Egyptian Journal of Artificial Intelligence
ISSN (Online): 2786-0213, ISSN (Print): 2786-0205
Volume 3, Issue 1, 2024, PP. 1-12

A Low-Cost Approach for Drowning Detection and Alertness

Mostafa A. Ibrahim[1*], Manal E. Ali[2], Mostafa N. Ahmed[1], Kareem A. Mohamed[1], Ali I. Ali[1], Ahmed M. Elsayed[1], Hossam H. AbdElHalim[1], Mona Alnaggar[3], Abdelmawla Yousef[1], Ali I. Siam[1]

[1]Dept. of Embedded Network Systems Technology, Faculty of Artificial Intelligence, Kafrelsheikh University, Egypt

[2]Department of Physics and Engineering Mathematics, Faculty of Engineering, Kafrelsheikh University, Kafrelsheikh, Egypt.

[3]Dept. of Robotics and Intelligent Machines, Faculty of Artificial Intelligence, Kafrelsheikh University, Egypt

[*]Corresponding Author: Mostafa A. Ibrahim, E-mail: ma4817334@gmail.com

Abstract:

In this paper, a unique approach has been developed to detect drowning by using a chest area belt equipped with a pressure sensor to monitor the breathing rate of the person wearing the belt. Additionally, MPU6050 module has been used to measure the cumulative displacement to track the swimmer's movement. These sensors have been integrated with a Remotexy application and a GSM module to create an SMS message that alerts lifeguards when someone is in danger of drowning. The chest area belt is equipped with carbon resistors that are sensitive to pressure changes, allowing us to monitor the breathing rate of the person wearing the belt. By monitoring the breathing rate, we can detect when a swimmer is in distress and alert lifeguards to the potential drowning. To track the swimmer's movement, an MPU6050 has been used, which is a motion sensor that measures the cumulative displacement of the person wearing the belt. This data is then integrated with a Remotexy application and a GSM module to create an SMS message that is sent to lifeguards when someone is in danger of drowning.

Keywords: drowning detection; esp32; GSM module, chest tracker; breathing rate; pressure sensor.

1. Introduction

Artificial intelligence techniques have opened up previously unthinkable possibilities and changed innovation in a number of fields. Such as applying these techniques in Thyroid classifications [1], heart conditions detection [2], facial expression recognition [3], automatic stress detection [4], level of consciousness detection [5] and Fatigue state detection [6].

Drowning is the 3rd leading cause of unintentional injury death worldwide, accounting for 7 percent of all injury-related deaths. There are an estimated 320,000 drowning deaths each year worldwide. Globally, the highest drowning rates are among children 1–4 years, followed by children 5–9 years.

Drowning is a silent phenomenon, with victims rarely exhibiting convulsive movements [7-10]. Instead, they often use all their energy trying to keep their heads above the surface of the water, making it impossible for them to call or signal for help. If water manages to reach their larynx or it can trigger a panic response in the victim that results in spasms, preventing them from being able to shout for assistance. Therefore, the only way to spot a potential drowning victim is by seeing them struggling in the water.

The wide area of the sea may create many blind spots. The lifeguard's view of the swimming area is obscured [7]. It is the lifeguard's responsibility to seek out every blind spot in the swimming area and to take the necessary action to reduce the amount of area that the lifeguard cannot see from his/her lifeguard station. It might make it difficult for lifeguards to monitor all the swimmers' safety in the sea [11]. The chance of drowning among children is the outcome of this issue. As a result, this deadly problem has attracted many specialists' attention.

2. Related studies

Drowning detection systems have been studied in detail in recent years. These systems can be divided into two primary categories: sensor-based systems and image processing systems. Sensor-based systems rely on a variety of sensors to detect drowning, such as pressure, heartbeat, motion, and depth. Image processing systems, on the other hand, use algorithms to detect drowning by analyzing images or videos. In both cases, the goal is to detect drowning in a timely manner and help prevent any further fatalities.

2.1 Drowning detection using computer-vision

Video based drowning monitoring systems have been proposed as an alternative to traditional drowning prevention systems, but the technology is very prone to visual and sensor disturbances [12]. This is especially true when the pool or waterbody is crowded, as the system is unable to differentiate between different individuals. To address this issue, several other electronic drowning prevention systems have been proposed. These systems make use of physiological features to detect a person in distress, such as a swimming cap that can detect changes in heart rate or breathing. However, many users are very selective about the drowning prevention systems they choose, as they need to ensure that they are both safe and effective.

Kam et al. [13] proposed a paper about drowning detection in swimming pools using computer vision. This paper proposes a drowning detection system that uses computer vision techniques to analyze video footage of swimming pools. The system detects drowning incidents by analyzing the movement patterns of people in the water and identifying instances where people are in distress.

Salehi et al. [12] proposed automated video-based surveillance systems that are used for real-time human behavior analysis and help to provide an effective way of monitoring any unusual activity in any surroundings. They noticed that due to the rapid lighting evolvement in the environment, there are high dynamic backgrounds and thus vague visibility of targets is a major difficulty and this continues to be faced by most advanced systems [8-10].

Heart rate (HR), blood oxygen saturation level (SpO_2), body temperature, photoplethysmography (PPG) signal, electrocardiography (ECG) signal, ambient temperature, and room humidity are just a few of the medical parameters that the designed system keeps track of. The measured data is shown on the built-in display or sent wirelessly to a mobile app for local monitoring or to cloud storage for distant monitoring. The proposed system proposed by Siam et al. [17] may be utilized to watch over the folks we need to take care of while allowing them to continue living their regular lives.

Alnaggar et al. [18] developed an architecture that involves watching a video stream of the patients while they are being monitored by a camera to extract the rPPG waveform from their faces. Eulerian Video Magnification (EVM) is used to first enlarge the movement and color in the video. The motion and color are then examined in two steps, one for HR estimate and the other for RR estimation.

2.2 Drowning detection using multi-sensors

Two systems which are wearable and used for anti-drowning have already been put in place:

• SenTAG [11] is an anti-drowning system that offers a safety solution for swimming pools which helps to check individual swimmers via a wristband given to the individual, it is used to monitor the individual's depth in water, motion, and time. Sensors are mounted on the wall of the pool to check if any swimmer wearing the wristband is approaching the preset limits for depth and time. It then evaluates whether the swimmer is spending a lot of time in water under a preset depth of field. If so, the wristband then sends a wireless alert signal through a radio or by an ultrasonic transmission frequency to a control unit alarm. At this point, the alarm, which is situated on the wristband begins to sound, and then the LED lights come on, which is used to indicate that the swimmer is to return to a safe point. If the swimmer fails to respond as demanded, the unit then sends a full emergency alert personnel only [19-20]. This really cuts down the probability of 6 false alarms. The alarm system is a wristband that triggers an alarm when a swimmer has been motionless underwater for twenty seconds.

• WAHOOO [5] A headband system was developed to send an alarm to a receiver if a swimmer stayed underwater too long. However, the system did not consider the complexities of in this case, the alarm could trigger too late, and the swimmer may not be able to be rescued in time; or it could trigger too early and create a false alarm [21]. To address this issue, the system must be further developed to better understand the nuanced scenarios of a near-drowning experience [22-24]. This may require the implementation of sensors that can judge the swimmer's movements and breathing patterns, as well as improved algorithms that are able to accurately assess the swimmer's well-being. Only then will the system be able to reliably alert the receiver in the event of a near-drowning [25-28].

3. Proposed System

. The proposed system uses a chest wearable device that measures the movement of the chest during inhalation and exhalation. As a result, we will be able to calculate the number of breaths produced by the individual. The MPU6050 sensor will be integrated into the wearable device to continuously monitor the movement and orientation of the wearer. Using this data, we can develop a model to predict drowning probabilities and then carry out rescue procedures based on that prediction. The system will be able to detect abnormal movements associated with drowning, alerting first responders or lifeguards, and allowing them to respond accordingly. Additionally, this data can be used to evaluate swimming performance and provide feedback to the swimmer [29].

Overall, this proposed system demonstrates the potential of wearable technology and motion sensors to improve public safety and save lives. By leveraging the capabilities of the MPU6050 sensor and a wearable device, we will get into every detail about their components in the upcoming sections. We can develop innovative solutions to prevent drowning and keep people safe in and around water [30-34].

3.1 Materials

(i) **ESP32**: It is a series of low-cost, low-power systems on a chip microcontroller with integrated Wi-Fi and dual-mode Bluetooth.

(ii) **Custom pressure sensors**: A pressure sensor is an electronic device that detects, regulates, or monitors pressure, and converts perceived physical data into an electronic signal [35]. The chest area belt is equipped with carbon resistors that are sensitive to pressure changes. In figure 1 a carbon resistor is shown.

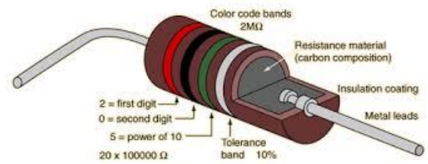

Figure 1: Carbon Resistor

There are many pressure sensors on the market that work well in a variety of research and experimental settings, it can be challenging to find one that meets all our specific requirements like small size, ignoring error and more sensitive to detect chest movement [36].

So, we created a pressure sensor that met our needs. The same idea of a carbon resistor but made it a variable resistor has been used in the proposed method [37].

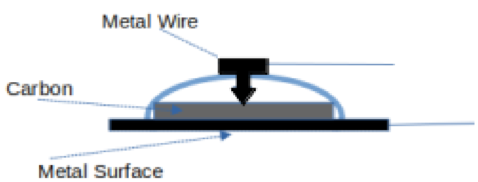

Figure 2: Custom Pressure Sensor

The general shape of our product, which is based on a simple idea such as common carbon resistors has been displayed in Figure 1 and 2. Carbon, being a resistive material, creates a voltage drop across it when an electric current is passed through it. Its ability to withstand high temperatures and provide stable resistance values makes it ideal for use. It is also relatively inexpensive and easy to use. Carbon resistors are also known for their durability and long-term performance [39]. The sensor in question is composed of a metal surface, which is covered by a layer of carbon powder. Attached to this surface is a metal pin, which when pressed, decreases the distance between the two metals. This decrease in distance alters the resistance of the sensor, allowing us to calculate the current passing through. By using this data, we can measure chest pressure and obtain valuable information regarding the patient's health. The proposed circuit diagram is shown in figure 3.

(iii) **MPU6050**: The MPU6050 is an incredibly popular accelerometer gyroscope chip, known for its six axes of sensing and 16-bit measurement resolution. This combination of precision and affordability has made it the go-to choose for the DIY community. Many commercially available products from a variety of industries are now equipped with the MPU6050 as well. In addition, the combination of gyroscope and accelerometers is referred to as an Inertial Measurement Unit (IMU). With its high accuracy and low cost, the MPU6050 is a great solution for any application requiring an IMU.

(iv) **SIM800L GSM/GPRS Module**: SIM800L is a miniature cellular module which allows for GPRS transmission, sending and receiving SMS and making and receiving voice calls. Low cost and small footprint and quad band frequency support make this module perfect solution for any project that requires long range connectivity.

(v) **TP4056 lithium cell charger module:** The TP4056 is a complete constant-current/constant-voltage linear charger for lithium-ion batteries. Its SOP package and low external component count make the TP4056 ideally suited portable applications.

(vi) **Power source**: a rechargeable 3.7V lithium batteries have been used.

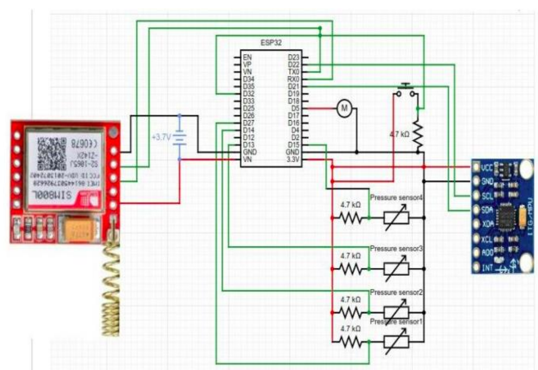

Figure 3: Circuit Diagram

3.2 System implementation

(i) Calculating breathing rate

We measure the breathing rate based on the data of the pressure sensors. Respiration rate is crucial in determining the condition of the victim. This algorithm calculates the breathing rate every 30 seconds using the data generated from the pressure sensors. these data will be previewed in figure 4.

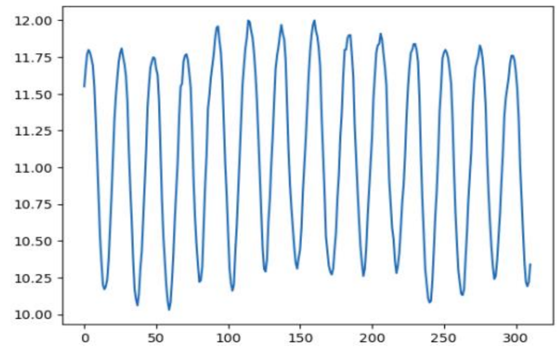

Figure 4: The **e**xtracted signal depending on the used sensor

• These data helped in calculation of breathing rate and do pattern recognition.

• First, the data was classified into 3 categories (exhalation, inhalation, and hold).

• We create a method that takes a difference between every two consecutive readings and checks if the difference is greater than 0.05, then this is inhalation, if smaller than -0.05, then this is exhalation, and between two these values there is hold.

• Think of this method as we are doing a slope for every Two consecutive points and check the previous conditions to classify data.

• 1 for inhalation, 0 for exhalation and 2 for hold

• The result is shown in figure 5

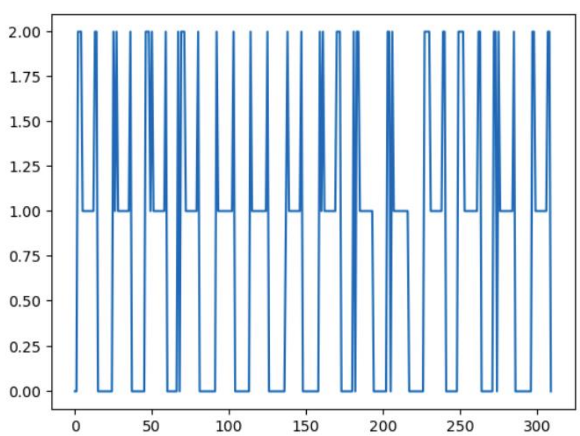

Figure 5: Data Classification

• There are some vibrations in the data, and this may negatively affect the calculation of the respiratory rate correctly, so we clean the data from these vibrations until it is ready to calculate the breathing rate.

• After analyzing the vibrations, I noticed that they are fast and often between 1 to 4 numbers. Therefore, if I notice a vibration in the data, I set the vibration values to the values that precede it. I can delete these vibrations, but I want to keep the data. I want to see the full breathing story. Because it will be possible in the future to calculate the length of the breath, it will be useful and accurate. Hence, the result after cleaning is shown in figure 6.

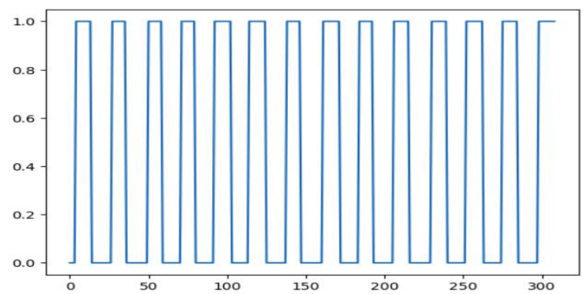

Figure 6: Data cleaning

• After that, we compile each period with what it expresses, if it was a group of singles, we take only one to be expressive of this period, and so on to calculate the respiratory rate in this step we have the shape of breathing.

• Then, the rate of breathing has been calculated by summing the array of breathing except 2(hold). Then the breathing array was used to check drowning patterns, I proposed two drowning patterns first: [2,0,1,2] which means hold> exhalation > inhalation > hold, the second is [2,1,2] which means hold > inhalation > hold.

• The number of times any of these breathing patterns occurred has been counted and previewed in Figure 7 as an example.

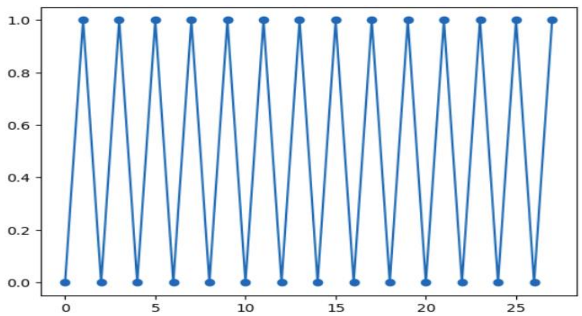

Figure 7: Aggregation

- Why this method?

There are several methods that we can use to calculate the respiratory rate, such as the average method as a threshold, and then if the number is greater than the threshold, we can measure it as inhalation and other exhalations. The data depends on the pressure of the sensors on the chest [18], and that can change every time a person wears these sensors, so relying on the threshold was very difficult and would lead to very bad results, so I preferred this method and I believe that God willing, it will work in all cases.

(ii) Implementing MPU6050 for Cumulative Displacement Measurement

In this section, the steps of using the MPU6050 to measure cumulative displacement, specifically detection of cumulative displacement the MPU6050 involves several steps.

The first step is data collection. This involves collecting data from the accelerometer and gyroscope sensors on the MPU6050. This data can be collected using a microcontroller or a computer.

Once the data has been collected, the next step is to filter and process the data. This involves using algorithms to remove noise and extract information from the data. The filtered data can then be used to calculate time.

In the context of drowning detection, the MPU6050 can be used to measure breathing rate.

- Data Collection and Processing

The first step involves acquiring a substantial amount of data from the MPU6050. We collected 1000 values using Pyserial and logged them in an Excel sheet. This data was then processed by applying mathematical equations and filters to the acceleration data, providing us with displacement values. It was observed that these values were cumulative, meaning that as we traversed further along the axis, the displacement increased. To measure cumulative displacement, we need to follow these steps.

A) Collect stable acceleration values

To begin with, we need to gather roughly 1000 samples of acceleration values from the MPU6050 device when it is in a stable position. To achieve this, we will focus on one axis and calculate the distance using its acceleration. We will select the most accurate axis to work with. To collect the data, we will utilize the Pyserial library in Python. This library allows us to connect our ESP32 port with Python and store the data in a format that is easily accessible for analysis. Once we have chosen the axis to work with, we will use the MPU6050 to measure the acceleration values. the Pyserial library has been used to transfer these values to our Python program. With the data collected, we can begin analyzing the results. By examining the acceleration values, we can determine the stability of the device and make any necessary adjustments. We can also use the collected data to calculate the distance traveled by the device. In summary, to gather approximately 1000 samples of MPU6050 acceleration values, we will focus on one axis to calculate distance using its acceleration. We will select the most accurate axis to work with and use the Pyserial library in Python to connect our ESP32 port with Python to store the data. With the data collected, we can analyze the results to determine the stability of the device and calculate the distance traveled.

B) Python Modules for Data Collection

To perform this step, we uploaded the MPU6050 code that calculates acceleration on ESP32, and then coded the following python modules:

• mpu_serial2.py: Connects our ESP32 with the same bandwidth as defined in the uploaded

ESP32 code and creates a text file containing our MPU6050 read accelerations on the three axes.

• Read_txt.py: Defines a function to loop on data saved in our text file, allowing us to apply regular expression on each line.

• re_and_csv_file.py: Contains a defined regular expression function for pattern extraction for the created text file, converting it to an Excel table that contains the three-axis acceleration as three columns.

• Pattern_matching_txt_to_csv.py: Applies the last two steps, ultimately providing us with a clean Excel sheet.

C) Calculate standard deviation:

After data has been collected on an Excel sheet, the next step is to calculate the standard deviation of all the readings. Fortunately, Excel provides mathematical formulas that make this calculation quick and easy. For our purposes, we will use the sample standard deviation formula to calculate the standard deviation of all acceleration readings. To perform this calculation, follow these steps: 1. Select the cell where you want to display the standard deviation. 2. Type "=STDEV.S(" into the cell. 3. Highlight the range of data you want to calculate the standard deviation for. 4. Type ")" and hit enter. Excel will then display the sample standard deviation for all the selected readings. This calculation can be performed quickly and accurately using Excel's built-in formulas.

After collecting 1000 values and calculating the standard deviation using the appropriate formula, we utilized the mathematical formulas provided by Excel to carry out the remaining steps. Upon completion, we found that the cumulative distance results were remarkably accurate. Encouraged by this outcome, we decided to put our findings into practice in real-time scenarios as the following.

D) Attaching the MPU6050 to a moving object:

To attach the MPU6050 to a moving object, first, fix the sensor to the object or person whose displacement you want to measure. In this case, we will send the MPU6050 to the chest of the person. To ensure accurate measurement, we will the chest so that the y-axis in the MPU mimics the altitude. This is because, through processing simulation with the MPU, it was found that the z-axis of the MPU is not accurate when in motion and can lead to a significant error in measurement. Therefore, to achieve accurate measurements, we will be rotating the chest to ensure that the y-axis in the MPU6050 mimics the altitude. This will help to minimize errors and ensure that the measurements are as accurate as possible. Once the MPU6050 is attached to the moving object and correctly positioned, it can be used to measure the displacement of the object accurately. With this information, you can analyze the motion of the object and use it for various applications, including tracking the movement of athletes or monitoring the movement of machinery.

E) Record acceleration and time:

In order to add acceleration values to array sample [2], we utilized the fillArray() function. This function was designed to populate the array with the necessary values. By calling the function and passing the appropriate parameters, we were able to successfully add the acceleration values to the desired location within the array. It is important to note that this process was crucial to properly analyze and interpret the data that was collected. The fillArray() function allowed us to efficiently and accurately store the acceleration values that were necessary for our analysis.

F) Calculate change in acceleration:

In this task, we will be working with the previous acceleration and the current acceleration. The main objective is to calculate the difference between these two values. However, we will only consider this difference if it is greater than the STD value that we calculated in step 2. Otherwise, we will neglect it. It is important to note that the acceleration values are critical in many applications. Therefore, we need to ensure that we are working with accurate and reliable data. The STD value that we calculated in step 2 provides us with a benchmark for the expected range of acceleration values. Any difference between the previous and current acceleration values that falls within this range can be considered normal and will be neglected. On the other hand, if the difference between the previous and current acceleration values exceeds the STD value, we will consider this as an anomaly. This could indicate a problem or an unexpected event. In such a case, we will take appropriate action to investigate the cause of the anomaly and address it accordingly. In conclusion, by working with the previous and current acceleration values and utilizing the STD value calculated in step 2, we can effectively monitor and detect anomalies in the acceleration data. This will help us ensure the accuracy and reliability of our applications.

G) Filter acceleration values:

Filter the acceleration values using the standard deviation calculated in step 2. Only consider the changes in acceleration that are greater than the standard deviation as true acceleration values.

H) Convert acceleration to m/s^2:

The MPU6050's output unit is in g, where 1g = 9.8 m/s^2. Multiply the change in acceleration value by 9.8 to convert it to m/s^2.

I) Calculate displacement:

Compute the displacement (d) using the below formula $d = 0.5 * a * t\ 2$ (EQ 3.1) where a = acceleration (from step 7) and t = time in seconds.

J) Calculate cumulative displacement:

Finally, compute the cumulative displacement (c) using the formula $c(n) = C(n-1) + d(n)$

K) Improving the Accuracy of MPU6050

Measurements After implementing the mathematical equations and steps mentioned above, it's essential to take additional measures to enhance the accuracy of the MPU6050 measurements.

- **Use High Thresholds to Cancel Unwanted Axes:** To measure the displacement in a specific axis, such as the vertical (y) axis for detecting if a person is drowning, use high thresholds to cancel the readings from the x and z axes.
- **Implement Calibration Techniques:** Calibrate the MPU6050 regularly to ensure accurate and reliable readings. For instance, combine your code with a calibration sketch, like the one developed by Luis Rodenas, to perform calibration at the beginning of your code.
- **Adopt Filtering Techniques:** Use filtering techniques, such as averaging multiple acceleration values or implementing a first-in, first-out (FIFO) buffer, to reduce the impact of outlier acceleration values on your data.
- **Maintain a Controlled Environment:** Ensure a controlled environment for your MPU6050 by using fans or other temperature control methods to maintain the accelerometer at room temperature (25°C). Utilize the mpu.getTemperature() function from the Jeff Rowberg MPU6050 library to monitor the temperature.
- **Increase the Data Stream Frequency:** Increase the frequency of the MPU data stream to receive more data in a shorter period, allowing for more accurate readings. The Jeff Rowberg library provides functions to adjust the data stream frequency.

Measuring cumulative displacement accurately with an MPU6050 requires a detailed understanding of the sensor's capabilities, mathematical equations, and filtering techniques. By following the steps outlined in this article and implementing the best practices for calibration, filtering, and maintaining a controlled environment, you can enhance the accuracy of MPU6050 measurements for various applications.

(iii) Software Integration

If an abnormality in breathing or vertical movement of a person is detected, is to alert the swimmer that the device has detected a potential danger. If the swimmer presses a button next to the vibration element, then he denies the device. Otherwise, text messages are sent to the person to save him. All these steps are showed in figure 8.

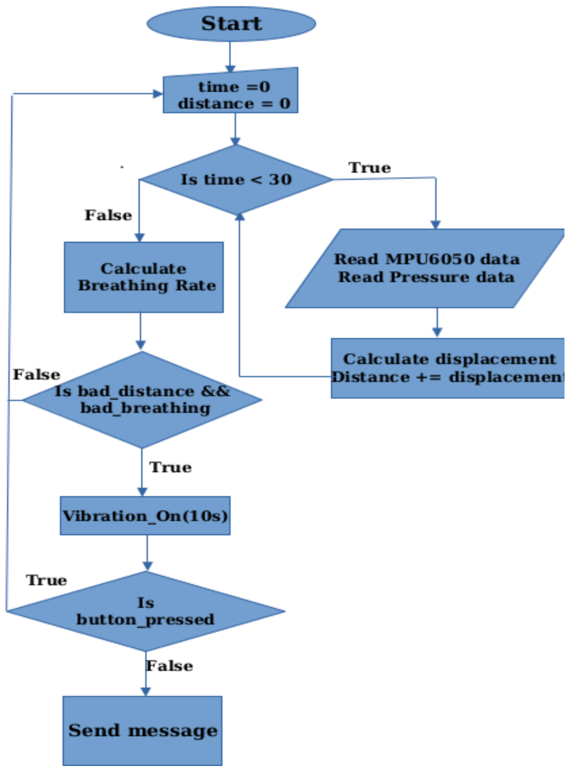

Figure 8: Flowchart of the algorithm

(iv) **RemoteXY app**

- Why this app has been used?

To display the value of the sensor in application rather than the serial and show swimmer information like breathing rate and drowning probability.

- What is RemoteXY ?

RemoteXY is a platform that allows you to create custom mobile applications for controlling your electronic projects remotely. The final form of application was displayed in figure 9.

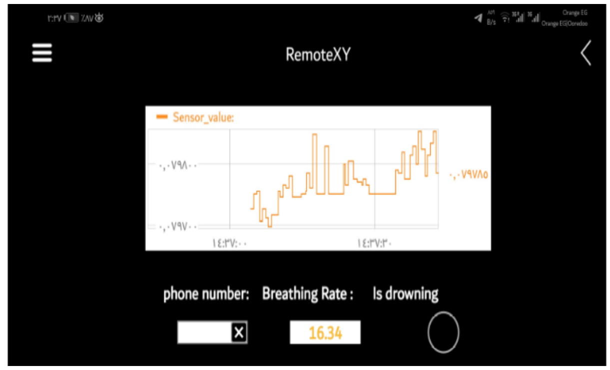

Figure 9: The final graphical user interface of the proposed application

4. Results

-The paper has successfully demonstrated the use of pressure sensors and theMPU6050 in detecting drowning, but we need to increase the accuracy and efficiency of the model in detecting drowning and add an airbag that inflates when drowning cases are detected.

To further enhance the capabilities of the belt, it is integrated with a RemoteXY application and a GSM module.

The proposed system can be further developed and implemented in various water-based activities to enhance safety and prevent accidents.

5. Conclusion

The has successfully demonstrated the use of pressure sensors and the MPU6050 in detecting drowning. By using pressure sensors to monitor the breathing rate of the person wearing the belt, we were able to identify if the person is in danger of drowning. Additionally, the MPU6050 enabled us to measure the cumulative displacement of the swimmer and track their movement in the water, and we hope that it can be further developed and implemented in real-world settings. To further enhance the capabilities of the belt, we integrated it with a Remotexy application and a GSM module. This allowed us to create an SMS message that alerts lifeguards when someone is in danger of drowning. This integration of advanced technology with the chest area belt has the potential to revolutionize water safety and prevent drowning accidents. We believe that our system can be further developed and implemented in various water-based activities to enhance safety and prevent accidents. The combination of multiple sensors, advanced algorithms, and communication technology has opened new possibilities for improving water safety. We hope that our system will inspire further research and development in this area, ultimately leading to a reduction in the number of drowning incidents around the world.

Competing interests: The authors declare that they have no competing interests.

Authors contributions: The authors declare that they contributed equally to all sections of the paper. All authors read and approved the final manuscript.

References:

1. Alnaggar M, Handosa M, Medhat T, Z Rashad M. Thyroid Disease Multi-class Classification based on Optimized Gradient Boosting Model. Egyptian Journal of Artificial Intelligence. 2023 Apr 1;2(1):1-4.
2. Alnaggar M, Handosa M, Medhat T, Rashad MZ. An IoT-based Framework for Detecting Heart Conditions using Machine Learning. International Journal of Advanced Computer Science and Applications. 2023;14(4).
3. Alhussan AA, Talaat FM, El-kenawy ES, Abdelhamid AA, Ibrahim A, Khafaga DS, Alnaggar M. Facial Expression Recognition Model Depending on Optimized Support Vector Machine. Computers, Materials & Continua. 2023 Jul 1;76(1).
4. Siam AI, Gamel SA, Talaat FM. Automatic stress detection in car drivers based on non-invasive physiological signals using machine learning techniques. Neural Computing and Applications. 2023 Jun;35(17):12891-904.
5. El-Rashidy N, Sedik A, Siam AI, Ali ZH. An efficient edge/cloud medical system for rapid detection of level of consciousness in emergency medicine based on explainable machine learning models. Neural Computing and Applications. 2023 May;35(14):10695-716.
6. Alharbey R, Dessouky MM, Sedik A, Siam AI, Elaskily MA. Fatigue state detection for tired persons in presence of driving periods. IEEE Access. 2022 Jun 22;10:79403-18.
7. Jalalifar, S., Kashizadeh, A., Mahmood, I., Belford, A., Drake, N., Razmjou, A. and Asadnia, M., 2022. A smart multi-sensor device to detect distress in swimmers. Sensors, 22(3), p.1059.
8. Vanegas, E., Igual, R. and Plaza, I., 2019. Piezoresistive breathing sensing system with 3d printed wearable casing. Journal of Sensors, 2019, pp.1-19.
9. Romagnoli, S., Ripanti, F., Morettini, M., Burattini, L. and Sbrollini, A., 2023. Wearable and Portable Devices for Acquisition of Cardiac Signals While Practicing Sport: A Scoping Review. Sensors, 23(6), p.3350.
10. Santos, M.O., Costa, J., Fernandes, T.R., Silva, C. and Faria, S.M., 2021, February. Wearable Inertial and Bio-signal Device for Real-time Swimmer's Monitoring. In 2021 Telecoms Conference (ConfTELE) (pp. 1-6). IEEE.
11. John, S.N., Ukpabio, I.G., Omoruyi, O., Onyiagha, G., Noma-Osaghae, E. and Okokpujie, K.O., 2019. Design of a drowning rescue alert system. International Journal of Mechanical Engineering and Technology (IJMET), 10(1), pp.1987-1995.
12. Salehi N, Keyvanara M, Monadjemmi SA (2016) An Automatic Videobased Drowning Detection System for Swimming Pools Using Active Contours. International Journal of Image, Graphics and Signal Processing 8(8):1-8
13. H. Kam, W. Lu, and W.-Y. Yau, “A Video-Based Drowning Detection System BT - Computer Vision — ECCV 2002,” 2002, pp. 297–311.

14. W. Lu and Y.-P. Tan, "A camera-based system for early detection of drowning incidents," in Image Processing. 2002. Proceedings. 2002 International Conference on, 2002, vol. 3, pp. III– III.

15. Ji Tao, M. Turjo, Mun-Fei Wong, Mengdi Wang, and Yap-Peng Tan, "Fall Incidents Detection for Intelligent Video Surveillance," 2005 5th Int. Conf. Inf. Commun. Signal Process. pp. 1590–1594, 2005.

16. Fei, Lei, Wang Xueli, and Chen Dongsheng. "Drowning Detection Based on Background Subtraction." Embedded Software and Systems, 2009. ICESS'09. International Conference on. IEEE, 2009.

17. Siam AI, El-Affendi MA, Abou Elazm A, El-Banby GM, El-Bahnasawy NA, Abd El-Samie FE, Abd El-Latif AA. Portable and real-time IoT-based healthcare monitoring system for daily medical applications. IEEE Transactions on Computational Social Systems. 2022 Nov 23.

18. Alnaggar M, Siam AI, Handosa M, Medhat T, Rashad MZ. Video-based real-time monitoring for heart rate and respiration rate. Expert Systems with Applications. 2023 Sep 1; 225:120135.

19. D. Jegede, "The Design and Construction of a Robotic Arm Using an 8051 Microcontroller Development of a Microcontroller Based Robotic Arm," 70 in 2007 Computer Science and IT Education Conference Development, 2007, no. June 2007.

20. Chaudhari T et al (2018) Anti Drowning system using remote alert. IOSR Journal of Engineering (IOSRJEN) 1:38-

21. Hemalatha S et al (2017) Automated Drowning Detection and Security in Swimming Pool. International Research Journal of Engineering and Technology (IRJET) 4(6):17-21

22. Pantelopoulos and N. G. Bourbakis, "A survey on wearable sensor-based systems for health monitoring and prognosis," IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev., vol.

23. M. Kharrat, A. Y. Wakuda, and A. S. Kobayashi, "Near drowning detection system based on swimmer's physiological information analysis," vol. 17, no. 201, p. 99635483, 2011.

24. Ramdhan, Muhammad, et al. "An early drowning detection system for Internet of Things (IoT) applications." TELKOMNIKA (Telecommunication Computing Electronics and Control) 16.4 (2018): 1870-1876.AlMahmoud, A.A., Khalifa University of Science and Research (KUSTAR), 2018. Sonar based drowning detection system, method and kit. U.S. Patent 9,972,188.

25. AlMahmoud, A.A., Khalifa University of Science and Research (KUSTAR), 2018. Sonar based drowning detection system, method and kit. U.S. Patent 9,972,188.

26. Jose A, Udupa G (2020) Gantry robot system for preventing drowning accidents in swimming pools. Materials Today: Proceedings 46:4975-4981

27. Kanchana, A. Kavya, G. R, Kavitha, C., Soumyashree, V. & Salila H. (2017). Automated drowning detection and security in swimming pool. International Research Journal of Engineering and Technology (IRJET), 4(6), 29-38.

28. Kulkarni, A., Lakhani, K. and Lokhande, S., 2016, September. A sensor based low-cost drowning detection system for human life safety. In 2016 5th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO) (pp. 301-306). IEEE.

29. B. J. Mortazavi, M. Pourhomayoun, G. Alsheikh, N. Alshurafa, S. I. Lee, and M. Sarrafzadeh, "Determining the single best axis for exercise repetition recognition and counting on smartwatches," in Wearable and Implantable Body Sensor Networks (BSN),

2014 11th International Conference on. IEEE, 2014, pp. 33–38

30. J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, "Activity classification using realistic data from wearable sensors," IEEE Transactions on information technology in biomedicine, vol. 10, no. 1, pp. 119–128, 2006. 71

31. L. Bao and S. S. Intille, "Activity recognition from user-annotated acceleration data," in International Conference on Pervasive Computing. Springer, 2004, pp. 1–17.

32. Li, D.; Yu, L.; Jin, W.; Zhang, R.; Feng, J.; Fu, N. An Improved Detection Method of Human Target at Sea Based on Yolov3. In Proceedings of the IEEE International Conference on Consumer Electronics and Computer Engineering, Guangzhou, China, 15–17 January 2021

33. Chan, Y.-T.; Hou, T.-W.; Huang, Y.-L.; Lan, W.-H.; Wang, P.-C.; Lai, C.-T. Implementation of deep-learning-based edge computing for preventing drowning. In Proceedings of the International Conference on Industrial Application Engineering, Taiwan, China, 26–30 March 2020.

34. Handalage, U.; Nikapotha, N.; Subasinghe, C.; Prasanga, T.; Thilakarthna, T.; Kasthurirathna, D. Computer Vision Enabled Drowning Detection System in Proceedings of the 3rd International Conference on Advancements in Computing (ICAC), Colombo, Sri Lanka, 9–11 December 2021.

35. C. Xu, K. Wang, P. Li, R. Xia, S. Guo, and M. Guo, "Renewable energyaware big data analytics in geo-distributed data centers with reinforcement learning," IEEE Transactions on Network Science and Engineering, vol. 7, no. 1, pp. 205–215, 2020.

36. Alotaibi, A. Automated and Intelligent System for Monitoring Swimming Pool Safety Based on the IoT and Transfer Learning. Electronics 2020, 9, 2082.

37. R. E. Johnson, S. Linderman, T. Panier et al., "Probabilistic models of larval zebrafish behavior reveal structure on many scales Current Biology, vol. 30, no. 1, pp. 70–82.e4, 2020.

38. J. Ren, Y. Guo, D. Zhang, Q. Liu, and Y. Zhang, "Distributed and efficient object detection in edge computing: challenges and solutions, IEEE Network, vol. 32, no. 6, pp. 137–143, 2018, 2020.

39. F. Ni, J. Zhang, and M. N. Noori, "Deep learning for data anomaly detection and data compression of a long-span suspension bridge," Computer-Aided Civil and Infrastructure Engineering Even vol. 35, no. 7, pp. 685–700, 2020