

# Intelligent information system for the determination of iron in coagulants based on a neural network

Andrii Safonyk<sup>a</sup>, Maksym Mishchanchuk<sup>a</sup> and Volodymyr Lytvynenko<sup>b</sup>

<sup>a</sup> National University of Water and Environmental Engineering, 11 Soborna St, Rivne, 33028, Ukraine

<sup>b</sup> Kherson National Technical University, Kherson, 73009, Ukraine

## Abstract

The construction of an intelligent system for determining the concentration of iron in the coagulant by its color on the basis of a neural network is considered. Based on the analysis of different types of neural networks, the most suitable neural network architecture was selected to solve the problem of determining the concentration of iron in the coagulant. The process of architecture design, analysis of teaching methods, data preparation for neural network training to determine the concentration of iron in the coagulant by its color is described. Developed the structural and functional scheme of the neural network, which consists of input, hidden and output layers, activation functions are described. Analyzed the accuracy of neural network learning by comparing data obtained using different optimizers using a TensorFlow library. The developed web application can be used as a component of information and analytical system of automated control electrocoagulation cleaning technological process.

## Keywords

Neural net, TensorFlow, iron concentration, coagulant, intelligent information system, photocolometer method.

## 1. Introduction

Wastewater treatment is one of the problems of modern humanity. Recently, small water treatment plants have become popular for water purification. Equipping water treatment facilities with such systems is especially important for agriculture, light industry, etc. But the use of reagents leads to the creation of services to ensure their delivery and storage, which is costly compared to electrochemical methods, which in turn allow to extract reagents on site from available raw materials. One of the most promising methods that provide this opportunity is the electrocoagulation method. Currently, more and more attention is paid to the study of this process using mathematical models, which helps to improve the design features of the device and operating costs, as well as to predict the efficiency of the process in a wide range of operations with minimal production costs. Also, these researches are the basis for the development of automated control systems for electrocoagulation processes. The process of obtaining a coagulant by electrocoagulation involves complex and expensive field experiments to determine the content of a useful element (iron) in the coagulant. One of the laboratory methods for finding iron in a coagulant is the photocolometer method. As for photocolometry technologies, it is the development of technology for the absorption of light by matter due to the visible spectrum, the conversion of light energy into electrical energy. This technology allows you to more accurately and quickly assess the quality of multicomponent compounds. In addition, this method is included in the standardized methods for determining the iron content in water, because the water color intensity and the optical density of the medium change with changing iron concentration. At present, almost no sensors for determining the concentration of iron coagulant in real time, so

---

IntelITSIS'2021: 2nd International Workshop on Intelligent Information Technologies and Systems of Information Security, March 24–26, 2021, Khmelnytskyi, Ukraine

EMAIL: a.p.safonyk@nuwm.edu.ua (A. Safonyk); mishchanchuk\_ak17@nuwm.edu.ua (M. Mishchanchuk);

immun56@gmail.com (V. Lytvynenko)

ORCID: 0000-0002-5020-9051 (A. Safonyk); 0000-0002-1197-4738 (M. Mishchanchuk); 0000-0002-1536-5542 (V. Lytvynenko).



© 2021 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

urgent task is to develop an automated information system for the electrochemical production of coagulant based on photocolimeter analysis.

Problem formulation. Given that the change in the concentration of iron in water changes the color and color intensity of the solution, the experimental setup for laboratory research was decided to supplement the automated information system that will analyze the color and intensity of light in real-time.

## 2. Related works

During the analysis of literature sources it was found that a number of authors conducted research using photocolometry and photocolometric analysis, so in [1] developed a digital colorimeter with a mobile application for iOS, using Euclidean distance theory to determine the content that detects color components, remembering the data of red, green and blue, as well as the value of hue, saturation, brightness calculated using standard color theory. A simple and quite effective method for determining the concentration of iron in water samples according to the rules of the United States Environmental Protection Agency is shown in [2]. In the research [3], the iron content in water was determined using digital colorimetry of the image using a webcam. The least squares regression was used to obtain an equation describing the dependence of iron content on color, based on shades of red, green and blue, as well as their saturation and transparency. Two methods for determining the concentration of chromium (Cr) and iron (Fe) are described in [4]. All this work is based on tests to determine the concentration in the laboratory, which is very expensive and time consuming. In [5], a portable device was developed that allows on-site colorimetric analysis and offers wide availability with limited resources. The article [6] describes the development of a mobile colorimetric analysis tool - PhotoMetrix, which uses simple linear regression methods for one-dimensional analysis and analysis of the main components for multidimensional analysis of research. These images are captured by the device main camera and converted into red, green and blue histograms.

Today there are a large number of different types of neural networks, as well as options for implementing networks of the same type. Among the different types of neural networks, the most common are:

1. Multilayer perceptron;
2. Convolutional neural network;
3. Recurrent neural network.

Consider in more detail the networks types.

Multilayer Perceptron (MLP) - is a class of direct artificial network that has at least three layers. The first is the input layer with neurons from  $I_1$  to  $I_n$ , the second is the hidden layer with neurons from  $H_1$  to  $H_m$  and the third is the output layer with neurons from  $O_1$  to  $O_k$ . These layers are connected so that each neuron of the previous layer is connected to each neuron of the next layer [7] (see Fig. 1). There can be more than one hidden layer, and networks that contain multiple hidden layers are called "deep" neural networks. MLP neural networks are often used to solve classification and forecasting problems [8].

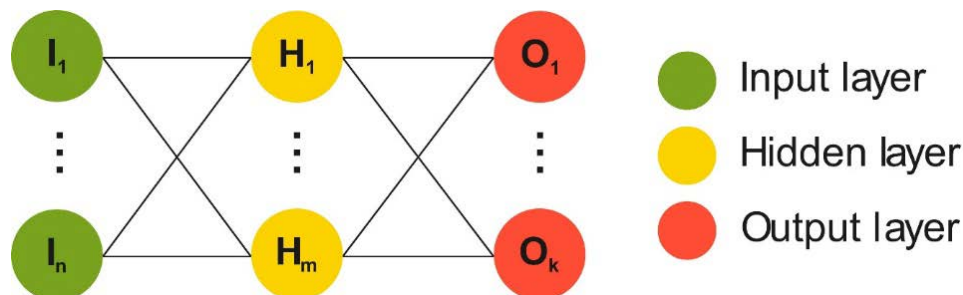


Figure 1: Multilayer perceptron

Convolutional neural networks are a class of deep neural networks that are widely used to analyze visual images. Such neural networks use mathematical convolution operations. These neural networks have the same structure as MPL networks.

Recurrent neural networks are a class of networks in which connections between nodes form a time-oriented graph. Such networks include long/short term memory networks [9] (LSTM). The structure of such a network is shown in the Figure 2, where neurons from  $I_1$  to  $I_n$  is input layer neurons, from  $H_1$  to  $H_m$  is hidden layer neurons with memory and repeat possibility, from  $O_1$  to  $O_k$  is output layer. This neural network is often used for word sequence analysis, speech recognition, or handwriting. These neural networks do not work well with data that can be represented as tables.

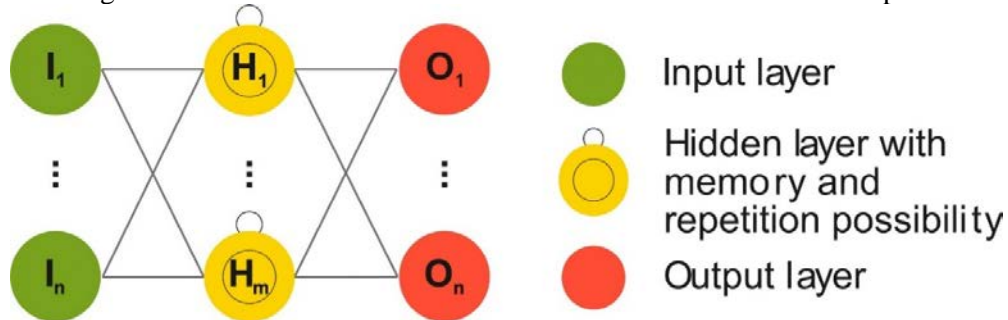


Figure 2: LSTM neural network

The problem of determining the concentration of iron in the coagulant by its color can be attributed to the class of problems of pattern recognition based on the classification of input data. The most common type of neural networks for this class of problems are multilayer perceptron.

The developed neural network has an input and output layer, between which there are two hidden layers (see Fig 3). Input layer has 5 neurons and activation function is hyperbolic tangent. In two hidden layers of 10 neurons and with the seul and relu activation functions, respectively. In the source layer 1 neuron and as an activation function used the exponential function.

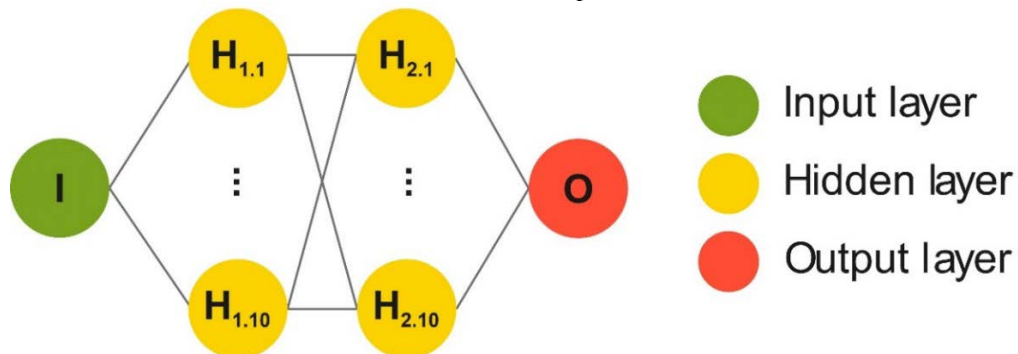


Figure 3: Scheme of the developed neural network

### 3. Proposed methodology

To solve this problem, the software for the Raspberry Pi 4 microcomputer and the connected TCS230 color sensor was developed. The developed software determines the concentration of iron in the coagulant using a neural network.

To select the type of network that will be used to solve the problem of determining iron in the coagulant, consider in more detail the networks types.

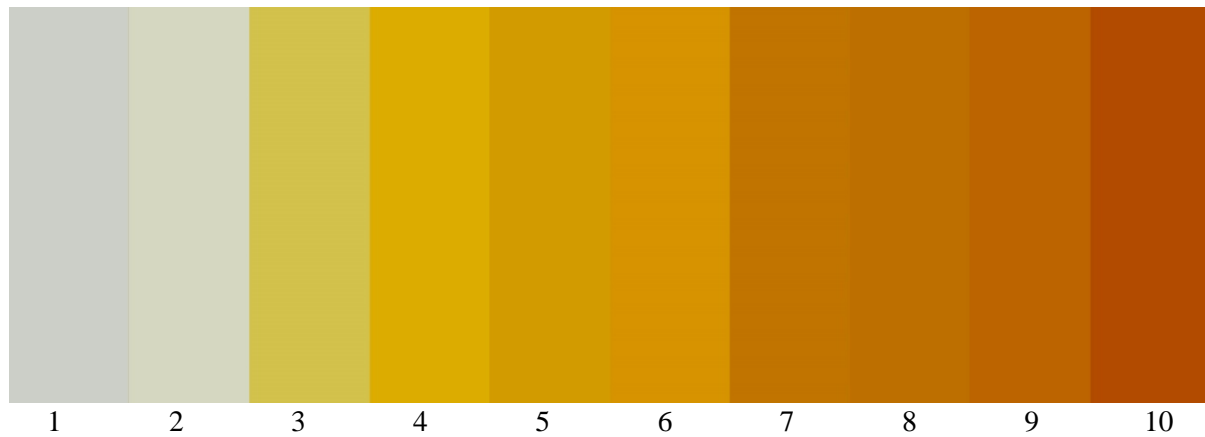
The input for the neural network is the hue (one of the three main color characteristics along with saturation and brightness) parameter from the HSL color model. The initial result of the neural network is the concentration of iron in the coagulant.

To generate a training and test dataset, a laboratory research of various samples of coagulant was performed, where the concentration of iron in the coagulant and RBG parameters for each of these samples were determined (see Fig 4).

On the table 1 shows the results of the process experimental research, namely: determining the concentration of trivalent iron and the color of the substance at different times.

**Table 1**  
Experiment data

№	Time (m)	Concentration (mg / dm <sup>3</sup> )	Red	Green	Blue
1	6	0.8	204	207	200
2	12	1.1	214	215	193
3	18	3.7	211	194	75
4	24	4.5	220	173	54
5	30	6.3	210	155	49
6	36	6.9	215	147	41
7	42	9.1	193	116	38
8	48	9.6	189	111	33
9	54	11.8	188	100	27
10	60	12.6	178	74	11



**Figure 4:** The color of a substance changes over time

The RGB parameters were converted to HSL color space and the hue parameter was selected to determine the concentration.

RGB color space is converted to HSL using the following relationships:

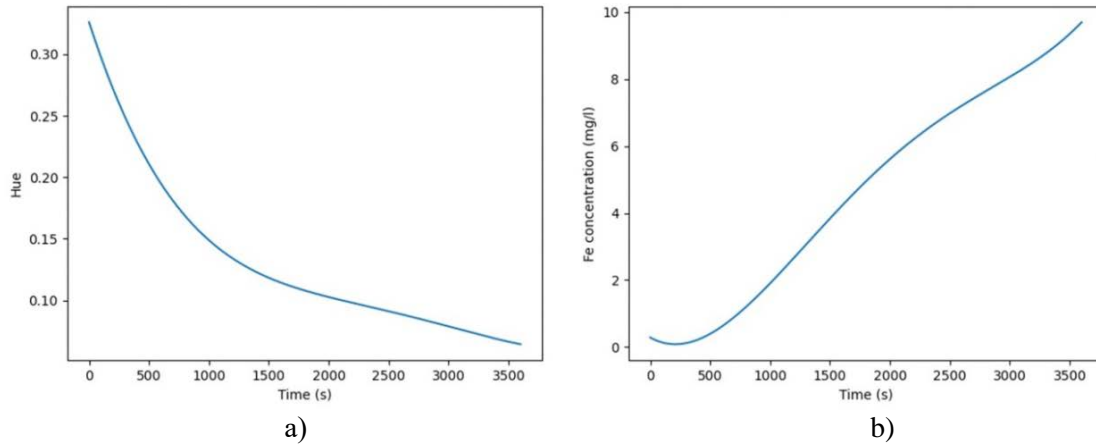
$$H = \begin{cases} \text{undefined} & \text{if } MAX = MIN \\ 60^\circ \times \frac{G-B}{MAX-MIN} + 0^\circ, & \text{if } MAX = R \\ & \text{and } G \geq B \\ 60^\circ \times \frac{G-B}{MAX-MIN} + 360^\circ, & \text{if } MAX = R \\ & \text{and } G < B \\ 60^\circ \times \frac{B-R}{MAX-MIN} + 120^\circ, & \text{if } MAX = G \\ 60^\circ \times \frac{R-G}{MAX-MIN} + 240^\circ, & \text{if } MAX = B \end{cases} \quad (1)$$

$$S = \frac{MAX-MIN}{1-|1-(MAX+MIN)|} \quad (2)$$

$$L = \frac{1}{2}(MAX + MIN), \quad (3)$$

In these expressions R, G, B are the color values in the color model RGB, MAX and MIN are the maximum and minimum values of the parameters R, G, B, respectively, H is a tone, S is a saturation, L is a lightness.

Next, the experimental data were approximated to determine the dependence of hue over time and the concentration of iron in the coagulant over time. Based on the obtained expressions, 3600 HUE points and iron concentrations were generated (see Fig 5 a, b).



**Figure 5:** Dependence: a) of hue change over time; b) of concentration change over time.

From the obtained points, 3240 learning points and 360 neural network testing points were selected. The table 1 shows the input and output parameters of the neural network.

**Table 2**

Example data

No	Hue	Concentration
1	0.2110	0.4009
2	0.1969	0.7229
3	0.1073	4.8701
4	0.0649	9.5388

To teach the neural network, optimizer features from TensorFlow library that improve learning by updating knowledge in the network are used. In order to choose the best optimizer for a neural network, several neural networks with different optimizers were developed and analyzed. Next, each neural network was trained using a generated dataset.

In particular, 8 optimizers available in the TensorFlow library were investigated. As can be seen from table 3, where the error is calculated as the standard deviation of the results obtained from the neural network and the approximate data, some optimizers, such as Ftrl, Adadelata, do not follow the neural network training, while others, such as SGD and RMSprop, teach the developed neural network well. The SGD optimizer performed the best results with an error of 6.91% on the test dataset.

As can be seen from the figure 6 of the SGD optimizer has achieved this result in 20 epochs of learning.

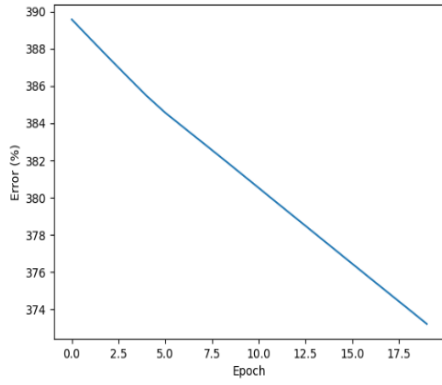
After analyzing the learning speed of the neural network, the SGD optimizer was selected for in software to determine the concentration of iron in the coagulant.

**Table 3**

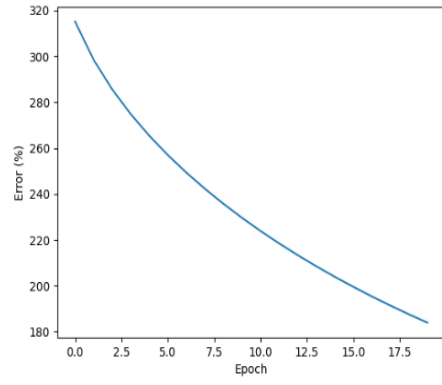
Optimizers mistake (standard deviation)

Optimizer	SGD	RMSprop	Adam	Adadelata	Adagrad	Adamax	Nadam	Ftrl
Mistake %	6.91	8.28	9.33	373.22	183.92	9.05	10.84	370.43

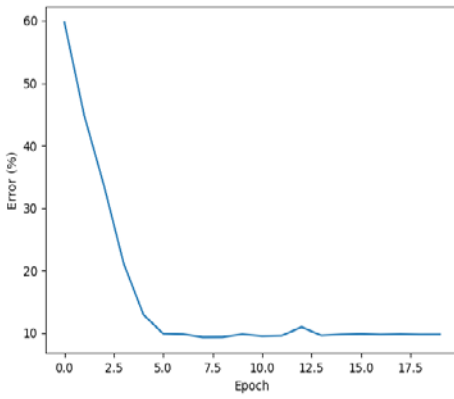
For easy access to the measurement results of the sensor and the concentration determined by the neural network, a web application which uses a Raspberry Pi 4 microcomputer as a server was developed. The access to control panel (see Fig. 7) of this device can be established after connecting this computer to the network at the network address of this device.



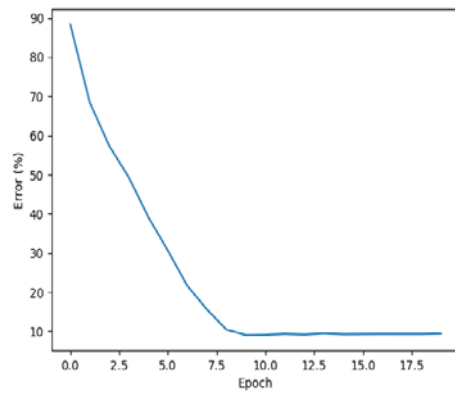
a. Adadelta



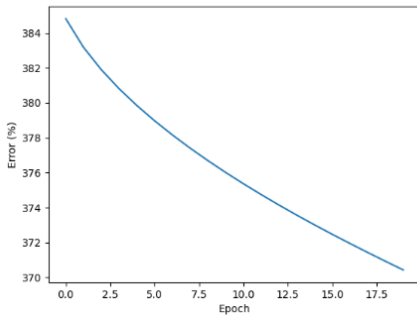
b. Adagrad



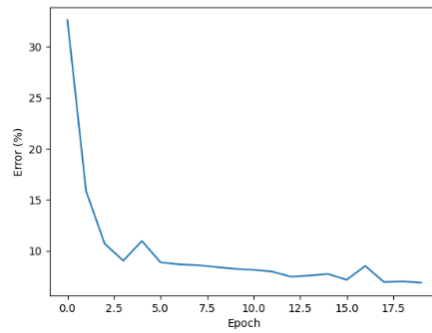
c. Adam



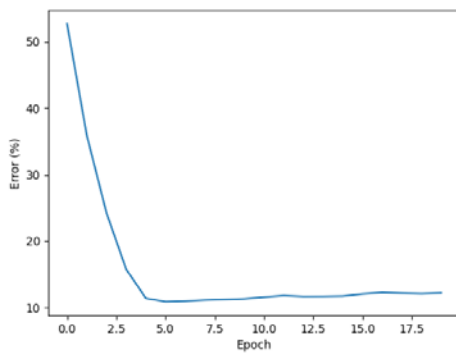
d. Adamax



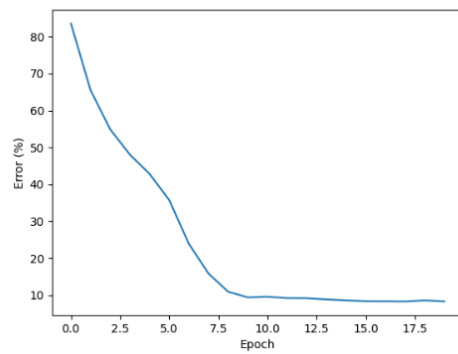
e. Ftrl



f. SGD

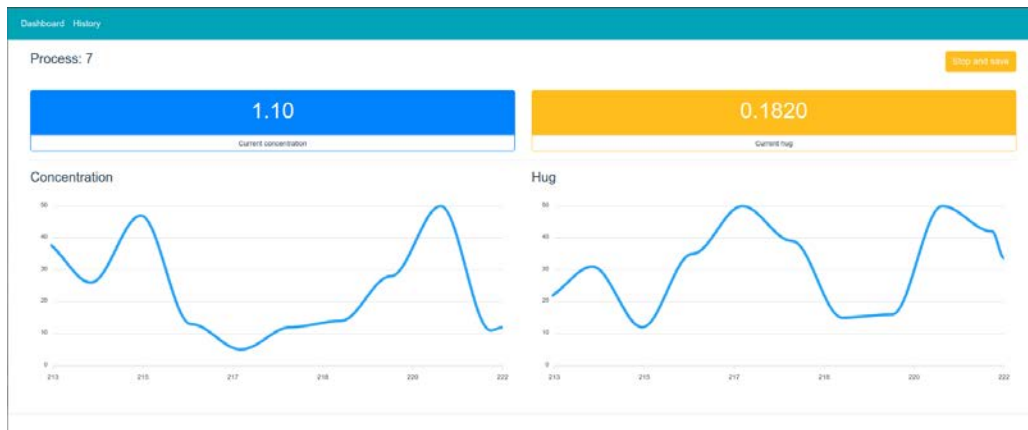


g. Nadam



h. RMSprop

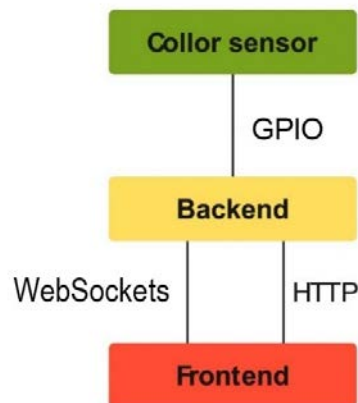
Figure 6: Scheme of the developed neural network



**Figure 7:** Program interface

Using the web interface, you can start the measurement process and see the history as well as the current measurement values.

The Vue.js [10] JavaScript framework was used to develop the web interface. The main selection criteria were ease of use and much smaller size compared to similar React.js and Angular. Vue.js uses a state manager to display data to the user.



**Figure 8:** System communication scheme

Vuex was used as such a state manager. States in such a manager are replaced by mutations that require sharing in their network. This mechanism allows you to split the code that modifies the data and retrieves it from the server.

To implement communication with the server (see Fig 8), two libraries are used: the axios library for executing asynchronous HTTP requests and the socket.io library for working with web-socket.

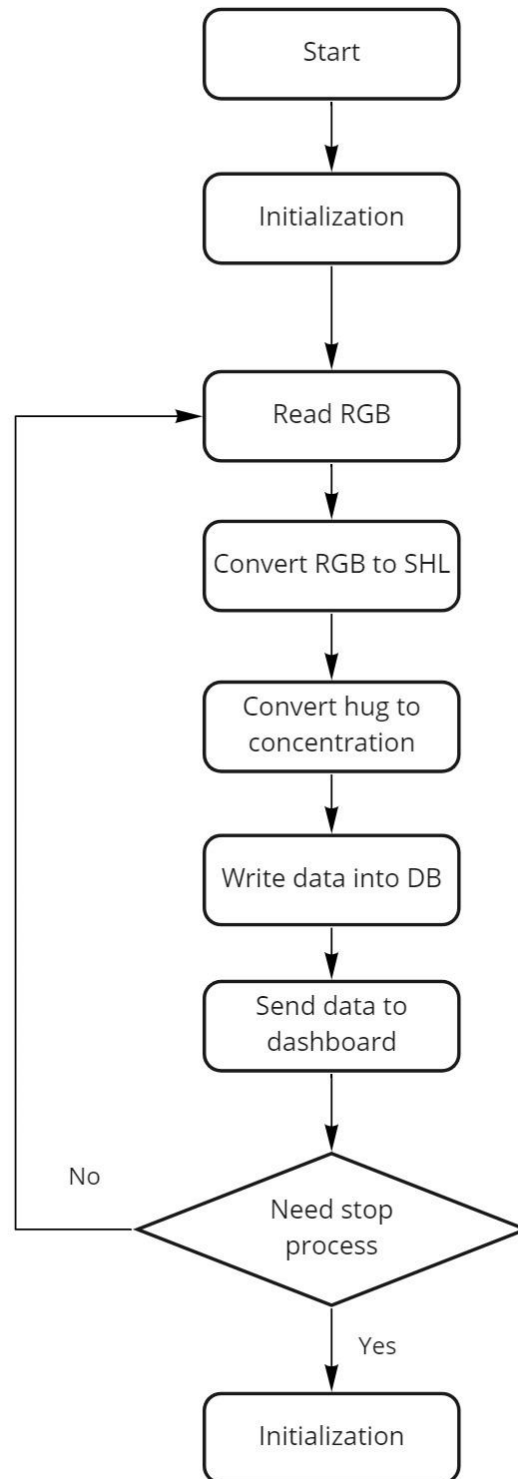
The server part of the program is responsible for polling the color sensor, determining the iron concentration based on the obtained color, processing user commands, recording the received and calculated data in the database.

The python framework Django was chosen to develop the server part [11].

With the help of this framework, a service for data definition and storage was developed. It was also used to create a web-socket server, which allows you to update the data on the web-panel in real time.

Consider in more detail the work of the server part of the program. When the user clicks the “Run” button in the web interface, a request is sent to the web server being processed by Django. During this process the data received in the request are validated, a check for an existing running process is made, and in case if all the conditions are met, a record in the database of the newly started process is created and the background process starts.

When the background process starts, the connection to the web-socket server channel to transfer data to the web panel is initialized. On the same communication channel, this process will stop after the user clicks the "Stop" button on the web panel.



**Figure 9:** Process block diagram

After initialization (see Fig. 9) the infinite cycle is started. In this cycle at 1 second intervals the color sensor is interrogated, the color parameters obtained from the RGB sensor are converted into HSL color space, then the hue parameter is transmitted to the input of the trained neural network, where it is converted into the concentration of iron in the coagulant. Then the parameters from the color sensor, the converted color parameters and the determined concentration are stored in the database and sent to the web-panel using a web-socket server.



The condition for ending the cycle is that the user stops it and the server processes the corresponding request with sending a message about the completion of the background process using web-socket and updating the record in the process history after its completion.

## 4. Conclusions

The development and application of neural networks to determine the concentration of iron in the coagulant by its color is considered. Software to determine the concentration of iron in the coagulant in the form of a web application for displaying measurement data in real time and saving the measurement history was developed. To solve this problem, the software for the Raspberry Pi 4 microcomputer and the connected TCS230 color sensor was developed. The developed software determines the concentration of iron in the coagulant using a neural network. The color sensor determines the color parameters: R, G, B, these parameters are translated into the color space HSL for easier further work with color parameters. Different types of neural networks were considered, among them the neural network that is best suited to solve the problem of determining the concentration of iron by the color of the coagulant was chosen. Based on the analysis of the TensorFlow optimizers, the SGD optimizer was selected with an error of 6.91% on the test dataset. An automated information system for determining iron-containing coagulant based on photocolometric analysis has been developed. It consists of a flowing opaque cell through which the investigated liquid is passed at a constant flow rate, and a unit for processing and storing data which allows to reduce human participation in the measurement process and to ensure the continuity of the measurement process due to the lack of need for the sampling of the test material, as well as to reduce the cost of the measurement process.

## 5. References

- [1] P. Masawat, A. Harfield, N. Srihirun, A. Namwong, Green Determination of Total Iron in Water by Digital Image Colorimetry, *Analytical Letters*, volume 50, issue 1 (2017) 173-185. doi:10.1080/00032719.2016.1174869.
- [2] Sreenivasareddy Annem, Determination of Iron Content in Water, *OPUS Open Portal to University Scholarship, Governors State University, Summer (2017)* 1-19.
- [3] Juan A. V. A. Barros, Fagner Moreira de Oliveira, Guilherme de O. Santos, Célio Wisniewski, Pedro Orival Luccas, Digital Image Analysis for the Colorimetric Determination of Aluminum, Total Iron, Nitrite and Soluble Phosphorus in Waters, *Analytical Letters*, volume 50, issue 2 (2017) 414-430. doi: 10.1080/00032719.2016.1182542.
- [4] M. L. Firdaus, W. Alwi, F. Trinoveldi, I. Rahayu, L. Rahmidar, K. Warsito, Determination of Chromium and Iron Using Digital Image-based Colorimetry, *Procedia Environmental Sciences* volume 20 (2014) 298 – 304. doi: 10.1016/j.proenv.2014.03.037.
- [5] G. S. Luka, E. Nowak, J. Kawchuk, M. Hoorfar, H. Najjaran, Portable device for the detection of colorimetric assays, *Royal Society Open Science* 4 11 (2017). doi: 171025. doi:10.1098/rsos.171025.
- [6] G. A. Helfer, V. S. Magnus, F. C. Böck, A. Teichmann, M. F. Ferrão, A. B. da Costa, PhotoMetrix: An Application for Univariate Calibration and Principal Components Analysis Using Colorimetry on Mobile Devices, *Journal of the Brazilian Chemical Society*, 28 2 (2017) 328-335. doi: 10.5935/0103-5053.20160182.
- [7] Multilayer perceptron, 2020. URL: [https://en.wikipedia.org/wiki/Multilayer\\_perceptron](https://en.wikipedia.org/wiki/Multilayer_perceptron).
- [8] J. Brownlee. When to Use MLP, CNN, and RNN Neural Networks, 2019. URL: <https://machinelearningmastery.com/when-to-use-mlp-cnn-and-rnn-neural-networks>.
- [9] Long short-term memory, 2021. URL: [https://en.wikipedia.org/wiki/Long\\_short-term\\_memory](https://en.wikipedia.org/wiki/Long_short-term_memory).
- [10] Vue.js, 2021. URL: <https://en.wikipedia.org/wiki/Vue.js>.
- [11] Django (web framework), 2021 URL: [https://en.wikipedia.org/wiki/Django\\_\(web\\_framework\)](https://en.wikipedia.org/wiki/Django_(web_framework)).