Detect User Falling Over and Abnormal Gait Using Arduino Nano 33

Chia-Cheng Kuo
ckuo37@asu.edu
Arizona State University
Tempe, Arizona, USA

ABSTRACT

In this article, I'm going to present my approach, using machine learning implemented on Arduino Nano 33 sense, to detect abnormal gaits and falling motion.

ACM Reference Format:
Chia-Cheng Kuo. 2022. Detect User Falling Over and Abnormal Gait Using Arduino Nano 33. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

My device is designed to solve the problem that, elderly or some patients have higher risk of falling over due to the lack of muscle strength. There might not be a person to look after the elderly or patient all the time, and the consequences of falling over might be serious if not treated immediately after the falling incident. As a result, I would like to propose a device which can track user's gait motion and raise alert when user falling over. There are two main functions of my device: 1) detect user's abnormal gaits and record the walking motions. 2) Send alert emails when falling motion detected. The system is implemented using Arduino Nano 33 Sense, Edge Impulse, Bluetooth, and Python.

2 RELATED WORK

2.1 Body Sway Measurement for Fall Risk Assessment Using Inexpensive Webcams [6]

This is a paper published in 32nd Annual International Conference of the IEEE EMBS Buenos Aires, Argentina, August 31 - September 4, 2010, by Fang Wang, Student Member, IEEE, Marjorie Skubic, Member, IEEE, Carmen Abbott, and James M. Keller, Fellow, IEEE. The authors of the paper tried to use 2 inexpensive webcams to monitor the movement of a person, and then they used the video captured from the webcams to build a 3D model, then calculate the swaying of the person in the videos.

2.2 Wearable Sensor System for Detecting Gait Parameters of Abnormal Gaits: A Feasibility Study [8]

This is a journal published in IEEE SENSORS JOURNAL, VOL. 18, NO. 10, MAY 15, 2018, by Guangyi Li, Tao Liu, Senior Member, IEEE, and Jingang Yi, Senior Member, IEEE. The authors of the journal used 3 sensors: force sensor, IMU sensor, and range sensor to detect abnormal gait without toe rotation. The force sensor is used to measure the pressure given onto the shoe sole. IMU sensor is used to detect the rotation and acceleration. Range sensor is used to detect the distance between the shoe and ground. The result of the journal has approximately 5 % relative errors.

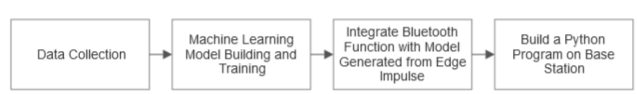
2.3 Walking Steadiness feature on Apple IOS

This is a new feature only on Apple IOS 15, which is the newest version of IOS, and was released in 2021. The walking steadiness feature can show the walking balance and stability of the mobile device user. The data is classified as OK, low, very low. The user of mobile device can get notification when the user's walking steadiness is low, which can warn the user about the risk of falling due to unstable walking.

The three related works focused on motion sensing using different methods. The first paper gave me the inspiration of using Arduino Nano 33 Sense to detect abnormal gaits. For the second paper, although it used three sensors in the experiment, it can only detect one kind of abnormal gait. For the walking steadiness feature on Apple IOS, it can only classify gait steadiness of Ok, low, and very low. Most important of all, these three related works do not have the falling motion detecting feature. As a result, I would like to use Arduino Nano 33 sense to design and build a device that can distinguish different kinds of gaits and detect falling motion.

3 SYSTEM DESIGN

There are two key features of my device, one is to recognize abnormal gaits, the other one is to raise warning when falling motion detected. There are four important steps in my progress of building the system. 1) Data collection 2) Machine Learning Model Building and Training 3) Integrating Bluetooth function with the machine learning model generated from Edge Impulse 4) Build a python program on base station.



3.1 Data Collection

I collected all the motion data by myself. I watched some Youtube videos [9] which explains the walking style of some abnormal gaits. After understanding the cause and characteristics of the abnormal gaits from the videos, I then imitated the abnormal gaits, with the Arduino board put in the pocket of the pants when collecting the data. There are 8 classes of data collected: idle, fall, normal walking, parkinsonian gait, slap gait, antalgic gait, diplegic gait, ataxic gait. I have totally collected 3 hours of data, with each class having approximately 20 minutes of data. The collected dataset was then uploaded onto Edge Impulse tool [5] for further model training. Figure 1 illustrates all features extracted from the dataset on Edge Impulse, with different color of dots representing different classes of data.

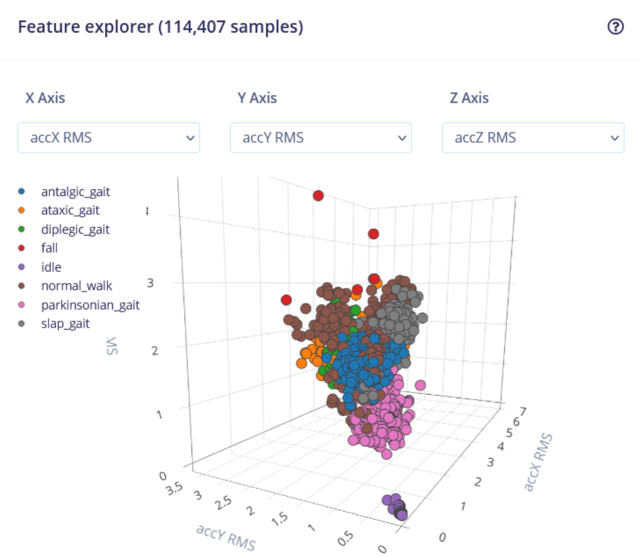


Figure 1: Plot of all extracted features from data.

3.2 Model Architecture and Training

The model was designed and trained on Edge Impulse tool [5], a development platform for embedded machine learning, which automatically helps us set up the machine learning model without writing the codes. I followed the official tutorial document on Edge Impulse website, which gives instructions on how to build and train a basic motion recognition model [1], with data acquisition, model architecture, and training parameters adjusted to my needs. The model architecture is shown in Figure 2. My model architecture had 3 layers of dense layers, with 20, 10 and 10 neurons respectively. The model was trained in 50 epochs, with the learning rate of 0.0005. The validation data was split from training data, with a portion of 20 %. After training the model on Edge Impulse, we can then download an Arduino library from Edge Impulse, which has the example code to perform data recognition on Arduino board. It is also worth mentioning that the original Edge Impulse tool has a working time limit of 20 minutes for each job. Edge Impulse will raise "ERR: DeadlineExceeded" error if the training time exceeds the 20 minutes time limit. I then emailed Edge Impulse customer service about the problem, and the working time limitation was then extended from 20 minutes to 60 minutes.

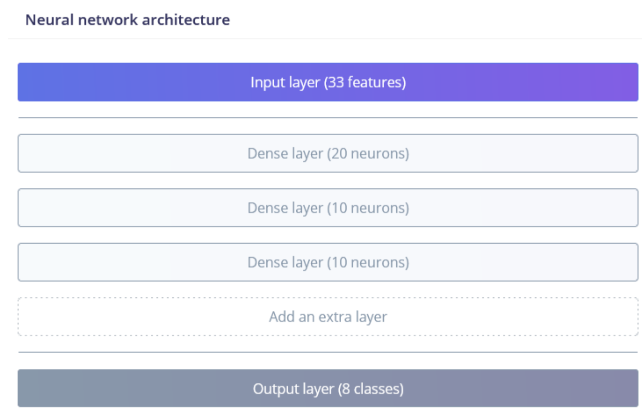


Figure 2: Model Architecture.

There is also an option for model quantization on Edge Impulse, enable us to quantize the model with int8 quantization automatically when training the model.

3.3 Bluetooth Function Integration

After obtaining the model and sample code from Edge Impulse which can recognize motion data, I then integrated the Bluetooth feature onto the sample code to make the Arduino Board to transmit prediction result to the base station. The original example code generated from Edge Impulse can recognize motion and will output the prediction result of the possibility of the detected motion belongs to each label and print the result in Serial port. I then retrieve the label which has highest possibility in the prediction result and transmit the predicted result to base station via Bluetooth. To integrate the Bluetooth feature with sample code generated from Edge Impulse, I referenced the mybledemo3.zip code uploaded on Canvas [7], which is the BLE Example 3 Program mentioned in the 9th course slide, Bluetooth Communication. I also designed to make the blue LED on Arduino Board Blink when the Board is searching for the base station to connect and will continue to be on when the Board is connected with a base station.

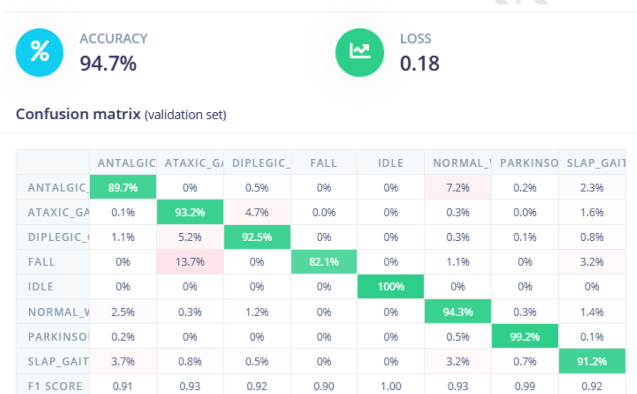
3.4 Functions on Base Station

There are several functions in the python program, which runs on base station. 1) receive Bluetooth signal from Arduino Board. 2) show illustration gif image. 3) record the prediction result on a csv file. 4) and send emails. I used python "Bleak" library to implement the function of receiving Bluetooth signal [4]. After receiving the Bluetooth signal, I used chrome driver and selenium library to show the gif images, which is easier to implement comparing with showing gif images by using tkinter library. To make the python program not to refresh the illustration video every time when it receives signal from Bluetooth, I add another variable to store the last prediction result received from Arduino Board. If the new signal received is the same as the last one, then the program won't refresh the illustration video. I also used pandas library to record the prediction result into a csv file, and then I used "smtplib" python library [2] [3], to send alert emails when falling motion detected. The email sender and receiver can be configured in "senderemail.txt" and "receiver email config.txt" file.

4 EVALUATION

4.1 Accuracy

The training result is illustrated in Figure 3, with an accuracy of 94.7 %, and all classes of data have accuracy over 90 %. Although the training result performed well on Edge Impulse, it did not perform so good on real-time prediction implemented on Arduino Nano Board. Here are some challenges that still need to be improved. In my real-time performance experiment, 1) the system would sometimes predict antalgic gait as other abnormal gaits. Figure 3 also shows that antalgic gait has the worst accuracy among all abnormal gaits on Edge Impulse, which corresponds to my real-time performance experiment. 2) During turning around and transition of different abnormal gaits, my system does not have good accuracy. 3) The system sometimes misestimates unreal falling motion which has small acceleration as ataxic gait. The reason of this error might be due to that the falling motion data I collected mostly have high acceleration, which I really used my bed mattress as a buffer when I was collecting falling data. The training result on Edge Impulse, which is illustrated in Figure 3, also shows that there is a high chance of 13.7 % to misestimate falling motion as ataxic gait. 4) The same reason would result in a similar error, which my system sometimes misestimates high acceleration motions, including jumping or other abrupt high acceleration motions, as falling motion.


ACCURACY
94.7%

LOSS
0.18

Confusion matrix (validation set)

	ANTALGIC	ATAXIC_GA	DIPLEGIC_	FALL	IDLE	NORMAL_\	PARKINSO	SLAP_GAIT
ANTALGIC_	89.7%	0%	0.5%	0%	0%	7.2%	0.2%	2.3%
ATAXIC_GA	0.1%	93.2%	4.7%	0.0%	0%	0.3%	0.0%	1.6%
DIPLEGIC_(	1.1%	5.2%	92.5%	0%	0%	0.3%	0.1%	0.8%
FALL	0%	13.7%	0%	82.1%	0%	1.1%	0%	3.2%
IDLE	0%	0%	0%	0%	100%	0%	0%	0%
NORMAL_V	2.5%	0.3%	1.2%	0%	0%	94.3%	0.3%	1.4%
PARKINSOI	0.2%	0%	0%	0%	0%	0.5%	99.2%	0.1%
SLAP_GAIT	3.7%	0.8%	0.5%	0%	0%	3.2%	0.7%	91.2%
F1 SCORE	0.91	0.93	0.92	0.90	1.00	0.93	0.99	0.92

Figure 3: Training Result on Edge Impulse.

For the optimization of the model, the model has 96.1 % accuracy with 22.2K flash usage before optimization. After int8 quantization, the model has 94.7 % accuracy, with 19.6K flash usage.

4.2 Power Consumption

The power consumption was tested using an external power source of 9v alkaline battery. A breadboard power supply module, which can turn 9v battery power into 5v, is used for plug type and voltage transition. The Arduino Nano Board was then powered with a 5v input source, connected with the Vin and GND pin port on Arduino. The result of the experiment showed that the system can operate normally (continue to predict motion and transmit prediction result to a base station via Bluetooth), for over 20 and a half hours using Amazon 9v alkaline battery. The power consumption experiment setup is illustrated in Figure 4. The reason why I did not use a portable charger as my power supply for this experiment, which is also a feasible power source, , was because the Arduino board draw too less electrical current from the portable charger that my charger would automatically turn off after a few second if it only supplies power to the Arduino Board. To make the portable charger continue to supply power to the Arduino Board, I need to connect my portable charger to the Arduino Board and another charging phone at the same time, so that there is enough electrical current drew from the portable charger. Although this is a feasible method to continue powering up the Arduino Board, this is not a good way to test the system's power consumption since the portable charger will charge a phone at the same time, makes it hard for me to estimate how much power is distributed to the Arduino Board, and how much power is distributed to the charging phone.

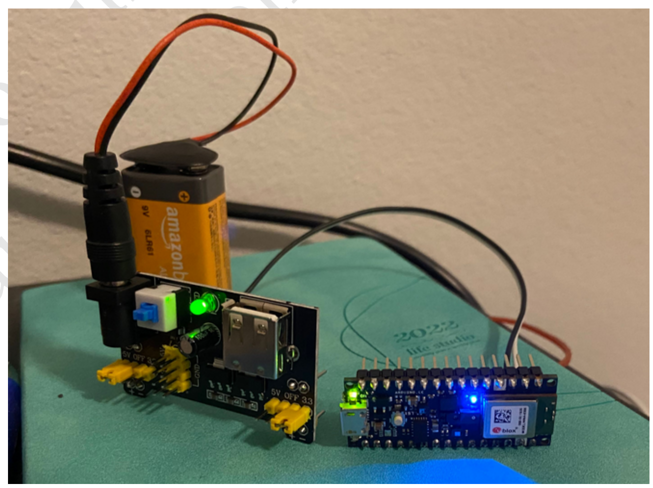


Figure 4: Power consumption experiment setup.

5 CONCLUSION

It is crucial to provide emergency treatment for elderly or patients when they fall over. It is also important to provide warnings if the elderly or patients have high risk of falling due to abnormal gait. Although my system does not have high accuracy in real-time performance and will sometimes misestimate a kind of abnormal gait as another kind of abnormal gait, it still has good accuracy on distinguishing normal and abnormal gait. Future works for the project may include: 1) analyzing the recorded walking motion and give warnings if the user have too much abnormal gaits detected and have high risk of falling. 2) collect more data, try more model architectures and training parameters to improve real-time performance accuracy.

ACKNOWLEDGMENTS

My thanks to Professor Hassan Ghasemzadeh for the lectures and course material design of the Embedded Machine Learning course, and the advice on the final project.

REFERENCES

[1] .. *An official tutorial document of how to use Edge Impulse to build and train basic a motion recognition model.* https://docs.edgeimpulse.com/docs/tutorials/continuous-motion-recognition
[2] .. *Python smtplib library.* https://docs.python.org/3/library/smtplib.html
[3] .. *A simple referenced python program using smtplib to send email.* https://www.afternerd.com/blog/how-to-send-an-email-using-python-and-smtplib/
[4] 2017. *Python Bleak library.* https://pypi.org/project/bleak/
[5] 2019. *Edge Impulse.* https://www.edgeimpulse.com/
[6] Carmen Abbott James M.Keller Fang Wang, Marjorie Skubic. 2010. *Body sway measurement for fall risk assessment using inexpensive webcams.* https://doi.org/10.1109/IEMBS.2010.5626100
[7] Hassan Ghasemzadeh. 2022. *mybledemo3.zip. Sample code of Bluetooth function posted on Canvas, which was also the BLE Example 3 Program mentioned in the 9th lecture slide.*
[8] Jingang Yi Guangyi Li, Tao Liu. 2018. *Wearable Sensor System for Detecting Gait Parameters of Abnormal Gaits: A Feasibility Study.* https://doi.org/10.1109/JSEN.2018.2814994
[9] Ontario School of Osteopathy. 2015. *Youtube video explaining normal and abnormal gait, used for reference when collecting gait data.* https://www.youtube.com/watch?v=PZBiv0uSXVg