selected line is written in the variable *nL*. The list of initial samples is stored in an *LS* array of size *lenLS*. The following situations are recognized in the machine: 0 - the line of annotation with the number *nL* does not exist; 1 - there are no errors, *LS* array is formed; -4 - for the *LS* array, a larger size is required. A corresponding function is created on the basis of this automaton. The implementation and invocation of this function is carried out in a similar way to the function discussed above.

After the list of initial counts of the selected line is formed, the user is given the opportunity to navigate through these samples in order to search for characteristic ECG sections and analyze them.

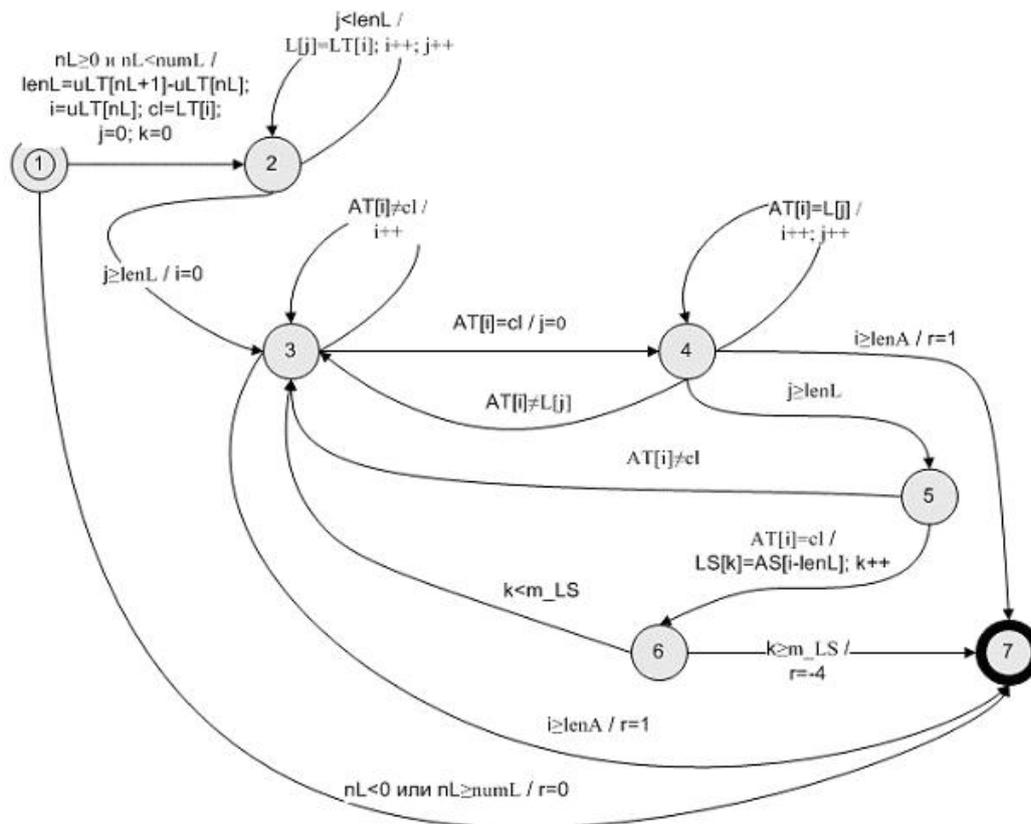


Fig.3. The state diagram of a finite automaton, generating a list of initial samples of a selected line

Thus, the process of functioning of two automata, which serve for organization of search and viewing of ECG lines, is considered. These machines were used in the design of automated means of searching for characteristic ECG regions. In the system of automated ECG processing, finite automata were also used in the design of other functions, for example, calculation of the ECG isoline, transformation of the input data formats, etc. The use of finite automata in the design of the system made it possible to visualize the logic of the system operation, avoid design errors, achieve high system speed and In general, speed up the process of its creation.

The main window of the ECG automated processing system is illustrated in Fig. With the help of element 1, the ECG file is selected. When the button 3 is clicked, the ECG data is read from the selected file and an automatic annotation is performed, using a set of filters from the directory 2. After the annotation is completed, the contents of the elements displaying information on the current ECG (elements 4,5,11,12, etc.) . Note that the annotation time takes no more than 2 seconds for 4 MB files.

Then, using the cursors 6, you can select the desired part (window) of the ECG and save it by pressing the button 10. You can move through the saved windows. The selected window is displayed in graph 5. (See Fig.4)



Fig.4.The main window of the ECG computer analysis system

The numbers of the annotated points of the current window are displayed in element 11. The same element moves through the annotated points. The cursor 8 is positioned on the current point. Similarly, the element 12 and the cursor 9 are used to display the RRS points and move along them.

For convenience of searching for specific ECG sections (lines). Using the group of elements 13, you can select the initial line code. After this selection, the lines are searched automatically. The lines found are displayed in the group of elements 14. The line can occur several times in the current ECG window. If you select the line of interest, then the starting point numbers of the selected line will be displayed in the element 15. Note that the search time of lines and start points of the selected line does not exceed 0.1 seconds. Moving along the starting points of the selected line is carried out in the same way as moving through

annotated points and RRS points. The cursor 7 is positioned on the starting point of the selected line.

Figure 4 illustrates the ECG window, on which two lines with the initial code "t" are visible. The line "t) (p P p) NQR s) (t T" is selected.) On the ECG chart, the cursor 7 is set to the start point of this line with the number 33230. The time of this point from the beginning of ECG removal is 259.609 seconds. The second line is "t) ((P) NQR) (t T », differs from the first one in that the s is absent (B.) In the first line, the tooth s is present (A.) It would be difficult to detect such ECG sections without using line search facilities.

SUMMARY

1. In the given article it was proposed an approach to the construction of the system of computer analysis of ECG, the characteristic feature of it is automatic recognition of singular points of ECG and automatic detection of ECG characteristic areas.
2. The prototype of the system was implemented
3. It is shown that the wide use of finite automata in the design and construction of a prototype the system of computer analysis allows, on the basis of a visual representation of the logic of the operation of the system, to avoid design errors, ensure high system speed, and accelerate the process of creating the system as a whole.

CONCLUSION

The analysis of test results developed by the team of the prototype of the computerized ECG analysis system with the advanced capabilities of automated search for characteristic sites allows us to conclude that the reliability of ECG interpretation by a specialist increases with a substantial reduction in the time required to perform this procedure. This positive effect is achieved through the implementation of the procedure for automatic recognition of special ECG points and the use of automated means of searching for ECG lines.

REFERENCES

1. Recommendations for the Standardization and Interpretation of the Electrocardiogram 2009. URL: <http://content.onlinejacc.org/cgi/reprint/49/10/1109.pdf>.
2. Petrov, S.P., E.V. Epishina and V.V. Voronin, 2014. Evaluation of Algorithms for Pattern Recognition Tasks of Automatic Analysis of ECG. Eurasian Scientists Union (ESU). 8: 27-29. URL: <http://euroasia-science.ru/files/arhiv/28-29.11.2014/p8.pdf#page=27>(in Russian).

3. Nikitin, V.M., Efremova, . . , Lipunova, . . , Ivanov, I.I., Anohin, D. . , Ermilov, .V., Gubareva, N. . , 2013. Setting algorithm for the diagnosis of coronary artery disease unwrapped programs ARM-cardiologist expert system. Belgorod State University Scientific bulletin Series Medicine Pharmacy, 16(135): 64-69 (in Russian).
4. PhysioBank Archive Index URL: <http://physionet.org/physiobank/database/>
5. Efremova, O.A., V.M. Nikitin, V.V. Muromtsev, E.A Lipunova and L.A. Kamyshnikova. 2016. ECG computer analysis system with the advanced features of automated search and identification of its diagnostically significant changes. International Journal Of Pharmacy & Technology IJPT. 8(2): 14174-14181.
6. Nikitin, V.M., V.V. Muromtsev, O.A. Efremova, E.A. Lipunova L.A. and Kamyshnikova. 2016. Software implementation of the system of computer analysis of ecg with advanced automated search for characteristic areas. Research results «Medicine and pharmacy». 2(2): 10-16 (inRussian).
7. National Instruments. URL: <http://www.ni.com/> .
8. ECG Annotation C ++ Library. URL: <http://www.codeproject.com/Articles/20995/ECG-Annotation-C-Library>.
9. Shalyto, A.A. 2008. The paradigm of automatic programming. Scientific and Technical Herald of the St. Petersburg State University of Information Technologies, Mechanics and Optics. Automaticprogramming. 53: 3–23.
10. Documents associated with Unified Modeling. URL: UML <http://www.omg.org/spec/UML/2.5/>.

How to cite this article:

Nikitin V M, Muromtsev V V, Efremova O A, Lipunova E A, Kamyshnikova L A. Automated detection and recognition of diagnostically significant ecg changes. J. Fundam. Appl. Sci., 2017, 9(1S), 929-938.