Healthy IoT: A Posture Monitoring System

Victor Chau
School of Engineering & Technology
University of Washington Tacoma
Email: hqchau@uw.edu

Akshdeep Cheema
School of Engineering & Technology
University of Washington Tacoma
Email: alone117@uw.edu

Abstract — The rate at which jobs are becoming remote has increased, as more individuals can work from home versus sitting at an office desk. This means that the average time a person sits has likely increased. Our objective for this project is to implement a monitoring system which takes pictures of an individual while they are sitting. This paper proposes a posture monitoring system which can detect bad posture and notifies the individual of incorrect posture using different notification methods in order to draw the individual's attention to correcting their posture. The data needed to make the decision is acquired through photos taken of the individual and processed using IBM Watson Visual Recognition model. Data is then analyzed and used to return a value which can be used to determine if the posture is incorrect or not and display correct notification to the user.

Keywords **— Raspberry Pi, posture, IBM Watson, pictures, notification, sensors**

I. *INTRODUCTION*

As remote jobs are getting popular, more and more people are working at home than in their own office. The average American adult usually sits around 6.5 hours per day, and this is increased to 8 hours a day for teens. From 2019 to the present, there was an unexpected turn when COVID changed many people's livelihood since many adults are now working from home, which may lead to a potential increase in the amount of time sitting down versus standing. Prolonged sitting can harm the spine and cause chronic issues that need long-term therapy. Maintaining the correct sitting posture is essential for a healthy back and spine. The project prototype aims to monitor an individual's posture by taking pictures using a camera module attached to the Raspberry Pi at 5 minute time intervals. These pictures will be stored and analyzed using the cloud and then appropriate notifications will be given to the individual by the easily accessible Node-RED and Grafana [1] user interface dashboards to improve their posture.

II. *RELATED WORK*

Michal Sieniawski et. al proposed a sitting posture monitoring system which identifies the degree of incorrect posture during work and gives the user an opportunity to improve their posture. There is a server which listens to incoming REST requests and a database management system to store measurements to be analyzed later. The user interface displays the latest query for sitting posture, the amount of time the user has spent sitting, and displaying additional statistics. The collected system data can be used for further research towards an optimal solution that can improve the user's faulty sitting habits [8]. Horned Sungem used the RPi 3b, buzzer sensor, and a HS device to create a system which detected between a healthy and unhealthy sitting posture using Python on the backend.

III. *COMPONENTS & IMPLEMENTATION*

A. *Raspberry Pi 3B+*

A high performing and affordable single board computer that utilizes free software to teach people how to create programs and harness the power of computing. The Pi contains extended 40-pin GPIO header, Full-size HDMI, 4 USB 2.0 ports, CSI camera port for connecting a Raspberry Pi camera, DSI display port for connecting a Raspberry Pi touchscreen display, Micro SD port for loading your operating system and storing data, and more. We will use this alongside the GrovePi+ board to capture physical data to analyze [2].

B. *GrovePi+ Board*

The GrovePi+ is an add-on board with 15 Grove 4-pin interfaces that brings Grove sensors to the Raspberry Pi. It is the newest version compatible with Raspberry Pi model B/B+/A+/2/3/4. It has a collection of more than 100 inexpensive plug-and-play modules that sense, control, and capture real physical world data. This will allow us to use the sensors we need without requiring us to mess with the physical RPi hardware [3].

C. *Camera Module*

The v2 Camera Module has a Sony IMX219 8-megapixel sensor, can take high-definition video and still photographs. It has better image quality, colour fidelity, and low-light performance. It supports 1080p30, 720p60 and VGA90 video modes, as well as still capture. Attached via a 15cm ribbon cable to the CSI port on the RPi. We will use this to capture images of the user while they are sitting [4].

D. *Buzzer Sensor*

Has a piezo buzzer as the main component. The piezo can be connected to digital outputs, and will emit a tone when the output is HIGH. We will use this sound as one method of notifying the user that their posture is incorrect [5].

E. *LCD RGB Backlight*

Allows you to set the color to whatever you like via the simple and concise Grove interface. Takes I2C as the communication method. Allows you to display text on the display and even supports user-defined characters. We will use this as another method to notify the user that their sitting posture needs correction [6].

F. *Button Sensor*

The button sensor is a momentary push button. It contains one independent "momentary on/off" button meaning that it will send a true signal only when pressed down otherwise the signal sends false. The button outputs a HIGH signal when pressed, and LOW when released. We will use it to stop the buzzer from emitting sound and to turn off the LCD RGB Backlight module [7].

G. *Required Setup*

Connect the GrovePi+ board to the RPi and make sure all GrovePi+ libraries are installed. Make sure Node-Red and IBM Watson are up and running. Our wiring to the ports on the GrovePi+ board are as follows: Button sensor -> D3, Buzzer sensor -> D6, LCD RGB Backlight -> I2C-1.

H. *Parts*

RPi 3 B+, GrovePi+ board, Multiple GrovePi+ connection wires, 1 x Button sensor, 1 x Buzzer sensor, 1 x LCD RGB Backlight, 1 x Raspberry Pi Camera Module v2

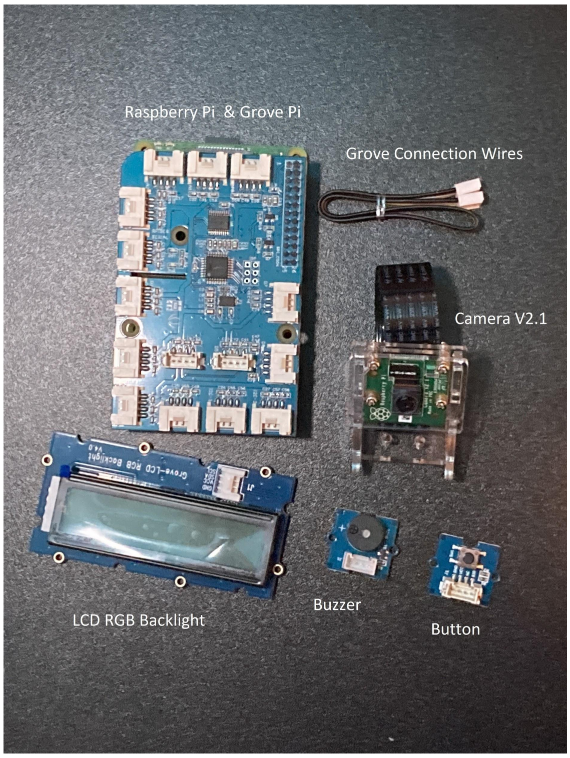

Figure 1: System Components

IV. *SYSTEM ARCHITECTURE & DESIGN*

The system is made up of two main components which is the front-end to display data visually for the end user, and the back-end which is to capture the necessary data, analyze it, and then send the appropriate information to be displayed on the front-end. The front-end consists of two parts that interact with the users: a dashboard such as Node-RED which can be accessed locally and an Grafana Dashboard which is cloud-based. There is also the LCD RGB Backlight alongside the buzzer module to notify the user with incoming warnings about incorrect sitting posture. For the back-end the two largest components are the IBM Watson Visual Recognition Model and the IBM Storage/Database.

The IBM Watson Visual Recognition Model is the core analysis module that evaluates images taken by the camera module on the RPi. IBM Storage is an effective way for storing uploaded images from the camera module as training data for the models, while the influx database [9] contains the information obtained through analysis of the images with timestamps, boolean values, confidence levels, and image url which can be fetched and displayed on the visualization dashboards. The design layout of our prototype is straightforward. We will have the RPi connected with the LCD RGB Backlight module, buzzer, camera sensor and have the entire setup located on a stand about 2 meters away from the desk where the user is sitting to capture images and send to our back-end for analysis. See *Figure 2* for a high-level design of our system.

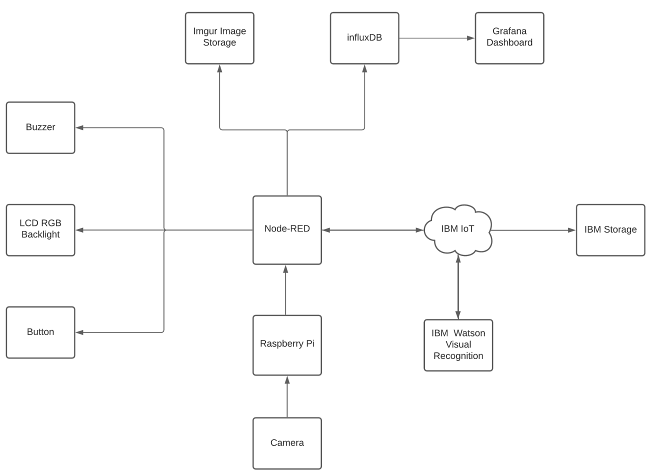

Figure 2: System Flow Diagram

A. *NODE-RED IMPLEMENTATION*

The primary function of the flow is to have the camera take pictures of the user in a sitting position in intervals of 5 minutes. After the picture is taken, a function node prepares a HTTP request to have the photo uploaded to an external photo storage called Imgur. Then, we extract the img link from Imgur, set up the necessary parameters needed for the machine learning model, and then send all this information through the IBM Watson node which will send it to the IBM Watson Visual Recognition service to see if a person is present in the image or not. There is a switch node which allows the flow to choose between two paths based on whether or not the person was present in the image. If the person was not present in the image, meaning the confidence level for a person detected in the image was less than 0.7, we send the information we extracted to a database to be stored and analyzed at a later time. The database table is called "Posture" and the columns it has are: time -> the timestamp at which image was taken by camera module, image_url -> link to image on Imgur, person_detected -> a boolean value, person_conf -> confidence level of person detected in image being a floating point number between 0 and 1, bad_posture -> a boolean value, posture_conf -> confidence level of user having correct sitting posture in image being a floating point number between 0 and 1. If a person is present in the image, confidence level greater than 0.7, then we will go through another IBM Watson machine learning model that checks the posture of the person in the image. If the confidence level is greater than 0.5 that means the person has bad posture while sitting down, otherwise they presumably have a good posture while sitting. The result is sent to the database for storage and future analysis. We also send the result to the buzzer sensor and the LCD RGB Backlight module. The buzzer sensor emits a high-pitched sound, while the LCD RGB Backlight module displays a warning against a red background with the string "Bad Posture". If the user wishes to stop the buzzer from emitting sound and clear the LCD RGB Backlight display, then they can press the button sensor to perform this operation. The second function this Node-RED flow is responsible for is to make a query to the influxDB which is triggered in intervals of 15 minutes. This query should grab all the results in the past 60 minutes from the database. We then analyze this information to check whether a single "false" value was in the person_detected column, and if this occurs, then the buzzer sensor should emit a high-pitched sound while the LCD RGB Backlight module should display the string "Take a break! Leave your seat!" against a yellow background. What this means is that the user has been sitting for too long and should get up and walk around for a bit to keep themself healthy. These are the two integral functions that this Node-Red flow performs and is the core component in making our project idea feasible. See *Figure 3* for the Node-RED flow layout to carry out these functions. See *Figure 4* to see how the information is being stored in the influxDB.

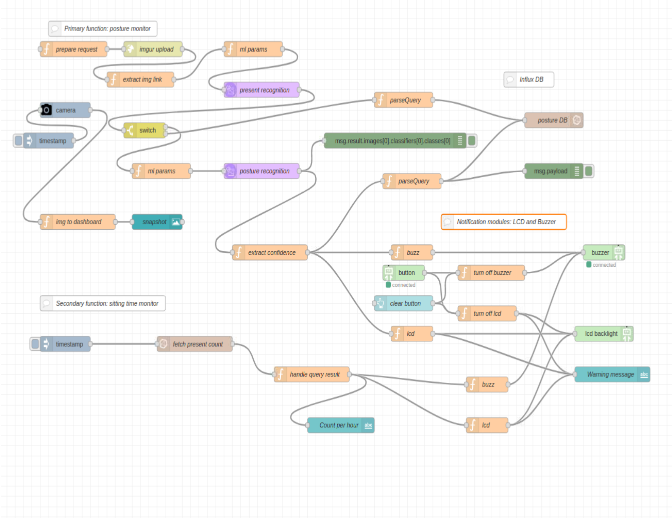

Figure 3: Node-RED Flow

name: Posture

time	bad_posture	img_url	person_conf	person_detected	posture_conf
1615795109316312900	true	https://i.imgur.com/ .jpeg	0.891	true	0.52
1615795115235547799	false	https://i.imgur.com/ .jpeg	0.87	true	0.181
1615795121761783775	false	https://i.imgur.com/ .jpeg	0.917	true	0.03
1615795128378030978	false	https://i.imgur.com/ .jpeg	0.76	true	0.174
1615795132873773639	false	https://i.imgur.com/ .jpeg	0.728	true	0.246
1615795137866565809	false	https://i.imgur.com/ .jpeg	0.776	true	0.48
1615795146507834227	false	https://i.imgur.com/ .jpg	0.918	true	0.054
1615795168515369422	false	https://i.imgur.com/ .jpeg	0.918	true	0.324
1615795175978296799	false	https://i.imgur.com/ .jpg	0.902	true	0.072
1615795211681635159	false	https://i.imgur.com/ .jpeg	0.843	true	0.27
1615795233080103529	false	https://i.imgur.com/ .jpg	0.008	false	0
1615795280727688836	false	https://i.imgur.com/ .jpeg	0.022	false	0

Figure 4: influxDB Table

B. *DASHBOARDS*

There were two visualization dashboards being used to display data to the end-user which are the Node-RED (edge-based) and Grafana dashboards (cloud-based). The Node-RED dashboard (*Figure 5*) features the latest snapshot from the camera, the warning message if applicable, and a button for the user to clear warning messages from buzzer as well as on the LCD screen/dashboard. For the cloud-based dashboard, we use Grafana (*Figure 6*) to retrieve data from the InfluxDB and summarize the recorded count of poor posture recurrences.

C. *MACHINE LEARNING*

We trained two classification models under the IBM Watson Visual Recognition service for two tasks: detect if the user is presented at desk, and poor posture recognition. The model for the first task was trained with 82 “Positive” images of the person sitting at desk, while the “Negative” class is a pool of 55 images of the desk without any people. The optimal threshold of detection for this model was chosen to be > 0.7 for positive

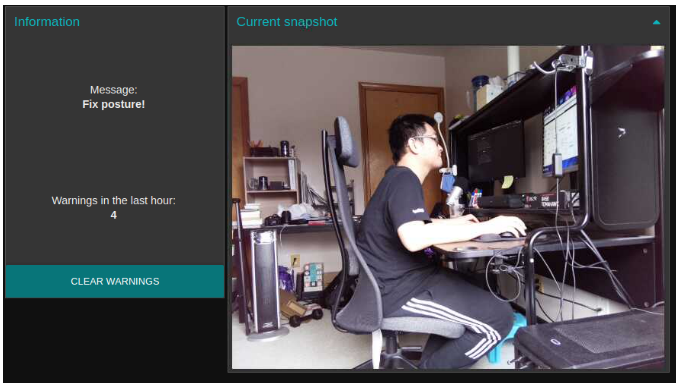

Figure 5: Node-RED Interface

detection. Posture recognition is a more complicated task, hence the training pool for the second model is larger, with 260 photos for “Positive” class of poor postures. The “Negative” class consists of 137 images of optimal sitting posture; and > 0.5 was chosen to be the positive threshold. We made an effort to collect diverse samples such as different camera angles, background/lighting, and people with various attire… but it proved to be a challenging task and no doubt that the models are currently biased towards a single particular setting.

V. *CONCLUSION*

Using the Internet of Things (IoT) concepts, smart systems have been built to deal with various real world challenges and problems in a multitude of ways. One such smart system is the posture monitoring system which we proposed in this report which has seen related work outside of our research. Using the camera module attached to the RPi alongside the buzzer and LCD RGB Backlight module connected to the GrovePi+ board, we managed to create a system that monitored a user’s sitting posture and notified them using the backlight module or via the visualization dashboards that their posture is incorrect. The diversity of needs which have arisen because of IoT have been achieved much more easily thanks to the simplicity and friendly user-interfaces of Node-Red, IBM Watson, Grafana, etc. to deliver quicker solutions to IoT related tasks. A decent amount of data was found and generated in order to ensure that the models we were trai. The finished prototype was tested using both images found online and pictures of the group members to achieve the highest accuracy possible. We mainly focused on making the buzzer and display module the primary ways the user will get notified that they have incorrect posture. The Posture Monitoring System provides a realistic and potentially feasible solution to help people of all ages improve their sitting posture and avoid chronic back issues in the future.

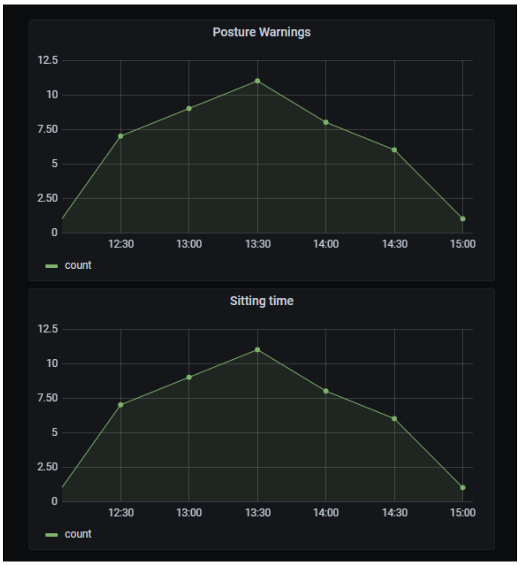

Figure 6: Grafana Dashboard

VI. *FUTURE WORK*

The Posture monitoring system can be enhanced further in order to provide the user a more fulfilling and positive experience. One addition that can be made is to generate suggestions on how to improve posture in general so the end-user is aware of the correct way of sitting. Another addition would be to create a reward system where the user can earn points for maintaining correct posture or not sitting down for 60 consecutive minutes and then redeem the points for a small prize. In terms of accuracy, the machine learning models could improve with more diverse data such as posture of various angles, different background/light intensity, and people with varied attires. Though this data is rather difficult to gather, an option for the user to review and evaluate captured snapshots in the dashboard is possible and could prove useful to improve the accuracy of the model, as it would be trained to recognize the settings of that specific user.

VII. *REFERENCES*

[1] "Grafana: The open observability platform," Grafana Labs. [Online]. Available: https://grafana.com/. [Accessed: 15-Mar-2021].

[2] Raspberrypi.org. 2021. [online] Available at: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/> [Accessed 15 March 2021].

[3] Seeedstudio.com. 2021. GrovePi+. [online] Available at: <https://www.seeedstudio.com/GrovePi.html> [Accessed 15 March 2021].

[4] Raspberrypi.org. 2021. [online] Available at: <https://www.raspberrypi.org/products/camera-module-v2/> [Accessed 15 March 2021].

[5] Zuo, B., 2021. Grove - Buzzer - Seeed Wiki. [online] Wiki.seeedstudio.com. Available at: <https://wiki.seeedstudio.com/Grove-Buzzer/> [Accessed 15 March 2021].

[6] Zuo, B., 2021. Grove - LCD RGB Backlight - Seeed Wiki. [online] Wiki.seeedstudio.com. Available at: <https://wiki.seeedstudio.com/Grove-LCD_RGB_Backlight/> [Accessed 15 March 2021].

[7] Zuo, B., 2021. Grove - Button - Seeed Wiki. [online] Wiki.seeedstudio.com. Available at: <https://wiki.seeedstudio.com/Grove-Button/> [Accessed 15 March 2021].

[8] Michal Sieniawski, Teodora Dimitrova-Grekow, Adam Klimowicz, "Low-cost Monitoring System of a Sitting Posture", Bialystok University of Technology, Bialystok, Poland, Measurement Automation Monitoring no. 01 vol. 64 ISSN 2450-2855, 2018

[9] InfluxData. 2021. InfluxDB: Purpose-Built Open Source Time Series Database | InfluxData. [online] Available at: <https://www.influxdata.com> [Accessed 15 March 2021].