Lying Posture Detection using IMU Sensor

Tulasi Sainath Polisetty
ASUID: 1223964586

September 20, 2023

1 System Design

1.1 Motivation

Lying posture tracking is paramount in clinical contexts. It offers insights into a patient's mobility, potential risks of hospital-acquired pressure injuries, and sleep quality. The primary objective of our system is to provide a real-time analysis of an individual's lying posture.

1.2 High-level Design

The design leverages an IMU sensor embedded in the Arduino board. The system is calibrated to detect three postures: supine, prone, and side. Depending on the detected posture, an LED indicator offers visual feedback.

1.3 Specific Observations and Difficulties

During the design phase, the positioning and orientation of the IMU sensor became crucial. Also, maintaining a balance between the sampling rate and power consumption was a concern.

1.4 Sampling Frequency

The IMU sensor (Accelerometer) sampled at 119.00 Hz, ensuring a good trade-off between data resolution and power consumption, and the baud rate use is 9600 Bd.

2 Experiment

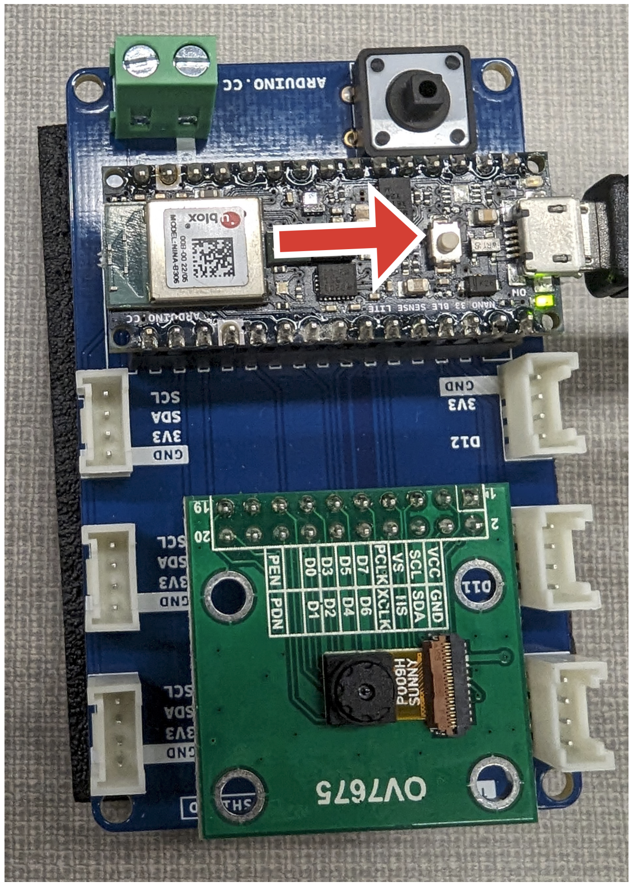

Figure 1: Picture of my setup. The arrow head indicates the direction taken as the "head".

2.1 Conduct of Experiments

Lying postures were simulated in a controlled environment, using the Arduino board with the IMU sensor. The board was strategically positioned on a flat surface in a manner that the side with the onboard and Pwd LED represented the head, while the opposite side was considered as the legs. This arrangement was set as a reference or 'compass' to ensure that the readings reflected the respective orientations accurately.

2.2 Data Collection

For each lying posture, data was collected continuously for approximately 1 minute and 10 seconds, with a possible variation of ± 10 seconds. This ensured a comprehensive set of readings for each posture while accounting for any minor deviations in collection time.

2.3 Experimental Environments

All experiments were carried out indoors to eliminate any potential external interference that might influence the sensor's readings.

2.4 Challenges

One of the main challenges faced was in emulating real-world lying postures without the nuances of human interaction. Moreover, maintaining a consistent sensor orientation, given our specific head-leg compass, was vital to ensure accurate and consistent readings throughout the experimentation phase.

3 Algorithm

My algorithm harnesses the nuances of acceleration variations across distinct axes to ascertain posture. Drawing inspiration from these dynamic shifts, the system manifests a symbiotic relationship between posture detection and LED signaling: a single blink every 0.8 seconds indicates a supine position; a double blink in the same time frame suggests a prone stance; and a threefold blink is reserved for the side position. Crucially, when the acceleration values veer outside the calibrated thresholds—perhaps when an individual stands—the LED remains unilluminated.

3.1 Design and Observations

Distinct acceleration patterns were noticed for each posture. For instance, the supine position showed specific readings in the Z-axis, different from the other postures. And, for the side position, the Y-axis reading was either a positive value above the taken threshold or negative depending on right side or left side.

3.2 Approach and Robustness

The algorithm effectively identifies the postures in most scenarios. However, deviations in the sensor's orientation can lead to mis-classifications.

3.3 Output Frequency

The LED output is updated every 0.8 seconds, ensuring real-time feedback without overburdening the system and to identify different blinking patterns without confusion.

4 Results

Below are the results of the project/experiment. Also the plots have been smoothed out using a rolling average/moving average filter.

4.1 Sensor Data Plots

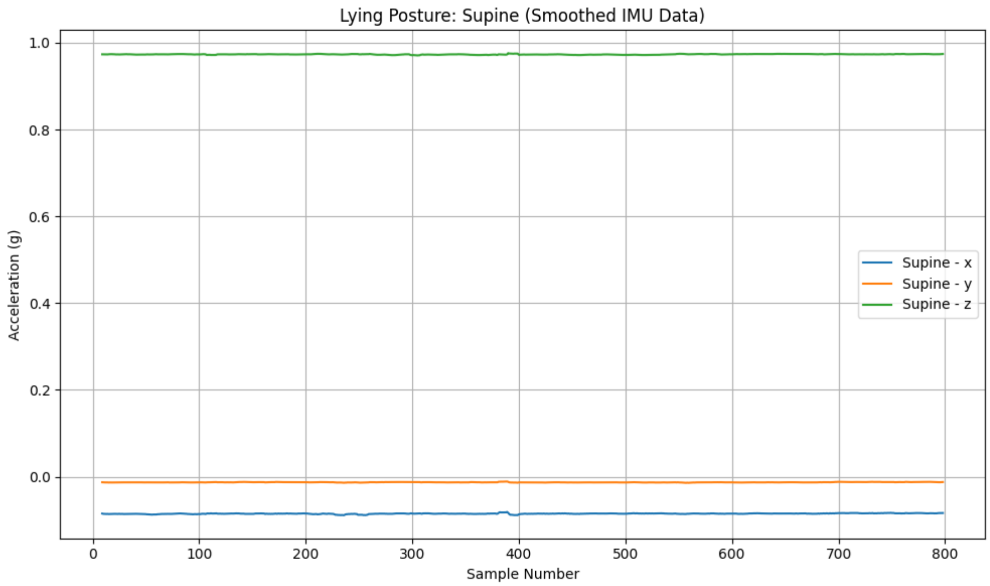

Figure 2: Sensor data for Supine Posture

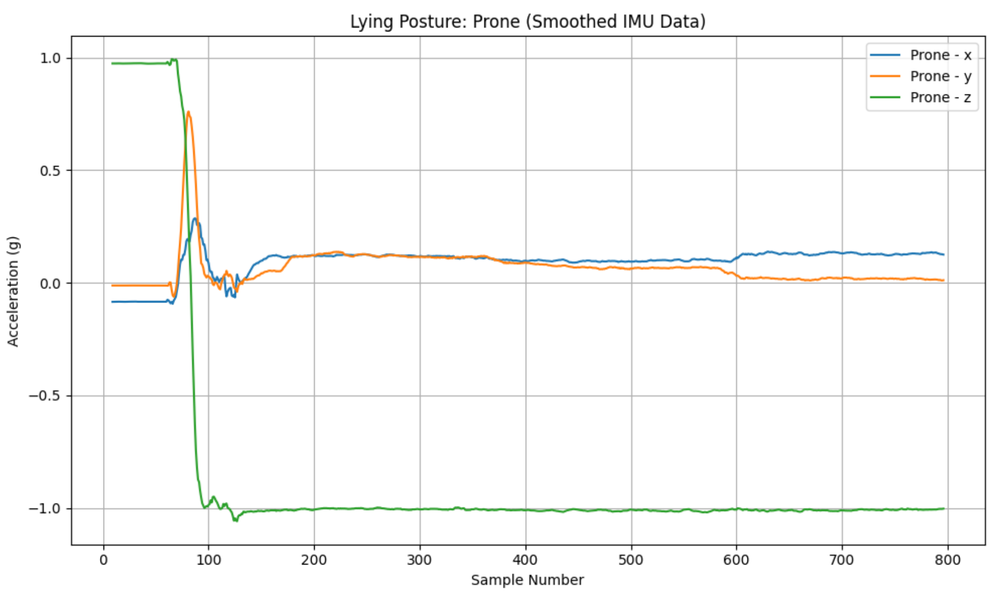

Figure 3: Sensor data for Prone Posture

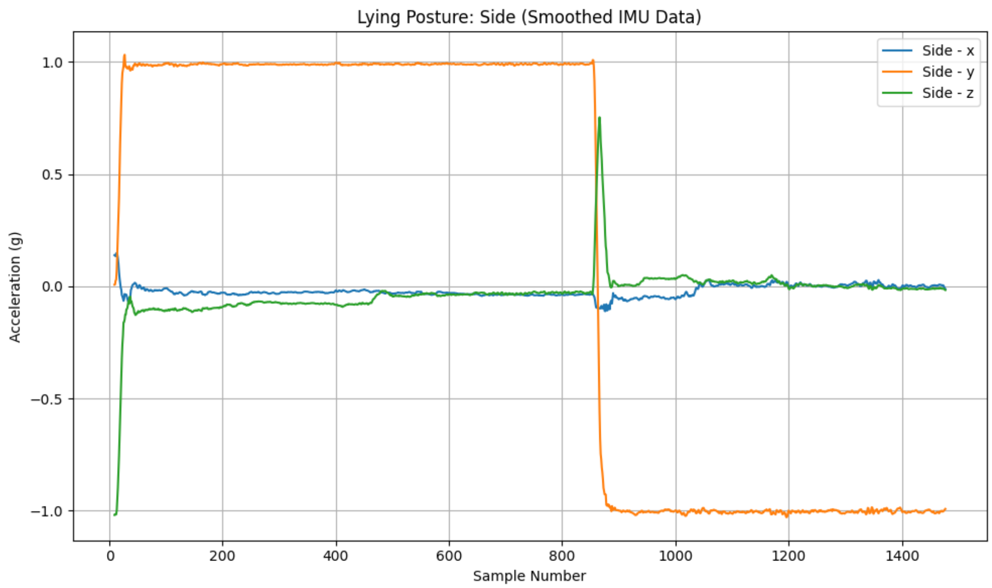

Figure 4: Sensor data for Side Posture

4.2 Accuracy Performance

The system exhibited varying performance for different postures. Utilizing the formula:

$$\text{Accuracy} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}$$

where TP represents True Positives, FP represents False Positives, and FN represents False Negatives, we derived the following accuracies:

- For the supine posture:

$$\text{Accuracy}_{\text{supine}} = \frac{8}{8+2+0} = \frac{8}{10} = 80\%$$

- For the side posture:

$$\text{Accuracy}_{\text{side}} = \frac{18}{18+2+4} = \frac{18}{24} = 75\%$$

- For the prone posture:

$$\text{Accuracy}_{\text{prone}} = \frac{8}{8+0+2} = \frac{8}{10} = 80\%$$

Thus, the system detected the supine, prone, and side postures with accuracies of 80%, 80%, and 75%, respectively.

During tests that strictly adhered to standard postures without introducing edge cases, the system achieved an accuracy of 100%. However, when subjected to edge cases, there were slight variations in detection accuracy, as indicated above.

4.3 Potential Failures

The system could potentially mis-classify postures if the sensor is tilted beyond a specific angle or subjected to abrupt movements. Such scenarios were evident in our tests, especially when edge cases were introduced. Future iterations of the system should focus on enhancing its robustness to such factors to achieve consistent and reliable posture detection.

4.4 Potential Corner Cases

The system might mis-classify if the sensor is tilted beyond a specific angle or if there's sudden motion.

5 Discussions

5.1 Summary of Results

My system offers a real-time analysis of lying postures with considerable accuracy. However, there's room for improvement, especially in terms of handling slight changes in sensor orientation.

5.2 Difficulties in System Design

Designing an algorithm that can distinguish between subtle changes in acceleration readings was a task. Also, emulating real-world scenarios without actual human subjects presented its own set of challenges.

5.3 Improvements for the Future

Future iterations could use machine learning models trained on a richer dataset, improving the system's robustness. Implementing sensor fusion, where multiple sensors are used, can also offer more precise posture detection.

5.4 Challenges and Real-time Approach

Achieving real-time performance without compromising on accuracy was the most challenging aspect of this project. My real-time approach was effective in most scenarios, though corner cases were observed where there was mis-classification.