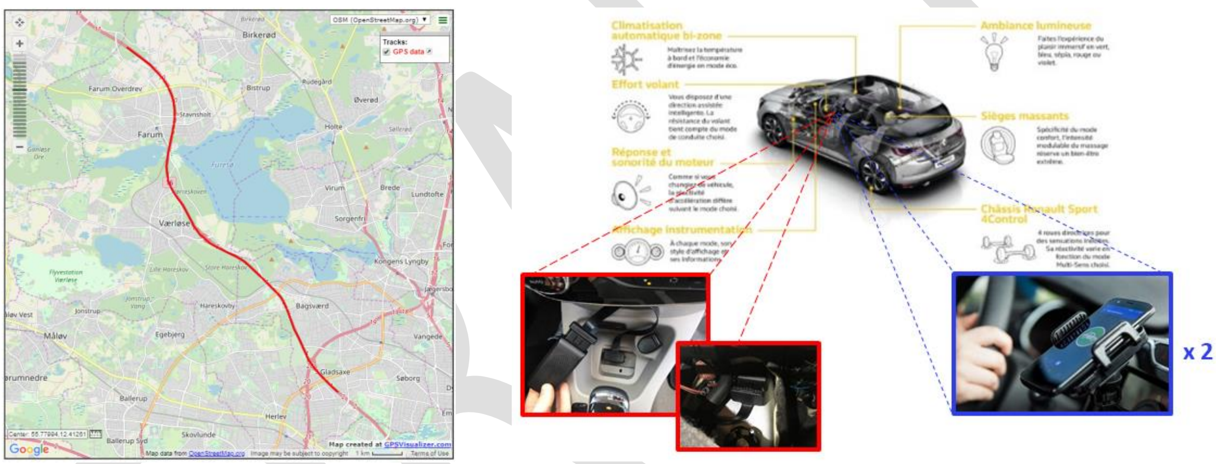
Cardata validation – Metadata, data quality and limitations

1 Introduction

The purpose of this document is to establish a basis for the initial phase of task "T2.1 – Training of the first hybrid model" and milestone "M2.1 - Usability of car sensors data". The study presented focuses on metadata, data quality and the current limitations and usefulness of data.

The cardata presented was collected on 28 May 2019, on a 14 km long section of Highway M13, northwest of Copenhagen (see Figure 1a). The test conditions was dry and the air temperature app. 20 °C. The dataset consists of six passes with a GreenMobility vehicle of type Renault Zoe, driving in the right lane, in both directions; three of the passes were carried out with a single driver and two passengers, and three passes with a single driver only. In total app. 120 km of road was covered during the test day.



(a) (b)

Figure 1. *(a) Overview of road section on Highway M13 marked with a red line and (b) location of external car sensors: AutoPi (red box) and smartphones (blue box).*

The AutoPi was located in the center of the car, close to the gearbox, as shown in Figure 1b. Moreover, one smartphone was placed in windshield in the center of the car (i.e., close to the AutoPi), and one smartphone close to the right front wheel.

The AutoPi collects cardata (in-vehicle sensor readings) from the Can bus transmitted via the ODBII-port (see Appendix A), as well as GPS and three-axis accelerometer data from sensors in the device. The smartphones collected independent three-axis accelerometer and three-axis rotation measurements utilizing the SmartRoadSense application [1].

Table 1 shows 6 rows and the first 10 out of 63 columns of a typical dataset from the AutoPi device. The column names are logical, e.g., AutoPi accelerometer data in z-direction at a given timestamp is "acc.xyz.z" and car yaw rate is "obd.acc_yaw.value". Group names are also found in row "@t"; in general we have "acc"=accelerometer data from the AutoPi, "rpi"=raspberry pi (i.e., AutoPi) information, "event"=event message from AutoPi and "odb"=cardata from ODB-port.

Table 1. Extract of 6 rows and first 10 out of 63 columns of cardata in csv format.

@rec	@t	@ts	@uid	@vid	acc.xyz.x	acc.xyz.y	acc.xyz.z	obd.acc_long.value	obd.acc_trans.value	obd.acc_yaw.value
2019-05-28T11:02:45.209472Z	acc.xyz	2019-05-28T09:05:33.004356Z	f8d815fe-ef73-6aad-063e-492ba05a6fd4	36	0.95	-0.19	-0.11			
2019-05-28T11:02:45.209486Z	acc.xyz	2019-05-28T09:05:33.024356Z	f8d815fe-ef73-6aad-063e-492ba05a6fd4	36	0.95	-0.19	-0.13			
2019-05-28T11:02:45.209500Z	acc.xyz	2019-05-28T09:05:33.044356Z	f8d815fe-ef73-6aad-063e-492ba05a6fd4	36	0.94	-0.2	-0.16			
2019-05-28T11:03:31.150481Z	obd.acc_long	2019-05-28T09:05:33.663505Z	f8d815fe-ef73-6aad-063e-492ba05a6fd4	36				390		
2019-05-28T11:03:31.150438Z	obd.acc_yaw	2019-05-28T09:05:33.663505Z	f8d815fe-ef73-6aad-063e-492ba05a6fd4	36						2246.1
2019-05-28T11:03:31.150502Z	obd.acc_trans	2019-05-28T09:05:33.663505Z	f8d815fe-ef73-6aad-063e-492ba05a6fd4	36					65514	

For this initial analysis, cardata was synchronized and georeferenced using the AutoPi timestamp (column "@ts") and GPS data with linear interpolation. Further, alignment of data between individual trips was based on GPS distances (see Appendix B), assuming that the car was driving on the exact same straight line for all car trips.

2 Metadata

Currently the CanZE application [2] is utilized to read Can bus data from the ODBII-port. The CanZE application includes a protocol for reading up to 378 individual sensors from Renault electrical cars. However, some of the data is not applicable/have not been reversed engineered for the Zoe car.

Overview of all the parameters currently collected, their units and conversion factors are given in Table 2. For the present study, the list was reduced to the app. 50 most relevant parameters. The table also shows typical sensor readings during the test ("Typical values"), if these values seems reasonable/physical ("Expected") and if the data have been validated using independent external sensors ("Validated").

Table 2. *CanZE protocol of most relevant car sensor data showing units and conversion factors as well as measured and expected values.*

ID (hex)	elements	startBit	endBit	resolution	offset	decimals	unit	options (hex)	Can ZE name	AutoPi name	Typical values	Expected	Validated
0c6	1	0	15	1	32768	1	°	ff	Steering Position	obd.strg_pos	65500	No	No
0c6	2	16	31	1	32768	1	°/s	ff	Steering Acceleration	obd.strg_acc	65500	No	No
0c6	3	32	47	1	32768	1	°	ff	SteeringWheelAngle_Offset	obd.strg_ang	65500	No	No
12e	1	0	7	1	198	0	#N/A	ff	LongitudinalAccelerationProc	obd.acc_long	400	No	No
12e	2	8	23	1	32768	0	#N/A	ff	TransversalAcceleration	obd.acc_trans	65500	No	No
12e	3	24	35	0.1	2047	1	°/s	ff	Yaw rate	obd.acc_yaw	2250	No	No
130	1	20	31	1	4094	0	Nm	ff	ElecBrakeWheelsTorqueRequest	obd.brk_trq_req_elec	8188	?	No
130	2	44	55	-3	4094	0	Nm	ff	DriverBrakeWheelTq_Req	obd.brk_trq_req_dvr	-8188	?	No
17a	1	48	63	0.5	12800	1	Nm	ff	Estimated Wheel Torque	obd.whl_trq_est	19000	?	No
186	1	0	15	0.125	0	2	rpm	ff	Engine RPM	obd.rpm	6000-8000	Yes	No
186	2	16	27	0.5	800	1	Nm	ff	MeanEffectiveTorque	obd.trq_eff	1200	No	No
186	3	28	39	0.5	800	0	Nm	ff	RequestedTorqueAfterProc	obd.trq_req	1200	No	No
1f8	1	16	27	1	4096	0	Nm	ff	TotalPotentialResistiveWheelsTorque	obd.whl_trq_pot_ri	6900	?	No
1f8	2	28	39	-1	4096	0	Nm	ff	ElecBrakeWheelsTorqueApplied	obd.brk_trq_elec	2	?	No
1f8	3	40	50	10	0	0	Rpm	ff	ElecEngineRPM	obd.rpm_elec	6000-8000	Yes	No
242	1	16	27	0.5	800	1	Nm	ff	ASR Dynamic Torque Request	obd.asr_trq_req_dyn	2847	?	No
242	2	28	39	0.5	800	1	Nm	ff	ASR Static Torque Request	obd.asr_trq_req_st	2847	?	No
242	3	40	51	0.5	800	1	Nm	ff	MSR Torque Request	obd.msr_trq_req	800	?	No
29a	1	0	15	0.041666667	0	2	rpm	ff	Rpm Front Right	obd.rpm_fr	0-800	Yes	No
29a	2	16	31	0.041666667	0	2	rpm	ff	Rpm Front Left	obd.rpm_fl	0-800	Yes	No
29a	3	32	47	0.01	0	2	km/h	ff	Vehicle Speed	obd.spd_veh	0-100	Yes	Yes
29c	1	0	15	0.041666667	0	2	rpm	ff	Rpm Rear Right	obd.rpm_rr	0-800	Yes	No
29c	2	16	31	0.041666667	0	2	rpm	ff	Rpm Rear Left	obd.rpm_rl	0-800	Yes	No
534	1	32	40	1	40	0	°C	5	Temp out	#N/A	#N/A	#N/A	No
5d7	1	0	15	0.01	0	2	km/h	ff	Speed	obd.spd	0-100	Yes	Yes
5d7	2	16	43	0.01	0	2	km	ff	Odometer	obd.odo	16000	Yes	Yes
5d7	3	50	54	0.04	0	2	cm	ff	Fine distance	obd.f_dist	0.25-0.6	?	No
5ee	1	8	10	1	0	0		ff	Front Wiping Request	obd.ww_f_req	0	Yes	Yes

ID (hex)	elements	startBit	endBit	resolution	offset	decimals	unit	options (hex)	Can ZE name	AutoPi name	Typical values	Expected	Validated
638	1	0	7	1	80	0	kW	ff	Traction Instant Consumption	obd.trac_cons	175	?	No
646	1	6	15	0.1	0	1	kWh/100km	ff	Average trip B consumpion	obd.trip_cons_avg	16	?	No
646	2	16	32	0.1	0	1	km	ff	Trip B distance	obd.trip_dist	13	Yes	Yes
646	3	33	47	0.1	0	1	kWh	ff	trip B consumption	obd.trip_cons	60-80	?	No
646	4	48	59	0.1	0	1	km/h	ff	Average trip B speed	obd.trip_spd_avg	25	Yes	No
653	1	8	9	1	0	0		ff	Driver Safety Belt Reminder	obd.sb_rem_fl	0-1	Yes	No
653	2	10	11	1	0	0		ff	Front Passenger Safety Belt Reminder	obd.sb_rem_fr	0-1	Yes	No
653	3	14	15	1	0	0		ff	Second Row Center Safety Belt State	obd.sb_stat_rc	0-1	Yes	No
653	4	16	17	1	0	0		ff	Second Row Left Safety Belt State	obd.sb_stat_rl	0-1	Yes	No
653	5	18	19	1	0	0		ff	Second Row Right Safety Belt State	obd.sb_stat_rr	0-1	Yes	No
654	1	52	61	0.1	0	1	kWh/100km	ff	Average Consumption	obd.cons_avg	16.8	?	No
656	1	48	55	1	40	0	°C	e2	External Temp	obd.temp_ext	92	?	No
673	1	16	23	13.725	0	0	mbar	ff	Rear right wheel pressure	obd.whl_prs_rr	2450	Yes	No
673	2	24	31	13.725	0	0	mbar	ff	Rear left wheel pressure	obd.whl_prs_rl	2450	Yes	No
673	3	32	39	13.725	0	0	mbar	ff	Front right wheel pressure	obd.whl_prs_fr	2450	Yes	No
673	4	40	47	13.725	0	0	mbar	ff	Front left wheel pressure	obd.whl_prs_fl	2450	Yes	No
699	1	8	15	0.5	0	0	°C	e2	Temperature	obd.temp	20	Yes	No
68c	1	21	31	1	0	0	min	ff	Local Time	obd.time	600-800	Yes	Yes
6f8	1	5	5	1	0	0		ff	Front Wiper Stop Position	obd.ww_f_stat	0	Yes	Yes
6f8	2	6	7	1	0	0		ff	Front Wiper Status	obd.ww_f_stop	1	Yes	Yes

It is observed from Table 2 that most sensors have not been validated. It is also found that the outside temperature sensor data is missing and that several measures does not give physical meaning compared to the units in the protocol.

Table 3 shows the AutoPi sensor data available. Relevant data, such as GPS location and accelerations, have been validated using smartphones.

Table 3. *AutoPi sensor data showing units and conversion factors as well as measured and expected values*

AutoPi name	@tag	Typical values	Expected	Validated
track.pos	#N/A	14-55	yes	yes
acc.xyz	#N/A	-0.25 - 0.95	yes	yes
rpi.temp	#N/A	#N/A	#N/A	No
event.system.device.ec2x.gnss	system/device/ec2x/gnss/assist_data_updated	#N/A	#N/A	No
event.system.minion	system/minion/restart	#N/A	#N/A	No
event.system.network.wwan0	system/network/wwan0/online	#N/A	#N/A	No
event.system.power	system/power/_booting	#N/A	#N/A	No
event.system.power	system/power/on	#N/A	#N/A	No
event.system.release	system/release/failed	#N/A	#N/A	No
event.system.release	system/release/pending	#N/A	#N/A	No
event.system.time	system/time/synced	#N/A	#N/A	No
event.system.time	system/time/uncertain	#N/A	#N/A	No
event.vehicle.engine	vehicle/engine/not_running	#N/A	#N/A	No
event.vehicle.engine	vehicle/engine/running	#N/A	#N/A	No
event.vehicle.engine	vehicle/engine/stopped	#N/A	#N/A	No
event.vehicle.obd	vehicle/obd/bus_connected	#N/A	#N/A	No
event.vehicle.obd	vehicle/obd/interface_connected	#N/A	#N/A	No
event.vehicle.position	vehicle/position/moving	#N/A	#N/A	No
event.vehicle.position	vehicle/position/standstill	#N/A	#N/A	No
event.vehicle.position	vehicle/position/unknown	#N/A	#N/A	No

Sensor locations and several vehicle mechanical parameters are unknown. Currently, it is assumed that the steering information is reporting the front wheel angles (incl. angle changes and rates), motion-tracking sensors report accelerations at the center of mass, and braking information is reported at the location of the brakes/wheels.

Based on the findings from this section it is recommended to start further investigations to define metadata, e.g., from reverse engineering in a controlled test environment. These tests could also be used to determine mechanical vehicle parameters (e.g., suspension system). Specifications on the specific cars used by GreenMobility is also required; e.g., weight, dimensions, cross sectional area, etc.

3 Data quality

This section present the initial assessment of the data quality and sensor behavior of cardata. The analysis is based on graphical observations and basic statistics; more advanced analysis will be carried out for final validation of data.

GPS sensor data is essential for georeferencing and to establish the vehicle speed (e.g., if cardata sampling frequency is low). Moreover, altitude measurements can be used to calculate the slopes of the road. Figure 2 shows the AutoPi GPS sensor data for six passes versus the reference data (i.e., the standard road condition measurements).

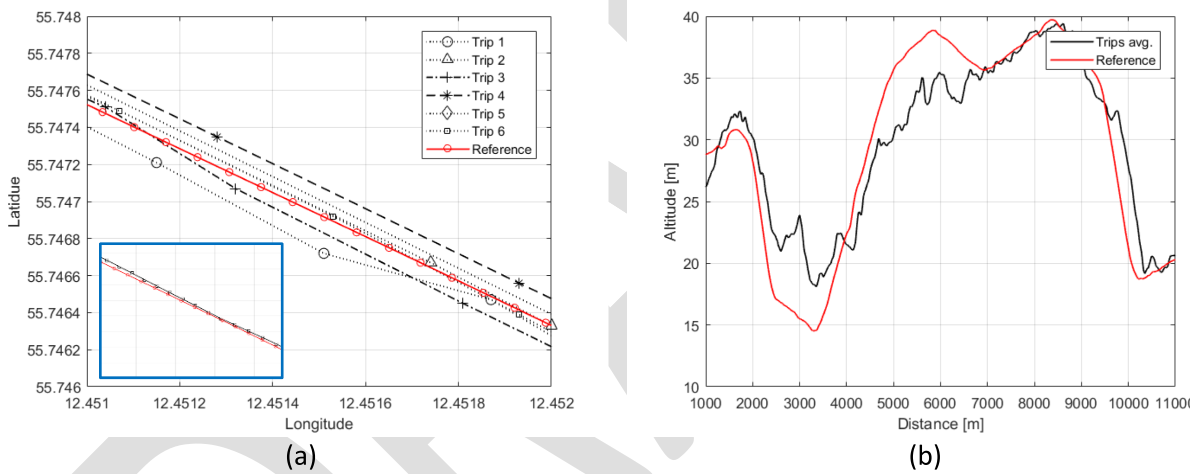


Figure 2. *Plot of AutoPi GPS sensor data (black) compared to reference GPS (red): (a) close-up of right lane northwest direction (app. 150 m) with offset in GPS of app. ±5 m and (b) altitude from chainage 1000 m to 11000 m of right lane northwest direction.*

It is observed from Figure 2a that there is an error of app. ±5 m in data between individual trips compared to the reference data. It is also observed that the average of six trips fit well with reference data (see blue frame). Figure 2b shows that the average altitude GPS readings from the AutoPi follows the reference. However, the data is not accurate enough for slope prediction.

Sensor data from accelerometers is essential input for development of models in the LiRA project. The quality of these sensor readings is therefore of high importance. The car accelerometer data is presented in Figure 3. In Figure 3a and Figure 3b y-axis accelerations versus car longitudinal accelerations is shown. In Figure 3c and Figure 3d x-axis accelerations versus car transverse accelerations is shown. Figure 3e shows z-axis accelerations for individual trips versus the average of all trips. Figure 3e shows a close-up of filtered z-axis accelerations on a part of the section between 3750 and 4550 m, where variation in data was observed. Prior to plotting, outliers in AutoPi accelerometer data were removed and sensor axis oriented using Euler's angle [3].

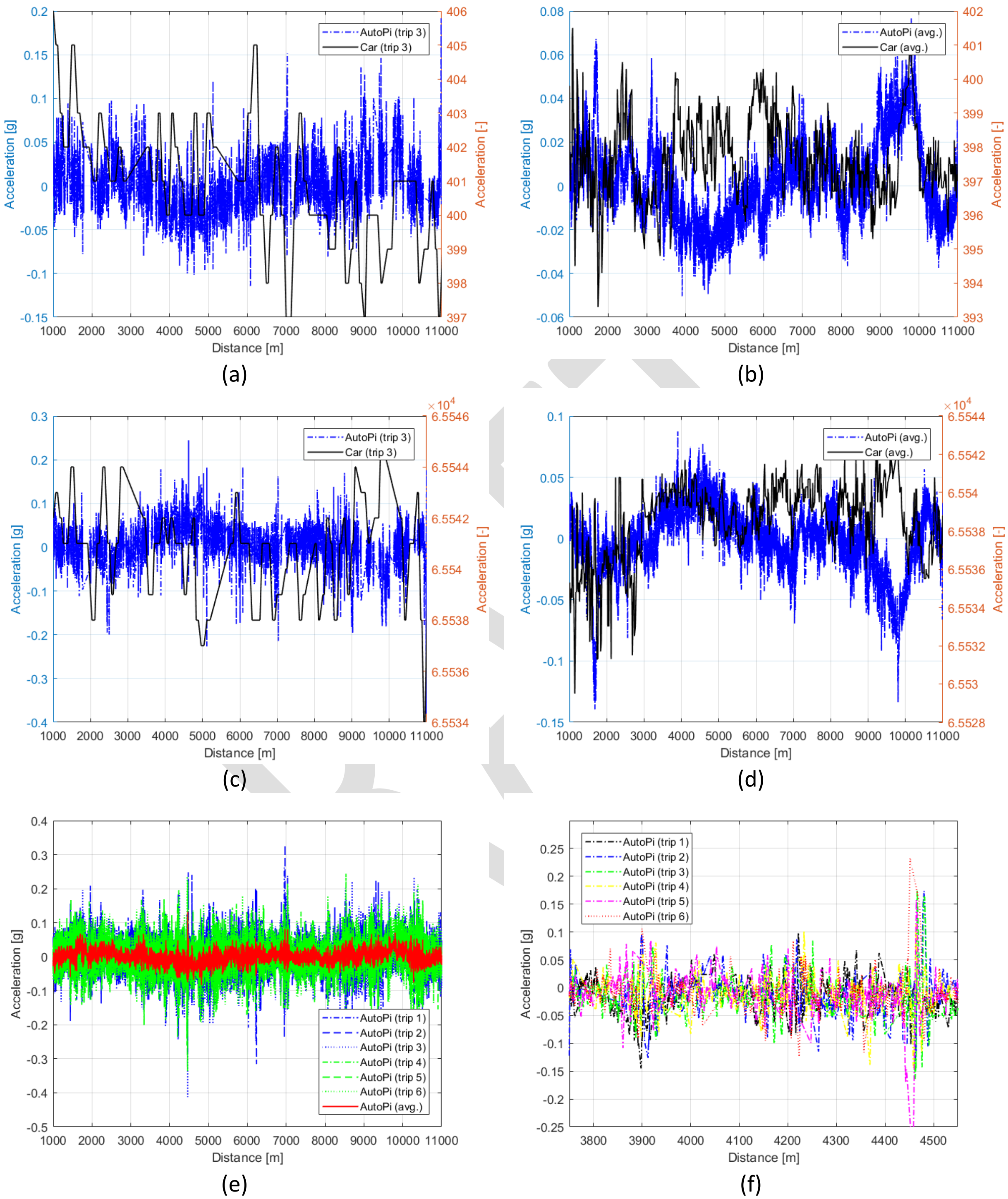


***Figure 3**. Car accelerations: (a) x-axis (longitudinal) accelerations for trip, (b) average x-axis (longitudinal) accelerations, (c) y-axis (transverse) accelerations for trip 3, (d) y-axis (transverse) accelerations, (e) z-axis accelerations for individual trips, and (f) close-up of filtered z-axis accelerations*

From Figure 3a and Figure 3c it is observed that the sampling frequency of car longitudinal and transverse accelerations is low. Thus, little can be concluded based on these plots. From Figure 3c and Figure 3d it is observed that there is some overlapping trends in average AutoPi y-axis versus longitudinal accelerations and average AutoPi x-axis accelerations and car and transverse accelerations, respectively. More data is required to further investigate if the two sensor types behaves similarly. From Figure 3e it is observed that the raw accelerometer signal from the AutoPi appears noisy. By filtering the signal it is possible to see more general trends in the data as exemplified in Figure 3f, indicating some level of repeatability in sensor signal. It was also found that outliers for all trips had constant values of -7.97 g, indicating a wrong reading or systematic error.

Development of reliable energy consumption models would be a valuable output from the LiRA project. It is therefore of interest to evaluate the various consumption sensor data from the electrical cars. Figure 4a and Figure 4b shows the trip energy consumption and the traction instant consumption versus speed, respectively.

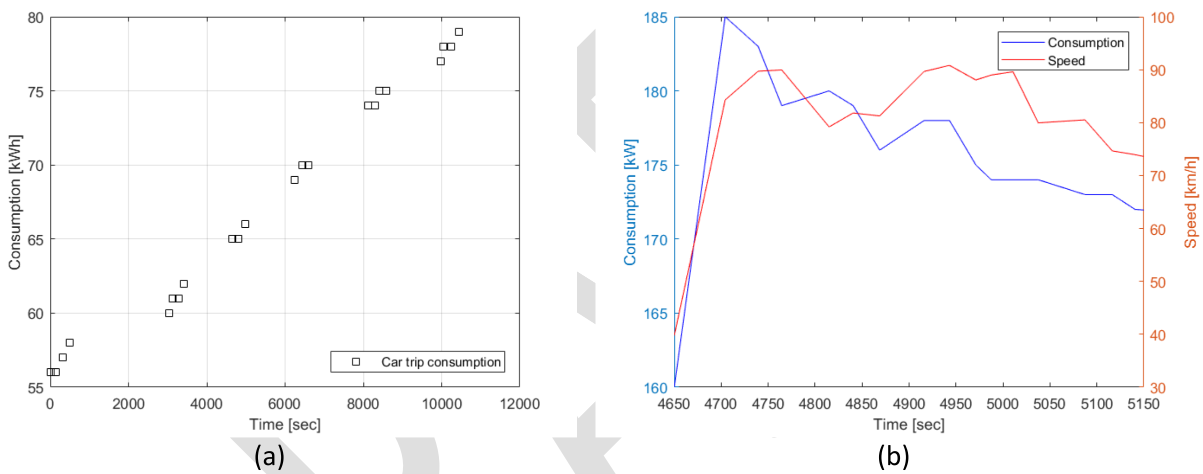
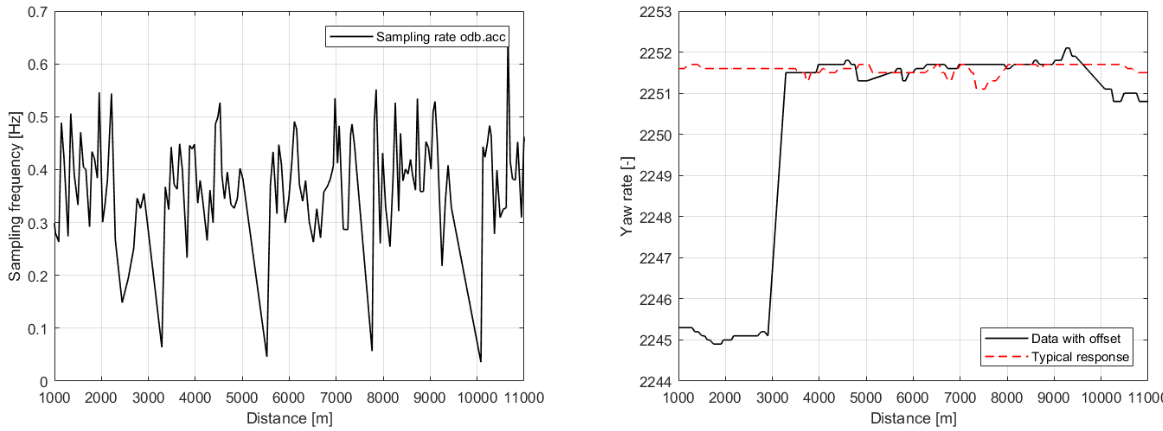


Figure 4*. Car energy consumption: (a) trip consumption and (b) close-up of traction instant consumption (blue line) versus speed (red line) for trip 3.*

It is observed from Figure 4a that the trip energy consumption increases linearly over the test day. Figure 4b shows that the traction instant consumption in general follow the speed of the car as expected. It is also found that the traction instant consumption has the unit of power.

In order to validate the data collected it is necessary to understand the behavior of individual sensors and how they are influenced by hardware and/or software configurations. For this purpose examples of unexpected sensor response is presented in Figure 5 and Figure 6. Typical cardata sampling frequency is shown in Figure 5a. Figure 5b shows an example of offset in motion tracking sensor data observed during one of the trips.

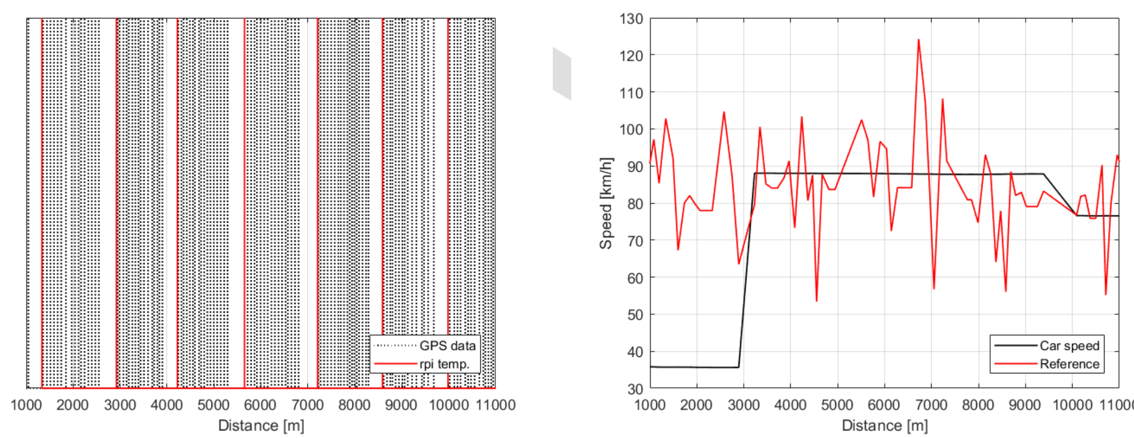


(a) (b)

***Figure 5**. (a) Typical sampling frequencies for cardata recorded and (b) example of offset in motion tracking sensor data.*

From Figure 5a it is observed that the variation cardata sampling frequency is relatively large. The Coefficient of Variation (CoV) is around 4 across all sensors. From Figure 5b offset/jump in yaw rate response for trip 2 (black solid line) compared to typical yaw rate data (red dashed line) is observed. It is not clear/validated if this offset was generated due to actual car response, wrong reading/systematic error or installation (e.g. if the sensor moved during braking or turning).

Figure 6a shows the positions of the car (trip 3) in the northwestern direction when AutoPi temperature (red line) and GPS sensor data (black dotted line) was recorded. Figure 6b shows the car speed reported from the in-vehicle sensor (black line) and GPS sensor (red line).



(a) (b)

***Figure 6**. (a) GPS sensor (black dashed lines) data versus AutoPi temperature data (red lines) and (b) car speed (black line) versus GPS speed (red line).*

From Figure 6a gaps in GPS sensor data is observed. This trend is general across all sensors. From Figure 6b it is observed that the speed measurements is almost constant between several readings. Similar trend is observed in odometer data. However, measures such as “trip distance” and “fine distance” are also available. Combining these measures in a rational manner was tested, but did not lead to any further conclusions.

The findings in this Section show that is likely that the AutoPi CUP/bandwidth limitations and/or other functions in the device influences the sampling frequency. It was not possible to identify a systematic pattern in the data. However, the temperature measurement in the AutoPi device is constant and typically followed by app. 15 GPS readings and then a pause. Speed and odometer cardata seems to be based on some averaging. However, the response observed could also be due to a fixed sampling frequency that is not supported by new data, e.g., due to the Can protocol (see Section 4).

Based on the findings from this section it is recommended to start further investigations to understand how AutoPi influences the data, e.g., test with different system configurations and compare with CPU usage, which is not available today. These tests could also be used to validate the permanent installation of the device, e.g. to ensure that there is no offsets in motion tracking data. Furthermore, potential improvements of the device and/or alternative systems, should also be investigated (see Section 4).

4 Limitations and usefulness of data

Sensors in modern cars typically sample at a frequency of min. 20 Hz up to 100 kHz (see e.g., [4], [5]). However, these vehicle-generated data are primarily of a technical nature. They exist only temporarily, are used locally within vehicle systems, and are never stored and transmitted to the ODBII-port.

In order to access all the data special software, and potentially hardware, is required in the cars (see e.g. [5], [6]). The sensor data transmitted to the ODBII-port are prioritized (see Appendix A). Messages with numerically smaller values of IDs have higher priority. Examples of average sampling/transmission frequencies of cardata from Highway M13 are shown in Table 4.

Table 4. *Example of average transmission rates of Can bus data to the ODB-port.*

Sensor type	Identifier (ID), [hex]	Sampling frequency, [Hz]
Steering	0c6	~0.4
Motion tracking	12e	~0.3
Braking	1f8	~0.3
Vehicle speed	29a	~0.2
Traction instant consumption	638	~0.05
Tire pressure	673	~0.05

In addition the AutoPi report on a number of other events, such as “power on”, “engine running”, “obd interface connected”, “system time synced”, etc. (see Table 3). These messages are typically saved when starting or stopping the vehicle. The sampling frequencies of the external sensors in the AutoPi and the smartphones are shown in Table 5.

Table 5. Example of average sampling frequency of AutoPi sensors. Smartphone sensors shown for reference with gray background.

Sensor type	Device	Sampling frequency, [Hz]
Three-axis accelerometer	AutoPi 1st generation	~50
GPS	AutoPi 1st generation	~0.3
Device temperature	AutoPi 1st generation	~0.015
Three- axis accelerometer	Huawei/ Vns-l31	~165
Three- axis accelerometer	Samsung/Sm-g930f	~105
Three- axis rotation	Samsung/Sm-g930f	~105

The AutoPi three-axis accelerometer of type MMA8451Q (14-bit/8-bit) is capable of sampling at a frequency of up to 800 Hz, but is currently restricted to a maximum sampling frequency of app. 50 Hz due all the running processes and the limited CPU/bandwidth. Thus, at the present time the AutoPi do not provide information at the same level as smartphones in terms of sampling frequency. However, optimization should be possible; Masino et al. [7] used a similar device sampling both three-axis accelerometer and gyroscope sensor data at a frequency of 250 Hz, as well as GPS data at 10 Hz.

Taking into consideration that the test sections will be passed maximum 10 times in the first measurement campaign (task T1.4), the data sampling frequencies summarized Table 4 and Table 5 raise some concerns about the limitations and usefulness of the datasets collected in the LiRA project.

For comparison, a brief summary of the hardware configurations used in similar studies reported in the literature compared to number of vehicle trips are listed below:

- Eriksson et al. [7] used a three-axis accelerometer and GPS sensor, which was sampled at a frequency of 380 Hz and 1 Hz, respectively. 174 km of road data was collected by using seven different vehicles passing over each section minimum 10 times. The data collected was used to classify potholes only.
- Seraj et al. [9] used smartphone three-axis acceleration sensor, gyroscope and GPS sensor, which were sampled at a frequency of 93 Hz. 45.9 km of road data were collected in two cities by using five different vehicles. The data collected were used to distinguish between three classes, namely severe (e.g. potholes, heavily patched road sections), mild (e.g. cracks) and span (e.g. speed bumps, pedestrian crossings) events.
- Asamer et al. [10] developed a vehicle consumption model based on 1 Hz data and 945 vehicle trips.
- Jiménez et al. [11] used smartphone three-axis acceleration sensor, gyroscope and GPS sensor, which were sampled at a frequency of 5 Hz. 22,000 car trips was used to evaluate the effect of driving events on electrical vehicles energy consumption. ODB cardata, from the software LeafSpy Pro (for Nissan Leaf cars), was used to validate the events detected.
- El-Wakeel et al. [12] used a deployable testbed with external sensors to simulate the behavior of in-vehicle sensors in order to by-pass this problem.

Table 6 summarizes the findings from this section. The table shows the success criteria for the LiRA project [13] versus the relevant sensors requirements and capabilities.

Table 6. *Summary of data limitations: LiRA project success criteria versus sensor requirements and capabilities. The table is provided for guidance only.*

Priority	Road condition measure	Primary sensors	Required frequency	Sensors available	Frequency sufficient	No. of sensors validated	Current estimated success rate
1	Friction	Braking; Accel.; Steering; Motion tracking	High	4/4	No	1/4	0%
2	Cracking density	Accel.	High	1/1	No	1/1	0%
3	Potholes	Accel.; Steering	Medium	2/2	Yes	1/2	90%
4	Noise	Microphone; Accel.	High	1/2	#N/A	#N/A	0%
5	IRI	Accel.	Medium	1/1	Yes	1/1	90%
6	Energy Expenditure / Rolling Resistance	Accel.; Gyro; GPS/Speed; Energy consumption	Low/Medium	3/4	No	2/4	0%
7	Patched area	Accel.	High	1/1	No	1/1	25%
8	Unevenness	Accel.	High	1/1	No	1/1	25%
9	Rutting depth	Braking; Accel.; Steering; Motion tracking	High	4/4	No	1/4	0%
10	Texture depth	Microphone; Accel.	High	1/2	#N/A	#N/A	0%

It is observed from Table 6 that the chances of meeting the success criteria for the LiRA project is below average at the present time. Moreover, it seems difficult, at best, to improve this outcome significantly, taking into consideration that the test sections, with a total length of app. 400 km, will be passed maximum 10 times (using 20 vehicles) during the first measurement campaign (task T1.4). Finally, 3-axis rotation measurements are not available with the current system. Thus, prediction of the road slope, relevant for development of energy consumption models, is not possible (see also Figure 2).

It is therefore, suggested to assess alternative hardware platforms and software systems for cardata collection. This in order to have "plan B" in place in good time before the second measurement campaign (task T1.5) if needed. In this relation, the following alternative providers of hardware platforms for reading cardata have been identified:

- National Instrument vehicle communication buses: https://www.ni.com/
- TTTech all-in-one data logger for the entire vehicle networking: https://www.tttech-auto.com/
- CSS electronics Can bus loggers: https://www.csselectronics.com/

References

[1] Klopfenstein et al. (2016). Mobile crowdsensing for road sustainability: exploitability of publicly-sourced data. *International Review of Applied Economics*, 1-22.

[2] https://github.com/fesch/CanZE

[3] Singh et al. (2017). Smart patrolling: An efficient road surface monitoring using smartphone sensors and crowdsourcing. *Pervasive and Mobile Computing*, 40, 71-88.

[4] Padarthy et al. (2020). Investigation on Identifying Road Anomalies Using In-Vehicle Sensors for Cooperative Applications and Road Asset Management. *Transportation Research Board* 2020, Washington DC.

[5] Ashlock. (2013). Synchronizing Controller Area Network and Analog Input Measurements for In-Vehicle Data Logging, *National Instruments.*

[6] Chen et al. (2020). Crowdsensing Road Surface Quality Using Connected Vehicle Data. *Transportation Research Board* 2020, Washington DC.

[7] Masino et al. (2016). Development of a Highly Accurate and Low Cost Measurement Device for Field Operational Tests*. IEEE International Symposium on Inertial Sensors and Systems,* 71-74.

[8] Eriksson et al. (2008). The pothole patrol: using a mobile sensor network for road surface monitoring. In *Proceedings of the 6th international conference on Mobile systems*, applications, and services (pp. 29-39).

[9] Seraj et al. (2016). RoADS: A road pavement monitoring system for anomaly detection using smart phones. In *Big data analytics in the social and ubiquitous context* (pp. 128-146), Springer, Cham.

[10] Asamer et al. (2016). Sensitivity Analysis for Energy Demand Estimation of Electric Vehicles. *Transportation Research Part D: Transport and Environment*, 46, 182-199.

[11] Jiménez et al. (2018). Modelling the effect of driving events on electrical vehicle energy consumption using inertial sensors in smartphones. *Energies*, *11*(2), 412.

[12] El-Wakeel et al. (2018). Towards a Practical Crowdsensing System for Road Surface Conditions Monitoring. *IEEE Internet of Things Journal*, vol. 5, no. 6.

[13] Exhibit 1 – Project plan, Innovation Fund Denmark, File number: 8090-00048B.

[14] http://www.flexautomotive.net/

Appendix A - Can bus in brief

CAN bus (for **controller area network**) is a vehicle bus standard designed to allow microcontrollers and devices to communicate with each other within a vehicle without a host computer.

Technology

CAN is a multi-master broadcast serial bus standard for connecting electronic control units (ECUs). Each node is able to send and receive messages, but not simultaneously. A message consists primarily of an ID (identifier), which represents the priority of the message, and up to eight data bytes. The improved CAN FD extends the length of the data section to up to 64 bytes per frame. It is transmitted serially onto the bus. This signal pattern is encoded in non-return-to-zero (NRZ) and is sensed by all nodes. The devices that are connected by a CAN network are typically sensors, actuators, and other control devices. These devices are not connected directly to the bus, but through a host processor and a CAN controller. If the bus is idle any node may begin to transmit. If two or more nodes begin sending messages at the same time, the message with the more dominant ID (which has more dominant bits, i.e., zeroes) will overwrite other nodes' less dominant IDs, so that eventually (after this arbitration on the ID) only the dominant message remains and is received by all nodes. This mechanism is referred to as priority based bus arbitration. Messages with numerically smaller values of IDs have higher priority and are transmitted first.

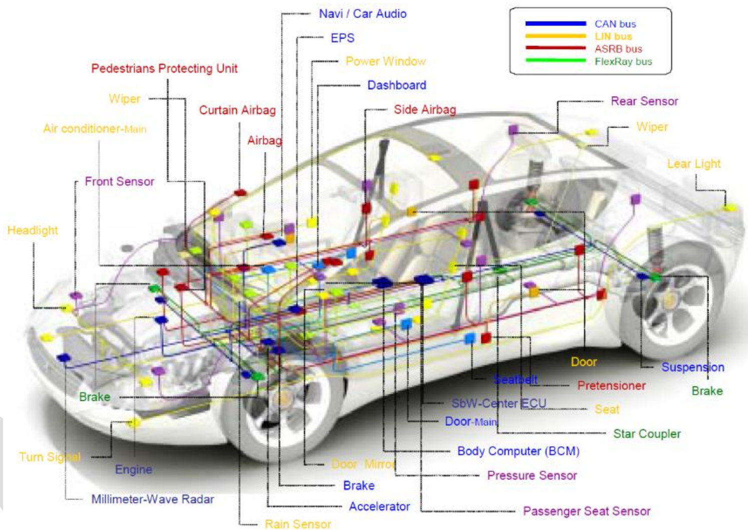


Figure 7. *Overview of vehicle networks (bus). From [14].*

Each node requires:

<u>Host processor</u>

- The host processor decides what received messages mean and which messages it wants to transmit itself.
- Sensors, actuators and control devices can be connected to the host processor.

<u>CAN controller</u> (hardware with a synchronous clock).

- Receiving: the CAN controller stores received bits serially from the bus until an entire message is available, which can then be fetched by the host processor.
- Sending: the host processor stores it’s transmit messages to a CAN controller, which transmits the bits serially onto the bus.

<u>Transceiver</u>

- Receiving: it adapts signal levels from the bus to levels that the CAN controller expects and has protective circuitry that protects the CAN controller.
- Transmitting: it converts the transmit-bit signal received from the CAN controller into a signal that is sent onto the bus.

Appendix B – Alignment between trips

In this preliminary study, alignment of data between individual trips was based on GPS distances. The distance between two coordinates/points (lat_i, lon_i) and (lat_{i+1}, lon_{i+1}), is given as

$$\Delta x_j = 2 \cdot \mathrm{atan}\left(\frac{\sqrt{a_j}}{\sqrt{a_j - 1}}\right) \cdot R \qquad (1)$$

, $R = 6378.137 \cdot 10^3$ is the radius of the Earth in meters, and

$$a_j = sin\left(\frac{\Delta lat_j}{2}\right)^2 + cos\left(lat_i \cdot \frac{\pi}{180}\right) \cdot cos\left(lat_{i+1} \cdot \frac{\pi}{180}\right) \cdot \left(sin\left(\frac{\Delta lon_j}{2}\right)\right)^2$$

$$\Delta lat_j = (lat_{i+1} - lat_i) \cdot \frac{\pi}{180}$$

$$\Delta lon_j = (lon_{i+1} - lon_i) \cdot \frac{\pi}{180}$$