



<http://www.diva-portal.org>

Preprint

This is the submitted version of a paper presented at *Mobile Apps and Sensors in Surveys (MASS) Workshop, 4-5 March, Mannheim, Germany.*

Citation for the original published paper:

Palmberg, R., Susilo, Y., Gidofalvi, G. (2019)

Developing and trialling an implicit interaction platform to monitor and aiding dementia travellers

In:

N.B. When citing this work, cite the original published paper.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-254187>

Developing and trialling an implicit interaction platform to monitor and aiding dementia travellers



Figure 1 - This figure shows the heart rate of a young man, using the bus to travel along Odengatan in Stockholm

Palmberg, R.C.O.¹, Susilo, Y.O.², Gidofalvi, G.²

¹robinpa@kth.se, Department of Urban Planning and Environment, KTH Royal Institute of Technology, Sweden

²Department of Urban Planning and Environment & Integrated Transport Research Lab, KTH Royal Institute of Technology, Sweden

Introduction

Age related cognitive diseases are becoming a growing problem in Sweden. With the fast ageing population and lowered mortality rate comes the spread of cognitive diseases related to dementia. In order to accommodate this growing target group in transport and the built environment, it is important to understand the mobility and travel behaviour of patients suffering from these diseases. One subset of this target group is travellers suffering from age induced illnesses related with dementia, which most often have fluctuating symptoms that are affecting the cognitive skills of the traveller. This makes it hard to use standardized forms and survey-based information that would require the traveller to actively respond retroactively, either in oral or written form, since the traveller might have forgotten or mixed up their past experiences, among other things, it becomes very hard to gain confidence in the results as it might be hard to tell in which condition the patient is during the collection.

We propose an automated collection of biometric data such as heart rate in combination with position. Since the validity of the information collected in this manner is directly

related to the quality of the sensors used it means that the precision and accuracy of the results could be virtually endlessly improved by upgrading the hardware and optimizing the software. To take a first step towards a solution like this we have started developing a smart watch application which is utilizing PPG technology to collect heart rate and combine it with positions collected through GPS technology.

Early testing has shown the possibility to correlate the heart rate of a traveller to their specific location. The implications of this must be validated through data labelling as we wish to utilize machine learning algorithms to analyse the data collected.

Aim of the project

The aim is to create a platform for the collection of biometric data in relation to position and validate it automatically by cross-referencing data from different sources. The purpose of this would be faceted, with several application areas. Some of the most immediate areas are: 1) to uncover how the built environment and transportation elements affect the travellers suffering from dementia related diseases, 2) to better understand the progression of conditions that the travellers suffering from dementia related diseases might experience, such as confusion, stress, etc. 3) to be able to aid the traveller in real time, both virtually and in reality by providing digital and physical support by knowing biometric data patterns related to critical conditions.

Design (of) Method

It might be hard for the travellers that are suffering from diseases related to dementia to perform surveys in an acceptable and accurate manner. In order to bypass this, the design of the collection needs to be made with implicit interaction in mind. What this means is that the user of the collection software, in this case the travellers, must be able carry out their everyday tasks without being burdened with additional tasks such as pressing virtual buttons in a digital survey in order to collect or not collect information. So just as the step counter in common phones are designed, our software would collect data on its own, and (unlike the step counter) transmit it to the researchers that will utilize the data for analysis.

Hardware

As this solution should be possible to scale up indefinitely, we decided to utilize common hardware, as opposed to build it from scratch. We looked for a smart watch-like device with an operating system that would be easily accessible to work with. Other requirements were that it had to have sensors making it possible to collect biometric data and position, as well as good communication possibilities on its own and not require the constant connection to another device to be useful.

After long consideration we found the Huawei Watch 2 LTE to fit our needs. It had the type of sensors we were looking for as well as standalone connection to the internet through a built in 4G modem. The specifications are as follows.

Operating System	Wear OS 2
Sensors	PPG, GPS + Glonass, 6-axis A+G Sensor, 3-axis Compass, Barometer, Capacitive Sensor, Light Sensor
Communication	4G LTE, BLE, NFC (RFID), UMTS, TD-SCDMA, GSM
Screen	1,2-inch Circular AMOLED with 326 PPI
Battery	410 mAh

Table 2 – The specifications of the Huawei Watch 2 LTE, used in this project.

As is clearly shown, it has a wide range of sensors in combination with a wide range of communication options. This makes it possible to use, once initial setup is completed, without any need of other devices. Since we are working with devices that needs to be attached to the user, we were aiming for as few as possible in order not to complicate the process too much.

The PPG sensor will be the main source for biometric data as PPG technologies allow for measurement of heart rate as well as blood pressure without physical contact. This makes it possible to collect information from this source, even if the watch might not be fitted in the exact same way every time it is put on. Huawei provides the heart rate from the PPG sensor as default, but further investigation will be made on the possibilities to utilize the raw data from that sensor to collect blood pressure as well.

One factor that limits the possibilities of data collection utilizing the Watch 2 is its battery which is relatively small in comparison with e.g. a smartphone. According to Huawei, this battery should be enough for two days of use without charging, but that is of course based on what kind of apps and services that are being used.

Software

The task of the software at this stage is to read sensor data, package it and record it to the external storage built in to the Watch 2 in a way which makes it possible to utilize it for analysis in a later stage.

ENTRUST V0.01

In the absolute first iteration, the goal was to make it possible to read the PPG data. Wear OS has a built-in function for translating the readings into heart rate, and this function was utilized for the first iteration. The sensor delay was set to the fastest which is defined by Google and made it possible to get constant real time information about the heart rate.

The possibility to display current device time was also added in order to be able to time stamp the information when stored on the device in a later stage.

Entrust V0.02

In the second iteration the goal was to add location utilizing the GPS and Glonass positioning systems that are available with the current hardware. Just as with the PPG data, this information was packaged in an easy-to-use manner by Google. The sensor was set to use either fine or coarse location, depending on availability, as to improve accuracy of location while maintaining consistent collection.

ENTRUST V0.1

At this stage, the software could locate the user, read its heart rate and time stamp it. So, for this iteration, the storing of information was added. This meant that we now had a first working prototype which collected all essential information and stored it to the device. A very simple UI was used in order to monitor the status of the sensors.



Figure 2 - The first three pictures show the progression within the app and the fourth show the results from the collection

As can be seen, to the first picture is the initial state of the software when it is loaded and on the second is the state which is monitoring the sensors. In order to monitor when information was collected, a simple popup was added which would show when data was being stored. The data was saved in a comma separated file, for easy usage later, as can be seen in the fourth picture.

The software was set to save any and all data whenever a sensor would change state, e.g. when the heart rate was updated. A trial run was made with these settings.



Figure 3 - Visualization of a segment of the data from the first run

The result were accurate recordings of position with clear changes in heart rate of the test subject. The first trial recorded information for 1 hour, 10 minutes and 21 seconds, measured from the first to last record. During this time, 5409 samples were collected and 42% of the battery was used. This means that the sample rate on average was about

$5409/4221=1,28$ samples per second. A rough estimation, assuming that the battery was mainly drained because of the software gives a run time of 2 hours and 45 minutes before the watch turns off. This is also assuming that all power save modes are turned off, as we do not know their effect on the data collection. This meant that it would not be a feasible solution, as our target user should be wearing the watch with the application running for at least a full day. We set our new goal to 30 hours and did a new iteration.

ENTRUST V0.2

In this iteration, the possibility to control the sample rate was introduced, together with the possibility to change sample rate from within the app, to make testing easier. A timer within the script would turn off all the sensors and recording of data for X second(s), then turn on the sensors, wait for Y seconds collect and store the data, then turn of the sensors and wait for X-Y seconds before the next run of the same functions. The reasons why the script have to wait Y seconds before collecting is that the sensors have some boot time that needs to be taken into consideration. For this iteration, Y was set to 30 seconds.

Another test was carried out with a sample rate of 1 sample per minute by setting X=60, making it possible for the watch battery to last for 30 hours. These findings seemed perfect at first glance since they met the goal of 30 hours, but when further analysed showed some reliability issues with the app. During this time, the app should have collected 1800 samples, but in reality, only collected about 130 samples. Further investigation showed that this was because of the ambient mode of the watch, something that had been used as a battery saver, but scripted incorrectly. This led to an investigation if it would be possible to utilize the ambient mode to collect information on a time basis.



Figure 4 - The first picture shows the last state of V0.2. The second picture shows V0.3 and the last picture shows V0.3 when running in ambient mode

ENTRUST V0.3

According to the documentation of Wear OS, ambient mode has a refresh rate of 1 update per minute. This means that between those refreshes, the watch is put to sleep, nothing is being utilized at all. But during the refresh, the watch wakes up and checks if there is anything that is supposed to be done. An initial idea was to utilize this refresh rate as the sample rate and run the collection every time it refreshes. This was implemented in the code and another trial was made.



Figure 5 - Visualization of the first data collected with V0.3

As you can see from this visualization, the software collected data, but not consistently. But after further investigation, the bugs were fixed, and a final run could be made to show the correlation between heart rate and position at a fixed sample rate. This iteration did not turn off the sensors in between collections, instead the sensor delay for the PPG sensor was set to normal, which made it so that the watch could run for 9 hours, resulting in 540 samples.

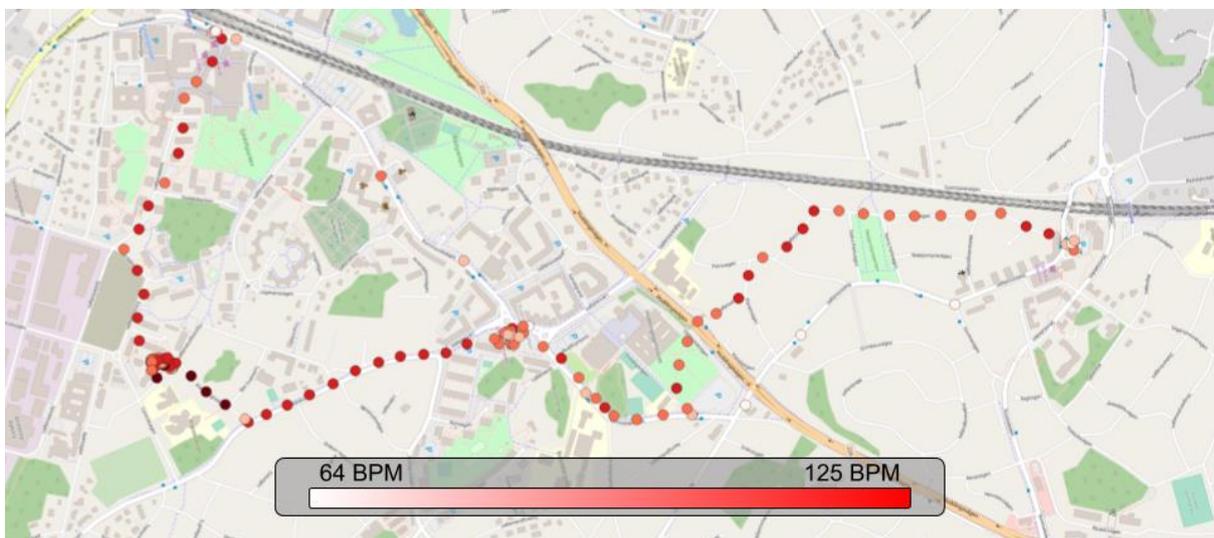


Figure 6 - Visualization of data collected with the fully functioning V0.3

A new idea for version V1.0 is to have one sample every other minute, switching the status of the sensors on and off at every other ambient update from the watch and collecting while the sensor is on, just before turning it off. This should result in a much longer battery life.

Position- vs time-based collection

Often when making automated travel diaries, which are closely linked with this, it is usable to collect data in a position-based manner, for example every 50 meter. But since we are looking for the affects of the built environment in general, we are more interested in the change of biometric data over time as biometric data can be collected when the user has just arrived at a park and have a high heart rate because of walking. If the collection would have been position-based, then the software would not have collected any more data until the person moved, which would give a rather high average heart rate of that position in comparison to when using time-based collection which would make it possible to collect several samples in the same place and (probably) lowering the average heart rate of a position.

Privacy concerns

As the data collected in this software is highly personal, each user has to give permission to the application before the software can start to collect the information. In the final tests utilizing this platform, the user will have full knowledge of how the data is used and why, as well as having the possibility to withdraw any and/all data they wish to destroy which is linked to their user number.

Real-time versus post collection

At this point, all the analysis and visualizations are done post collection as the information is stored on the watch itself and needs to be transferred with the watch being physically close to the computer that will utilize its information. This is also beneficial from a privacy concern viewpoint as it ensures that the data will not be hacked when transferred to server. This makes it possible to optimize the sampling and collection process before having to deal with internet security related to transferring the information.

Once the software for collecting the data is finalized, the information will be transferred in real-time to be able to work as a real-time aid, since our aim is to analyse the information server side and find anomalies in the data in order to find conditions and situations in which it might be useful to aid the user in some way, for example by sending a caregiver to the location of the user if they seem to have big issues. In theory it should be possible to detect critical conditions through machine learning, which in return might save lives. In order to find out if this would work, we need to continue developing this app and try it in reality.

Continued work

Further optimization and updating need to be conducted before this would be a reliable and feasible solution for monitoring and aiding dementia travellers. But we can now, with more certainty than before, confirm that there is a possibility to collect biometric and position data in order to visualize the correlation between position and heart rate.