

Bewegingsherkenning met een smartphone

Arne De Brabandere

arne.debrabandere@student.kuleuven.be

Menno Keustermans

menno.keustermans@student.kuleuven.be

Abstract

1 Inleiding

2 Afzonderlijke activiteiten

Het eerste probleem is om van een gegeven reeks samples van de accelerometer en gyroscoop van een smartphone de activiteit van een persoon te bepalen. We veronderstellen hier dat telkens één afzonderlijke activiteit gemeten wordt.

We willen tien verschillende activiteiten kunnen herkennen:

- wandelen,
- lopen,
- fietsen,
- een trap opwandelen,
- een trap afwandelen,
- springen,
- niets doen (zitten, liggen, staan),
- een lift naar boven nemen,
- een lift naar beneden nemen,
- tanden poetsen.

In bovenstaande lijst hebben we tanden poetsen als moeilijke activiteit toegevoegd. De beweging lijkt sterk op niets doen en zal waarschijnlijk minder goed te herkennen zijn. Ook voor lift naar boven en trap opwandelen bestaan er gelijkaardige bewegingen, respectievelijk lift naar beneden en trap afwandelen.

Het proces om een afzonderlijke activiteiten te herkennen verloopt in drie stappen. Ten eerste moeten er gegevens verzameld worden. Als tweede stap worden er features berekend op de verzamelde gegevens. Deze zijn nodig om tenslotte modellen te leren met behulp van classificatiemethodes. Als criterium om de verschillende modellen te vergelijken, gebruiken we de accuraatheid als percentage van het aantal juiste geclassificeerde samples ten op zichte van het totaal aantal samples met behulp van cross-validatie.

2.1 Gegevensverzameling

Alle gegevens werd opgemeten door MotionTracker tool . Dit is een Android-applicatie die de versnelling en rotatie (respectievelijk gemeten door de accelerometer en gyroscoop van de smartphone) aan 50 Hertz. Als uitvoer geeft de applicatie een .log-bestand met de gemeten versnellingen (in de x-as, y-as en z-as met de z-as evenwijdig met de gravitatie) en rotaties (in quaternion notatie) met bijhorende timestamps.

Voor elke meting werd de applicatie gestart alvorens de smartphone in de broekzak gestopt werd en gestopt na het uithalen. Waardoor er bij het begin en einde van elke meting altijd enkele seconden niet-activiteit bevat. Om zo het fout labelen te vermijden, werd na elke meting een stuk van de start en het einde van het .log-bestand weggeknipt. Zodat elke meting exact één activiteit bevat van vier á twintig seconden.

We hebben voor elke activiteit 22 metingen verzameld, opgemeten door twee verschillende personen. Om voldoende variatie te hebben, gebeurden de metingen op verschillende dagen. Ook hebben we ervoor gezorgd dat we niet telkens dezelfde broek droegen, aangezien de gemeten versnelling kan variëren in verschillende broekzakken. Na elke meting werd het uitgevoerde .log-bestand geknipt en gelabeld met de juiste activiteit.

2.2 Dataverwerking: features berekenen

Voor we classificatiemethodes kunnen gebruiken, moeten we eerst features berekenen. Dit zijn parameters die we uit de samples van de accelerometer en gyroscoop kunnen halen. Om de verschillende features te berekenen, maakten we gebruik van MotionFingerPrint.jar tool

MotionFingerPrint berekent in tool 134 features verdeeld onder vier soorten:

- *Statistische features:* dit zijn gemiddelde, standaardafwijking van zowel z- als xy-versnellingen en vermogen. En correlatie tussen z- en xy-versnelling. In het algemeen kunnen deze tegen een lage kost berekend worden.
- *Fourier-transformatie:* deze worden berekend in het frequentie domein van de metingen, zoals amplitudes volgens de verschillende assen.
- *Wavelet-transformatie:*
- *Hidden Markov models:* log-likelihood voor het model van elke activiteit

Het resultaat van de features berekening geeft een voor elke sample een set van parameters. Elke set vormt een instantie van de training set om een model te leren.

2.3 Classificatiemethodes

We gebruiken classificatiemethodes om een model te zoeken om afzonderlijke activiteiten te herkennen. We vergelijken enkele veel voorkomende methodes: beslissingsbomen, Random Forest, k-Nearest Neighbours, Naive Bayes en Support Vector Machines. De methodes leren telkens een model uit de instanties met de hierboven beschreven features voor de verschillende activiteiten. Hieronder wordt kort de werking van de vernoemde methodes beschreven:

- *Beslissingsbomen* zijn bomen waarvan de interne knopen features voorstellen en de bladknopen één (of meerdere) labels bevatten. Een tak in de boom stelt een test voor op de feature van de knoop waaruit de tak vertrekt. Om een instantie te classificeren worden de juiste takken gevolgd tot in een bepaalde bladknoop.

2.4 Experimenten en resultaten

We evalueren de methodes met 10-fold cross-validatie. Hierbij worden de instanties op 10 verschillende manieren opgesplitst zodat telkens 90% van de instanties als trainingset wordt gebruikt om een model uit te leren. Op de overige 10% wordt het model geëvalueerd. De accuraatheid van elke methode wordt dan berekend als het gemiddelde percentage van correct geclassificeerde instanties in de verschillende testsets.

In figuur X wordt de accuraatheid van de verschillende methodes vergeleken. We zien dat Random Forest de hoogste accuraatheid geeft voor onze metingen. Ook beslissingsbomen (J48) geeft een goede accuraatheid, net zoals Naive Bayes. De hogere accuraatheid van Random Forest ten opzichte van beslissingsbomen is waarschijnlijk te verklaren door het feit dat Random Forest minder kans op overfitting heeft.

3 Sequenties van activiteiten

3.1 Datacollectie

3.2 Dataverwerking

3.3 Algoritme

3.4 Experimenten en resultaten

4 Conclusie

5 Verder werk