International Journal of Engineering Trends and Technology
ISSN: 2231–5381 / https://doi.org/10.14445/22315381/IJETT-V71I5P234
Volume 71 Issue 5, 328-335, May 2023
© 2023 Seventh Sense Research Group®

Original Article

Machine Learning Trained Drowsiness Detection using Gyroscope on a Microcontroller

Bagas Aditya Putra[1], Valerius Owen[2], Antoni Wibowo[3]

[1,2,3]*Department of Computer Science, Master of Computer Science, BINUS University, Jakarta, Indonesia.*

[1]*Corresponding Author : bagas.putra001@binus.ac.id*

Received: 23 February 2023 Revised: 02 April 2023 Accepted: 13 April 2023 Published: 25 May 2023

Abstract *- In more recent years, drowsy driving has resulted in 328,000 incidents annually. Drowsiness itself is usually a sign of fatigue, and drivers who experience this may also experience temporary episodes of microsleep that can be fatal while driving. Realizing that some car manufacturers have implemented this, it has only been applied to higher-end cars. From this, a drowsiness detection device needs to be created that is portable and can be used by anyone of any background. A system is designed using a microcontroller and implemented as an Artificial Neural Network with a small size and high accuracy, ported into a header file using Tensorflow Lite. With this, the device is able to detect the drowsiness of the user using the gyroscope from the head movement and uses an LED as well as a buzzer to alert the user or others of drowsiness for real-time predictions.*

Keywords *- Artificial neural network, Drowsiness, Gyroscope, Machine learning, Microcontroller.*

1. Introduction

A microsleep is a brief period of sleep or tiredness that can last a few seconds or up to several seconds, during which a person loses consciousness and is unable to react to arbitrary sensory input. Drivers and motorists may not feel if they are feeling drowsy because it is hard to recognize the signs of exhaustion. Microsleep, which refers to brief, uncontrollable bouts of inattention, can also occur in some persons. The car will travel the length of a football field at highway speed in the 4 to 5 seconds a driver spends in microsleep.

Driving while drowsy is similar to driving under the influence of alcohol:

- The more sleepy a driver is, the worse their reaction times, awareness of hazards, and capacity for sustained concentration are.
- Driving after lasting more than 20 hours without sleep equates to having a blood alcohol content of 0.08%, the legal limit in the United States.
- If you are tired, your risk of being in a car accident increases by three times.

According to estimates from the US Department of Transportation and National Highway Traffic Safety Administration, around 100,000 police-reported crashes involving drowsy driving in 2017 caused close to 800 fatalities and close to 50,000 injuries. The true number, however, might be substantially higher because it can be challenging to tell whether a driver was intoxicated during a collision.

According to a survey by the AAA Foundation for Traffic Safety, 328,000 accidents involving fatigued driving happen annually. That is more than three times the quantity the cops reported. According to the same data, 6400 of these drowsy driving accidents ended in fatalities, while 109,000 others caused injuries. The researchers contend that the number of fatal crashes caused by tired driving is more than 350% more than previously thought [27].

The data provided above shows that a system to detect drowsiness would be useful not only for the driver but also to help authorities to have an immediate response in case of an accident.

Previously there have been attempts to create a system to detect drowsiness, such as the Mercedes-Benz Attention Assist by using sensors and an algorithm to analyze driving behavior by using over 70 parameters [7]. However, this is only implementable on Mercedes-Benz manufactured automobiles, so a system would still need to be made that can be portable enough and versatile for any user, any make, and any type of vehicle. This research aims to get a reading of drowsiness with small, cheap, but accurate enough to identify driver drowsiness. This research decided to put the device and the sensors as wearables to then further research could implement better design and results.

The research that has been done most research uses heart rate and cameras to detect a driver's drowsiness levels. These devices' limitations are the sensors' placement and whether they obstruct the driver's field of view. With the MPU6050 sensor, the sensor can be placed around the head of the driver just like a reading glass or embedded within the driver's helmet.

This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

2. Literature Review

2.1. Discussion

2.1.1. Microsleep

Generally speaking, microsleep can be described as a brief loss of consciousness or a sleep-like state. Microsleep can be separated into phases: an episode of drowsiness (ED), a microsleep episode candidate (MSEc), a unilateral microsleep episode (MSEu), and a bilateral microsleep episode (MSE).

Bilateral microsleep episodes (MSE) are characterized by irregular and polymorphic theta brain signals in the setting of sluggish background activity or low-voltage EEG, which differ from missing alpha activity, the simultaneous decline in beta activity, and eyelids that are more than 80% closed. For 1 to 14 seconds, bilateral microsleep episodes can last. Minor symptoms include sluggish eye movement and head drooping due to a lack of muscle tone. While drowsiness episodes (ED) can last anywhere from one to thirty seconds, they are characterized by an EEG that alternates between regular and erratic activity patterns.

These brief loss of consciousness that may happen from time to time would be fatal during the operation of heavy machinery or at high speeds. Such as, during the operation of industrial equipment or motorized vehicles at highway speeds, it would be fatal if the driver experiences drowsiness. Although there are other methods of preventing microsleep, it would still need interaction from other parties or to have strict habits which may not always be fulfilled. Due to this, a new method of microsleep prevention is needed, which may be performed using a microcontroller.

2.1.2. Microcontroller

Esp32 was chosen because of its low cost, low power, could handle multiple GPIOs, and built-in WiFi and Bluetooth. FreeRTOS is already embedded in ESP32. This enables us to use the functions from the FreeRTOS libraries.

The ESP32 processor, which was created for mobile, wearable, and Internet of Things applications, uses a variety of proprietary software to achieve incredibly low power consumption. The ESP32 also has contemporary technologies like dynamic power scaling, several power modes, and fine-grained clock gating.

With this, the microcontroller of choice, the ESP32, is able to be programmed to read the movements of the user and also perform simple tasks to detect the state of the user. The programming of the ESP32 could also be done using the Arduino IDE using a C++ programming language, making it relatively easy to set up. In addition, a buzzer and LED are attached to create an alert for the user. However, the ESP32 itself does not have a gyroscope to detect the head tilt of the user, which means that it would need an extra module to be attached to detect the head tilt.

2.1.3. Gyroscope

Gyros are devices that measure or maintain rotational motion. Degrees per second or rotations per second are used to quantify angular velocity. Simply said, angular velocity is a measurement of rotational speed.

An x, y, and z rotation may be measured by a triple-axis MEMS gyroscope. Single and two-axis gyros are available in some cases, but the triple-axis gyro in a single chip is getting smaller, more affordable, and more common.

Gyros are frequently applied to items that are just slightly spinning on each axis, such as an. Gyros aid in stabilizing the aircraft's flight by spotting these minute shifts. Also, take notice that the measurement of the gyro is unaffected by the aircraft's acceleration or linear velocity. Gyros solely track angular speed.

The MEMS has a small gyroscope sensor (between 1 to 100 micrometers). Variations in angular velocity cause a tiny resonant mass to move when the gyro rotates. A host microcontroller may amplify and decode the movement's converted ultra-low-current electrical impulses. The gyroscope being used that is compatible with the ESP32 is an MPU6050 module. It is simple enough to use and compact. In addition, this module has a 3-axis gyroscope and a 3-axis accelerometer which is suitable for the use case of this proposed method.

2.1.4. Accelerometer

Acceleration, or the rate at which an object's velocity changes, is what will be measured by an accelerometer. They measure in G-forces or m/s2, or meters per second (g). On Earth, one G-force is comparable to 9.8 m/s2, but height affects this slightly (and will be a different value on different planets due to variations in gravitational pull). Applications requiring orientation or the detection of system vibrations benefit from the use of accelerometers.

Capacitive plates are typically found inside accelerometers. While some of them are affixed to tiny springs that move within the sensor as acceleration forces are given to them, others are fixed in place. As these plates move in relation to one another, the capacitance between them changes. These fluctuations in capacitance can be used to compute the acceleration.

2.1.5. Early Developments of Drowsiness Detection

A drowsiness detection system has been developed since 1994; since the publication of a study report in 1994 by NISSAN researchers in Japan, drowsiness detection technology has been created. Techniques used by [9] include physiological signals and reactions, which are two factors that they sense in human physiological phenomena. Signals include variations in brain waves, blinking, heart rate, pulse rate, and skin electric potential. Physical responses were based on changes in the driver's head inclination, slumping posture, eye blink frequency, and grip force on the driving wheel. These two detections are both quite accurate. However, according to their research [9], physiological signals are less useful than physical reaction detection. Even though both are equally extensible, physical reactions nonetheless outperform physiological signals.

2.1.6. Machine Learning

Machine learning is a program that predicts outcomes based on learning data and statistics. Machine learning could be implemented in devices such as computers, laptops, and even as small as a microcontroller. Although constrained by small processing power, various machine learning models could be run in real-time. Based on [18], it demonstrates how there are numerous machine learning algorithms, including gradient descent, linear regression, logistic regression, decision trees, support vector machines, bayesian learning algorithms like naive bayes, K nearest neighbor, K means clustering, and backpropagation, each with their own benefits and drawbacks depending on the classification task at hand.

With the use of machine learning, it would be able to further increase the accuracy of the drowsiness detection by using the data gathered on the gyroscope MPU6050 module attached to the ESP32 microcontroller. However, there is a drawback to using machine learning on a low-powered processor since machine learning typically requires a powerful processor, especially with larger datasets or the use of many variables. Aside from being heavy to train, this may also result in a long prediction time which may reduce the performance and the overall aim to detect drowsiness in real-time. This would mean that the proposed method would require lightweight machine learning for low-powered processors.

2.1.7. TinyML

A recent advancement is machine learning for the low-end Internet of Things. This means that microcontrollers can now perform lightweight machine learning where they usually need high computing power and memory, unlike microcontrollers with small resources. [19] TinyML is possible to be based on Tensorflow Lite (TFL), which is an open-sourced deep learning framework that can be deployed on low-power edge devices such as a microcontroller. [20]

With the use of TinyML, it is possible to train a simple machine-learning algorithm and extract the model to be inserted into the code of a microcontroller using the Arduino IDE. From this, the microcontroller would not need to train the algorithm from the built-in processor, but it is possible to train on stronger, more capable computers. The model can then be extracted and inserted into the microcontroller for prediction.

2.2. State-of-the-Art Methods

There has been previous research on drowsiness using wearable sensors, such as using sensors that read electroencephalography electrodes, accelerometers, and gyroscope data. The data obtained was then used in feature extraction, feature fusion and selection, and classification. It achieved an accuracy of 86.5%; however, if feature selection was used, it could be optimized to 92%. [10].

This research [1] made a multimodal device that runs in real-time and outputs data to a computer. The platform itself is called Elapse. The wearable device records EEG, ocular video, and head movement and sends the information in real time to a distant computer. As part of the development of the related software, an expandable framework for real-time, multimodal data processing and categorization was also demonstrated.

In [2] discusses the current state of driver tiredness detecting technology. A review of the scientific literature and commercially available tools for measuring driver alertness and sleepiness includes a usability, intrusiveness, and detection accuracy evaluation. According to research, video-based technology is more user-acceptable and simple to use than other technologies.

Modern techniques were compared with two well-known procedures, ElSe and ExCuSe. The best method is the one using the geodesic distance and the suggested detector. The suggested technique [3] has a negligible inaccuracy of just 5.5 pixels on the BioID dataset in particular. In contrast to the other assessed approaches, the suggested solution is made to work in pictures with eyebrows. Additionally, their solution demonstrated a lower degree of error (6.2 pixels) for the given dataset when compared to state-of-the-art techniques.

In this [4] article, they proposed a portable, all-encompassing device that could automatically detect a driver's levels of fatigue and distraction and alert them to their unsafe condition via equipment alerts. This technology can be put in any car without the requirement for additional hardware. A hardware component and a software component make up the SOMN_IA. The system is optimized to meet all requirements using all methods, including face identification, the CNN-based classifier, and correction using subsequent classification results. Five stages make up the software component, including frame separation and region detection, using a shallow convolutional neural network (S-CNN), and correcting prediction errors based on actual driving conditions. The sounding of pertinent alarms can tell distracted driving from the driver's regular behavior and tiredness from normal blinking.

The following are the contributions of this [5] research work: An ideal location for a video camera is on the far left of the dashboard without zoom, according to extensive analysis after testing three camera locations with and without zoom with two people. They suggest techniques for accurately detecting faces and eyes, even when the driver is not looking directly at the camera or has their eyes closed.

The aim of this [6] research is to deploy a helmet prototype for construction workers. EEG, acceleration, and gyroscope were used to get the data that needed to be forecast. The state of the employees will then be determined using a Random-Forest Classifier. First, a fifth-order Butterworth bandpass filter was applied to the data. The data is filtered to retain the frequency range between 3 Hz and 25 Hz because the study in the next sections mostly concentrates on the theta and alpha frequency bands. The features are placed into a machine learning model to determine the class to which the data belongs. The random forest classifier is employed in this study. The accuracy of the internal, real experiments was 98%.

This [10] research involves detection not only while driving but also doing activities. Accuracy rates of 98% were achieved in the real internal experiments. A modest acceleration kurtosis and standard deviation point to a little departure from the normalized mean distribution. This is because the movements were abrupt. After all, these tasks were carried out quickly.

The research conducted [28] shows that using a brain-machine interface (BMI) with an SVM classifier could detect the user's state with 93.67% accuracy. The brain-machine interface would include several data taken from the user: EEG readings, transcranial direct current stimulation (tDCS), and a gyroscope paired with a watch for operation. The proposed method acts locally or in a closed loop. However, the device itself is bulky. In addition to that, since they used a transcranial direct current stimulation, this may not be safe if there was an error on the device. Furthermore, the paper also stated that with an accurate prediction of the user's state, the use of the tDCS was not able to improve the driver's alertness, but only to warn them to take a break.

In [22], a proposed method was to use a multimodal approach for fatigue drivers by using an EEG, gyroscope, and computer vision to detect if a driver is fatigued. It achieved an accuracy of 93.91% by using a deep neural network consisting of a recurrent neural network with 400 LSTM nodes, whose result is then passed onto a neural network with 3 hidden layers trained using an adam optimizer. Furthermore, the neural network classifies the driver's fatigue state by using a sigmoid function for classifying 6 classes.

In [23], the researchers used an ANN model to detect and predict the driver's state where its inputs were using heart rate, respiration rate, head and eyelid movements, as well as driving behaviour. The proposed method of using the ANN model achieved a mean square error of 0.22 and prediction with a mean square error of 4.18 min. However, this research was conducted using a car simulator where its condition on the road has been created to be monotonous, and the model was able to detect the driver's drowsiness. This means that it may not perform accurately in real-life scenarios where there is more dynamic driving behaviour.

To create a passive brain-computer interface, this [24] effort analyzed the effectiveness of sleepiness detection using functional near-infrared spectroscopy and deep learning algorithms. A deep neural network was used to assess driver weariness using four windows (0 to 1s, 0 to 3s, 0 to 5s, and 0 to 10s). This research also discovered a new region comprising the channels with the highest classification accuracy and thirteen distinct channels most active when fatigued utilizing the convolutional neural network on the functional brain maps, which yielded an accuracy of 99.3%.

An effective hybrid technique is put out in this paper [25] to diagnose tiredness based on EEG. The suggested approach employs three feature extraction methods to characterize the EEG signals that indicate fatigue precisely. Thus, parameters such as energy and zero-crossing distributions, spectral entropy, and instantaneous frequency are extracted during the first feature extraction process. Deep features are extracted from trained AlexNet and VGG16 models using the second feature extraction approach. During the third feature extraction process, the statistical properties of the instantaneous frequency of the TQWT decomposed EEG signals are retrieved. Each LSTM network's output is merged based on a majority vote. According to data, individual categorization outcomes varied from 86.46% to 88.47%. The accuracy score after a majority vote was 94.31%.

3. Data and Methods

Data gathered from ESP32 attached with MPU 6050 that has been chosen are a gyroscope and accelerator, which can be obtained by using the quaternion to convert into yaw, pitch, and roll or x, y, and z. Using a gyroscope and accelerometer data, it can detect the yaw, pitch, and roll from the user's head movement. But a couple of seconds after ESP32 is powered on, the MPU 6050 has to calibrate its offset position to improve the accuracy of the head position.

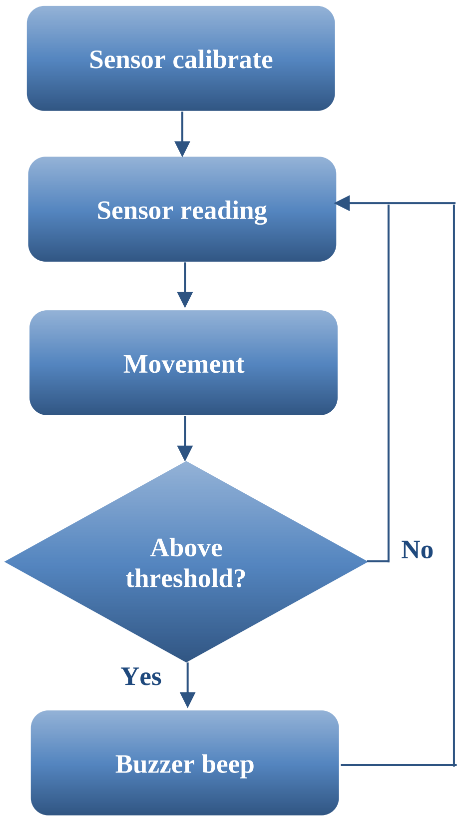

Fig. 1 Detection system flowchart

At first, the sensor will calibrate itself for a couple of seconds. This will ensure the offset is correct. The sensor will then read real-time data from head movement, whether it is above the threshold that has been set or not. If the movement is above the threshold, a buzzer and LED will activate and produce both visual and audio warnings. If the movement is below the threshold, then the sensor will continue reading.

The state machine diagram that can be seen in Figure 1. represents ESP32 activity. At first, ESP32 and MPU 6050 will calibrate the offset.

Machine learning can be trained using sample data gathered from previous readings. Parameters considered are yaw, pitch, roll, and target (shown as binary, 1 for DROWSY and 0 for AWAKE). Data is then pre-processed by using a MinMaxScaler to normalize the data first before feeding it into the classifier. There are 6 different classifiers that will be implemented to compare which are superior for this research scenario: Artificial Neural Networks, Support Vector Machine, K Nearest Neighbors, Multinomial Naive Bayes, and the Random Forest Tree. There are other classifiers to be used, such as Logistic Regression.

Classification is used because of the nature of the data collected and the target. Machine learning is trained with sequential/time series data of gyroscopes which then are classified by machine learning if the values are classified as *awake* or *drowsy*, called Binary Classification. But the testing found that Multi-Class Classification is more fitted to the use case. The Multinoulli distribution [12], a discrete probability distribution, covers an event with a categorical result, such as K in 1, 2, 3,..., K. This suggests that the model predicts the probability that a given an example will fall under a particular class label in terms of classification.

Naive Bayes is a parametric algorithm, which means that to streamline the machine's learning process, it needs a predetermined set of parameters or assumptions. KNN is an algorithm for supervised machine learning that can be applied to both classification and regression issues. It memorizes the training data rather than learning a discriminative function from it.

The model that is extracted from TensorFlow has then converted the model to a TensorFlow Lite format without quantization which includes a few functions such as starting the model, predicting, and also to get the score of the model, which also has the weights and data attributes to perform TinyML on an ESP32 microcontroller. The result was that the model could predict the user's state by using a machine learning algorithm only using a microcontroller running locally with a gyroscope and extracting the tilt of the head.

4. Result and Discussion

From the experiments that have been performed, the MPU6050 can correctly detect the head's angle by using the yaw, pitch, and roll combined by using a threshold angle to detect if the user is drowsy. After detecting a drowsy user, the microcontroller turns on an LED and a buzzer to alert the user or other passengers.

Machine learning is then implemented and runs an inference model that was chosen. The given data from MPU6050, the machine learning model, then predicts the user's awareness. This is done in real time by ESP32, even with its small processing power.

As stated in the previous chapter, several machine learning algorithms resulted in only 4 models that provided high accuracy for this research scenario. The 4 best-performing models were Random Forest Tree, KNN, ANN, and Naive Bayes, which obtained accuracies of 100%, 99.85%, 97.58%, and 95.22%, respectively. This data is also shown in Figure 2.

In addition, the confusion matrix of each model has also been compiled into one figure to show the difference between each model and why each model is not accurate. This can be seen in Figure 3.

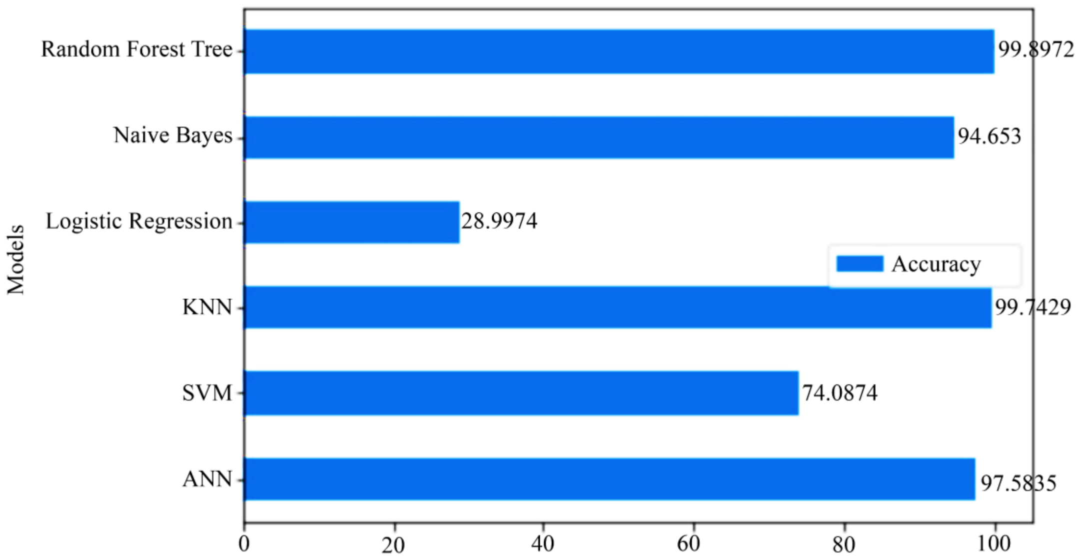

Fig. 2 The accuracy obtained from each model

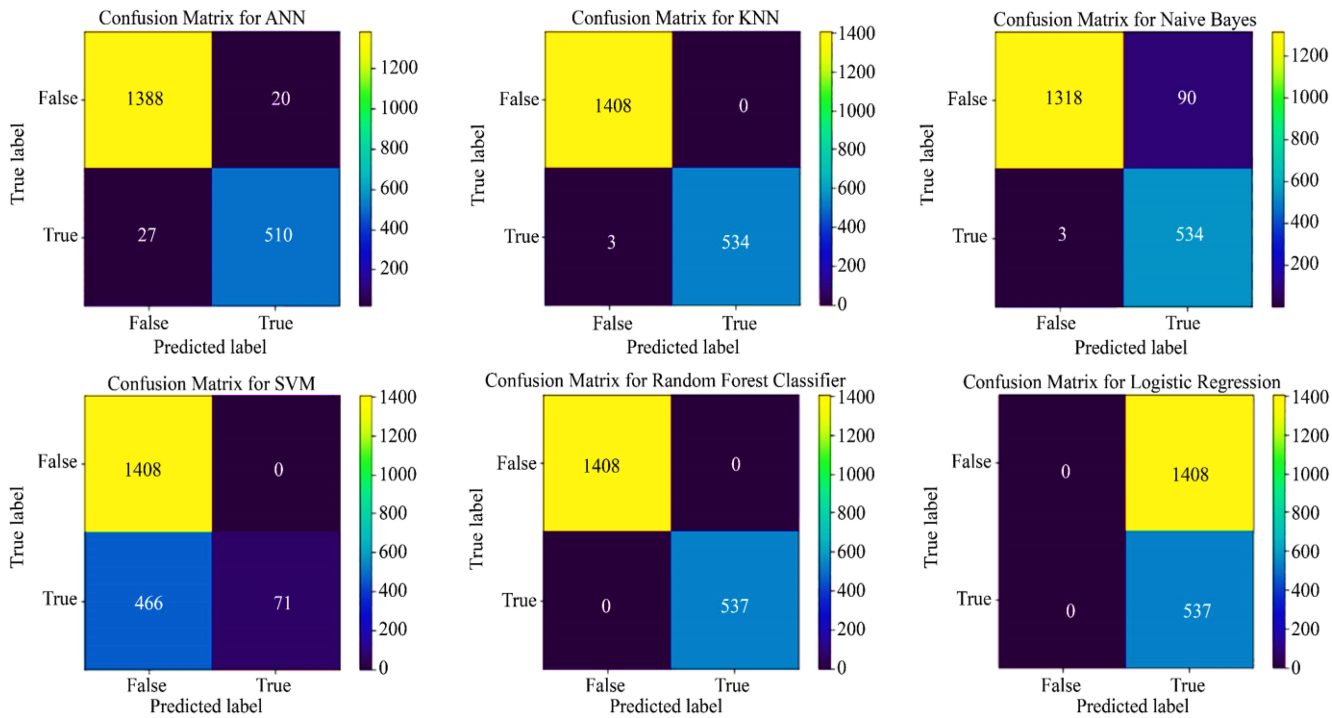

Fig. 3 Confusion matrix of each model used

The results of the proposed models show that this research requires a classification model, compared to regression, since the model only needs to classify between 2 classes being drowsy or awake. This can be seen as the Logistic Regression model performing poorly towards the classifiers. In addition to that, SVM is unable to distinguish between drowsy and awake, as there are many false predictions. This may be due to poor distinction between drowsy and awake. As from the paper by M. Ameliasari, A. G. Putrada, and R. R. Pahlevi showed that SVM can accurately classify gestures if they are highly distinctive and do not have too many features. [26] So, the fewer the features and the more distinctive the features are, the more accurate the SVM can perform classification.

Furthermore, Random Forest Tree, K-Nearest Neighbors, and Naive Bayes have an accuracy of over 95%, which means that it is reliable for classification. Naive Bayes itself has an accuracy of 95.21%. However, this model has a weakness where if the distribution of train and test data are different, it may affect the whole model to predict accurately. [13] In addition to that, based on R. Ashwin, the model also assumes that all the features are independent of each other, which is not true in real-life scenarios such as drowsiness detection mounted on a helmet. [14]

In the case of using K-Nearest Neighbors (KNN) for classification, after trying it with K values from 1 to 50, the most accurate is when using the K value of 1. There is one major setback toward using KNN as the classifier for the microcontroller since, based on [15], KNN has a high memory usage to store all the training data. Due to this, KNN will not be implemented on a microcontroller that only has 320 KiB of SRAM. In addition to KNN, the Random Forest tree has been tested for classifiers. However, there is a major issue during prediction. Based on [16], it also states that it is fast to train. However, larger trees may make the predictions slower than other methods since they have to traverse through the tree. This is a huge issue since the need for drowsiness detection requires a fast response from the gyroscope reading to the prediction of the drowsy state of the user to prevent dangers whilst driving.

However, a custom Artificial Neural Network model is chosen for the proposed model classification on the ESP32 Microcontroller since the proposed model was able to be extracted into a header file and then used for prediction in the microcontroller, especially since it was able to compete with the other classifiers and still produced an accuracy of 97.58%, a precision of 97%, a recall of 97%, as well as an f1 score of 97% on average.

The architecture of the proposed model was small, being only 3484 bytes, so it would be able to fit in the memory of the microcontroller, which consisted of 2 hidden layers, which had 16 neurons for each, as well as an input layer consisting of 3 neurons and a single output layer with a sigmoid function. Additionally, the mean squared error loss function and a rmsprop optimizer were used with the metrics of mean absolute error as well as accuracy. The network ran for 50 epochs with a batch size of 16.

5. Conclusion

From the data gathered ANN was implemented in a microcontroller as it is accurate enough, especially after comparing it with other models. The model can be exported into the microcontroller. In addition, the ANN architecture chosen was small enough that it has a small package size so that the model can deploy into a microcontroller. Real-time predictions are one of the goals for this research because the model has to be fast and accurate enough to predict if the user is drowsy or not. Using the proposed ANN architecture, the model can predict the user's state with an accuracy of 97.58% with fast predictions where the features extracted from the gyroscope are the tilt of the head.

Further studies and improvements can be made to this research by using more powerful microcontrollers to be able to use more advanced models. It can also be improved using a multimodal system, such as integrating computer vision or EEG, for more accurate and reliable drowsiness detection. The prototype created may also be miniaturized into a simple accessory

References

[1] Simon James Knopp, "*A Multi-Modal Device for Application in Microsleep Detection*," PhD Thesis, University of Canterbury, Electrical and Computer Engineering, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[2] M. Doudou, A. Bouabdallah, and V. Berge-Cherfaoui, "Driver Drowsiness Measurement Technologies: Current Research, Market Solutions, and Challenges," *International Journal of Intelligent Transportation Systems Research*, vol. 18, no. 2, pp. 297–319, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[3] Radovan Fusek, "Pupil Localization Using Geodesic Distance," *Advances in Visual Computing*, vol. 11241, pp. 433–444, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[4] Jonathan Flores-Monroy et al., "SOMN_IA: Portable and Universal Device for Real-Time Detection of Driver's Drowsiness and Distraction Levels," *Electronics*, vol. 11, no. 16, p. 2558, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[5] Bahjat Fatima et al., "Driver Fatigue Detection Using Viola Jones and Principal Component Analysis," *Applied Artificial Intelligence*, vol. 34, no. 6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6] Sameer Raju Dhole et al., "A Novel Helmet Design and Implementation for Drowsiness and Fall Detection of Workers on-Site Using EEG and Random-Forest Classifier," *Procedia Computer Science*, vol. 151, pp. 947–952, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[7] F. J. M. Newport, What is Attention Assist®? | Mercedes-Benz Safety Features, *Fletcher Jones Motorcars*, 2019. [Online]. Available: https://www.fjmercedes.com/mercedes-benz-attention-assist/

[8] Anneke Hertig-Godeschalk et al., "Microsleep Episodes in the Borderland between Wakefulness and Sleep," *Sleep*, vol. 43, no. 1, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[9] H. Ueno, M. Kaneda, and M. Tsukino, "Development of Drowsiness Detection System," *Proceedings of VNIS'94 - 1994 Vehicle Navigation and Information Systems Conference*, pp. 15-20, 1994. [CrossRef] [Google Scholar] [Publisher Link]

[10] Aqsa Mehreen et al., "A Hybrid Scheme for Drowsiness Detection Using Wearable Sensors," *IEEE Sensors Journal*, vol. 19, no. 13, pp. 5119–5126, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] D Mahalingam et al., "Real Time Monitoring for Drowsiness Detection using EEG System," *SSRG International Journal of Electronics and Communication Engineering*, vol. 2, no. 5, pp. 42-44, 2015. [CrossRef] [Google Scholar] [Publisher Link]

[12] Marco Taboga, 'Multinoulli Distribution', Lectures on Probability Theory and Mathematical Statistics, 2021. [Online]. Available: https://www.statlect.com/probability-distributions/multinoulli-distribution

[13] A. Aylin Tokuç, How to Improve Naive Bayes Classification Performance? | Baeldung on Computer Science, *www.baeldung.com*, 2021. [Online]. Available: https://www.baeldung.com/cs/naive-bayes-classification-performance

[14] Ashwin Raj, The Not So Naive Bayes, *Medium*, 2021. [Online]. Available: https://towardsdatascience.com/the-not-so-naive-bayes-b795eaa0f69b

[15] Marina Chatterjee, The Introduction of KNN Algorithm | What is KNN Algorithm?, *GreatLearning*, 2020. [Online]. Available: https://www.mygreatlearning.com/blog/knn-algorithm-introduction/

[16] Niklas Donges, A Complete Guide to the Random Forest Algorithm, *Built In*, 2021. [Online]. Available: https://builtin.com/data-science/random-forest-algorithm

[17] Pham Anh Tuan et al., "Micro-Opto-Electro-Mechanical Transducers for Measurement and Control Systems," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 5, no. 10, pp. 7-11, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[18] Susmita Ray, "A Quick Review of Machine Learning Algorithms," *IEEE Xplore*, pp. 35-39, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[19] Swapnil Sayan Saha, Sandeep Singh Sandha, and Mani Srivastava, "Machine Learning for Microcontroller-Class Hardware: A Review," *IEEE Sensors Journal*, vol. 22, no. 22, pp. 21362–21390, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20] Pete Warden, and Daniel Situnayake, *TinyML: Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers*. "O'Reilly Media, Inc., 2019. [Google Scholar] [Publisher Link]

[21] Ngan T.T. Le, Huyen T.T. Tran, "Robot Control Based On Image Processing To Follow The Target," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 7, no. 4, pp. 5-8, 2020. [CrossRef] [Publisher Link]

[22] Naveen Senniappan Karuppusamy, and Bo-Yeong Kang, "Multimodal System to Detect Driver Fatigue Using EEG, Gyroscope, and Image Processing," *IEEE Access*, vol. 8, pp. 129645–129667, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[23] Charlotte Jacobé de Naurois et al., "Detection and Prediction of Driver Drowsiness Using Artificial Neural Network Models," *Accident Analysis & Prevention*, vol. 126, pp. 95–104, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[24] M. Asjid Tanveer et al., “Enhanced Drowsiness Detection Using Deep Learning: An fNIRS Study,” *IEEE Access*, vol. 7, pp. 137920–137929, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[25] Umit Budak et al., “An Effective Hybrid Model for EEG-Based Drowsiness Detection,” *IEEE Sensors Journal*, vol. 19, no. 17, pp. 7624–7631, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[26] Maya Ameliasari, Aji Gautama Putrada, and Rizka Reza Pahlevi, “An Evaluation of SVM in Hand Gesture Detection Using IMU-Based Smartwatches for Smart Lighting Control,” *Jurnal Infotel*, vol. 13, no. 2, pp. 47–53, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[27] Brian C Tefft, “Acute Sleep Deprivation and Culpable Motor Vehicle Crash Involvement,” *Sleep*, vol. 41, no. 10, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[28] Gang Li, and Wan-Young Chung, “Combined EEG-Gyroscope-tDCS Brain Machine Interface System for Early Management of Driver Drowsiness,” *IEEE Transactions on Human-Machine Systems*, vol. 48, no. 1, pp. 50–62, 2018. [CrossRef] [Google Scholar] [Publisher Link]