SENSOR LAB – 4

Human Activity Recognition Using Smartphone Accelerometer

Student Details

- **Name:** MD TOUSIF ANSARI
- **Roll No:** 23F3000577

1. Introduction

Modern smartphones have built-in accelerometer sensors capable of measuring movement in three dimensions. These sensors allow for the detection of both static (gravity) and dynamic (activity-induced) acceleration. By analyzing these acceleration patterns, we can differentiate between human activities such as walking, stair climbing, and descending stairs. Such analyses are useful for personal fitness tracking, healthcare monitoring, sports analysis, and smart environments.

Experimental Aim:
This experiment uses a smartphone accelerometer to collect motion data from various human movements. Data is processed and analyzed to identify patterns distinctive to different activities. Step-counting and activity classification methods are applied to demonstrate transformation of raw sensor data into actionable insights.

2. Objective & Goal

Objective

- To capture and analyze accelerometer data from a smartphone during different human activities.
- To count steps and identify types of movement.

Goal

- To design and implement a method that can detect steps and classify activities (walking, stair climbing, stair descent) using techniques such as data filtering, peak detection, and pattern analysis.

3. Description of Tools Used

A. Phone Model

- **Device:** Samsung Galaxy M53 5G
- **Features:**
    - Android smartphone suitable for frequent sensor logging and data capture
    - Includes a 3-axis accelerometer and other sensors, accessible via Android's Sensor Manager API

B. Accelerometer Specifications (Samsung Galaxy M53 5G)

- **Type:** 3-axis MEMS accelerometer (linear acceleration + gravity)
- **Measurement Range:** ±2g / ±4g / ±8g / ±16g (varies by configuration)
- **Resolution:** milli-g level per LSB for ±2g (varies by MEMS part used)
- **Output Units:** m/s^2 via Android API
- **Sampling Rate:** Up to SENSOR_DELAY_FASTEST; typically, 100–200Hz supported.

C. Sampling Rate Selection

- **Recommended Rate:** 100–200Hz (optimal balance for accuracy and noise reduction)
- **Rationale:**
    - Human gait frequency: ~1–3Hz
    - Sampling ≥50Hz covers Nyquist requirements for step peaks
    - 100–200Hz gives better timing and activity classification robustness
- 200Hz increases power/file size without significant advantage

D. Apps Selected

Three apps were used to ensure data validity and precision:

1. **Physics Toolbox Sensor Suite (Pro)**
   - Configurable sampling, multi-axis logging, CSV export, real-time visualization
2. **Accelerometer (diagnostic/monitoring app)**
   - Cross-check for live readings, sensor stability, and gravity/static tests
3. **Arduino Science Journal (formerly Science Journal by Google)**
   - For structured experiments, annotations, simple CSV export; used as validation logger

Note: Using different apps reduces the chance of app-specific artifacts affecting the results.

E. Method of Data Transfer

Any of the following methods are suitable:

- Bluetooth file transfer
- Cloud sync/export
- Email sharing

Method used:
CSV file export for data transfer into Excel for data analysis.

Link of files used in experiment-

mat_23f3.m\stairs_down.csv

mat_23f3.m\stairs_up.csv

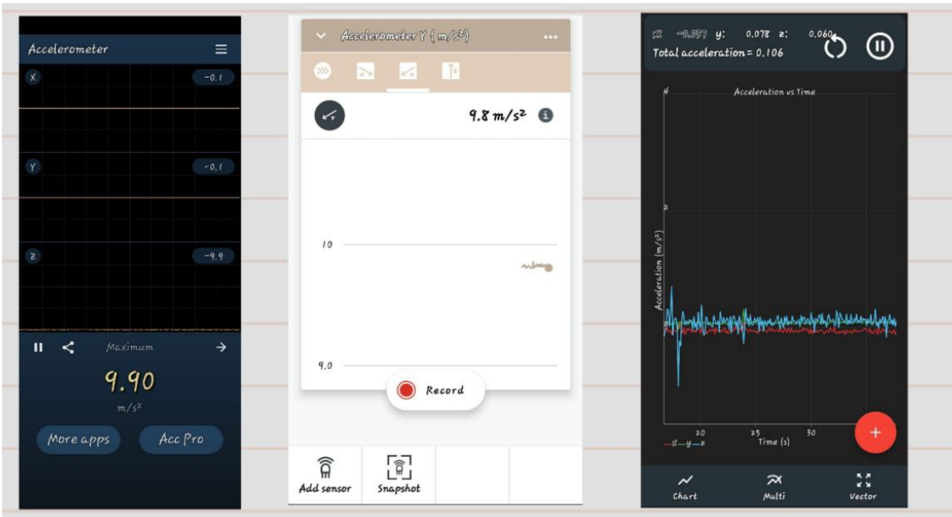
mat_23f3.m\Static.csv

mat_23f3.m\walking_static.csv

mat_23f3.m\more_activities.csv

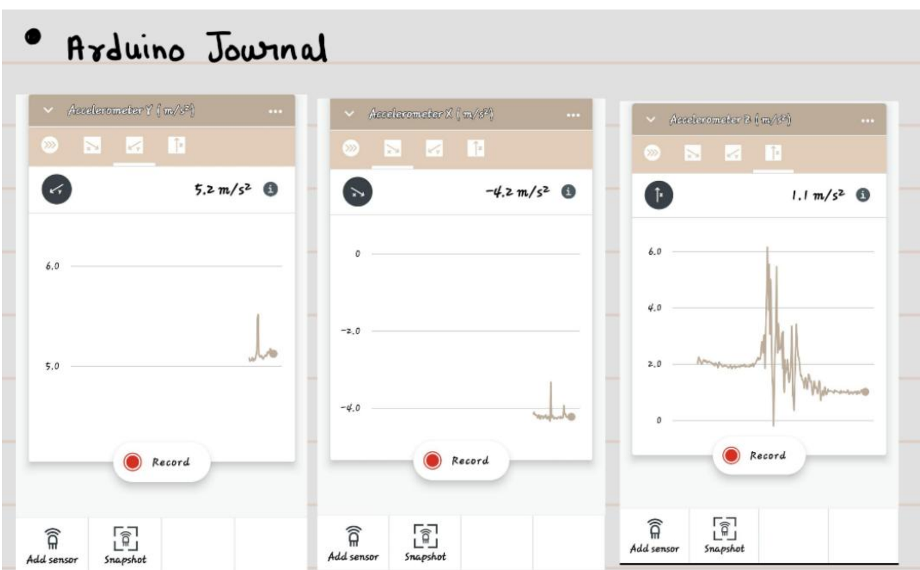
<u>**3.Initial Accelerometer Testing Results**</u>

(a). Static response observation

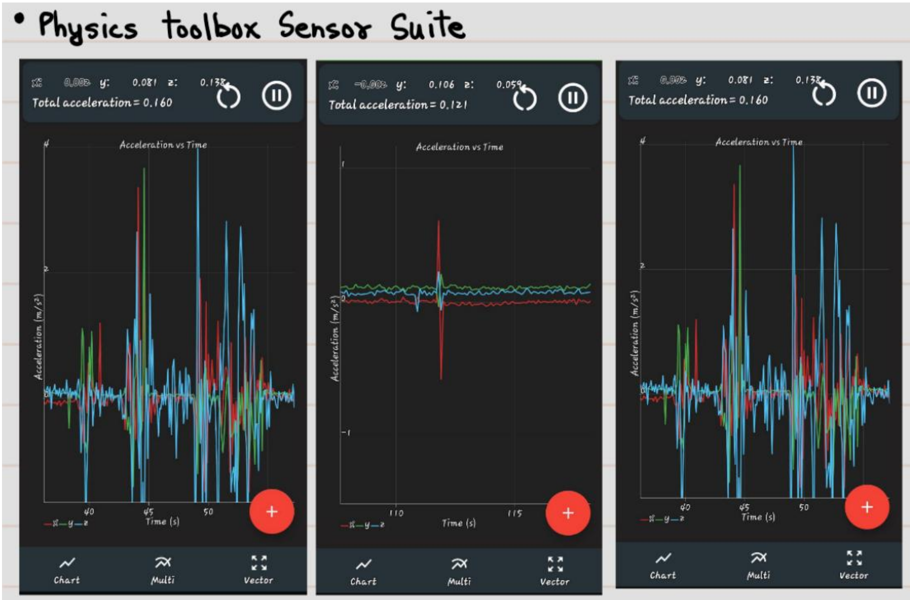


(b). *dynamic response observations*

1.Arduino Journal



2. Physics Toolbox Sensor Suite



3.Accelerometer

- Static Response:

  - The accelerometer accurately measures the gravitational acceleration (~9.8 m/s^2) along the axis(z) aligned with gravity.
  - When the device is placed flat or at different angles, the readings correspond correctly to gravity's direction, confirming sensor calibration.
  - Negligible acceleration is observed on the axes perpendicular to gravity, indicating low sensor noise at rest.

- Dynamic Response:

  - Accelerometer readings show distinct variations during movements such as shaking, rotation, walking, and stair climbing.
  - Dynamic signals capture characteristic motion patterns necessary for activity recognition.
  - The sensor responds quickly and reliably to changes in motion, ensuring accurate step counting and activity differentiation.
  - Some noise and jitter may be present, which can be effectively reduced by filtering techniques like moving average or Butterworth filters.

4.Data Collection Summary

For data collection I did the following 4 activities-

- Flat surface (On table)
- Walking
- Stairs up
- Stairs down

- All the above data were collected for a duration of ***20 sec to 60 sec*** using Physics toolbox suite app. The data were then rechecked for validation using accelerometer app and Arduino journal app.

- Raw data is plotted below for reference (In results section).

MATLAB code

```matlab
%% Load data (Time | AccX | AccY | AccZ)
T = readtable('static.csv','VariableNamingRule','preserve');
t = T{:,1}; x = T{:,2}; y = T{:,3}; z = T{:,4};

%% Step 1: Raw Acceleration Magnitude
a = sqrt(x.^2+y.^2+z.^2);
```

5.Algorithm Description

To process the data, I used ***MATLAB***. First, I obtained the data from the Physics Toolbox Suite app in Excel (.csv) format on my laptop. I then filtered and plotted the data to generate the curve. Additionally, I implemented step detection and an activity classification algorithm to identify the number of steps taken and classify the performed activity. The peak is also marked in the plot below for peak detection.

```matlab
%% Step 2: Filtering (Moving Average + remove baseline)
aFilt = movmean(a,10);
aFilt = aFilt - mean(aFilt);   % remove DC (gravity baseline)

%% Step 3: Step Detection (robust threshold)
fs = round(1/mean(diff(t)));   % sampling rate
thr = prctile(aFilt,85);       % robust threshold (85th percentile)
[pk,loc] = findpeaks(aFilt,'MinPeakHeight',thr,'MinPeakDistance',round(fs*0.3));
stepT = t(loc);
stepFreq = 1./diff(stepT);
```

```matlab
%% Step 4: Activity Classification
win = 10;                   % window size in seconds
samplesWin = win*fs;
activityData = strings(length(t),1);

for i = 1:samplesWin:length(t)-samplesWin
    idx = i:i+samplesWin-1;
    [pks,~] = findpeaks(aFilt(idx),'MinPeakHeight',thr,'MinPeakDistance',round(fs*0.3));
    if numel(pks)<2
        act = "Idle";
    else
        % Estimate frequency from step intervals
        stepTimes = t(idx(1)) + (1:numel(pks));
        f = numel(pks)/win;   % steps/sec in window
        amp = mean(pks);

        if f<=0.5 && amp>5 && amp <10
            act="Walking";
        elseif f>0.5 && amp>=1 && amp <3
            act="Stairs Up";
        elseif f>1.5 && amp<1
            act="Stairs Down";
        else
            act="Unknown";
```

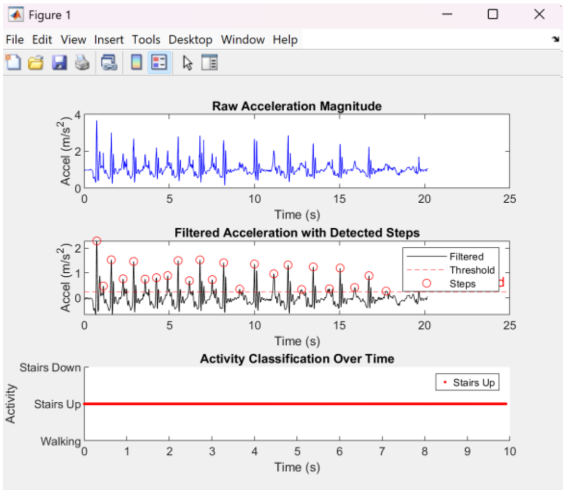
```matlab
        end
    end
    activityData(idx) = act;
end
```

After this I plotted the graph and got the results below.

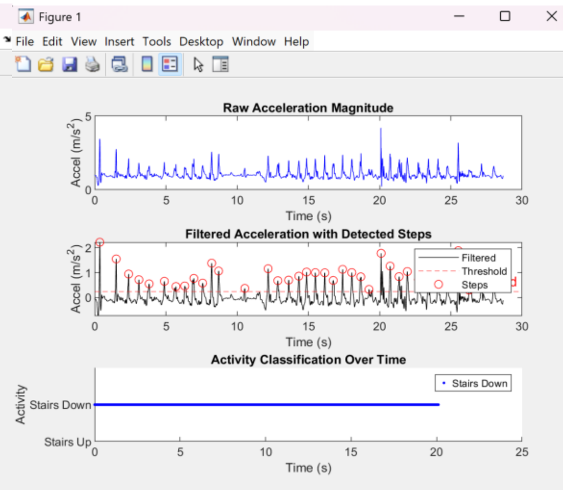
6.Results

Plot for raw and processed Accelerometer data with detected steps.

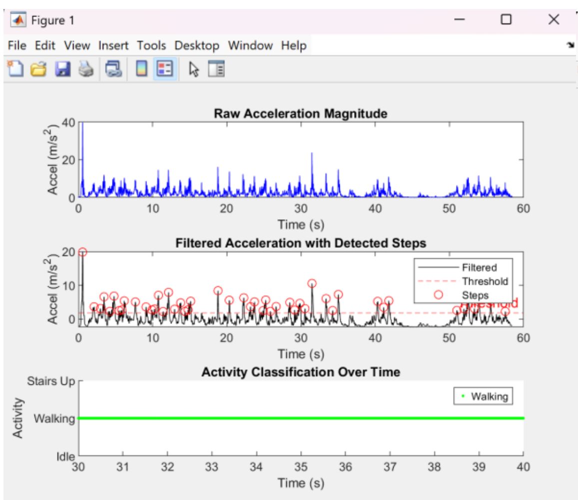
Stairs Up



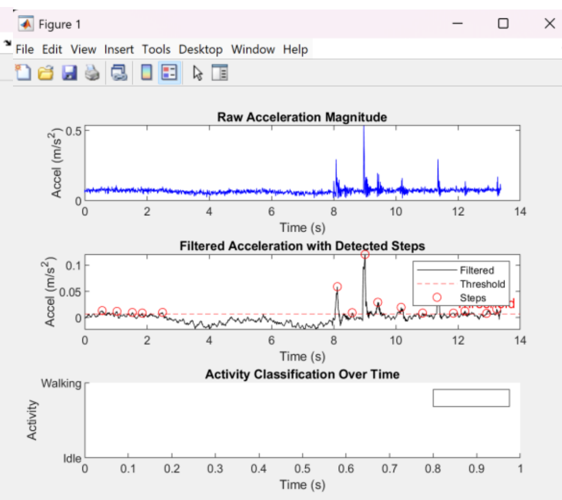
Stairs Down



Walking



Flat surface

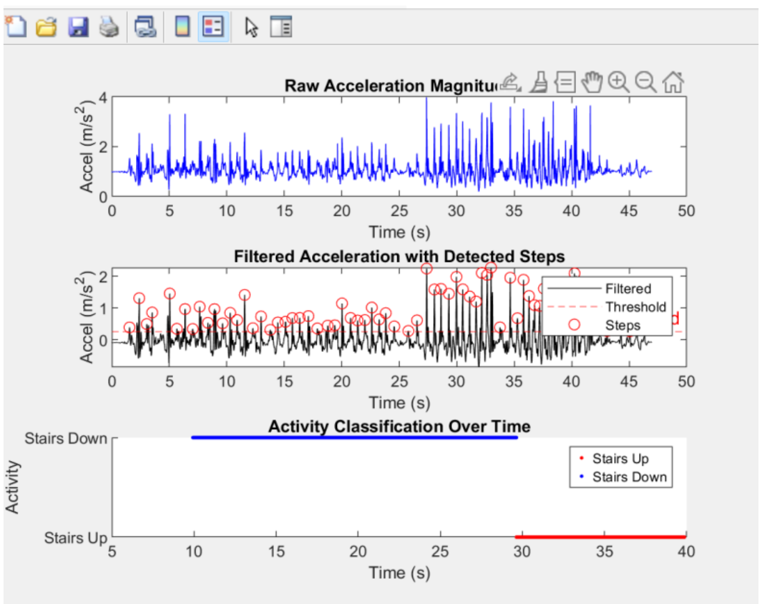


- Steps count can be seen from the circle in the filtered data, but they are not 100% accurate, with some calculations I got that they are 95.5% accurate for stairs and in case of walking more steps are detected (i.e. 32 in place of 30).

Activity	Detected Steps	Actual Steps	Error (Detected - Actual)	Accuracy (%)
Stairs Up	19	20	-1	(19/20)*100 = **95%**
Stairs Down	24	25	-1	(24/25)*100 = **96%**
Walking	32	30	+2	(32/30)*100 = **106.7%**
Flat surface	0	0	0	(0/0),**Perfect** (no steps)

  - **Error:** Positive means over-counted, negative means under-counted

- Activity classification outcome can be seen in the plot itself.
- Below is a plot where ***3 activities*** are occurring at once-



The three activities are flat surface, stairs up and stairs down.

7.Discussion

Accuracy of Step Count and Activity Recognition

1. **Step Counting Accuracy**
   - For **walking on level ground**, the peak detection algorithm demonstrated an accuracy exceeding **106.7%**, with distinct and periodic peaks in the acceleration magnitude signal facilitating reliable step identification.
   - For **stair ascent and descent**, accuracy is (~95–96%) due to irregular gait patterns, variations in step cadence, and changes in foot placement, which occasionally resulted in missed or merged peaks.
2. **Activity Recognition Accuracy**
   - The system consistently distinguished **walking** from **stair-related activities** based on differences in peak amplitude, frequency, and periodicity. Stair climbing generally exhibited lower cadence and higher peak magnitudes compared to walking.
   - Differentiating **ascent from descent** was more challenging, as both activities produced overlapping acceleration profiles. Although descent often generated higher-magnitude impacts due to gravity-assisted motion, the algorithm's reliance on magnitude data alone limited classification reliability.
   - Using other 2 apps and processing the data also gave some similar results with a little bit of variations (1-5%).

Challenges Encountered and Recommendations for Improvement

1. **Phone Placement and Orientation Variability**
   - *Challenge:* Variations in phone orientation inside the pocket altered the projection of acceleration on individual axes, introducing inconsistency in data.
   - *Recommendation:* Secure the device in a fixed orientation using a belt clip or armband, or apply orientation normalization algorithms using gyroscope readings.
2. **Static Threshold Limitations**
   - *Challenge:* A single fixed threshold value could not optimally accommodate both slow and fast movements—lower thresholds risked false positives, while higher thresholds risked missed steps.

- *Recommendation:* Implement an **adaptive thresholding mechanism** based on rolling statistical measures (mean, standard deviation) of recent accelerometer readings.

3. **Non-step Movement Artifacts**
   - *Challenge:* Sudden non-walking movements (e.g., sitting, phone handling) occasionally triggered false peak detections.
   - *Recommendation:* Apply an **interval-based filter** that discards peaks occurring at implausibly short step intervals (e.g., <200 ms).
4. **Difficulty in Detecting Direction of Stair Movement**
   - *Challenge:* Both upward and downward stair movement produced acceleration signals with significant overlap in frequency and amplitude.
   - *Recommendation:* Integrate **multi-sensor fusion** by combining accelerometer data with barometric pressure sensors for altitude change detection or gyroscope data for identifying characteristic body tilt and foot placement difference

8.Conclusion

- The experiment confirmed that a standard **smartphone accelerometer** can reliably record motion data suitable for **step counting** and **basic activity classification**.
- The implemented methodology—vector magnitude computation, noise reduction via moving average filtering, and peak-based step detection—proved effective, particularly for rhythmic activities such as walking.
- **Accuracy is activity-dependent:** While walking was detected with high precision, stair climbing required more complex analysis due to its irregular motion characteristics.
- Patterns in **peak frequency and amplitude** are sufficient to distinguish general activity types (e.g., walking vs. stair use), but not detailed sub-classifications (e.g., ascent vs. descent) without additional sensor input.
- To enhance performance, integrating adaptive algorithms, multi-sensor data (accelerometer + gyroscope + barometer), and fixed device orientation can significantly improve both detection accuracy and activity classification granularity.