Generating a personalized diet and fitness routine using emotion, activity, and food tracking

D.S. Jiffry
Software Engineering Department
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka.
dsjiffry@gmail.com

Premarathna K.S.N.
Software Engineering Department
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
sathiranimhana@gmail.com

Yahathugoda S.R
Software Engineering Department
Sri Lanka Institute of Information Technology
Malabe, Sri Lanka
radishanyahathugoda@gmail.com

***Abstract*—The evolution of technology has reduced the need for physical activities, because of this we see that people feel the need to allocate time in their day for physical activities. To assist in this requirement, we propose an android app which will monitor a user's lifestyle and emotional wellbeing, and based on that data generate workouts and diet plans. Emotional wellbeing is monitored via the front camera, by performing analysis on the user's facial expressions. Lifestyle is tracked by performing activity tracking on the user throughout the day. Accellorometer readings from a smartwatch is used as the input to predict the user's activities. Then the app would identify common unhealthy practices and attempt to assist the user in avoiding them. Images of the user's food is used to track their calorie input, the app would identify the type of food and then using the calories of each food item, calculate user's daily calorie intake. All these features are combined in order to deliver a dynamic fitness app, which will assist the user in maintaining their health.**

Keywords—fitness app, emotion analysis, activity tracking, deep learning, diet plan, workout plan,

I. Introduction

Technology has made modern life less physical, the introduction of motorized transport and the internet means the physical activities a person needs to perform daily have reduced.

Studies show that lack of physical activity can even lead to a premature exit from employment [1]. Obese and overweight individuals are more likely to exit their paid employment via disability pension, the study recommends long-term interventions to encourage physical activities in the population [1]. Lack of physical activity can also be a primary cause for chronic diseases [2], The study says that the body can mal-adapt to insufficient physical activity and can result in a decrease in the total and the quality of years of life [2].

Due to these kinds of health concerns, we see a trend of people purposely allocating time in their schedule for physical fitness. We aim to assist in this regard by offering the user an app that can dynamically adjust to how they perform their physical activities, and how their emotional state varies. The app is able to learn the user's current lifestyle via sensors and also obtain their emotional state via the camera. The user will be given a diet and workout plan and their commitment to it will be monitored. Based on this data along with the emotional analysis and lifestyle tracking, our future recommendations will change. Furthermore, if the app identifies any habits that would adversely affect the user's health, then it would notify the user so that they may act. We hope that by using our app the user would be able to avoid the adverse health issues that could arise due to a lack of physical activity.

II. Methodology

A. *Obtaining the Dataset*

The app consists of three main sections, identifying the user's emotion, activities, and the food consumed. In order to do this, we needed three separate datasets, since we would be building three separate Models to identify each of these.

In order to obtain the user's activities, we decided to require them to wear a smartwatch. While it would be possible to track the activities from a smartphone, the accuracy would not be to our satisfaction. Looking at research done with smartphone activity tracking showed that their accuracy would max out at about 72% [3]. Whereas studies that had used both a smartphone and a smart-watch for activity tracking would boast a 19% better accuracy when using smart-watch sensors [4]. Therefore, we decided to use extraction from the WISDM

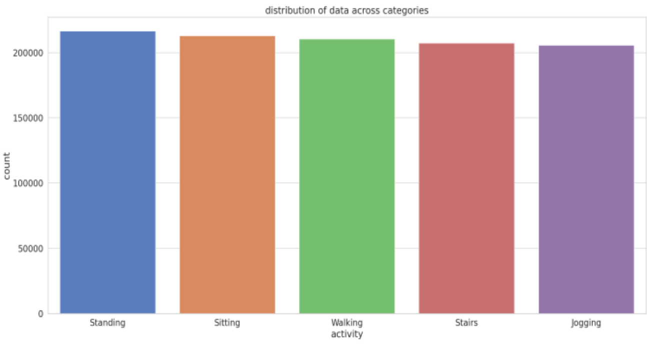

Fig 1. Distribution of smart-watch sensor data

dataset [5], which contains accelerometer readings along the x-axis, y-axis, and z-axis taken from a smartwatch. The data distribution is shown in Fig 1.

The dataset consisted of readings from 51 different subjects each of them had performed the 5 activities shown and sensor readings had been recorded. Readings were taken at a rate of 20Hz as each participant did each activity for 4 minutes apiece. This gives us many readings which would be beneficial when training the model. Furthermore, we will be considering only the smart-watch accelerometer readings. Since that sensor is what we decided to use, considering that power consumption of the sensor must also be given attention. Running a gyroscope would consume more power than an accelerometer, hence we decided to only use the accelerometer to predict the user's activity.

To capture the user's emotion, we use the smartphone's camera, which is the most suitable way to track the emotion of users. A research done in emotion extraction shows the accuracy of their project as 91% [9]. In this research they used a picture to extract the emotion [9]. The research used 8 emotions, Angry, Contempt, Disgust, Fear, Happy, Sad, Surprise, and Natural [9]. Research regarding the Development of Deep Learning-based Facial Expression Recognition System shows that, 7 emotions are more accurate than 8 emotions [10]. The 7 emotions were obtained by dropping the "contempt" emotion. The emotion dataset distribution is shown in Fig 2.

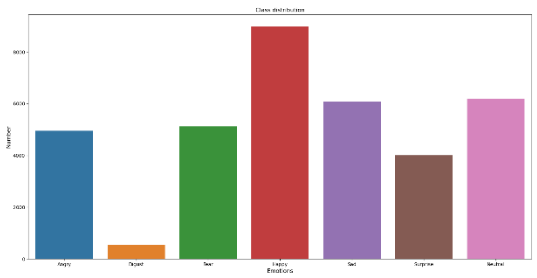

Fig 2. Class distribution of 7 emotions

The dataset includes 7 classes of emotions. Which are as follows,

1. Angry
2. Disgust
3. Fear
4. Happy
5. Sad
6. Surprise
7. Neutral

The current data set has enough data to improve the accuracy when extracting users' emotions. The smartphone camera will take the input as a picture of the user's face. The model will analyse the input to make a prediction.

In the process of recommending the food diet, we are getting data regarding user's Age, Gender, Height, and Weight. From this data, we can calculate the Body Mass Index (BMI) of the user and categorize them into three main body types as

1. underweight
2. healthy
3. overweight

depending on the category, the system will recommend a food diet. To give the user the freedom of consuming food not in the diet, we made a function where the user can capture a picture of the food item, using the camera of their smartphone. Using the image, we can calculate the calorie amount in the food item and consider it when generating the diet, to make the app more dynamic.

we took images of 11 food categories which are popular in Sri Lanka,

1. dairy products
2. donuts
3. eggs
4. hoppers
5. Koththu
6. Meat
7. Noodles
8. Pittu
9. Rice
10. Soup
11. String Hoppers

We are able to predict the food calorie count using the image.

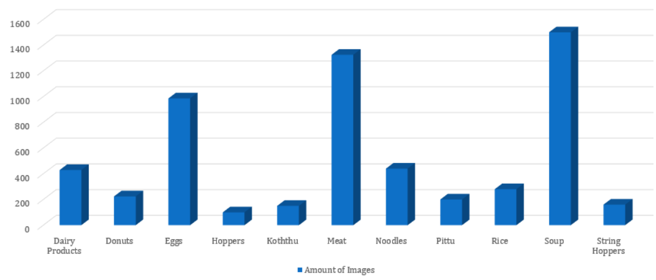

Fig 3. Distribution of food category data

We created 3 main food diet plans for each of the 3 categories. We also created additional diet plans which will be used if the user deviates from the given plan.

B. Building the Model

The creation of deep learning models is a resource-intensive task, therefore we decided to make use of Google Colab. Which is a service by Google which offers free GPU access for the creation of Machine Learning models.

To build a model for the emotion tracking section, the dataset data was given 7 labels. Architecture used for this model was Convolutional Neural Network. When Building the layers for the model, we went for the sequential Keras model. The neural network consist of four convolutional neural layers and two fully connected layers.

1. The first layer has a 3X3 filter and learns 64 filters.
2. The second layer has a 5X5 filter and learns 128 filters.
3. The third layer has a 3X3 filter and learns 512 filters.
4. The fourth layer has a 3X3 filter and learns 512 filters.

The batch normalization technique was used for increasing the rate of training in the neural network. Max Pooling was put to decrease the spatial dimensions in output. In the network, the input shape is 48x48. Dropout was used for network generalization and to not cause overfitting of training data. The Softmax classifier is used to get the output from the layer. The Relu function was used as an activation function.

1. The first fully connected layer has 256 neurons.
2. The second fully connected layer has 512 neurons.

To the fully connected layers, we added batch normalization, dropout, Relu, and softmax functions. For hyperparameters, the learning rate was set to 0.01. for regularizations, 1e-7 was set. The neural network consist of batch size of 128 and 40 epoches. This neural network provided better accuracy than previous researches. The current model has 90% of accuracy. The loss and confusion matrix is given below.

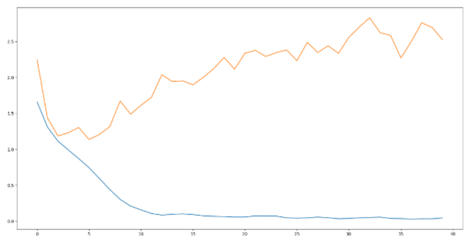

Fig 4. Emotion tracking model loss graph

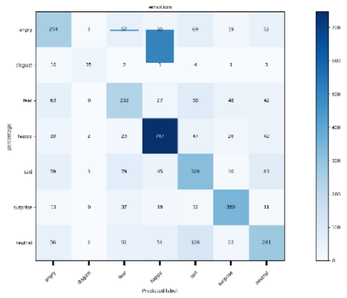

Fig 5. Emotion tracking model confusion matrix

When building the model for the activity tracking section, initially the dataset was loaded and then split into training and validation subsets. The splitting was done via the user id column, this column values range from 1600 to 1650, so we took the readings with user id less than 1641 for training and the others for validation. When constructing the layers of our model, we went with a sequential Keras model for simplicity, then a Long Short-Term Memory (LSTM) layer was used. This layer was selected since it is suitable for making predictions based on time series data, and helps overcome the vanishing gradient issue that can occur in traditional Recurrent Neural Networks (RNN). The next layer was a dropout layer, intended to reduce overfitting to the training data. A dense fully connected layer was used to interpret features of the LSTM layer and then an output layer would give us the predictions.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_2 (LSTM)                (None, 128)               67584
_________________________________________________________________
dropout_2 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_4 (Dense)              (None, 128)               16512
_________________________________________________________________
dense_5 (Dense)              (None, 5)                 645
=================================================================
Total params: 84,741
Trainable params: 84,741
Non-trainable params: 0
```

Fig 6. Layers in the activity tracking model

The model was then trained, and we observed an accuracy of about 88%. The loss graph and confusion matrix are given in Fig 7 and Fig 8.

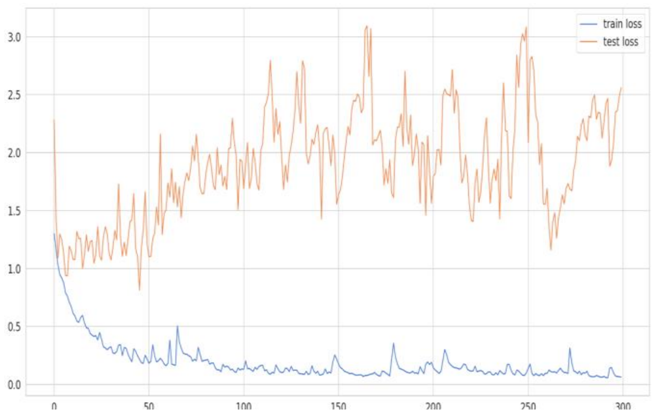

Fig 7. Activity tracking model loss graph

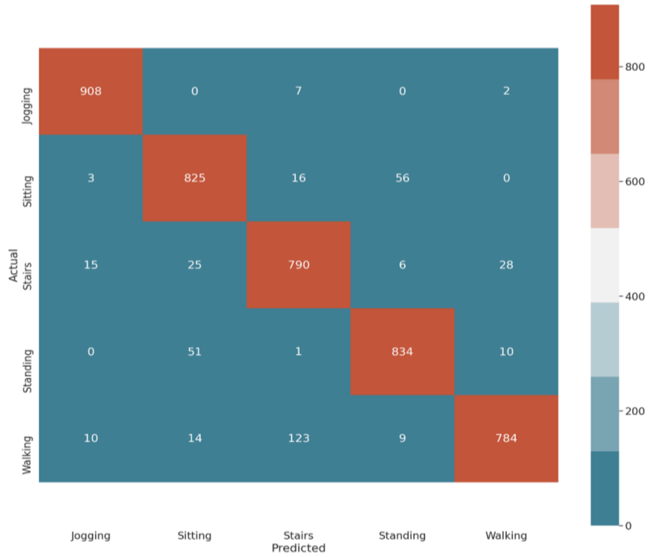

Fig 8. Activity tracking model confusion matrix

When developing the Food Recommendation function, we divide it into 3 main parts.

1. Identify the food from a picture taken by the user's camera and predict the average calorie amount.
2. Analyze and identify the user’s body type and recommend a food diet.
3. Analyze and identify the remaining calorie amount in the past 4 days and adapt the next diet.

For the first part, we use Convolutional Neural Network (CNN)

Layer	Parameters
Convolution 2D	3 x 3 @ 64
ReLu	
Pooling	2 x 2
Convolution 2D	3 x 3 @ 32
ReLu	
Pooling	2 x 2
Dense	64
Dense	2

Fig 9. CNN Architecture of the model

Then, we used FFNN

Type	Value
Loss	0.0018
Validation Loss	0.0010
Accuracy	0.99
Validation Accuracy	0.99

Fig 10. FFNN Architecture of the model

Next we used this LSTM architecture for each of the 3 body types. For the ‘Over weight’ category the LSTM Architecture is as follows.

Type	Value
Loss	0.0018
Validation Loss	0.0022
Accuracy	0.99
Validation Accuracy	0.99

Fig 11. “Over weight” LSTM Architecture

for the ‘under weight’ category the LSTM Architecture is as follows.

Type	Value
Loss	0.0016
Validation Loss	0.0010
Accuracy	0.99
Validation Accuracy	0.99

Fig 12. “Under weight” LSTM Architecture

for the ‘regular weight’ category the LSTM Architecture is as follows.

Type	Value
Loss	0.0061
Validation Loss	0.0140
Accuracy	0.99
Validation Accuracy	0.99

Fig 13. “Regular weight” LSTM Architecture

C. Building the App

When it came to building our app, we decided to have a Django server as our backend. The server would contain each of our exported models and when we send the inputs required via POST request, the server would make the prediction and respond with the result. This would help us conserve battery life on the user’s phone since the task of making predictions would require the phone to do more work and be constantly awake. Fig 14 shows a diagram of how our app would work.

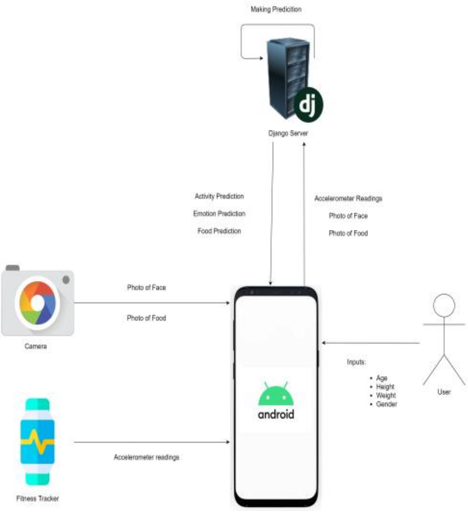

Fig 14. Architecture of the app

For the app to function we need the user’s age, height, weight, and gender, so upon the first launch of the app, they would be asked to fill in these details.

In the lifestyle tracking section, the watch would supply accelerometer readings which the phone would then send to the server and obtain an activity prediction. These activities would then be stored on the user’s phone on a local database. For privacy reasons it was decided that the server would not store any kind of log of the user’s activity. The Lifestyle tracking section would run in the background for one week initially. During this time, it would attempt to identify what the user’s current lifestyle pattern is. Using a combination of the activity history and the other smartphone sensor readings, the app would identify the user’s:

- Wake-up and sleep time
- Home and work location
- Hours spent at the workplace
- Exercise Time and type
- How they get to work

After the one-week analysis period is complete the app would start analyzing this data to see where the user can improve their lives so that they may be more healthy.

- By analyzing the wake-up and sleep time we can find out how many hours of sleep the user gets. Ideally, an adult should get at least 7 hours of sleep [6].
- Knowing the work hours and location is beneficial for the workout and diet sections of the app, since we can avoid disturbing the user when they are at work.
- Sitting continuously for long periods of time has been shown to have adverse effects on a person’s health [7].
- Meditation is shown to help with managing stress and anxiety [8].
- To track the user’s progress the app would be requesting a weekly update on the user’s weight

When considering the recommending food diet, Using the obtained details we calculate the Body Mass Index of the user. After that by using the BMI value we categorized the user as either underweight, healthy or overweight. Then, depending on the category, the app would recommend a diet plan every week. In that diet plan, each and every day has a specific calorie intake amount. That amount differs according to the body category. But we do not know whether the user is properly following the diet that we recommend. Therefore, the user can take a picture of food using the camera. Using that image we can identify the food item, and identify the average calorie amount of that item. Then we consider that value as a Calorie Intake. At the end of the day, the system will calculate the total calorie intake amount of that day.

Then by considering the calorie burn, which is obtained from the workout function, is utilized by this function.

- Remaining Calories = Calorie Intake – Calorie Burn

Using above formula, the system calculates the remaining calorie amount of the user at the end of the day. Afterward, the system collects and stores the remaining calorie amount of the past 4 days and analyzes the pattern. If the user belongs to ‘overweight’ category, they needs reduce the weight. But if they are ‘underweight’ they need to increase calorie intake. the system will detect the pattern and automatically change the diet plan.

The Emotion tracking sector includes two main functions,

1. Facial expression analysis function
2. Mobile game function

The facial expression analysis function is done on the backend server and android game is implemented in the app. To play the game, the game function checks whether the user has completed the daily task. After validating that parameter, the game function allows access to the game. At the start of the game, difficulty is set to beginner level. The gameplay is limited to 25 minutes because the daily average time spent on mobile games is 25 minutes and 30 minutes [11].

When the game is started the system will take a picture of the user by using the Phone's camera. After that, taken pictures are send to the facial expression analysis function that runs on the backend server. The Function analyzes the picture, and gives a prediction. The result is the user's emotion. Result is sent back to the app. According to the result, the mobile game's difficulty is changed from beginner to hard, challenging, or epic. The result is also used to display quotes during gameplay. Quotes are choosen according to the user's current emotion. The purpose of these display quotes is to assist the user’s mood [12]. This process runs until the end of the gameplay.

Previous research has used video frames for emotion extraction [13]. In the previous research, the emotion extraction process required more resources and took a long time to train. Research shows that by using two methods [14], users can play games without disturbing player experience. According to Research [15], if the game assists the player, the player willingly more interacts with the game. It helps to establish a stable player mindset. Most computer games assist the player by giving hints. In this component, the mobile game will display quotes according to the emotion.

III. Results and Discussion

When considering emotion tracking, we can track the emotions of the user while they are playing the game. When the user is playing the game, the app detects their automatically changes the difficulty according to the emotion. The system then pops up motivational quotes that help to set the user’s mind at ease. All 7 emotions can be recognized within a short time because the current model is based on the picture train model. Previous researches mostly used 1 to 6 convolutional layers, but our model used 4 convolutional layers and 2 fully connected layers. As for the result, we gained better accuracy overall. Furthermore, overfitting was reduced by adding dropout and batch normalization. The model accuracy is 90%. This neural network doesn’t need a huge amount of resources, because our model runs on Django server, making it efficient for android devices.

When considering the food diet recommendation, we are able to detect and analyze the body type of the user and recommend a food diet weekly according to their detected body type. The user can take a picture of the food, using the camera. The system will predict the average calorie amount of the food in the picture. And will calculate the total calorie intake at the end of each day. Then the system detect, analyze and predict the pattern of calories remaining, in past 4 days and change the upcoming food diet accordingly.

In the lifestyle tracking section, the watch would supply accelerometer readings which the phone would then send to the server and obtain an activity prediction. These activities would then be stored on the user’s phone on a local database. For privacy reasons it was decided that the server would not store any kind of log of the user’s activity. The Lifestyle tracking section would run in the background for one week initially. During this time, it would attempt to identify what the user’s current lifestyle pattern is. Using a combination of the activity history and the other smartphone sensor readings the app would identify the user’s

- Wake-up and sleep time
- Home and work location
- Hours spent at the workplace
- Exercise Time and type
- How they get to work

After the one-week analysis period is complete the app would start analyzing this data to see where the user can improve their lives so that they may be more healthy.

- By analyzing the wake-up and sleep time we can find out how many hours of sleep the user gets. Ideally, an adult should get at least 7 hours of sleep [6], so by looking at the user’s wake up time, our app would notify the user about 7 hours in advance that if they go to sleep shortly they will be able to get the required hours of sleep, while still waking up at their usual time.
- Knowing the work hours and location is beneficial for the workout and diet sections of the app since we can avoid disturbing the user when they are at work.
- Sitting continuously for long periods of time has been shown to have adverse effects on a person’s health [7]. If the app detects the user sitting continuously for one hour, it would send a notification suggesting that they stand up and move about for a bit.
- Meditation is shown to help with managing stress and anxiety [8]. The app would consider suggesting the user try out meditating if it detects they are at home and are not engaging in any taxing activities.
- To track the user’s, progress the app would be requesting a weekly update on the user’s weight, which would be used to plot a Body Mass Index (BMI) history. This would be considered when updating the user’s diet and/or workout plan.

In this way, the app would assist the user in maintaining their health.

IV. Conclusion

Health and fitness are a major concern in modern society, with technological advancements making life easier. So People use apps to track their daily fitness and there are no existing apps that can be used to make decision making learn considering the user’s lifestyle and expressions as the main input factors. Also, there is not an app that combines daily tasks, routines, nutrition patterns, and user expression monitoring. For a particular person to examine all these tasks mentioned, they need to use multiple apps. Thus, then all these functionalities cannot be obtained.

Due to the static nature of such apps, people tend to move away from such fitness apps and ultimately impact a healthy life. So, in the end, we have developed an app that takes the user’s habits and expressions as inputs and leads to a better decision-making platform in combining daily tasks routines, nutrition patterns, and emotion prediction. Furthermore, considering the inputs, the system adapts to match the user’s different lifestyles, dynamically changing food diet and analyze and detect user’s moods. So, in this way, the user will be motivated to keep using the app and thereby leading to a healthy life.

Figures and Tables

- Fig 7. Activity tracking model loss graph
- Fig 8. Activity tracking model confusion matrix
- Fig 9. CNN Architecture of the model
- Fig 10. FFNN Architecture of the model
- Fig 11. “Over weight” LSTM Architecture
- Fig 12. “Under weight” LSTM Architecture
- Fig 13. “Regular weight” LSTM Architecture
- Fig 14. Architecture of the app

ACKNOWLEDGMENT

We wish to show our appreciation to Mr. Prasan Yapa (Lecturer at SLIIT) for acting as the supervisor of this project and for his valuable insights.

We thank Dr. Evone Jayaweera (Senior Registrar in Clinical Nutrition at Teaching Hospital Peradeniya, Sri Lanka.) for her knowledge and expertise in the nutrition section of our project.

REFERENCES

[1] S. J. Robroek, K. G. Reeuwijk, F. C. Hillier, C. L. Bambra, R. M. V. Rijn, and A. Burdorf, “The contribution of overweight, obesity, and lack of physical activity to exit from paid employment: a meta-analysis,” Scandinavian Journal of Work, Environment & Health, vol. 39, no. 3, pp. 233–240, 2013.

[2] F. W. Booth, C. K. Roberts, and M. J. Laye, “Lack of Exercise Is a Major Cause of Chronic Diseases,” Comprehensive Physiology, 2012.

[3] X. Sheng, J. Tang, J. Wang, T. Li, G. Xue, and D. Yang, “LIPS: Lifestyle Learning via Mobile Phone Sensing,” 2016 IEEE Global Communications Conference (GLOBECOM), 2016.

[4] G. M. Weiss, J. L. Timko, C. M. Gallagher, K. Yoneda, and A. J. Schreiber, “Smartwatch-based activity recognition: A machine learning approach,” 2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), 2016.

[5] G. M. Weiss, K. Yoneda, and T. Hayajneh, “Smartphone and Smartwatch-Based Biometrics Using Activities of Daily Living,” IEEE Access, vol. 7, pp. 133190–133202, 2019.

[6] H. H. Publishing, “Repaying your sleep debt,” Harvard Health. [Online]. Available: https://www.health.harvard.edu/fhg/updates/Repaying-your-sleep-debt.shtml. [Accessed: 05-Mar-2021].

[7] H. H. Publishing, “Why you should move - even just a little - throughout the day,” Harvard Health. [Online]. Available: https://www.health.harvard.edu/heart-health/why-you-should-move-even-just-a-little-throughout-the-day. [Accessed: 05-Mar-2021].

[8] H. H. Publishing, “How meditation helps with depression,” Harvard Health. [Online]. Available: https://www.health.harvard.edu/mind-and-mood/how-meditation-helps-with-depression. [Accessed: 05-Mar-2021].

[9] Liliana, D Y (2019). Emotion recognition from facial expression using deep convolutional neural network. Journal of Physics: Conference Series, 1193(), 012004–. doi:10.1088/1742-6596/1193/1/012004

[10] Jung, Heechul; Lee, Sihaeng; Park, Sunjeong; Kim, Byungju; Kim, Junmo; Lee, Injae; Ahn, Chunghyun (2015). [IEEE 2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV) - Mokpo, South Korea (2015.1.28-2015.1.30)] 2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV) - Development of deep learning-based facial expression recognition system. , (), 1–4. doi:10.1109/FCV.2015.7103729

[11] C. Gough, "Time spent with mobile games worldwide 2016 | Statista", Statista, 2020. [Online]. Available: https://www.statista.com/statistics/667679/time-spent-mobile-games-world/.

[12] *Eprints.lancs.ac.uk*, 2020. [Online]. Available: https://eprints.lancs.ac.uk/id/eprint/12587/1/Gilleade_Affective_Gaming_DIGRA_2005.pdf.

[13] Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," ISCAS 2010 - 2010 IEEE Int. Symp. Circuits Syst. Nano-Bio Circuit Fabr. Syst., pp. 253–256, 2010

[14] J. Moniaga, A. Chowanda, A. Prima, Oscar and M. Tri Rizqi, "Facial Expression Recognition as Dynamic Game Balancing System", 2020.
.

[15] *Eprints.lancs.ac.uk*, 2020. [Online]. Available: https://eprints.lancs.ac.uk/id/eprint/12587/1/Gilleade_Affective_Gaming_DIGRA_2005.pdf. [Accessed: 18- Jul- 2020].