

CenceMe(Lite) Dataset Explanation

Below there is a description of the data collected with the phones using the CenceMeLite application, where CenceMeLite is a subversion of CenceMe able to sense and store raw accelerometer data, accelerometer facts, and GPS positions.

The files are called “*CenceMeLiteLogXX.txt*” where XX is the user number.

Be careful:

The sizes of the files are pretty big (up to 500Mb one), that’s the reason why they are stored into compressed archives. The archives names are “*CenceMeRawDataX.rar*”, where X indicates the number of the archive. In each archive there are 4-5 data files.

Users:

Users 1,3 are undergraduate students.

Users 2,8,9,10,16,19 are graduate students from the Sensorlab.

Users 4,12,13,15,17 are graduate students from biology departments.

Users 6,11,14,20 are other CS graduate students.

Users 18,5,7 are employees and professors.

“Bad-behaved” users:

Because users were not restricted in any way, some users did not collect very good data (especially in terms of GPS samples). The first “bad” user is number 20: from the size of the traces (about 20 Mb) it is obvious the user did not use the phone at all. Another bad user is number 18, but just in terms of GPS traces (not a lot). Other users travelled a lot during the experiment period so you can see GPS user traces covering different countries such as United States, Canada, and countries in Europe.

File Format:

In the files there are both information lines (about system configuration) and data lines. Information lines are written when the system is started. It is possible to distinguish them using the keyword INFO or DATA.

The only exception is when the system starts. A preamble is written into the files: the preamble consists of three lines starting with a line symbol. An example:

```
-----  
----- NEXT LOG -----  
-----  
1217004653781 INFO (3) - CenceMeLite: Configuration:  
1217004653903 INFO (3) - CenceMeLite: CONFIG: 0 * 2 * 1 * 1 *  
socket://127.0.0.1:8100 * 0 * 0 * 0 * built-in accelerometer in N95  
1217004653919 INFO (3) - CenceMeLite: CONFIG: 1 * 501 * 2 * 1 *  
socket://127.0.0.1:8101 * 0 * 0 * 0 * Symbian-based sound  
1217004653925 INFO (3) - CenceMeLite: CONFIG: 2 * 3 * 1 * 1 * void * 180000 *  
10000 * 0 * built-in GPS in N95
```

...

Data lines can be distinguished by the keywords: ACC, ACT, GPS.

ACC lines contain the accelerometer raw data and are in the format:

Timestamp DATA (0) - ACC: Xacc,Yacc,Zacc*Xacc,Yacc,Zacc*.....

where:

- Timestamp is the time when the line has been written into the log file.
- Xacc, Yacc, Zacc are respectively the accelerations on the three axes. Every accelerometer sample is separated by the symbol *.

The data set also contains information about inferred activities in the format:

Timestamp DATA (0) - ACT: AccSamplingStart,AccSamplingEnd,Fact

where:

- Timestamp is the time when the line has been written into the log file.
- AccSamplingStart is the time the accelerometer starts the sampling.
- AccSamplingEnd is the time the accelerometer starts the sampling.

The codes of the different facts ("Fact" field) are as following:

Sitting: 0

Running: 1

Walking: 2

Standing: 5

"AccSamplingStart" and "AccSamplingEnd" are the start and end times of the interval during which the accelerometer data used for classification of that particular activity were collected.

GPS lines could be tricky to process because there are 3 possible cases:

A GPS line can contain:

- No samples
- N samples
- the string "GPS-Skipped: user sitting".

The latter case was designed for power issues: every time a user is sitting for more than 15 minutes the GPS sensor is not used. These lines can be replaced using the last known position.

GPS lines containing samples can contain up to N samples about the same measure.

Format:

Timestamp DATA (0) - GPS:

altitude,latitude,longitude,hdop,speed*altitude,latitude,longitude
,hdop,speed*

Even in this case every sample is separated by the symbol *.

Experiment start and duration:

Because the phones have been given to the users at different times please use the timestamps to figure out the exact time intervals.

As lower and upper bound dates we used:

Start date = 2008-7-28 00.00.00

End date = 2008-8-11 23.59.59

Between these 2 dates every user was certainly in possession of a phone.

Note: all the timestamps and time indications are in Java time format, which is the time expressed from 1970 in milliseconds!