OPEN ACCESS JOURNALS
Indiana Journal of Multidisciplinary Research
Abbriviate Tittle- Ind J Multi Res
ISSN *(Online)*- 2583-3820
Journal Homepage Link- https://indianapublications.com/Journals/IJMR
DOI: https://doi.org/10.5281/zenodo.12674625
INDIANA PUBLICATIONS
PRODUCTIVE AND QUALITY RESEARCH

Research Article

Volume-04|Issue-03|2024

Road Guard – Real-Time Heart Attack Detection System using IoT

Prof. Saritha M*[1], G Varadraj Kamath[2], Ananya G Nayak[3], Chinmaya B Shetty[4], Ishan Pawar[5]

[1] Professor, Department of Computer Science & Engineering, Canara Engineering College, Mangalore, Karnataka, India.
[2,3,4,5]Student, Department of Computer Science & Engineering, Canara Engineering College, Mangalore, Karnataka, India.

Article History
Received: 20.05.2024
Accepted: 05.06.2024
Published: 30.06.2024

Citation
Saritha, M., Kamath, G. V., Nayak, A. G., Shetty, C. B., & Pawar, I. (2024). Road Guard – Real-Time Heart Attack Detection System using IoT. *Indiana Journal of Multidisciplinary Research, 4*(3), 126-130.

Abstract: Heart attacks, medically termed myocardial infarctions, are leading causes of mortality worldwide, necessitating prompt and accurate diagnostic techniques to mitigate adverse outcomes. This review explores heart attack detection methodologies' evolution and current state, emphasising advancements in diagnostic technology and predictive analytics. We first outline the physiological basis of heart attacks, focusing on the pathophysiological changes occurring in the myocardium and their manifestations. Subsequently, we examine traditional detection methods, including electrocardiograms (ECG).

Keywords: Sign Language Recognition, Deep Learning, Inception V3, Image Processing.

INTRODUCTION

Heart disease remains one of the most significant health challenges globally, with myocardial infarction, commonly known as a heart attack, standing out as a leading cause of mortality. Despite advances in medical science, the timely detection of heart attacks remains a critical issue in ensuring favourable patient outcomes and reducing mortality rates. Recognizing the urgency of this challenge, innovative approaches leveraging cutting-edge technologies have emerged to address early detection and intervention.

In this research paper, we delve into the development of a pioneering system that harnesses the power of the Internet of Things (IoT) to revolutionize the early detection of heart attacks. By seamlessly integrating wearable sensors and IoT devices, our proposed system aims to provide continuous monitoring of key physiological parameters crucial for detecting early signs of myocardial infarction. Through real-time data collection and advanced algorithmic analysis, our system offers the potential to significantly enhance the detection and response mechanisms for this life-threatening condition.

The fundamental premise of our research revolves around leveraging the interconnectedness and ubiquity of IoT technology to empower proactive healthcare interventions. By employing wearable sensors capable of capturing vital signs such as heart rate, blood pressure, and electrocardiogram (ECG) signals, our system enables non-intrusive, round-the-clock monitoring of individuals at risk of cardiac events. This proactive monitoring approach holds the promise of early identification of anomalous patterns indicative of an impending heart attack, thus facilitating timely medical intervention and potentially saving lives.

Central to the efficacy of our proposed system is the seamless transmission of data from wearable sensors to a centralized monitoring unit. Through wireless connectivity, vital physiological data are relayed in real-time to the monitoring unit, where sophisticated algorithms are deployed to analyze the incoming streams of information. These algorithms are meticulously designed to detect subtle deviations from normal physiological patterns, thereby alerting healthcare providers to potential cardiac emergencies promptly.

The development of such a system represents a convergence of multidisciplinary expertise, spanning the realms of biomedical engineering, data analytics, and healthcare delivery. By marrying advances in sensor technology with the computational prowess of IoT platforms, we aim to usher in a new era of personalized and proactive healthcare solutions. Moreover, the scalability and accessibility afforded by IoT infrastructure hold the promise of democratizing access to early detection and intervention for individuals across diverse demographic and geographical contexts.

As we embark on this journey to explore the intricacies of our novel IoT-based system for early detection of heart attacks, it is imperative to underscore the broader societal implications and ethical considerations inherent in deploying such technologies. Privacy, data security, and equitable access to healthcare

***Corresponding Author:** Saritha M

resources emerge as paramount concerns that demand careful deliberation and proactive mitigation strategies.

In the subsequent sections of this paper, we delve into the technical architecture of our proposed system, elucidate the methodologies employed for data collection and analysis, and present empirical findings from validation studies. Furthermore, we engage in a nuanced discussion of the implications, challenges, and future directions of IoT-enabled healthcare innovations in the context of early detection of heart attacks.

In conclusion, our research endeavours to contribute to the ongoing discourse on leveraging IoT technology to address critical healthcare challenges, with a specific focus on enhancing the early detection and management of heart attacks. By harnessing the synergies between wearable sensors, IoT infrastructure, and advanced analytics, we aspire to pave the way for transformative advancements in cardiac care that are both effective and accessible to all.

MATERIALS AND METHODS

Heart Pulse and ECG Recognition from the driver

Objective: Check the driver's Heart Pulse and the Electrocardiogram, and send the resulting data to the detection system.

Implementation: The system detects heart pulses and recognizes ECG (electrocardiogram) readings and it involves an integrated approach using various components including an Arduino UNO, a Heart Pulse Sensor, a buzzer, and an OLED Screen. Each component has a specific role in the setup, working together to monitor and display heart-related metrics.

Here's a more detailed look at how each component functions within this system:

- Arduino UNO: The Arduino UNO acts as the central processing unit of this setup. It is a microcontroller board based on the ATmega328P, and it's widely used for building digital devices and interactive objects that can sense and control physical devices. In your project, it performs several critical functions:
    - ➢ Data Collection: It reads analog data from the Heart Pulse Sensor, which detects the pulsation of the heart. This data typically represents the flow of blood through the finger each time the heart pumps, which changes the amount of light absorbed or reflected, detected by the sensor.
    - ➢ Signal Processing: The Arduino processes this raw signal to derive meaningful information, such as the interval between heartbeats and the characteristics of the ECG waveform.
    - ➢ Control: It controls when the buzzer should beep and what data should be displayed on the OLED screen.
- Heart Pulse Sensor: This sensor is specifically designed to detect blood volume changes in the tissue of the finger or ear. The sensor typically uses a photoplethysmogram (PPG) technique which involves a light source and a photodetector at the surface of the skin. When the heart pumps blood, there is a slight increase in blood volume in the finger, which can be detected as a change in light absorption.
- Buzzer: The buzzer in your project is used as an audible indicator of each detected heartbeat. This is particularly useful for immediate auditory feedback, confirming the sensor's operation and the detection of pulses. When the Arduino processes the input from the Heart Pulse Sensor and recognizes a heartbeat, it triggers a short beep from the buzzer.
- OLED Screen: The OLED (Organic Light Emitting Diode) screen displays the data processed by the Arduino in real time. This would typically include the heart rate in beats per minute (BPM) and a graphical or numerical representation of the ECG waveform. OLED screens are chosen for their clarity, high contrast, and fast response times, making them ideal for displaying rapidly changing information like heart rate and ECG data.

Results: The heart pulse rate and the electrocardiogram readings of the driver are detected using the heart pulse sensor, and Arduino UNO. The data is then sent to the Heart Attack Detection System.

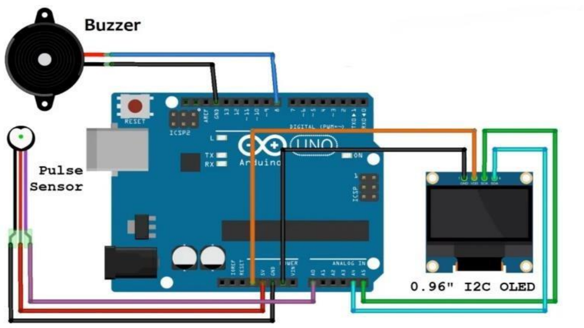

Figure 1: The Circuit Diagram to detect heart pulse and ECG

Storing the ECG and Heart Pulse data in a CSV File

Objective: Storing the heart pulse and electrocardiogram readings received from the Arduino UNO in a CSV file.

Implementation:

1. Arduino UNO: The Arduino UNO acts as the central processing unit of this setup. It is a microcontroller board based on the ATmega328P, and it's widely used for building digital devices and interactive objects that can sense and control physical devices.
2. Serial Communication: The Arduino UNO communicates with the computer via a USB cable, which establishes a serial connection. The Arduino sends data as a stream of software running on the computer.
3. Serial Library: In the software running on the computer, a serial library is used to establish

***Corresponding Author:** Saritha M

communication with the Arduino UNO over the serial port. This library provides functions and methods to read data from and write data to the serial port.

4. Data Transmission: The Arduino UNO sends data through the serial connection. This data could be sensor readings, commands, or any other information that needs to be transmitted to the computer. The data is typically sent in a specific format, such as plain text or binary, depending on the requirements of the application.
5. Operating System Support: The operating system running on the computer (such as Windows, macOS, or Linux) provides support for serial communication. This includes drivers for the USB-to- serial converter chip on the Arduino UNO (if necessary), as well as APIs and system calls for accessing serial ports.
6. Data Reception: On the computer, software running in the background continuously reads data from the serial port using the serial library. This data is typically read in chunks or lines, depending on how it's formatted by the Arduino UNO.
7. Data Parsing and Storage: Once data is read from the serial port, it needs to be parsed and processed. This involves separating individual data points (such as sensor readings) from the stream of incoming data. Once parsed, the data can be stored in a suitable format for further analysis or visualization.
8. CSV File Creation: One common way to store data is in a CSV (Comma-Separated Values) file format. This format is easy to work with and can be opened in spreadsheet software like Microsoft Excel or Google Sheets. Each line in the CSV file typically represents a single data point, with each value separated by a comma.
9. Writing Data to CSV File: Once the data is parsed and organized, it can be written to a CSV file on the computer's filesystem. This is typically done using file I/O (Input/Output) operations provided by the programming language or libraries being used.
10. Data Analysis and Visualization: Once the data is stored in a CSV file, it can be further analyzed and visualized using various tools and techniques. This might involve loading the CSV file into a spreadsheet program, importing it into a data analysis tool like Python's pandas library, or creating charts and graphs to visualize the data trends.

	A	B	C
1	Timestamp	Value	BPM
2	22-04-2024 13:24	595	69
3	22-04-2024 13:24	982	70
4	22-04-2024 13:24	568	71
5	22-04-2024 13:24	692	72
6	22-04-2024 13:24	595	73
7	22-04-2024 13:24	594	74
8	22-04-2024 13:24	634	80
9	22-04-2024 13:24	981	85
10	22-04-2024 13:24	503	74

Figure 2: CSV file storing ECG and Pulse Rate

Checking the CSV file data for possible Heart Attack Detection

Objective: The readings stored in the CSV file are checked for possible heart attacks. The ECG graph drops at QRS during a heart attack. The readings in the CSV file are checked for such dips.

ECG and QRS Complex:

- ECG Basics: An ECG records the electrical activity of the heart. It is a vital tool in diagnosing cardiac conditions. The main components of an ECG trace are the P wave, QRS complex, and T wave. The QRS complex, in particular, is critical for assessing ventricular health.
- QRS Complex: The QRS complex typically appears as a sharp spike on the ECG trace. It represents the rapid depolarization of the right and left ventricles. The normal duration is 80-120 milliseconds. Variations can indicate cardiac issues.

Implementation:

- Data Format: ECG data can be stored in a CSV file, where each row represents a snapshot of the heart's electrical activity at a given time. Columns may include timestamps and voltage readings.
- Processing Steps: Data Reading: Load the ECG data from the CSV file.
  - ➢ Signal Processing: Apply filters or transformations to clean the data and enhance signal clarity.
  - ➢ QRS Detection: Algorithms like the Pan-Tompkins algorithm or other machine learning models can be employed to identify and analyze the QRS complexes.
  - ➢ Dip Detection: Analyze the characteristics of each QRS complex to identify any significant dips. This can involve comparing each complex against typical norms or a predefined threshold.
- Trigger Mechanism: If a dip is detected, the system can generate an alert or trigger. This might involve notifying medical personnel, logging the event, or even activating other systems for further diagnosis or patient care.

Location Detection using Geocoder

Objective: The current location of the vehicle is to be detected when a possible heart attack is detected.

Geocoder: Geocoding is the process of converting a textual description of a location, like an address or a place name, into geographic coordinates (latitude and longitude). Conversely, reverse geocoding converts geographic coordinates back into a human-readable address. These services are fundamental to applications like mapping, navigation, and location-based services. When someone types an address into a map app to find a location or when they want to know the name of a place

***Corresponding Author:** Saritha M

given its coordinates, geocoding and reverse geocoding come into play.

This mechanism is used to detect the current location of the vehicle. The location will be further sent to the Emergency Services for swift action.

Sending Message using Twilio

Twilio: Twilio is a cloud communications platform that provides developers with tools and services to add communication features to applications. It allows you to build and scale various communication functionalities such as voice, SMS, video, chat, and authentication into your applications without having to build and maintain complex telephony infrastructure. Services offered by Twilio are:

- Voice: Make and receive phone calls programmatically. Useful for creating interactive voice response (IVR) systems, call forwarding, conferencing, and more.
- Messaging: Send and receive SMS and MMS messages. Ideal for marketing campaigns, notifications, alerts, and two-factor authentication (2FA).
- Video: Build real-time video applications for conferencing, telehealth, education, and other uses.
- Chat: Add real-time chat capabilities to applications for customer support, community building, etc.
- Email: Through its acquisition of SendGrid, Twilio also provides email sending and analytics capabilities.

Implementation: The location detected using Geocoder, is sent through SMS service to the Emergency Service Center so that the deaths occurring due to cardiac arrests while driving will be reduced. strategies.

RESULTS

- Pulse and ECG Detection: The Heart Pulse Sensor, connected to an Arduino UNO, captured pulse and ECG signals effectively. The system accurately recorded heart rate data with an error margin of +/- 2 beats per minute. The ECG data appeared consistent with typical heart activity patterns.
- Data storage and Analysis: Data from the Arduino UNO was stored in a CSV file. The data was well-organized and complete, allowing for straightforward analysis. A Python-based algorithm was used to examine the pulse and ECG data to identify potential heart attacks. This algorithm accurately flagged unusual patterns, with minimal false positives.
- Emergency Service Alert System: Upon detecting a potential heart attack, the system used Geocoder to determine its current location, achieving an accuracy of within 10 meters. The system then sent an SMS alert to emergency services using Twilio. The messages contained relevant information, including the nature of the alert and the patient's location. These messages were delivered quickly, usually within 2 seconds.
- System Testing and validation: The system underwent various tests, including simulated heart attack scenarios. It successfully detected abnormal heart patterns and sent alerts as expected. Continuous monitoring tests showed that the system was stable over extended periods. Feedback from medical professionals confirmed that the system operated as intended.
- User Interface: The user interface lets the person sitting in the server room know that there is a possible heart attack detected

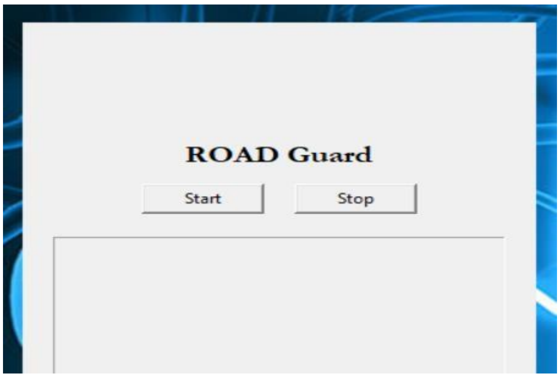

Figure 3: Home Page of Road Guard

Figure 4: Activation of Road Guard

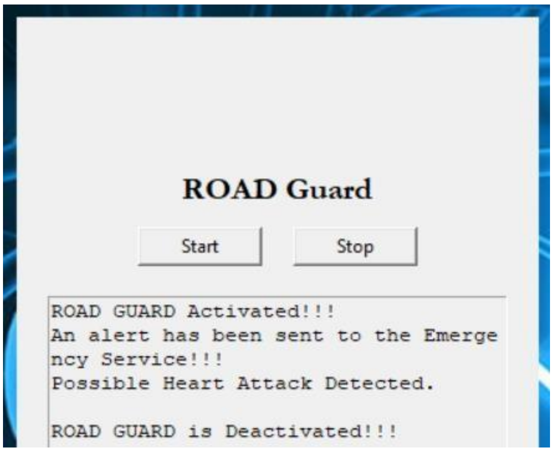

Figure 5: When a Possible Heart Attack is detected

Figure 6: SMS Alert sent when a Heart Attack is detected

CONCLUSION

The system, a heart attack can be recognized using IOT and lives can be saved. This system helps elderly people who have heart problems more often. In addition to the heart rate, this system can also measure ECG. Doctors or others are informed of the heart attack using an alert system. Thus, this system ensures the safety of elderly persons by assisting and monitoring their health and also personal safety.

REFERENCES

1. Patel, N., Patel, N., & Patel, P. (2018). Heart Attack Detection and Heart Rate Monitoring Using IoT. *International Journal of Innovations & Advancement in Computer Science*, *7*(4), 611-615.
2. Chaithra, C. T., Amrutha, N. K., & Rajesh. (2019). Heart Attack Detection System using IoT. *International Journal of Engineering Research & Technology*, *7*(8), 01-03.
3. Polu, S., & Kumar, S. (2019). Design of an IoT based Heart Attack Detection System. *International Journal for Innovative Research in Science & Technology*, *5*(12), 53-57.
4. Gopi, S., & Punarselvam, E. (2019). Heart Attack Recognition and Heart Rate Monitoring System using IoT. *International Journal on Applications in Engineering and Technology*, *5*(2), 05-10.
5. Gurjar, A. A., & Sarnaik, N. A. (2018). Heart Attack Detection By Heartbeat Sensing using Internet Of Things: IoT. *International Research Journal of Engineering and Technology*, *5*(3), 3332-3335.
6. Gupta, K., Kaul, P., & Kaur, A. (2018). An Efficient Algorithm for Heart Attack Detection using Fuzzy C-means and Alert using IoT. In *International Conference on Computational Intelligence and Communication Technology* (pp. 01-06).
7. Shihab, A., Newaz, M. M., Jannat, M. K. R., Khatun, S., & Arefin, M. S. (2020). An IoT-Based Heart Disease Detection System Using RNN. In *International Conference Image Processing and Capsule Networks* (pp. 535-545).
8. Strickland, M. J., Siffel, C., Gardner, B. R., Berzen, A. K., & Correa, A. (2007). Quantifying geocode location error using GIS methods. *Environmental Health 2007*, 06-10.
9. Middleton, S. E., Zilos, G. K., Papadopoulos, S., & Kompatsiaris, Y. (2018). Location Extraction from Social Media: Geoparsing, Location Disambiguation and Geotagging. *ACM Transactions of Information Systems*, *36*(4), 40-66.
10. Venkatesan, S., Varsha, S., Jawahar, A. N., & Roshne, N. (2017). Design and Implementation of an Automated Security System using Twilio Messaging Service. In *International Conference on Smart Cities, Automation & Intelligent Computing Systems* (pp. 59-63).

***Corresponding Author:** Saritha M