UDC 618.3 – 004.896

DATA READING SOFTWARE

Meshkova K.E.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Davydov M.V. – PhD, associate professor, First Vice-Rector of the BSUIR

Annotation. Software has been developed for reading the impedance characteristics from the immitansometer control program. It has been found that the use of data mining reduces the time of data recording during research and simplifies data storage.

Keywords: immitansometer, control program, impedance characteristics.

Introduction. The study of the impedance characteristics of biological fluids includes software modeling and calculation of characteristics with subsequent testing in practice. In this article, the author has developed a system for automated reading of impedance characteristics from an immitansometer, which increases the processing speed and further data analysis.

The main part.

To prepare and conduct the experimental part to determine the impedance characteristics of the amniotic fluid, it is necessary to solve the following tasks:

– set up the device and make a solution for the study;

– to conduct research at different frequencies according to the developed methodology;

– record the received data.

The adjustment of the device of the E7-20 immitansometer is carried out according to the technical documentation. According to the previously developed technique, a solution of amniotic fluid and deionized water is made, and a circuit for measuring with two types of electrodes is also assembled [1]. The recording of the received data was previously performed without the use of software, which extremely slowed down the work.

There is a program for this series of immitansometer with the ability to perform all the functions of a stationary immitansometer using the screen and control buttons [2]. The immitansometer control program externally represents a device display with unlimited access to the interface. It works only when a personal computer is connected to the device using an RS232 wire without the possibility of saving data and recording it.

To solve the third problem, a program code has been developed with the ability to quickly read digital characters from the built-in immitansometer control program and record the results.

There are four integrated modules in the program:

1) OpenCV (this program works on the basis of artificial intelligence, designed to read data from a picture by color, create a picture based on a written text query, and more) [3];

2) PyTesseract (trained on the basis of OpenIA to perform machine learning workflows for data transformation and creation of semantic search applications);

3) NumPy (built-in math library in python);

4) Os (built-in library in python, which is responsible for all inputs and outputs).

The algorithm of the code. The program provides 2 modes of operation: the first mode is test, the second is full-fledged. In the test mode, the correctness of the code operation is checked at the current location of the immitansometer control program on the computer desktop. The full-fledged mode duplicates the behavior of the test scenario, excluding the display of the screenshots received. It has a built-in loop that has 14 clock cycles, starting from zero, and a waiting time of 6 seconds. After each clock cycle, it saves to a file.

We will describe the first mode of operation, since it is taken as the basis of the second mode. The readable logical process starts in the main() method. The first thing the program does is take a screenshot of the desktop. To do this, the ImageGrab.grab function is used, in which a bbox frame is set and an image in RGB is returned from pixels. In the listing of the program, the parameters of the x and y coordinates of the beginning of the frame area and the x and y of the end of the frame

are set in such a way as to get a ¼ of the screen in the left corner of the monitor. The OpenAI used was initially trained to search for green fields, which are the immitansometer screen. At the moment, our snapshot does not satisfy this condition, so the GetGreen(img) method is called next. The method allows you to convert the color to COLOR_BGR2HSV by setting up a filter with a range of green color. Next, we transfer cv2 (OpenCV) to processing, which, using a filter and computer vision, outputs the coordinates of the location of the green color in the image. Save it to the mask variable. Next, in the getScrLocal(mask, img) method, we set the display area of the mask immitansometer and cut out two display rectangles from the img snapshot. One image contains information on the capacitance parameter, the second on the tangent parameter of the dielectric loss angle. We call the GetGreen() method to get a mask of all green pixels and output it to the screen using cv2.imshow(). Then the analysis of the correctness of the work is performed. Good readability of parameters and their values is visually checked. With cropped values, it is necessary either to shift the control program of the immitansometer, or to change the coordinate parameters in the code. After closing the screenshots, the program starts counting the results. The resulting masks from the capacitance parameters and the tangent of the dielectric loss angle are sent to pytesseract for decoding into a machine-readable byte language. Data is recorded in the form of numeric characters, therefore we apply the conversion operation to the string format and output the results to the terminal.

Rules for working with the code. First of all, you need to run the program code and call the python command in the terminal main.py. Next, the text is displayed with the option to select the operating mode. In the terminal, we prescribe what mode of operation is needed at the moment. When the test mode is selected, the code is executed according to the algorithm specified above, and all screenshots are shown to the user at the moment. Closing the screenshot window data, the values are read, which are transferred to the terminal for evaluation. At the same time, the program has finished its actions and offers to select the operating mode again with a new line in the terminal. When selecting the full-fledged operation mode, a text is displayed with a note about the correct location of the window of the immitansometer control program. Next, the program asks you to enter the data of the voltage of the field being created, the measurement frequency, the offset voltage and the name of the file folder. We can name the folder using numbers and letters of the Latin alphabet. After filling in, the code is run and the values of the capacitance, the tangent of the dielectric loss angle and the clock cycle number are displayed in the terminal. After recording 14 values in a row, the program issues a notification about the termination of data recording and the termination of the work cycle. To write again, you need to enter the python command main.py.

Conclusion. A program code was used to conduct studies of the impedance characteristics of the amniotic fluid. The proposed application simplified the recording of research data. Storing the history numbers, the type of electrodes used and the frequency of the study in separate folders with notes helps to quickly search among a huge database. Each new measurement is saved as separate files with the txt extension, convenient for conversion to Microsoft Excel. In the future, this program code can be changed to automatically fill in data in the desired form in Microsoft Excel tables and plotting.

References

1. Meshkova, K.E. A software model of a system for analyzing the impedansometric characteristics of biological fluids / K.E. Meshkova // Collection of scientific articles / XII International. Sci.-tech. Conf. – Minsk, 2022. – Issue : Medical electronics and new medical technologies. – pp. 93-97.

2. MNIPI [Electronic resource] / Immitance meter E7-20. – Moscow, 2022. – Access mode: http://www.mnipi.ru . – Access date : 10.03.2023.

3. OpenCV [Electronic resource] /Library. – USA, 2023. – Access mode: http://www.opencv.ai . – Access date : 03.16.2023.