

Smart Watch-based Stress Recognition to Control Difficulty Levels in Games

Raul-Alexandru Gorgan
Technical University of Cluj-
Napoca
Str. G. Barițiu 28, 400027,
Cluj-Napoca, România
gorganraul@yahoo.com

Adrian Sabou
Technical University of Cluj-
Napoca
Str. G. Barițiu 28, 400027,
Cluj-Napoca, România
adrian.sabou@cs.utcluj.ro

ABSTRACT

This paper describes a system for stress recognition based on data provided from sensors included in commercial smart watches. Nowadays, stress has become an important concern and the ability to accurately detect and manage those feelings is necessary. The proposed system uses the Photoplethysmography sensors to extract the intervals between heartbeats. By analyzing the variations in these intervals, a feature named Heart Rate Variability (HRV) can be extracted. HRV serves as an indicator of stress in terms of the balance between the sympathetic and parasympathetic nervous system. The data is collected, analyzed and interpreted in order to obtain information about the health status of the user. The system offers a non-intrusive and convenient method to identify stress and to manage the well-being of the user. To emphasize the practical utility of the system, the recognized intervals of stress are used to control the difficulty level of a game. This form of user interaction can be integrated in many different types of application, for instance health and wellness monitoring.

Author Keywords

Human-computer interaction; Stress recognition; Smart watch; Photoplethysmography; Heart rate variability; Unity;

ACM Classification Keywords

H.5.m. Information interfaces and presentation: Human – Computer Interaction

DOI: 10.37789/rochi.2023.1.1.7

INTRODUCTION

In today's fast-paced world, stress has become a problem that affects human beings regardless of age, occupation or social status. This reaction of the body, when it becomes chronic, can have a negative impact on the health and the quality of life of an individual. WebMD [13] presents that this can lead to different diseases such as anxiety, depression, sleep disturbances, high blood pressure or other cardiovascular or digestive problems. As wearable technologies evolve and smart watches are equipped with various sensors, there is an opportunity to develop innovative applications for stress recognition and

management. In this work we present a system that uses the capabilities of sensors embedded in smart watches to identify stress and to adapt the difficulty level of a game based on that information. The data is collected using the sensors, analyzed and interpreted to obtain statistics and information about the health status of the user.

Almost all recent smart watches contain Photoplethysmography sensors (PPG). There are three sensors combined, PPG Red, PPG Infrared and PPG Green that work together, send light waves to the skin and measure the changes in the absorption of the light. This information is then used to extract different parameters specific to every individual.

One of the main parameters computed and observed is heart rate variability (HRV). Peabody et al. [10] consider HRV one of the best metrics used in the approximation and identification of stress level and health status. This represents the variation of the time intervals between successive heartbeats. In a previous study regarding this parameter, Tiwari et al. [14] identified that a high value for the HRV represents a healthy state and a low level of stress and anxiety. A low value for HRV indicates problems in reactions and body adaptability in the environment induced by stress and anxiety.

Without history about the user and information about age, gender, actual health or other relevant parameters, the system uses a base value of HRV collected before the game starts as "ground truth". After the game starts, the reference value is compared to the new values in order to make decisions regarding difficulty level settings.

The system consists of four modules that communicate in order to offer the required functionality. The main component is represented by the application that extracts the HRV parameter from the sensors, which is installed on an Android smart watch that runs Wear OS, namely the Samsung Galaxy Watch 5 Pro because it provides all the necessary sensors and developer access to them. The adaptive game is developed using Unity engine. It uses a WebSocket connection to receive commands to control the game's difficulty level. The communication between the smart watch application and the game is mediated by a server application used as a REST service. This is used for

storing data and sending commands based on the HRV values. The web application is used for better monitoring and analysis of the data. It consists of 2D graphics and statistics in order to post-analyze data.

This work presents an efficient method for monitoring and managing stress using the sensors of a smart watch. In the context of a society that generates many stressful situations, managing stress can offer great benefits to the user.

The next section discusses related work, and the one after provides information about the technical aspects used in this system. After that, we present a system overview followed by the description of the implementation, along with the methodology used to extract the HRV values and to interpret them accordingly. Finally, we present an evaluation of the system and the conclusions and future development.

RELATED WORK

In the last decades, wearable technology evolved surprisingly and became indispensable in our lives. This offers the opportunity to monitor and to manage a wide range of physiological and behavioral information. This section presents methods and approaches used by other works and offers a perspective of the current state of this field of research. The analyzed characteristics represent methods to approximate the stress level, to compute HRV and to interpret it.

Siirtola [12] concludes that the majority of studies in this field use sensors that are not available in commercial smart watches. They use EDA (Electrodermal Activity) sensor known as Galvanic Skin Response. This sensor is available only in special devices designed for laboratory conditions. The device Empatica E4 [3] is used to identify stress levels. It contains a variety of sensors as EDA, BVP (Blood Volume Pulse), ST (Skin Temperature) and many of their combinations are tested in order to identify the one that offers a good accuracy, along with three machine learning algorithms to classify the information and identify a good combination of sensors: Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA) and Random Forest (RF). Results suggest that LDA classifier using the combination of ST, BVP and HR (Heart Rate) performs better, having an accuracy of 87.4% on 15 subjects.

Gjoreski et al. [4] analyzed a system composed of three different modules, a detector used in laboratory conditions that uses no context, a module that identifies the context and a module that approximate the stress levels using the context identified by the second component. They use machine learning algorithms and a combination of sensors: skin temperature, HR, EDA, BVP and accelerometer. The system performed well in laboratory conditions, having an accuracy of 83%.

To classify stress episodes, Xu et al. [16] used a clustering method. This algorithm proved to provide better results than similar works that do not use it. The data is collected and, after that, the algorithm of clustering takes place:

- The subjects are divided into clusters based on many criteria. This step is known as k-means clustering process.
- Every cluster is evaluated using a regression analysis machine learning algorithm.

Can et al. [1] use HRV and a classifier algorithm to identify stress in traffic conditions. The work separates the results into three classes, low stress, medium stress and high stress by using Minimum Distance Classifier. The algorithm obtained an accuracy of 90% in real life conditions.

A non-linear method is used by Seungjae et al. [11] to compute HRV and to analyze the level of stress. Data is collected from 74 people, 26 features about the physiological status are extracted, using intervals of 5 minutes and less than 5 minutes, and results are compared. To classify the data, Support Vector Machine (SVM) is used. The results showed that for the period of 5 minutes, the accuracy is 86.5% and for a period less than 5 minutes the accuracy is 90.5%.

Kang et al. [8] use a variety of methods to compute HRV and after that, the results were compared in order to identify the one that offers the best input for classifiers. The techniques used are SDNN (Standard Deviation of NN/RR intervals), RMSSD (Root Mean Square of Successive Differences), LF (Low Frequency signals), HF (High Frequency signals) and the ratio between LF and HF. SDNN offered the best results and the ratio between LF and HF the worst.

Kageyama et al. [7] use only frequency domain methods to compute HRV: LF, HF and LF/HF. LF represents the heart rate variability between 0.04-0.15 Hz and the influence of the sympathetic nervous system and HF represents the heart rate variability between 0.15-0.4 Hz and the influence of the parasympathetic nervous system. The results showed that this method does not provide the necessary accuracy for classifying the data.

The algorithm RMSSD is used by Vrijkotte et al. [15] to identify stress at work. Data was collected in two working days and in one non-working day. It was observed that there are variations in the value of HRV and heart rate in the working days. As mentioned before, a decrease in HRV value means a higher level of stress.

CONCEPTS

This section contains the theoretical information for understanding the concepts and technical components used in this stress recognition system. We will describe the sensors available in a smart watch, particularly the

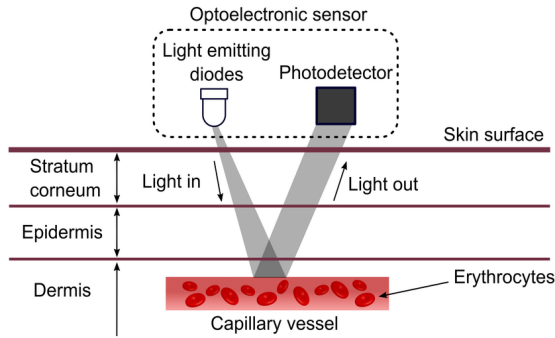


Figure 1 PPG working principle [2]

embedded BioActive Sensor, concepts about PPG, Samsung Privileged Health SDK (Software Development Kit) and computing HRV using RMSSD (Root Mean Square of Successive Differences).

Wearable devices such as smart watches or fitness bracelets evolved in order to fulfill the requirements in the Healthcare domain. They embed a variety of sensors that can be used to extract physiological information, even stress states. Almost all devices contain a heart rate sensor, an accelerometer, a blood oxygenation sensor, a light sensor or an ECG (Electrocardiogram) sensor. Using blood oxygenation or ECG data combined with heart rate data could give important insights into the human behavior related to stress. The problem that appears is that the usage of ECG or oxygen level sensor involves actions made by the user that disturbs the purpose of the project. The user needs to place one or two fingers on the smart watch electrodes and in this way, it is impossible for the user to play the game and to collect relevant information. Recent smart watches contain an embedded BioActive Sensor. This contains many sensors combined on a single chip, sensors like PPG Green PPG Red or PPG Infrared. These sensors can be used to collect data about heart rate and heart rate variability and use photoplethysmography to do this.

Photoplethysmography is a non-intrusive technology that uses infrared light to measure the variations in blood volume. It consists of a light emitting diode and a photodetector. The diode emits light and the photodetector measures the quantity of light reflected. The working principle of photoplethysmography is presented in Figure 1. The variations of the blood volume from the vessels during a heartbeat are visible in the amount of reflecting light back to the photodetector. The absorption is different at the exact moment of a heartbeat and the time between two heartbeats can thus be identified. This technology provides some features that can be used to collect information about human reaction and behavior based on the heart activity. It offers parameters such as the systolic amplitude, the pulse area, the RR interval (the interval between two successive heartbeats) and the arterial stiffness. The parameter used in

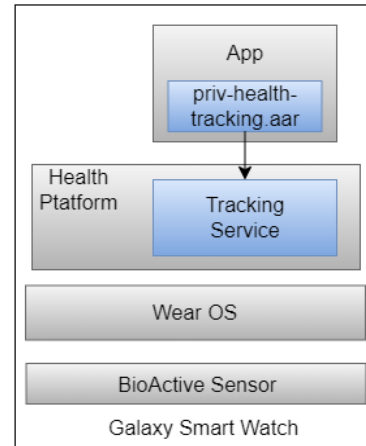


Figure 2 Architecture of the system using Samsung Privileged Health SDK

our work is RR interval because HRV is measured based on those values.

The wearable device does not provide direct access to the raw data collected by the sensors unless the Samsung Privileged Health SDK is used to allow this. Developers need to send a request to Samsung to use the SDK in order to collect the data. The main library of the SDK is then imported into the project files. Figure 2 presents the architecture of a system that uses the SDK. It collects data from sensors such as Heart Rate, PPG, Oxygen level, accelerometer, ECG, BIA (Bio-electric impedance analysis) but the sensors of interest for this project are those in Table 1, along with the type of retrieval and the amount of data gathered. The SDK offers a set of APIs (Application Programming Interface) to access sensor data, including BioActive Sensor that contains PPG sensors. The main parameter collected is the interval between every successive heartbeat, which can be used to compute HRV using several equations and methods.

Table 1 Sensor data specifications

| Sensor | Type | Description |
|------------|-----------|-----------------------------------|
| Heart Rate | Batching | 1 Hz, 1 value / second |
| PPG Green | Batching | 25 Hz, 300 values / 12 seconds |
| PPG IR | On-demand | 100 Hz, 1 value / 10 milliseconds |
| PPG Red | On-demand | 100 Hz, 1 value / 10 milliseconds |

Heart rate variability is a measure of the variation in time of successive heartbeats and represents the ability of the nervous system to respond to different stimuli. It reflects the balance between the sympathetic and parasympathetic nervous system and can be computed in the time domain or in the frequency domain. In the time domain, the analysis consists of the quantity of time between the heartbeats and results in an absolute value. In the frequency domain, the signal is passed through a Fourier transformation and after that is analyzed based on frequencies.

The main methods used to compute HRV are presented by Kubios [6]. The RR intervals are the intervals between two successive heartbeats:

- Mean RR – the mean value of the RR intervals
- SDNN – standard deviation of the NN / RR intervals
- SDD – standard deviation of the differences between successive RR intervals
- RMSSD – root mean square of the successive differences between heartbeats.

Other methods that belong to frequency domain which transform time series data into frequency data are:

- LF – low frequency signal analysis
- HF – high frequency signal analysis
- LF/HF – ratio of the low frequency and high frequency signals

RMSSD is used to collect the absolute value of the heart rate variability based on an arbitrary amount of data and this offers stability and accuracy.

Farnsworth [5] illustrates the difference of heart rate variability measured with RMSSD in two phases: non-stressed and stressed. In both moments, the heart rate value registered is 92bpm, the mean RR is 650ms and making a distinction is impossible. Figure 3 and Figure 4 demonstrate the importance of HRV. In the first case, HRV value is 313ms, a fact that suggests a good activity of the nervous system. In the second case, HRV registered is 31ms. This low value suggests a low adapting activity of the nervous system, stress or anxiety.

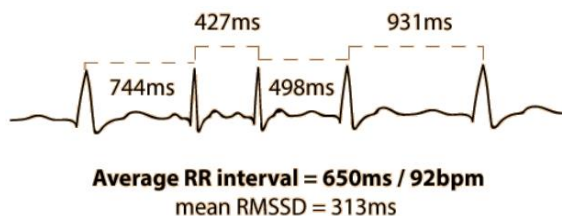


Figure 3 RMSSD in non-stress condition [5]

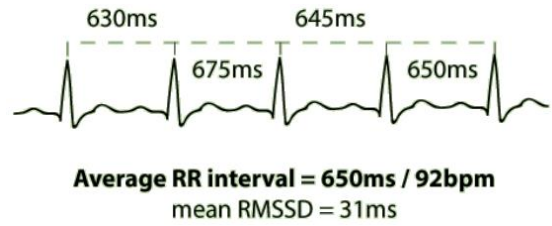


Figure 4 RMSSD in stress condition [5]

SYSTEM OVERVIEW

This section presents an overview of the recognition system and its components. The conceptual architecture is presented because it offers a high-level perspective of the structure and the way that the modules communicate. Figure 5 illustrates the components of the system. It consists of four main modules and the persistence layer:

- Smart Watch Application
- Server Application
- Unity Client Application (the game)
- Web Application
- Database

The Smart Watch Application represents the main component of the system. It has the functionality of connecting to the Health Tracking Service of the Samsung Privileged Health SDK, collecting the data from sensors, computing HRV and sending it to the Server Application.

The Server Application has the functionality of retrieving the HRV data from the Smart Watch Application, interpreting the value and sending commands to the game. The Server Application also performs the management of the data in terms of transforming in appropriate formats and storage. The data is persisted in a relational database of type PostgreSQL.

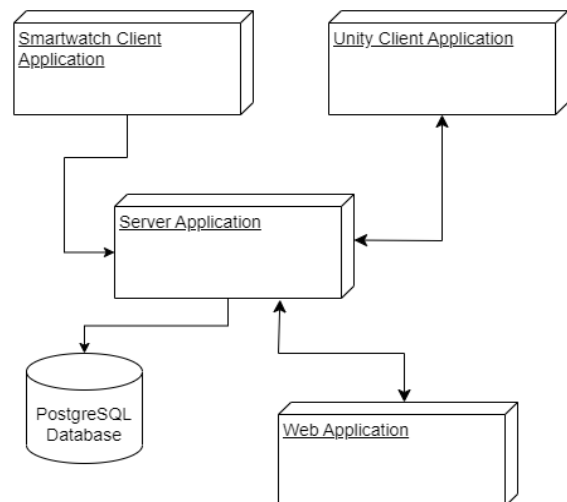


Figure 5 Conceptual architecture of the system

The Unity Client Application represents the game that receives commands from the server based on data collected by the smart watch application and is implemented as an “Endless Runner” type game with increasing speed. The player has three lives that are lost at an impact with obstacles. As the speed increases, the user is more involved into the game and changes in nervous system should appear.

The Web Application is implemented to offer a better perspective on the data collected from the Smart Watch Application. The real time data collected can be visualized and analyzed using this application that communicates with the Server Application.

DESIGN AND IMPLEMENTATION

This section presents the implementation of the recognition system and the approaches used in the interpretation of the values collected from the sensors.

The main component of the system is represented by the Smart Watch Application. The application is developed using Android Studio and the Kotlin programming language. The Smart Watch used is Samsung Galaxy Watch 5 Pro with Wear OS. The architecture of the application is MVC (Model-View-Controller). This design pattern is described in Figure 6. This module uses the MVC in a slightly different way and modifies the functionalities of the components to match the requirements. The activities and fragments that compose the application represent both the View and Controller components.

The Model is used to serialize and deserialize the data and to be able to transport the information between the modules. We use a Passive Model because it does not have access to any interface, it is only used for communication. The passive model behavior is presented in Figure 7.

The View component has the responsibility to render the elements on the screen and to ensure human-computer interaction. This is represented by the design of the fragments used in the application.

The controller represents the business logic and the connection between the Model and the View. This approach offers benefits on the testing side and eases future development.

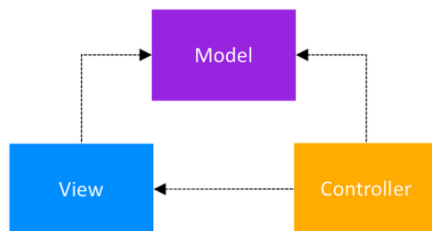


Figure 6 MVC structure [9]

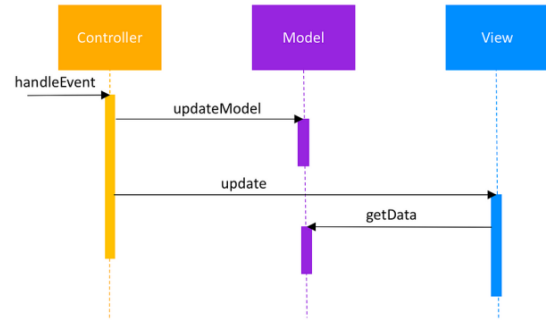


Figure 7 Passive Model MVC – behavior [9]

The Server Application represents the component that retrieves data processed by the Smart Watch Application, interprets it and sends commands to adapt the difficulty of the game based on that interpretation. It is designed as a RESTful Service and the applications use HTTP request to communicate with it. It is developed using Python and the framework Django. Rest Framework is used in order to handle the incoming requests. The server contains a method that interprets the value. This method is described later in the paper.

HRV Extraction Approach

The component responsible for the extraction of the data and computation of HRV is the Smart Watch Application. It uses Samsung Privileged Health SDK to be able to access the raw sensor data. The steps that have to be performed to access the data are to connect to the Health Tracking Service and to track the desired sensors. This service holds the sensor data and offers methods to access, retrieve and process it.

The first step is establishing the connection. After the connection to the Health Tracking Service succeeds, tracking takes place. In our project, the tracker object is bound to the Heart Rate type sensor that combines the functionality of many sensors from BioActive Sensor. The Observer Design Pattern is used to notify the main classes when new values from sensors are available.

The data gathered from the Heart Rate sensor contains:

- Heart rate value (HR)
- Heart Rate Status
- Inter-beat Interval (IBI)

Because the data can be disturbed by noises, we use a mask to verify if the IBI data is correct. If the data is identified as incorrect, it is not taken into consideration when computing HRV. This ensures the consistency of data and a correct value of HRV.

HRV is computed using the RMSSD formula. It offers an absolute value of HRV and is an efficient method to capture the variation regarding the stress and anxiety. The RMSSD formula presented by Kubios [6] is illustrated in Equation 1:

Equation 1 RMSSD formula

$$RMSSD = \sqrt{\frac{1}{N-1} \sum_{n=1}^N (RR_{n+1} - RR_n)^2}$$

where N is the total number of IBI values, RR_n and RR_{n+1} are two successive intervals between heartbeats.

Then, this value is sent to the Server Application using the Retrofit library. This library offers the functionality for sending the data using HTTP requests to the server.

HRV Interpretation

The component responsible for interpreting the HRV value is the Server Application. The data is received from the Smart Watch Application, interpreted, transformed into commands and sent to the game.

The server interrogates the database to obtain the baseline value of HRV for a certain session. When new data comes across, it is compared to the reference value. If the new value is lower than the reference and the difference is considerable, the server sends a command to the game using WebSocket. This technology is used in order to be able to send real time commands to the game, without the need of long polling from the Unity client. Figure 8 illustrates an overview of the process that involves the smart watch application and server application.

Game difficulty adjusting

The Unity Client Application contains a class that creates a WebSocket connection to the server. This class has assigned the player C# Script that provides access to the speed.

When new data is received from the server, the command is used to maintain or to decrease the speed of the player. If the message from the server is "Decrease", the speed of the player is decreased. If the message is "OK", the speed is maintained as the user is not considered to be stressed.

EVALUATION AND RESULTS

In order to validate the system and to measure the performance obtained, different parameters are used and modified. The modified parameters are:

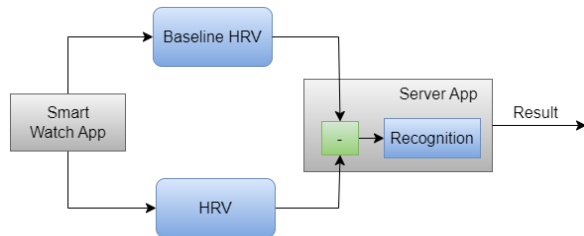


Figure 8 General overview of the recognition steps

- The amount of time needed to compute the reference value for HRV.
- The amount of time needed to compute HRV after the reference value is identified.
- The speed of the character (i.e. the difficulty level of the game).

The first step was to collect a baseline for comparison. It is then used to identify big variations in HRV after the game starts and the difficulty increases.

The first series of tests covered the length of the array on which the HRV is computed after the game starts. The baseline is collected using a 120-length array of IBI values. The reference value is collected before the game starts using 15, 30, 45, 60 and 120 valid values. The results are presented in Table 2. The varying of the sliding window before the game starts showed that the methods using 15, 30, 45 and 60 values capture only a limited part of the information and the variations are too prominent. This happens because the new values of HRV are computed using only the last collected IBI intervals that may vary fast. A good result is offered by the length of 120 for both the baseline and the new values registered when playing the game. It starts with an increase, a normal reaction of the human body during a game. After the difficulty increases, it is observed that HRV tends to lower the value and there is a considerable difference in HRV between the speed of 10 and 40.

Table 2 HRV values using different sliding windows and 120 values for the baseline

| Speed Length \ | 5 | 10 | 15 | 20 |
|----------------|--------|--------|--------|--------|
| 15 | 50.02 | 90.4 | 101.24 | 110.3 |
| 30 | 211.28 | 184.79 | 166.48 | 102.83 |
| 45 | 188.62 | 183.6 | 124.2 | 56.81 |
| 60 | 140 | 139.44 | 143.48 | 88.19 |
| 120 | 194.67 | 224.3 | 237.69 | 182.82 |
| Speed Length \ | 25 | 30 | 35 | 40 |
| 15 | 115.2 | 116.84 | 205.07 | 191.37 |
| 30 | 36.94 | 45.57 | 48.10 | 50.71 |
| 45 | 59.07 | 62.95 | 61.78 | 61.45 |
| 60 | 85.9 | 81.95 | 70.5 | 67.94 |
| 120 | 193.59 | 172.32 | 163.14 | 162.98 |

Table 3 HRV values using different amount of time for collecting the baseline

| Speed Time | 5 | 10 | 15 | 20 | 25 |
|-----------------------|-----------|-----------|-----------|-----------|-----------|
| 30 | 53.79 | 60.04 | 55.86 | 81.91 | 78.47 |
| 60 | 106.24 | 104.56 | 74.05 | 75.26 | 51.52 |
| 120 | 192.36 | 214.93 | 185.16 | 154.29 | 54.73 |
| Speed Time | 30 | 35 | 40 | 45 | 50 |
| 30 | 74.9 | 75.48 | 76.31 | 43.28 | - |
| 60 | 54.46 | 50.66 | 49.53 | 48.24 | - |
| 120 | 51.55 | 55.1 | 56.27 | 53.8 | - |

During the first series of tests and experiments, a limitation was identified. This limitation is represented by the fact that collecting the 120-length array of valid values, even if the frequency of receiving data is 1Hz, requires approximately 3 minutes, even more because of incorrect data received. The second series of tests covered the usage of time instead of number of values when identifying the baseline. For the baseline, 30, 60 and 120 seconds are considered and when the game starts, 120 seconds are used to compute the new values. In 120 seconds, approximately 75 valid values were collected. The results are presented in Table 3. The intervals of 30 and 60 seconds are too short because the number of IBI values is small. The interval of 120 seconds tends to perform better but there are identified big variations because of the lack of values.

The third series of tests address the limitation of the time-based computation of HRV. This is an addition to the first series of tests. We used the same length of the IBI array for the baseline and after the game started. Because the first series analyzes the length of 120 for both, Table 4 presents the results for using 30 and 60 valid values. This approach offers approximately the same results as the time-based analysis as 30 and 60 values are insufficient to identify the baseline.

Table 4 HRV values using different sliding window for baseline and new values

| Speed Length | 5 | 10 | 15 | 25 |
|-------------------------|-----------|-----------|-----------|-----------|
| 30 | 151.92 | 60.73 | 55.39 | 54.9 |
| 60 | 243.57 | 235.32 | 107.97 | 126.25 |
| Speed Length | 30 | 35 | 40 | 45 |
| 30 | 60.9 | 63.34 | 56.54 | 48.0 |
| 60 | 156.22 | 249.86 | 256.08 | 237.68 |

Experimental results indicate that the performance of the system is influenced by the number of values used or the allocated time to compute HRV. The time-based measurement does not provide optimal results due to the unpredictable and small number of values collected. HRV values fluctuates and there is no correlation with the difficulty levels because the data collected is not stabilized. The approach that uses a fixed number of valid values for HRV computing offers the same performance if the data is not sufficient to capture the stress patterns. Employing smaller window sizes, such as 15, 30, 45 or 60 values, does not provide sufficient stability and does not establish a strong correlation with game difficulty levels. In contrast, a larger window of intervals between heartbeats shows promising outcomes. Specifically, we found that using 120 values for both reference and comparison significantly improves stress recognition performance. By adopting this approach in HRV analysis, the accuracy and reliability is enhanced and lead to a more effective correlation between the difficulty level of the games and the HRV patterns.

CONCLUSION

The main objective of this paper was to analyze the data provided from sensors integrated in smart watches to identify the moments of stress and the human body behavior with regard to stimuli. The results are used in a practical way by adapting the difficulty of a game based on the users' level of stress.

The system consists of four components that communicate through several protocols in order to offer the required functionality. The main component is represented by the Smart Watch Application that is responsible for collecting the data and computing the HRV parameter used to measure the level of stress. The Server Application component is used to receive the data from smart watch, to interpret it and to send commands to another component, the game. The game receives commands from the server and adjusts the difficulty based on them. The fourth component is represented by the Web Application which is used for visualizing the data in 2D format.

The results obtained by the recognition system are those expected. It was demonstrated that it is possible to identify stress and to study the human body behavior and reactions by using only the sensors embedded in commercial smart watches.

As future development, the system can be improved with optional machine learning algorithms to perform the classification, to combine the data gathered with other sensor data such as accelerometer or to improve the computing method of HRV.

REFERENCES

1. Can, Y.S., Chalabianloo, N., Erkiz, D. and Ersoy, C. Continuous Stress Detection Using Wearable Sensors in Real Life: Algorithmic Programming Contest Case Study. *Sensors* 19, 8 (2019), 5-19.
2. de Moraes, J., Id, M., Rocha, Vasconcelos, G., Vasconcelos, F., Hugo, V., Albuquerque, V. and Alexandria, A. Advances in Photoplethysmography Signal Analysis for Biomedical Applications. *Sensors* 18, 10 (2018).
3. Empatica E4.
<https://www.empatica.com/research/e4/>.
4. Gjoreski, M., Gjoreski, H., Luštrek, M. and Gams, M. Continuous stress detection using a wrist device: in laboratory and real life. In *Proc. of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2016).
5. Heart Rate Variability – How to Analyze ECG Data.
<https://imotions.com/blog/learning/best-practice/heart-rate-variability/>.
6. HRV Analysis Methods.
<https://www.kubios.com/hrv-analysis-methods/>.
7. Kageyama, T., Nishikido, N., Kobayashi, T., Kurokawa, Y., Kaneko, T. and Kabuto, M. Self-reported sleep quality, job stress, and daytime autonomic activities assessed in terms of short-term heart rate variability among male white-collar workers. *Industrial health* 36, 3 (1998), 263–272.
8. Kang, M., Koh, S., Cha, B., Park, J., Woo, J. and Chang, S. Association between job stress on heart rate variability and metabolic syndrome in shipyard male workers. *Yonsei medical journal* 45, 5 (2004), 838–846.
9. Muntenescu, F. Android Architecture Patterns Part 1: Model-View-Controller.
<https://medium.com/upday-devs/android-architecture-patterns-part-1-model-view-controller-3baecef5f2b6>.
10. Peabody, J. E., Ryznar, R., Ziesmann, M. T. and Gillman, L. A Systematic Review of Heart Rate Variability as a Measure of Stress in Medical Professionals. *Cureus* 15, 1 (2023).
11. Seungjae, L., Bin Hwang, H., Seongryul, P., Sanghag, K., Jung Hee, H., Yoojin, J., Sejin, H., Hoon-Ki, P., Jongshill, L. and In Young, K. Mental Stress Assessment Using Ultra Short Term HRV Analysis Based on Non-Linear Method. *Biosensors*, 12, 7 (2022), 465.
12. Siirtola, P. Continuous Stress Detection Using the Sensors of Commercial Smartwatch. In *Adjunct Proc. of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and the 2019 International Symposium on Wearable Computers*, (2019).
13. The Effects of Stress on Your Body.
<https://www.webmd.com/balance/stress-management/effects-of-stress-on-your-body>.
14. Tiwari, R., Kumar, R., Malik, S., Raj, T. and Kumar, P. Analysis of Heart Rate Variability and Implication of Different Factors on Heart Rate Variability. *Current cardiology reviews* 17, 5 (2021).
15. Vrijkotte, T., van Doornen, L. and de Geus, E. Effects of work stress on ambulatory blood pressure, heart rate, and heart rate variability. *Hypertension* 35, 4 (2000), 880–886.
16. Xu, Q., Nwe, T. and Guan, C. Cluster-based analysis for personalized stress evaluation using physiological signals. *IEEE journal of biomedical and health informatics* 19, 1 (2015), 275–281.