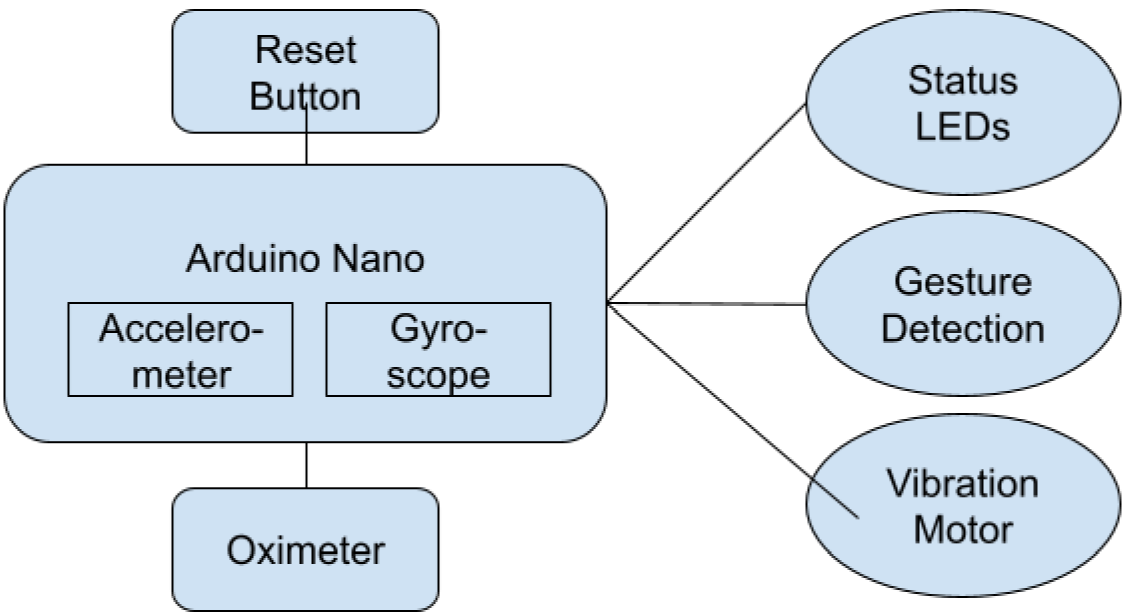
Bangjie Xue, Qi Luo
Professor Youssfi
18848C Embedded Deep Learning
13 Dec. 2024

Final Project Report: Gesture Detection on Wrist

Abstract

This project aims to replicate the Apple Watch double-tap feature using on-device machine learning with an Arduino Nano BLE Sense 33. The device leverages its built-in accelerometer, gyroscope, and an externally attached MAX30102 oximeter sensor [1] to detect gestures such as double-tap, clench, and rotate. The hardware is housed in a 3D-printed wearable enclosure, mimicking a smartwatch form factor. This system has the potential to improve accessibility, especially for individuals with disabilities or those needing single-handed operation. The final system achieves a gesture detection accuracy of 94%, with robust classification performance across five gesture classes (including idle and other).

High-level block diagram



[1] Possible ethical issues could include: user's health data (blood oxygen level) will be collected during this experiment. However, we are only getting the raw voltage value from the sensor instead of using it to interpret the heart rate or blood oxygen level. Therefore this ethical issue should not be a concern. There could be a potential bias due to the small sample size collected and may be overfitting to our own specific gestures. If we were to deploy this project, more data should be collected on a more diverse set of people. Skin colors could potentially also be a factor because it will affect the emission rate from the IR oximeter sensor. Since this is a class project, and only the two of us working on the project are going to be the users, we believe the ethical issue would not be a problem.

Dataset Source and Method of Collection and Data Cleaning

1. **Data Collection**

   We collected the data ourselves using a custom Arduino script [2] to collect sensor data, including acceleration (x, y, z), gyroscope (x, y, z), and IR values from the MAX30102 oximeter through I2C protocol. Data was forwarded to Edge Impulse via their data-forwarder.

   The collection process ensured:
   - Sampling frequency of 50 Hz.
   - Real-time monitoring of sensor outputs.
   - Data was captured under realistic conditions with gestures like clench, pinch, and idle performed multiple times.
2. **Data Cleaning**
   - Filtering: Raw sensor data was filtered to remove noise and outliers.
   - Scaling: Sensor values were mapped to a range of [-20, 20] to standardize inputs.
   - Normalization: Features were normalized to ensure consistency across sessions.

Feature extraction

1. **Time-Series Features**
   - Seven input axes (accelX, accelY, accelZ, gyroX, gyroY, gyroZ, IR) were used to create input features for the classifier.
2. **Spectral Analysis**
   - Processing Method: FFT with logarithmic scaling and overlapping frames.
   - Output Features: Frequency-domain spectral power for all input axes.
3. **Edge Impulse Configuration**
   - Zero-padding was applied to ensure uniform input length across samples.
   - Spectral features from all seven axes were selected for model training.

[2] https://github.com/tomqiluo/Gesture-Detection/blob/main/gesture_detection.ino

Classifier architecture and rationale

1. Neural Network Architecture

2. Rationale

The architecture was chosen to balance computational efficiency and classification accuracy, ensuring compatibility with embedded hardware. Dropout layers mitigated overfitting, while quantization reduced model size for embedded deployment.

Confusion matrix and overall accuracy metrics

1. Confusion Matrix

	CLENCH	IDLE	OTHER	PINCH	ROTATE
CLENCH	100%	0%	0%	0%	0%
IDLE	0%	100%	0%	0%	0%
OTHER	0%	9.1%	72.7%	0%	18.2%
PINCH	0%	0%	0%	100%	0%
ROTATE	0%	0%	0%	0%	100%
F1 SCORE	1.00	0.95	0.84	1.00	0.88

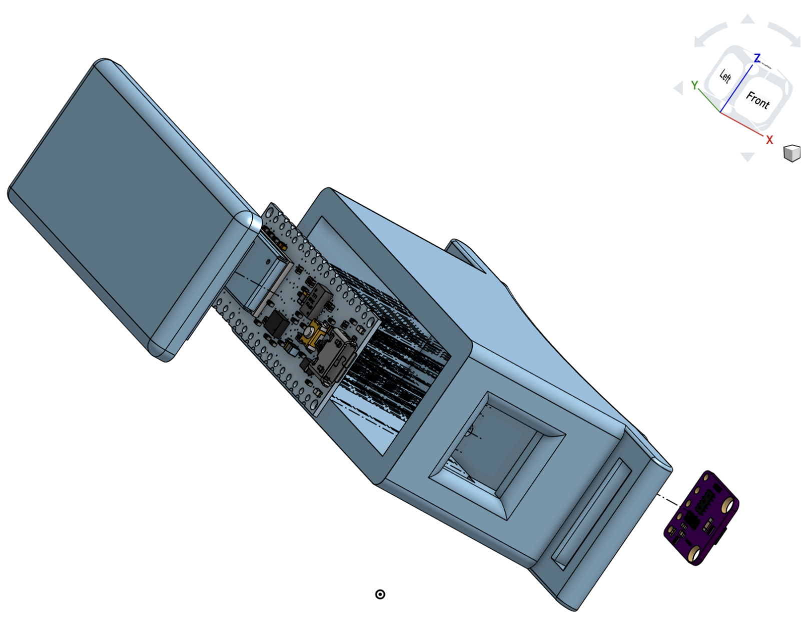
2. **Overall Metrics**

Area under ROC Curve ⓘ	1.00
Weighted average Precision ⓘ	0.95
Weighted average Recall ⓘ	0.94
Weighted average F1 score ⓘ	0.94

Deployment method: hardware target and software techniques

1. **Hardware**

In order to ensure repeatability, an enclosure that houses the oximeter sensor and the Arduino board was designed and 3D printed. The enclosure turns the whole setup into a watch you can wear on your wrist. The watch is strapped onto the wrist using velcro to ensure the oximeter sensor is tightly pressing against the skin on the wrist, which is essential for accurate detection. All parts are precisely designed so that all components can be friction fitted together without the need for screws, which significantly reduces the assembly efforts. On the top of the watch are three LEDs to indicate different statuses, and a reset button to restart the program. Within the housing, a vibration motor is attached to the side of the wall to alert the user of a new simulated message pending review. The oximeter interacts with the Arduino through I2C pins, the LEDs and the vibration motor are controlled through GPIO pins, the reset button is connected to the reset pin and ground. Notability, the vibration motor requires higher current to operate, therefore two GPIO ports are connected in parallel to provide higher current to drive the motor.



2. **Software**
    - A neural network trained in Edge Impulse.
    - Deployed as a quantized Arduino library for real-time execution. [3]
3. **User Manual**

    The whole setup is a “watch” attached to the wrist with the oximeter pressing on the skin. First, turn the reset switch off and back on to start a new session. After a few seconds, a new message will arrive. When a new message arrives, the watch will vibrate and flash the blue light. The user can then raise their hand to “read” the message. You can acknowledge the message by double tapping the finger, and the system will light up the green LED to indicate success acknowledgment. If you wish to delete the message, you can clench your fist once and the yellow LED will light up to indicate successful deletion.

Significant challenges and lessons learned

1. **Challenges**
    a. Small Dataset
    b. Hardware Constraints
2. **Lessons Learned**
    a. Ways to mitigate overfitting whilst having a small dataset: dropout layer.
    b. Ways to efficiently enhance a system's performance: quantization.

Parts List

- Arduino BLE Sense 33
- Oximeter MAX30102 [4]
- 3D printed enclosure
- Green, Blue, Yellow LEDs
- Push button
- Velcro
- Vibration motor

Reference

Apple. "Apple Watch Double Tap Gesture Now Available with watchOS 10.1." *Apple Newsroom*, October 2023, https://www.apple.com/newsroom/2023/10/apple-watch-double-tap-gesture-now-available-with-watchos-10-1/.

[3] Source Code: https://github.com/tomqiluo/Gesture-Detection
[4] https://www.amazon.com/HiLetgo-MAX30102-Breakout-Oximetry-Solution/dp/B07QC67KMQ