

Multiple Fault Detection and Smart Monitoring System Based on Machine Learning Classifiers for Infant Incubators Using Raspberry Pi 4



Maryam A. Mahdi*, Sabah A. Gittaffa, Abbas H. Issa

Department of Electrical Engineering, University of Technology, Baghdad 10066, Iraq

Corresponding Author Email: eee.20.60@grad.uotechnology.edu.iq

<https://doi.org/10.18280/jesa.550609>

ABSTRACT

Received: 13 October 2022

Accepted: 31 October 2022

Keywords:

fault detection, infant incubator, Raspberry Pi 4, decision tree (DT), support vector machine (SVM), neural network (NN)

The most significant and sensitive component of the biomedical field is the care of newborn newborns. Because of their gestational age or birth weight, certain newborn newborns are at a higher risk of death. Because of their unmet demand for warmth, the majority of preterm newborns born between 32 and 37 weeks of gestation die. The neonatal incubator is a device that provides a regulated and closed environment to preterm newborns. In this paper and depending on technological progress, an intelligent system has been designed to monitor the performance of the incubator sensors depending on four features (temperature, humidity, fan current, and heater current) to detect any fault in the system. The intelligent system employs a low-power computing device to detect the fault in sensors, like the Raspberry Pi 4, which delivers the data from the incubator's sensors. For classification, tasks adopted many algorithms like Decision Tree (DT), Support Vector Machine (SVM), and Neural Network (NN), then send alerting messages (e-mail, text message) to the nurse or doctor via Wi-Fi. The promising results of the proposed method accuracy come as 98% of DT and SVM while 97.3% of NN.

1. INTRODUCTION

The incubator provides a constant temperature as well as a certain level of moisture to keep the baby in a safe and healthy condition. Any temperature change will turn on or off the fan to adjust the optimum temperature [1, 2]. Any fault in any of these parameters will lead to an imbalance in the system and the loss of a safe environment. Therefore, must work on monitoring and detecting the input and output faults in the various sensors of the incubator [3]. A fault indicates a system malfunction, which leads to unsatisfactory performance and, consequently, the deterioration of the system [4, 5]. The output of the system falls between two specific limits (the upper and lower); when one of the two is exceeded, that means there is a failure in the sensor, which is known as a fault [5, 6]. In this paper, an intelligent system is designed to monitor the baby incubator's most essential sensors, including temperature and humidity, as well as the fan current and heater current to detect the fault and alert about it. Providing constantly monitor to manage these conditions to preserve temperature and humidity in the baby incubator is crucial for the appropriate development of premature babies. The main purpose of an infant incubator is to keep the temperature of a baby's care at (25-35.5) degrees Celsius and the relative humidity at (20-30%) percent. This device can keep the infant's temperature comfortable and maintain a high relative humidity to prevent thermal loss. The proposed system depends on machine-learning algorithms to classify the fault and non-fault parameters and the microprocessor Raspberry Pi 4 to make it as simple to use, cost-effective, and accurate. And providing continuous monitoring and sending an alarm message through Wi-Fi to the corresponding person.

This paper is organized in the following manner: Section 2

explains the related works. Section 3. Describes the theoretical background. Then, the proposed system in describes Section 4. The experimental results are explained in Section 5. Finally, Section 6 presented the conclusion.

2. RELATED WORKS

Because the safety of the incubator's work is a crucial matter, and because of the technological development in the field of monitoring and detecting the faults, there have been many works that have been interested in proposing a monitoring system to perform the parameters of the incubator and ensure their safety, will review some of it.

Ali et al. [7] presented an innovative control system that is utilized to monitor several critical factors that impact an infant's life. This technology monitors and controls several parameters at the same time using sophisticated control and delivering smooth operation, which helps to improve system accuracy. The proposed system had four temperature sensors that were later extended to monitor skin temperature and two humidity sensors. An application page was created to make it easier for the user to monitor the service.

Masi et al. [8] proposed a method that employs the Internet of Things (IoT) to constantly monitor parameters in the incubator, such as temperature, humidity, gas leakage, light intensity, and neonate pulse rate. When these parameters surpass the threshold level, an alert message will be sent via SMS or e-mail to the doctor or the designated medical person. This system can continuously update the parameters in the cloud and avoids the need for a person to monitor the neonates constantly.

Koli et al. [9] proposed an intelligent infant incubator to

attain monitoring and controlling the environment in the infant incubator. They utilized Arduino, a temperature sensor, to monitor and control the baby's body temperature. In a newborn nursing facility with many incubators, the buzzer is used to alert staff about temperature changes. IoT web design is utilized to monitor temperature changes anywhere and at any time.

Sendra et al. [10] submitted a suggestion for an infant incubator system depending on temperature and humidity sensors, as well as a group of weight sensors that enable monitoring of the baby's progress. Each incubator equipped with this technology is connected to a central network depending on Long Range Networks (LoRa) that enable medical data to be stored in a database. Finally, the system has a Near Field Communication (NFC) interface that enables doctors to be identified, tablets to be utilized to track patient progress, and the doctor to enter new data. The system was tested in a laboratory setting to confirm optimal system operation.

Radhika et al. [11] proposed a method to monitor specific parameters inside the incubator, such as the baby's pulse rate, temperature, humidity, and light. If there are any configuration parameters (pulse, temperature, and humidity) changes, the lights and fans inside the incubator will control the parameters. Additionally, the infant will be continually monitored via a webpage, the readings will be continuously shown on the LCD, and the information will be transferred to the proper doctor or nurse via GSM.

Shabeeb et al. [12] proposed a method by which the humidity levels and air temperature in the infant's incubator are remotely monitored as a result of using an Arduino microcontroller with various sensors (DHT11/DHT22) and Internet of Things (IoT) applications. Using a wireless (ESP8266Wi-Fi) connection, the system connects to a network and is linked to a smartphone or computer application.

Fahmi et al. [13] concentrated on how to create baby incubators that can listen to a baby's cries, record them, and then interpret them using artificial intelligence. By using energy signal and spectrum analysis, more than 40 (forty) voice datasets were effectively categorized into five different categories of the baby's condition: burping, sleeping, hungry, uncomfortable, and pain.

El-Aziz and Taloba [14] developed a cheap and safe technology for real-time monitoring of the newborn incubator in order to solve the disadvantages of an unmonitored incubator. A low-cost yet effective device for monitoring vital factors such as the preterm baby's heart rate, temperature, humidity, gas, and light within an incubator was used. The Arduino UNO wirelessly transmits the detected data to the physicians or nurses via the Internet of Things (IoT) so that they may take the required measures to maintain an acceptable environment for the protection of preterm newborns' lives.

Kristya et al. [15] created a newborn incubator temperature and humidity monitoring system rapidly and practically. The Nextion TFT display connected to the WIFI network will show the data collected from the temperature sensor and DHT 22 utilized in the baby incubator temperature and relative humidity monitoring center module. The findings demonstrated that the device has an acceptable average error as the highest allowable error limit, under ECRI415-20010301-01, is 1°C. The outcomes of this research can be used to help nurses monitor preterm newborns and prevent neglect.

El-Aziz et al. [16] proposed a method by which the neonatal

incubator is monitored in real time, and the vital signs of a premature baby in an incubator, such as temperature, pulse rate, humidity, light, and gas. The Arduino UNO wirelessly transmits the detected data to nurses or doctors through the Internet of Things (IoT) so that the needed actions can be taken at a suitable time to preserve a proper environment for the safety of premature babies' lives.

Kolhe et al. [17] discussed the planning of the hardware component of the IoT-based baby equipment monitoring system. The hardware module is made up of a microprocessor, a sensor data collection submodule, and a sensor data transmission submodule. The Arduino Uno Rev3 microcontroller and other sensor components were used in this work. The Arduino board, in conjunction with a Sim 800 GSM Module as an information transmission submodule, delivers SMS to a medical expert the baby body's sensor data monitored by the system. In summary, this study provided new ways for designing a dependable baby hatchery using the UNO microcontroller.

3. THEORETICAL BACKGROUND

This section presents the theoretical concepts that have been relied upon in this work, as follows:

3.1 Fault detection and isolation (FDI)

FDI of physical systems, particularly critical systems that deal with human life and make decisions about whether a situation is safe, have recently become increasingly important, owing to the sensor, computer technology, and communication advances [18]. In light of the severe and competing requirements, the high chance of correct detection and isolation, and the low error probability, it remains a significant challenge [19].

3.2 Classification techniques

Unknown objects are classified into discrete categories using classification, which is a supervised machine learning approach. This study used the binary classification method to distinguish between faulty (abnormal) and normal classes [20]. The classification methods utilized in this study are as follows:

3.2.1 Decision Tree (DT)

It is a supervised learning approach that is non-parametric for classification and regression. To build a model that forecasts the value of a target variable, it is necessary to learn straightforward decision rules from data properties. A tree is an approximation to a piecewise constant [21]. The decision tree's main factors are [22]:

- **Entropy:** A node's entropy is used to gauge its degree of information content as Eq. (1):

$$Entropy = \sum_{i=1}^c p_i \log p_i \quad (1)$$

where, C is the maximum number of nodes, and p_i is the probability of the current node.

- **Information Gain:** The dataset was divided by

information gain according to an attribute with lower entropy. Find the characteristic in the decision tree with the biggest information gain that divides the dataset into more homogenous classes or sub-datasets.

$$Gain(S,A) = Entropy(S) - \sum_{i=1}^n \frac{|S_i|}{|S|} * Entropy(S_i) \quad (2)$$

where,

$\{S_1, \dots, S_i, \dots, S_n\}$ =partition of S according to the value of attribute A

n=number of attribute A

$|S_i|$ =number of cases in the partition S_i

$|S|$ =total number of cases in S

3.2.2 Support Vector Machine (SVM)

This method of supervised machine learning can address classification and regression issues. However, it is mostly used to address categorization issues. Each feature in the SVM method is represented by the value of a particular coordinate, and each feature in n-dimensional space is represented as a point (where n is the number of available features) [23]. Then, categorization is achieved by identifying the hyper-plane that effectively separates the two groups [24].

3.2.3 Artificial Neural Network (ANN)

ANNs are a kind of machine learning (ML) technology that is effective in various applications, including machine translation and diagnosis in medical and other domains. ANNs were motivated by neuroscience and were used for feature extraction and classification tasks. A neuron is hence regarded as being an ANN's primary building block [25]. The output layer is the last layer in an input layer when many layers combine to generate hidden layers. Each layer is made up of one or more neurons. The neurons supply the characteristics, which evaluate certain output values using the related weights and biases. The output is obtained by combining these outputs and multiplying them by the activation function [26].

4. THE PROPOSED SYSTEM

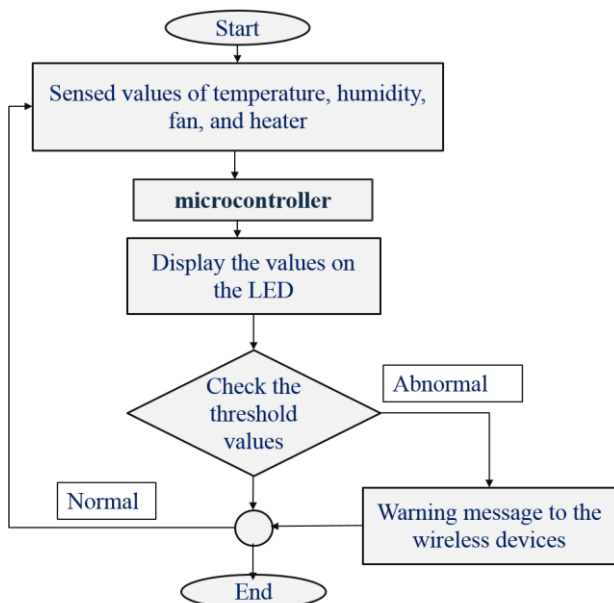


Figure 1. Flow chart of the proposed system

The incubator provides a constant temperature as well as a certain level of humidity. This is done by setting a certain level for these parameters. Any temperature change will lead to turning on or off the fan to adjust the optimum temperature, as well as changing the humidity ratio and turning on the heater. Any fault in any of these parameters will lead to an imbalance in the system and the loss of a safe environment. Therefore, the proposed method is to provide continuous, real-time monitoring of the sensors of these parameters. The proposed system works on monitoring and detecting the input and output faults of the transducer by using various sensors. A transducer model is created consisting of a microcontroller (NodeMCU ESP8266 and Raspberry Pi4), a power supply that connects to the relay to operate the heater element, a fan to equalize the temperature, and a digital temperature and humidity sensor. The temperature and humidity sensors convert physical energy (heat and humidity) into an electrical signal that can be read, and the fan converts electrical energy into mechanical energy. All this data is collected by an intelligent system based on the Raspberry Pi. The information is continuously collected by the sensors and displayed on the LCD screen (connected to Wi-Fi), in addition to sending it to the concerned person via text messages or email. Therefore, as shown in Figure 1, the system will continuously work when there is no fault in these parameters, but as soon as it finds any abnormal situation, it will send an alarm message through Wi-Fi if it exceeds the threshold values (Temperature=25-35.5Celsius, Humidity =20%-30%, Fan current =1.8-2A, Heater current=0.12-0.2A.

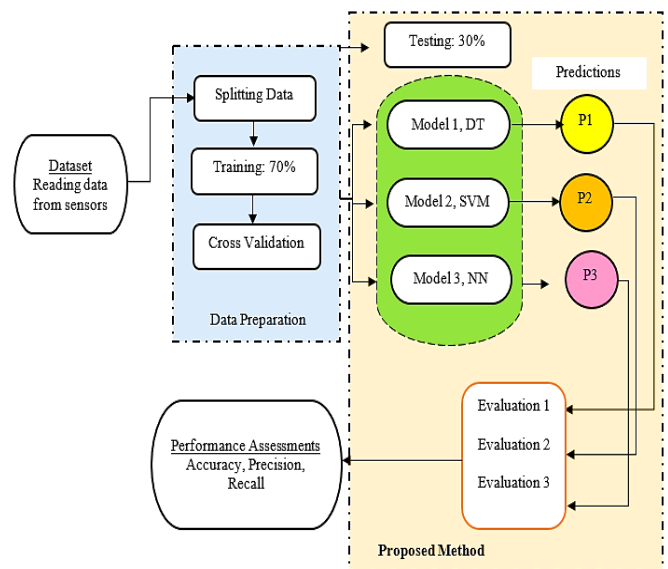


Figure 2. Diagram of the proposed system

4.1 Dataset

Data is recorded in real-time from sensors of the designed virtual infant incubator. It consists of four important parameters (temperature, humidity, fan current, and heater current) from which to identify the situation of the incubator when it is a class 2 fault (abnormal) or class 1 non-fault (normal). Samples of these data are shown in Table 1. In this paper, the data was split into two sets, the training, and testing sets, by the split ratio of 70:30 as shown in Figure 2. In the simulation stage (offline stage), we used four features (temperature, humidity, heater, and fan), but in the practical stage (online), we used only two features (temperature, and humidity), which are heat and humidity.

Table 1. Samples of data from the sensor in the virtual incubator

Serial No	Temp (C)	Hum (RH%)	Heater (A)	Fan (A)	Class
150	34	27	1.86	0.13	1
151	34	25	1.87	0.131	1
152	34	14	0.6	0.054	2
153	35	13	0.5	0.709	2
154	36	13	4.6	0.193	2
155	36	14	1.5	0.16	2
156	31	15	3.5	0.114	2
157	35	27	1.8	0.13	1
158	37	28	3	0.132	2
159	35	27	1.81	0.131	1
160	35	26	1.79	0.14	1
161	34	25	1.8	0.129	1
162	31	15	1.1	0.24	2
163	32	25	1.82	0.14	1
164	32	25	1.79	0.132	1
165	31	26	1.87	0.14	1
166	30	26	1.81	0.133	1
167	30	26	1.82	0.131	1
168	31	28	1.8	0.131	1
169	29	25	1.79	0.132	1
170	29	26	1.83	0.131	1
171	31	25	1.82	0.129	1
172	32	26	1.83	0.13	1
173	30	26	1.79	0.131	1
174	30	25	1.8	0.133	1

4.2 System components

In this section, we present the hardware components of the proposed system as follows:

4.2.1 DHT22 sensor

Its dependability and stability are ensured by a patented digital signal-collection technique and humidity sensor technology. Its sensors are coupled to a single-chip 8-bit CPU. Each sensor in this design has been temperature corrected, calibrated, and kept in a precise calibration chamber. When a sensor detects, it will quote the calibration coefficient from memory that has been saved in OTP memory in the form of a program. The DHT22 is suitable for a wide range of demanding application areas thanks to its small size (as shown in Figure 3), low power consumption, and high bandwidth distance (20m).



Figure 3. DHT22 sensor

4.2.2 NodeMCU ESP8266

Figure 4 depicts NodeMCU, an open-source Internet of Things platform. It consists of firmware running on Espressif Systems' ESP8266 Wi-Fi SoC and hardware depending on the ESP-12 module. The moniker "NodeMCU" refers to the firmware rather than the DevKit by default. The Lua programming language is used in the firmware. It is constructed utilizing the Espressif Non-OS SDK for ESP8266 and is depending on the eLua project. It makes extensive use of open source programs like lua-cjson and spiffs.



Figure 4. NodeMCU ESP8266

4.2.3 Raspberry Pi 4

The Raspberry Pi 4 Model B, seen in Figure 5, is the most recent update to the popular Raspberry Pi computer family. It exceeds the previous-generation Raspberry Pi 3 Model B+ in terms of CPU speed, multimedia performance, memory, and connectivity while keeping backward compatibility and power consumption. It was employed in the planned system to transform data into a format that could be programmed and subsequently classified.



Figure 5. Raspberry Pi 4

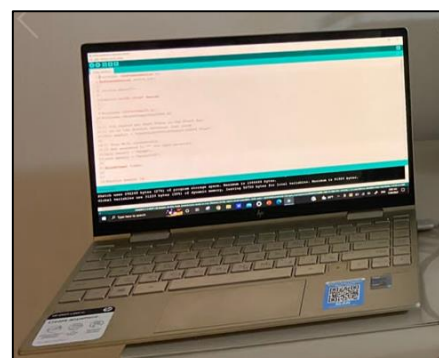


Figure 6. Personal Computer

4.2.4 Personal computer

A personal computer as shown in Figure 6 with the following properties has been used.

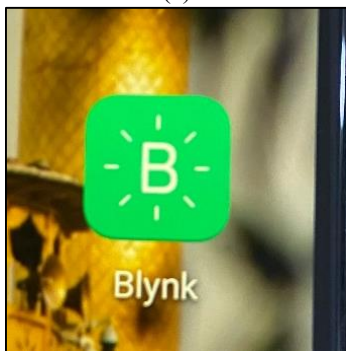
- Intel (R) Core i7- 5200U CPU @2.80 GHz
- 8 GB Random Access Memory (RAM),
- 64-bit operating system.

4.2.5 Smartphone

A Huawei Andr8 GB Random Access Memory (RAM), the phone which is shown in Figure 7 (a) was used to run the application in Figure 7 (b) and display sensor values so that people could monitor the incubator from a distance.



(a)



(b)

Figure 7. (a) Huawei phone, (b) Blynk App

4.3 Virtual incubator design



Figure 8. Virtual incubator

Figure 8 shows the incubator that was designed as a virtual environment for the work of the proposed system, which was made of glass and linked to a group of sensors and smart devices.

4.4 Evaluation method

The proposed system will be evaluated by using these metrics:

4.4.1 Accuracy

Measure the ratio of the correct prediction to the total number of forecasts in model performance. The higher accuracy means, the lower the error rate [27].

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

4.4.2 Precision

The result of dividing all true positive predictions by the sum of the true positive and false-positive predictions, which means how many predictions were correct. Therefore, it is called positive predictive [28].

$$\text{Precision} = \frac{TP}{TP+FP} \quad (4)$$

4.4.3 Recall

How many true positives were predicted or how many right hits were found, referred to as recall, also called sensitivity [29].

$$\text{Recall} = \frac{TP}{TP+FN} \quad (5)$$

4.4.4 F1-Score

The F1 score is a statistical metric for evaluating the performance of the model via harmonic precision and recall [29].

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (6)$$

where, TP=True Positive (predicted value is positive and actual one is positive).

TN=True Negative (predicted value is negative and actual one is negative).

FP=False Positive (predicted value is positive and actual one is negative).

FN=False Negative (predicted value is negative and actual one is positive).

5. RESULTS AND DISCUSSION

The virtual incubator that was designed was connected to a personal computer (PC) and the computer was linked to a smartphone. The proposed system works as follows:

- 1) Sensors are connected to the NodeMCU ESP8266 unit for temperature, humidity, heater, and fan readings.
- 2) The NodeMCU ESP8266 is connected to a Raspberry Pi 4, which in turn converts the signals into data that can be read by a computer.
- 3) After the data arrives at the computer, it is categorized using three different machine learning techniques that have been programmed in the Python programming language.
- 4) Then the data is sent to the smartphone and read by an application called Blynk which is created on a smartphone that allows you to easily interact with microcontrollers or even full computers such as the Raspberry Pi.

Figure 9 shows the proposed system environment in all its details.

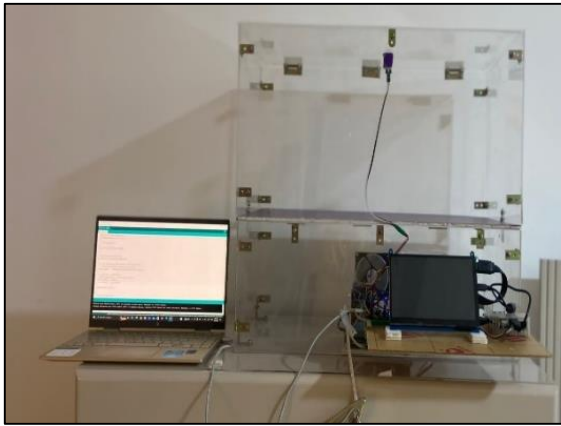
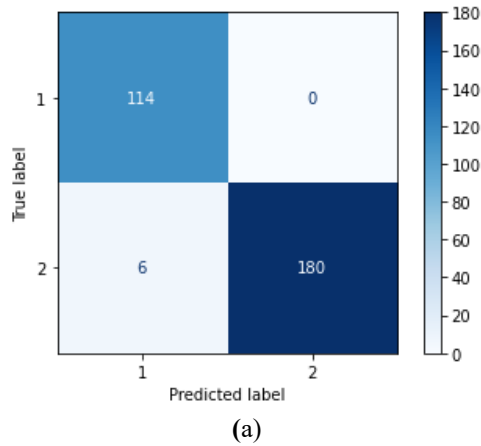
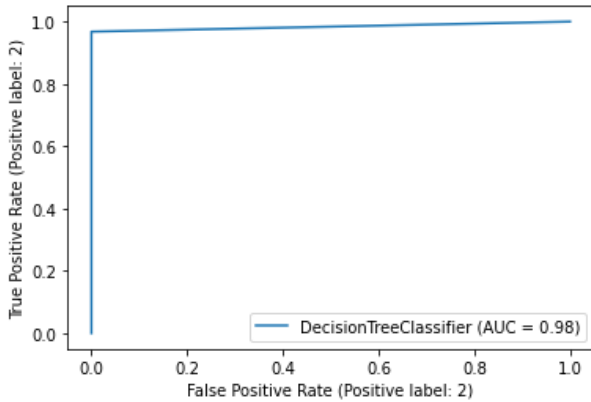


Figure 9. The proposed system environment

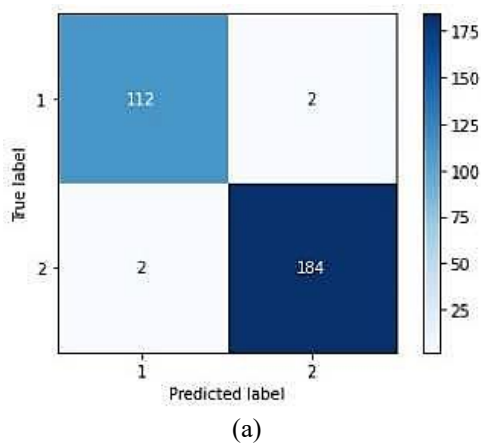


(a)

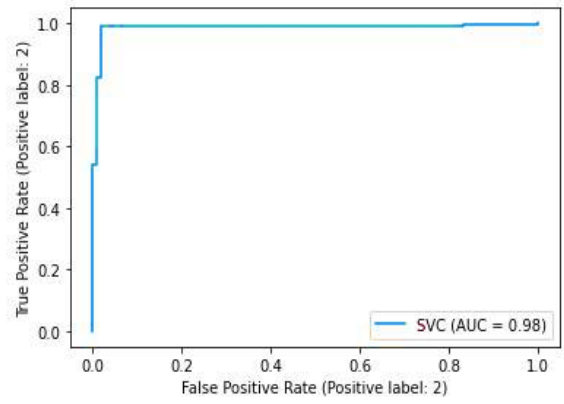


(b)

Figure 10. (a) DT classifier Confusion matrix, (b) DT Classifier Region of Convergence

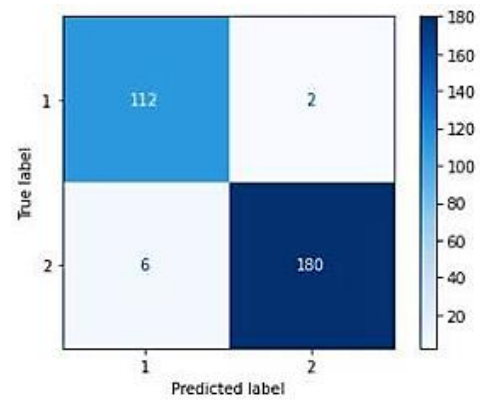


(a)

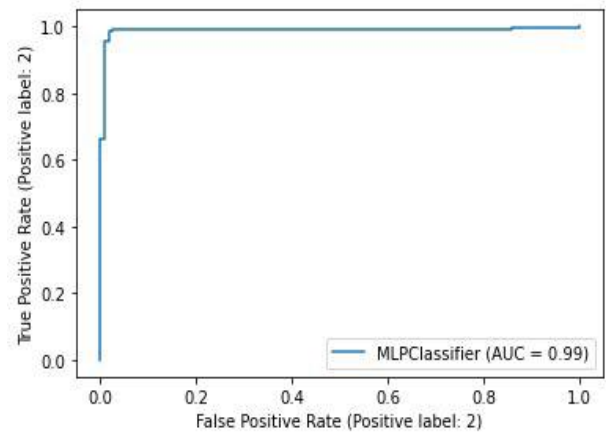


(b)

Figure 11. (a) SVM classifier Confusion matrix, (b) SVM Classifier Region of Convergence



(a)



(b)

Figure 12. (a) ANN classifier Confusion matrix, (b) ANN Classifier Region of Convergence

The scikit-learn machine learning library from the Python language was used to train classifier algorithms on the Raspberry Pi 4. In addition, the results of the evaluation method come as shown in Figures 10, 11, and 12, representing each classification algorithm (a) The Confusion Matrix. (b) The region of the Convergence Curve (ROC Curve) and the evaluation values are presented in table 2. After training the classification algorithms on 70% of the data. Then testing result gives an excellent performance; as clarified in Figure 12, the confusion matrix and Area under Curve (AUC) of the DT algorithm come with an accuracy rate of 98%. Then, in Figure 11, the confusion matrix and (AUC) of the SVM algorithm

come with 98% accuracy. While in Figure 12, the confusion matrix and (AUC) of the NN algorithm come with 97.3% accuracy.

6. CONCLUSIONS

Monitor the Sensors inside the incubator to ensure they do not exceed the safe limit and detect the fault in real-time. This technology will provide continuous monitoring and thus reduce the effort and time of the medical staff involved in monitoring the incubator. As well as taking the necessary measures to protect and not lose the lives of children due to excessive exposure to moisture or high temperature. The intelligent monitoring design takes all the cases of fault, not only temperature and humidity like all works but also pays attention to fan and heater current to make the work comprehensive and valuable. For several reasons, the imbalance in the input and output voltage or the overheating of the transducer can occur for many reasons that can lead to significant malfunctions that may lead to disasters, loss of equipment, and lives. The results were promising and highly accurate with a different machine learning algorithm. The next step will connect the system to the secure server to provide integrated performance through the Internet of Things (IoT). The contributions of the paper: Experiment with more than one machine learning algorithm (DT, SVM, and NN) to classify incubator sensor readings (features) to arrive at a perfect and accurate system. A Raspberry Pi 4-based system with minimal computational resources was utilized to identify sensor faults, making it a lightweight solution, which modifies AI applications so they are affordable, simple to install, take up less space, and have adequate computational power.

Table 2. The evaluation values for each classifier are according to the Confusion Matrix

Method	Accuracy%	Precision%	Recall%	F1-score%
DT	98	95	100	97.4
SVM	98	98.9	98.9	100
ANN	97.3	98.9	96.7	97.78

REFERENCES

[1] Kumar, P.A., Akshay, N., Kumar, T.A., Sama, A. (2013). Real time monitoring and control of neonatal incubator using LabVIEW. *International Journal of Application or Innovation in Engineering & Management*, 2(4): 375-380. <https://doi.org/10.48550/arXiv.2112.02071>

[2] Araújo, J.M., Menezes, J.M., Albuquerque, A.A., Almeida, O.D., Araújo, F.M. (2013). Assessment and certification of neonatal incubator sensors through an inferential neural network. *Sensors (Basel, Switzerland)*, 13(11): 15613-15632. <https://doi.org/10.3390/s131115613>

[3] Marafa, I. (2020). Fault tolerance and real-time monitoring infant incubator model. *Lautech. J. Comput. Informatics*, 1(1): 40-48.

[4] Abed, A.M., Gitaffa, S.A., Issa, A.H. (2021). Robust geophone string sensors fault detection and isolation using pattern recognition techniques based on Raspberry Pi4. *Materials Today Proceedings*, <https://doi.org/10.1016/j.matpr.2021.04.360>

[5] Abed, A.M., Gitaffa, S.A., Issa, A.H. (2021). Quadratic support vector machine and k-nearest neighbor based robust sensor fault detection and isolation. *Engineering and Technology Journal*, 39(5A): 859-869. <https://doi.org/10.30684/etj.v39i5a.2002>

[6] Hadi, M.H., Issa, A.H., Sabri, A.A. (2021). Design and FPGA implementation of intelligent fault detection in smart wireless sensor Networks. *Engineering and Technology Journal*, 39: 653-662. <https://doi.org/10.30684/etj.v39i4A.1951>

[7] Ali, M., Abdelwahab, M.M., Awadekreim, S.D., Abdalla, S.A. (2018). Development of a monitoring and control system of infant incubator. 2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), Khartoum, Sudan, pp. 1-4. <https://doi.org/10.1109/ICCCEEE.2018.8515785>

[8] Masi, S., Banu, R., Deepika, L., Indu, R. (2018). Neonatal healthcare monitoring in incubator using IoT. *International Journal of Electrical Electronics and Data Communication*, 6(6): 2321-2950. <https://doi.org/10.26562/IJRAE.2019.SI.MRAE10095>

[9] Koli, M., Ladge, P., Prasad, B., Boria, R., Balur, N. J. (2018). Intelligent baby incubator. Proceedings of the 2nd International Conference on Electronics, Communication, and Aerospace Technology (ICECA), Coimbatore, India, pp. 29-31, <https://doi.org/10.1109/ICECA.2018.8474763>

[10] Sendra, S., Romero-Díaz, P., Navarro-Ortiz, J., Lloret, J. (2018). Smart infant incubator based on lora networks. 15th International Conference on Computer Systems and Applications (AICCSA), Aqaba, Jordan, PP. 1-6. <https://doi.org/10.1109/AICCSA.2018.8612863>

[11] Radhika, B., Sheshagiri Rao, V. (2019). Incubator baby parameter sensing and monitoring. *International Journal of Innovative Technology and Exploring Engineering*, 8(7): 2945-2947.

[12] Shabeeb, A.G., Al-Askery, A.J., Nahi, Z.M. (2020). Remote monitoring of a premature infants incubator. *Indonesian Journal of Electrical Engineering and Computer Science*, 17(3): 1232-1238. <https://doi.org/10.11591/ijeecs.v17.i3>

[13] Fahmi, F., Shalannanda, W., Zakia, I., Sutanto, E. (2020). Design of an IoT-based smart incubator that listens to the baby. *IOP Conference Series: Materials Science and Engineering*, 1003(1). <https://doi.org/10.1088/1757-899X/1003/1/012153>

[14] El-Aziz, R.M., Taloba, A.I. (2021). Real time monitoring and control of neonatal incubator using IOT. *International Journal of Grid and Distributed Computing*, 14(1): 2117-2127. <https://doi.org/10.48550/arXiv.2112.02071>

[15] Kristya, F., Luthfiyah, S., Hari Wisana, I.D., Thaseen, M. (2021). Baby incubator monitoring center for temperature and humidity using wifi network. *Journal of Electronics, Electromedical, and Medical Informatics (JEEEMI)*, 3(1): 8-13. <https://doi.org/10.4028/p-392j82>

[16] El-Aziz, R.M., Taloba, A.I. (2021). Real time monitoring and control of neonatal incubator using IOT. *International Journal of Grid and Distributed Computing*, 14(1): 2117-2127. <https://doi.org/10.48550/arXiv.2112.02071>

[17] Kolhe, S., Mishra, M.S., Shah, M.S. (2021). Improved design of IoT based infant incubator monitoring and

- control system. *International Research Journal of Engineering and Technology (IRJET)*, 8(9): 2837-2842.
- [18] Said, B.M., Eddine, K.D., Salim, C. (2020). Artificial neuron network based faults detection and localization in the high voltage transmission lines with Mho distance relay. *Journal Européen des Systèmes Automatisés*, 53(1): 137-147. <https://doi.org/10.18280/jesa.530117>
- [19] Hadi, M.H., Issa, A.H., Sabri, A.A. (2021). Improvement of salp swarm algorithm (SSA2) for intelligent fault detection in smart wireless sensor networks. *Materials Today: Proceedings*, <https://doi.org/10.1016/j.matpr.2021.07.248>
- [20] Shruthi, U., Nagaveni, V., Raghavendra, B.K. (2019). A review on machine learning classification techniques for plant disease detection. 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, pp. 281-284. <https://doi.org/10.1109/ICACCS.2019.8728415>
- [21] Jiao, S.R., Song, J., Liu, B. (2020). A review of decision tree classification algorithms for continuous variables. *Journal of Physics: Conference Series*, 1651(1): 012083. <https://doi.org/10.1088/1742-6596/1651/1/012083>
- [22] Saeed, U., Ullah Jan, S., Lee, Y., Koo, I. (2020). Machine learning-based real-time sensor drift fault detection using Raspberry Pi. 2020 International Conference on Electronics, Information, and Communication (ICEIC), Barcelona, Spain, pp. 1-7, <https://doi.org/10.1109/ICEIC49074.2020.9102342>
- [23] Jeong, K., Choi, S.B., Choi, H. (2020). Sensor fault detection and isolation using a support vector machine for vehicle suspension systems. *IEEE Transactions on Vehicular Technology*, 69(4): 3852-3863. <https://doi.org/10.1109/TVT.2020.2977353>
- [24] Mohamed, M.B., Meddeb-Makhlouf, A., Fakhfakh, A. (2019). Intrusion cancellation for anomaly detection in healthcare applications. 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, pp. 313-318. <https://doi.org/10.1109/IWCMC.2019.87665922>
- [25] Abdullah, F.S., Hamoodi, A.N., Mohammed, R.A. (2021). Performance Improvement in Steam Turbine in Thermal Power Plants Using Artificial Neural Network. *Journal Européen des Systèmes Automatisés*, 54(6): 891-895. <https://doi.org/10.18280/jesa.540611>
- [26] Japkowicz, N. (2006). Why question machine learning evaluation methods? (An illustrative review of the shortcomings of current methods). in *AAAI Workshop on Evaluation Methods for Machine Learning*, 6-11, https://doi.org/10.1007/978-3-642-39712-7_15
- [27] V Starovoitov, V.V., Golub, Y.I. (2020). Comparative study of quality estimation of binary classification. in *Informatics*, 17(1): 87-101. <https://doi.org/10.37661/1816-0301-2020-17-1-87-101>
- [28] Tharwat, A. (2020). Classification assessment methods. *Applied Computing and Informatics*, 15(2): 81-93. <https://doi.org/10.1016/j.aci.2018.08.003>
- [29] Luque, A., Carrasco, A., Martín, A., Heras, A.D. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognit*, 91: 216-231. <https://doi.org/10.1016/j.patcog.2019.02.023>

NOMENCLATURE

°C	Celsius
A	Ampere
RH%	Relative humidity
GB	Gigabyte
GHz	Gigahertz
MB	Megabyte