Activity detection and localization using KNN

Code used: KNN Classifier

Group 9 - Xiaomi Redmi 5A (Android 7.1.2, API 25)

Sonnya Dellarosa
4783344

Pranjal Rajput
4788087

I. Introduction

In this assignment, we have to develop an Android application that provides the location and activity information of the user. The application shows the user where they are among four different cells and also tells whether they are walking or standing still.

II. Method

In this assignment, we used KNN classifiers for both activity recognition and localization. For activity recognition, after collecting the samples and analyzing the data, we saw that using the minimum and maximum value of the accelerometer within a window might not provide us with an accurate classifier. Then, we visualized the data and saw that the difference between the maximum and minimum values are clustered together for both walking and standing still. Thus, we decided to use a three-dimensional KNN classifier with the max-min difference of the X, Y, and Z-axis accelerometer sensor values. The k-value used is 3, as it seems to output good results and the distance measured is Euclidean.

For localization, the feature used is the number of times a Wi-Fi station is detected within a sample window. For each cell, we collected the samples and chose the Wi-Fi station that appears the most and is the strongest in that cell. We were able to do this as the cells that we have chosen are not too small. The KNN classifier is a four-dimensional classifier where the variables are the frequency of appearance of a particular Wi-Fi station related to each cell. We decided to use only one Wi-Fi station per cell as we would like to limit the dimension of the KNN classifier, considering it suffers from the curse of dimensionality. The k-value used is also 3 and Euclidean distance is again used for distance measurements. Both k-values are chosen as 3 as it is suitable with the size of the training samples.

When the application is running, the accelerometer sensor values are constantly checked as their values change, while the Wi-Fi scan is only performed when a movement is detected, i.e. when the user is recognized as walking.

III. Evaluation Setup

A. Training method

For the training part of activity recognition, we collected 1000 samples of the X, Y, and Z-axis values of the accelerometer sensor. Using a window of size 20, we find the maximum and minimum value in the window and calculate the difference for each axis.

For the training part, we chose 4 cells of size approximately $20m \times 20m$ at first floor of the EWI building. We performed a Wi-Fi scan every 100 ms for a total of 20 seconds, providing us with a total of 200 samples. We decided to give a delay of 100 ms as a Wi-Fi scan tends to be an expensive process.

B. Testing method

We tested our app in 4 cells that were chosen for training. Most of the times, the app successfully detected the correct cell in which the user is present. Also, the user activity is for the most part successfully recognized and is displayed as ”Standing” if user is standing still or ”Walking”, if user is moving.

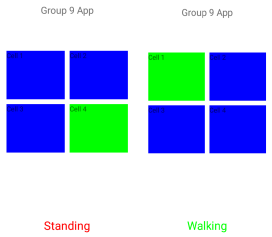

Fig. 1. App screenshots

IV. Analysis of results

As the KNN classifier was implemented using Weka, specifically the stripped version that is suitable for Android applications, a function is available that evaluates the performance of the model through the use of a confusion matrix. The matrices for both activity recognition and localization are shown in equations (1) and (2).

	$Walk$	$Still$
$Walk$	30	0
$Still$	0	19

(1)

	$Cell1$	$Cell2$	$Cell3$	$Cell4$
$Cell1$	20	0	0	0
$Cell2$	1	19	0	0
$Cell3$	0	1	16	3
$Cell4$	0	0	1	19

(2)