YOUR NATIONAL TRANSPORT RESEARCH ORGANISATION

CAV Probe Vehicle

Interim Report

Phase 3: Testing of vehicle. High-level report - Learnings and Opportunities

ARRB Project No.: 016730

Author: Amolika Sinha
Marcus John
Macgregor Buckley
Edward Dann
Daniel Siu
Dr Sepehr G. Dehkordi

Prepared for: iMOVE Australia Ltd

March 2022

Final version

Summary

This progress report provides insight into the hardware and software integration using the iSCAN probe vehicle. The on-road data collection has been done and the post-processing was applied for different applications where the iSCAN vehicle can be potentially used.

The report is divided into four sections. ***Section 1*** provides a brief description of the collected data and the locations where the iSCAN probe vehicle was operated. ***Section 2*** gives the perception information provided by the iSCAN vehicle with the extrinsic calibration between LiDAR and Camera, following some post-processing and using machine learning algorithms to detect vehicles, making sure that the system can be used for AI applications. We investigated the data synchronisation and provided insight into synchronising different sensors in a post-processing level. ***Section 3*** provides insight into the localisation accuracy of iSCAN system within the on-road collected data. We also generated a map for the driven locations and their corresponding localisation accuracy. ***Section 4*** investigates the data associated with the driving monitoring system and approaches that can be used to synchronise its data with other sensors. Finally, ***Section 5*** outlines some commentary about the learnings and opportunities with the iSCAN vehicle.

Contents

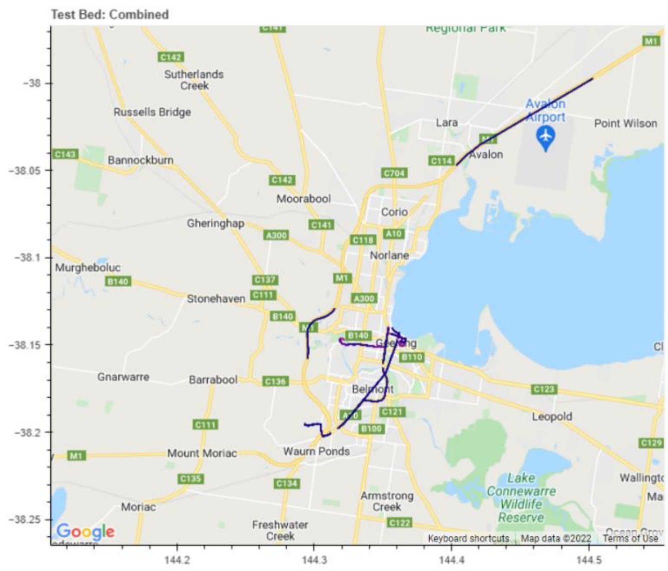
Tables

Figures

1 Data collection with iSCAN Probe Vehicle

This section outlines data collection using hardware-software integration with the iSCAN vehicle. The data were collected in Geelong, Victoria with a combination of the driving scenario from the motorways to urban driving. All the raw data from the sensors and their associated timestamps were recorded as a Robotic Operation System *Bag (RosBag*[1]*)* File. RosBag allows us to replay the driving scenario as it was recorded in the real world. Total of 316Gb of data for 8 hours of driving were collected. Figure 1-1 shows all the trajectories which have been driven with the iSCAN vehicle.

Figure 1-1 Trajectory map of collected data.



During the data collection, some qualitative information and incidents were recorded and labelled by one of the team members. Table 1-1 shows examples of the recorded meta-information which facilitates us with easy search in the database for post-processing.

Table 1-1 Metadata recorded during iSCAN data collection

Start	End	Location	Speed	Traffic	Notes
10:05 AM	10:07 AM	Deakin University	Low (40)	Low	N/A
10:11 AM	10:22 AM	Deakin – Geelong	Medium (70)	Medium	Ambulance with lights, pole landmark in Geelong
12:05 PM	12:16 PM	Inner Geelong	Medium (50)	Medium	Underpass, train lines, tunnel
2:10 PM	2:23 PM	Westgate Bridge	Low/High (20-80)	High	Traffic jam, road works

While RosBag is an engineering solution to record and save data, it is challenging for common users to access information written in RosBag. We did an effort to transform RoSbag into a human-readable format like a CSV tables. Figure 1-2 shows an example of transformed data into CSV from the positioning system. It provides information related to latitude, longitude, altitude and heading with their expected errors.

1 http://wiki.ros.org/rosbag

Figure 1-2 An example of IMU/GNSS data recorded in CSV format

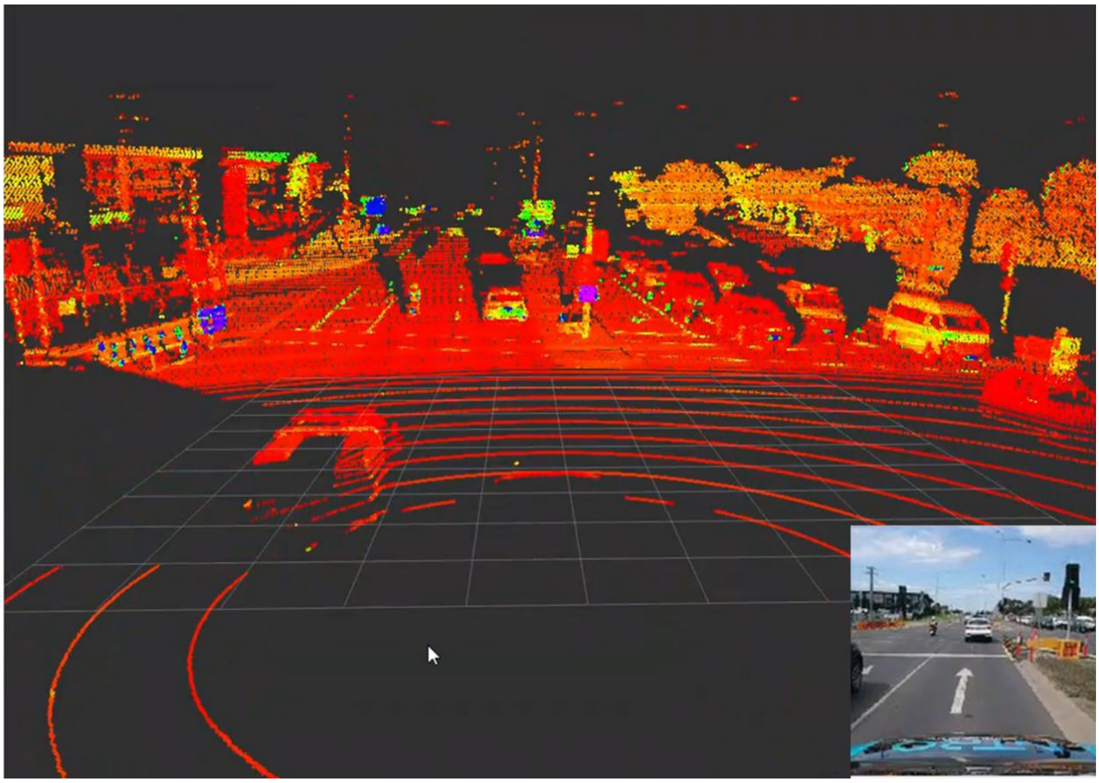
Time	header.se	header.sta	header.sta	header.fra	nov_head	nov_head	nov_head	nov_head	nov_head	nov_head	nov_head	sol_status	pos_type.	lat	lon	hgt	undulatio	datum_id	lat_stdev	lon_stdev	hgt_stdev	stn_id
1644371331	30849	1.6E+09	2.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.6921	2.5	61	0.44429	0.32357	0.65878	
1644371331	30850	1.6E+09	3.2E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.7298	2.5	61	0.44429	0.32357	0.65878	
1644371331	30851	1.6E+09	4.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.7678	2.5	61	0.44429	0.32357	0.65878	
1644371332	30852	1.6E+09	5.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.8059	2.5	61	0.44429	0.32357	0.65878	
1644371332	30853	1.6E+09	6.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.8452	2.5	61	0.44429	0.32357	0.65878	
1644371332	30854	1.6E+09	7.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.8858	2.5	61	0.44429	0.32357	0.65878	
1644371332	30855	1.6E+09	8.5E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.9264	2.5	61	0.44429	0.32357	0.65878	
1644371332	30856	1.6E+09	9.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	41.9653	2.5	61	0.47213	0.34705	0.6902	
1644371332	30857	1.6E+09	1.3E+07	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	42.0031	2.5	61	0.47213	0.34705	0.6902	
1644371332	30858	1.6E+09	1.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	42.0415	2.5	61	0.47213	0.34705	0.6902	
1644371332	30859	1.6E+09	2.1E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	42.0803	2.5	61	0.47213	0.34705	0.6902	
1644371332	30860	1.6E+09	3.2E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	42.1178	2.5	61	0.47213	0.34705	0.6902	
1644371332	30861	1.6E+09	4.2E+08	gps	BESTPOS	42	0	0	180	2196	2.7E+08	0	53	-38.151	144.351	42.1535	2.5	61	0.47213	0.34705	0.6902	

The camera data is also re-played and recorded as sequences of images for post-processing. This approach is a common practice for data management and vision processing applications.

2 Perception system

Figure 2-1 is a snapshot of a driving scenario recorded for this report, showing the camera and lidar capture the same driving scenario. Noted that the perception data and positioning data has the same timestamp (i.e. ROS timestamp) and they can replay together. We explored how extrinsic calibration between the camera and LiDAR can be done to overlay point clouds on the image. We also run pre-trained detection models with our sample data to make sure the vision data provided with the iSCAN vehicle can be used in deep learning and artificial intelligence applications. Finally, we provide insight into data synchronisation as a post-processing step and a fundamental part of sensor fusion applications.

Figure 2-1 Sample data from LiDAR point cloud and front-view camera



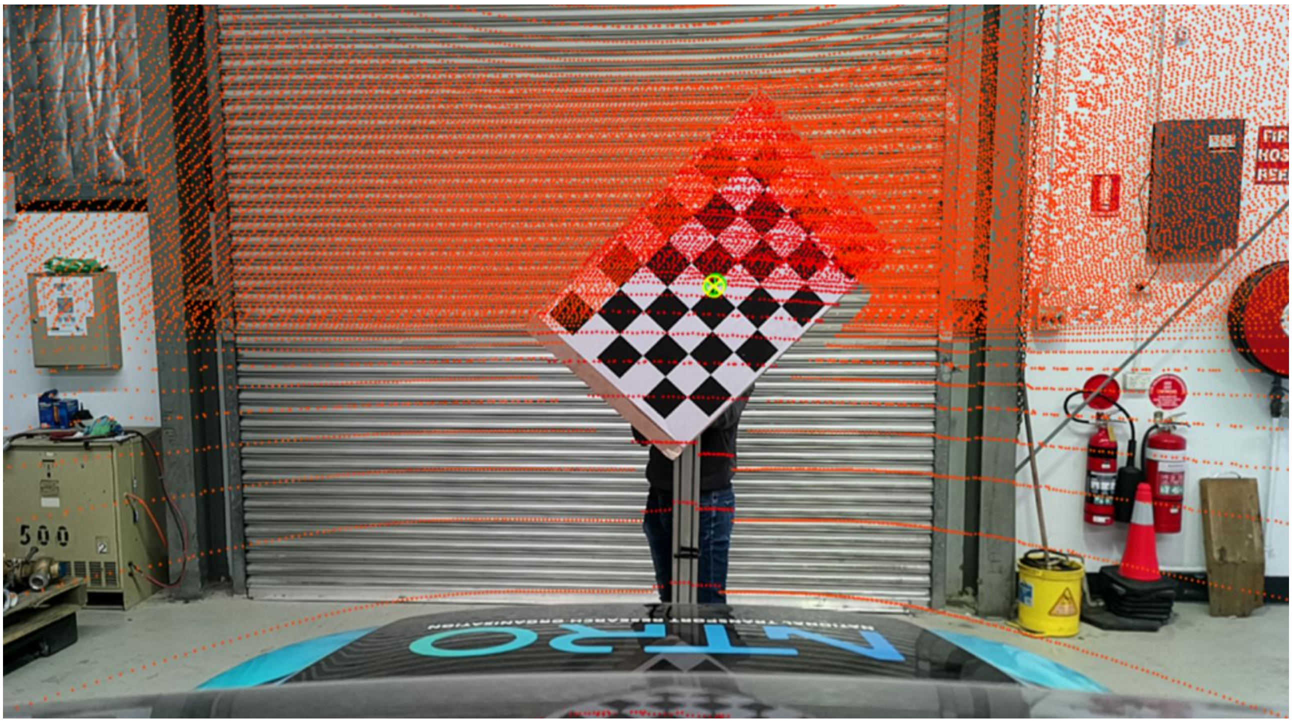
2.1 Extrinsic camera calibration

Extrinsic camera calibration involves understanding where the camera exists in the world and its relation to other sensors. It is an essential step for being able to aggregate the data from cameras and other sensors to create depth maps, 3D mapping and more (i.e. sensor fusion applications). More specifically, extrinsic calibration is achieved by calculating the relative rotation (roll/yaw/pitch) and translation (x,y,z) between each camera and a common reference frame. For our application, the common frame will be the LiDAR. There are open-source software available to obtain the extrinsic calibration compatible with the ROS platform[1]. Using a checkerboard target, the software is able to identify the board centre, edges and orientation of the board in both the camera and LiDAR data sources and fit a transformation between the two. Collecting a range of samples with the target in different positions and orientations will improve the accuracy of final estimates. A good test is to use the final parameters to project the LiDAR points onto the image. The result shown in Figure 2-2 demonstrates a successful calibration with clearly defined edges of the LiDAR on the checkerboard and

[1] https://github.com/Deephome/Awesome-LiDAR-Camera-Calibration

LiDAR rings following the contours of background architecture. The next step in this domain is to use derived parameters to create 3D mapping and even fuse the depth cloud from the stereo camera to the LiDAR point cloud greatly increasing the 3D mapping resolution by providing Red-Green-Blue-Depth images (RGBD images) which is achievable in the short term.

Figure 2-2 Extrinsic calibration between camera and LiDAR



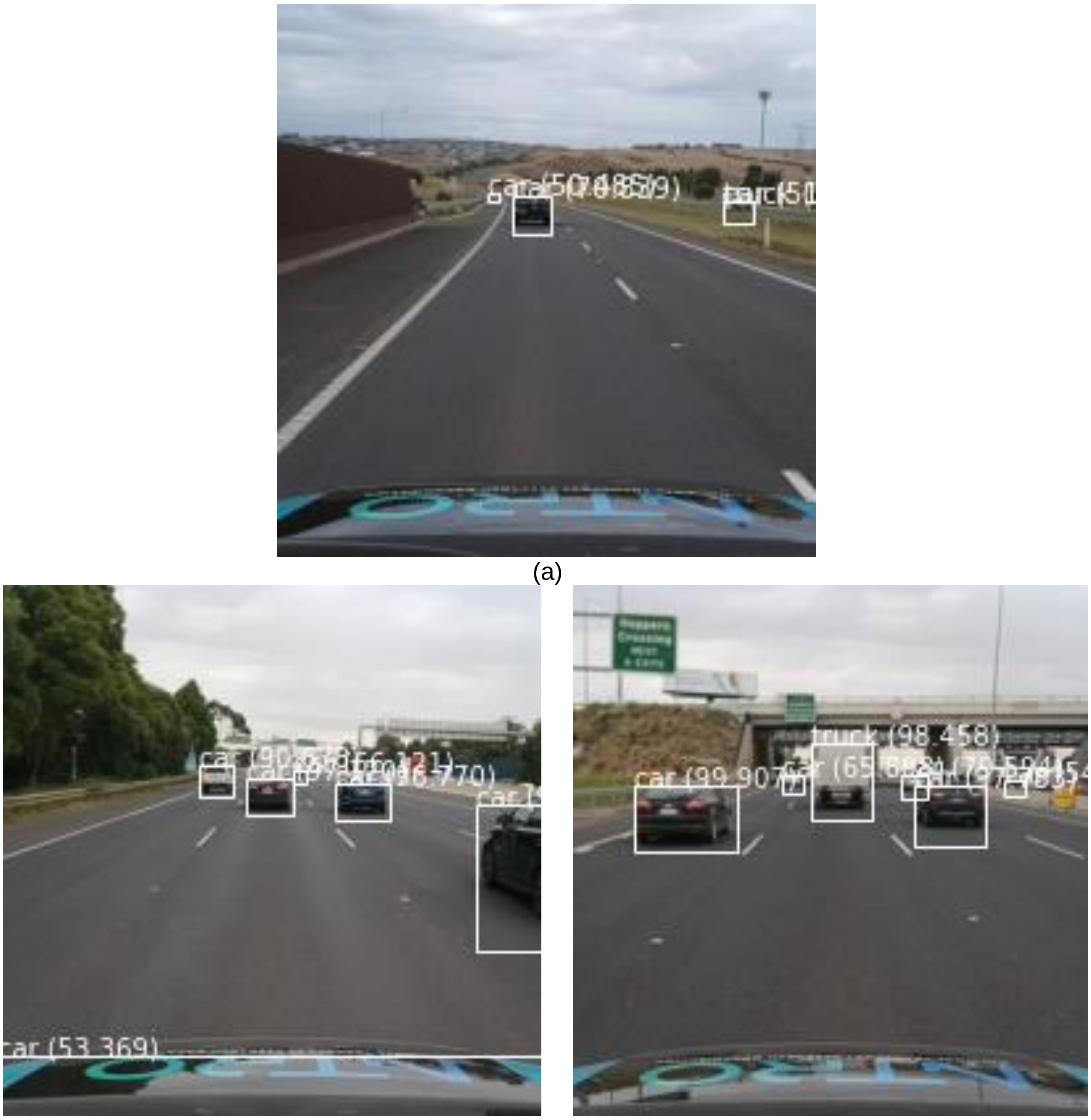
2.2 Object detection applied to the camera footage

The object detection/classification model required training data for modelling. We used a pre-trained model to test the iSCAN vision data.YOLOv4 model which is pre-trained with the COCO dataset were used in our evaluation (Lin et al. 2015). COCO dataset images of complex everyday scenes containing common objects in their natural context. Objects are labelled using per-instance segmentations to assist in precise object localization. The dataset contains photos of 91 object types. There is a total of 2.5 million labelled instances in 328k images in COCO dataset. The creation of this dataset drew upon extensive crowd worker involvement via novel user interfaces for category detection, instance spotting and instance segmentation. We only used the pre-trained model and training a new system is outside the scope of iSCAN's compatibility test.

Figure 2-3 shows samples taken from test drives throughout the Greater Geelong area. An object detection model has been deployed on the recorded data, generating bounding boxes around detected objects. Figure 2-3 shows the YOLO model's strong ability in object detection, detecting multiple objects and identifying them as cars. Figure 2-3 also shows the model's ability to identify different objects from each other, even at a distance, identifying both cars and a truck from the image.

Overall, the pre-trained YOLOv4 model and iSCAN's collected images identify the practicality and strong ability to utilise video analytics and object detection models. Opportunities exist to custom train models to identify a wide variety of objects, which may prove useful in identifying road signs, lanes, and obtaining other information necessary for both drivers and autonomous vehicle systems with data collected by iSCAN probe vehicle.

Figure 2-3: iSCAN camera data used as input for a YOLO4 model for object detection



(a)

(b)

(c)

2.3 Data synchronisation

The iSCAN uses ROS as a middleware (*ROS: Home* 2022), and the recorded data is timestamped with ROS across different sensors using the Unix time[1]. In addition to ROS timestamps, some sensors such as IMU have their internal clock with its own sensor-specific timestamp. The ROS time stamps and the IMU internal clock timestamps were compared to estimate the lag. Figure 2-4indicates the distribution of lag (histogram). The X-axis represents the lag bins (milliseconds) and the Y-axis represents the frequency. The lag bin size is 25 milliseconds. As observed in Figure 2-4 that the distribution of lag is primarily close to 0 milliseconds, indicating the accuracy of the system. After this preliminary analysis of the data, post-processing synchronisation has been done to the recorded data based on the sensors' frequency listed in Table 2-1.

[1] Unix time (also known as Epoch time, Posix time, seconds since the Epoch, or UNIX Epoch time) is a system for describing a point in time. It is the number of seconds that have elapsed since the *Unix epoch*, excluding leap seconds. The Unix epoch is 00:00:00 UTC on 1 January 1970 ('Unix time' 2022).

Figure 2-4 The time lag between of IMU/GNSS internal clock and ROS time stamps for a sample pilot data

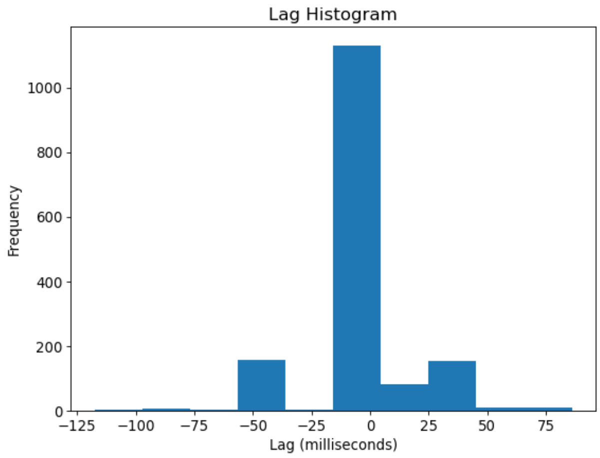
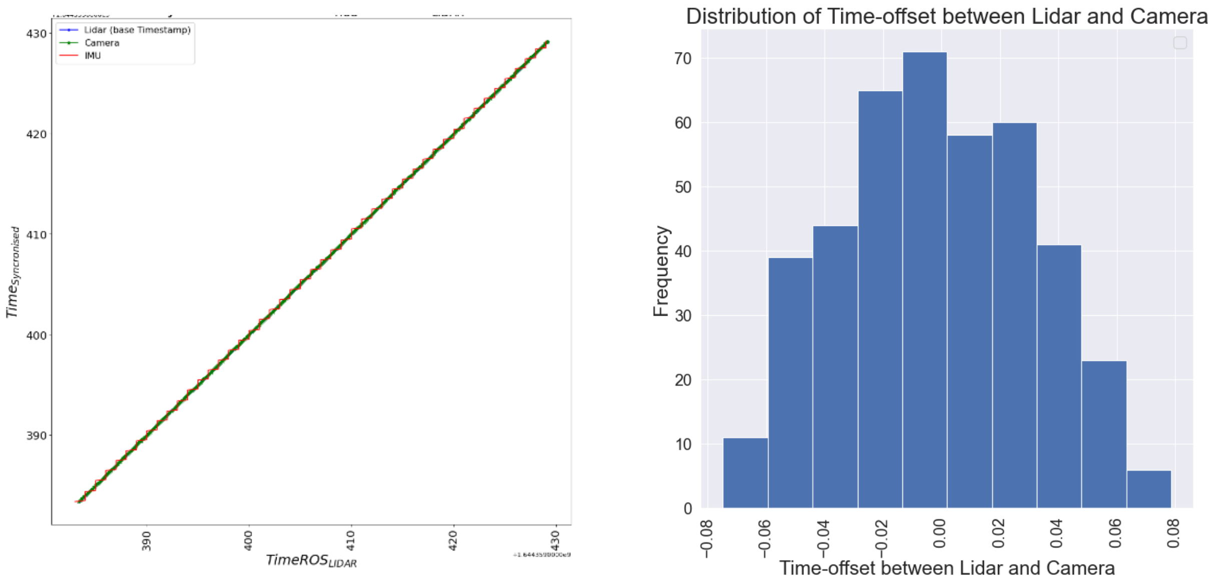


Table 2-1 Frequency of data collection for different sensors

Sensor	Frequency	Timestamps
IMU/GNSS	100 Hz	TROS (UNIX) & GPS week-seconds
Camera	10 Hz	TROS
Lidar	10 Hz	TROS

It is a common practice to synchronise all the sensors against LiDAR frequency because LiDAR usually has the lowest frequency among other sensors. The synchronisation can be done by matching the nearest data with the closest timestamp (i.e. nearest interpolation approach). Figure 2-5(a) shows the incremental timestamp across different sensors. The figure also shows how consistent is the ROS timestamp in iSCAN data. Applying the nearest interpolation approach between LiDAR and camera provides an expected error for matching the synchronised data. Figure 2-5(b) shows the histogram of the time offset between two matched frames of the LiDAR with a camera which is normally distributed around zero. The synchronization analysis which has been done on the iSCAN data helps for future application in sensor fusion and ultimately generates the sensor's abstraction layer for automated vehicles.

Figure 2-5 Synchronised timestamp for all the sensors, a) $Time_{ROS}$ vs $Time_{ROS}$ and Distribution of time-offsets between different sensors.



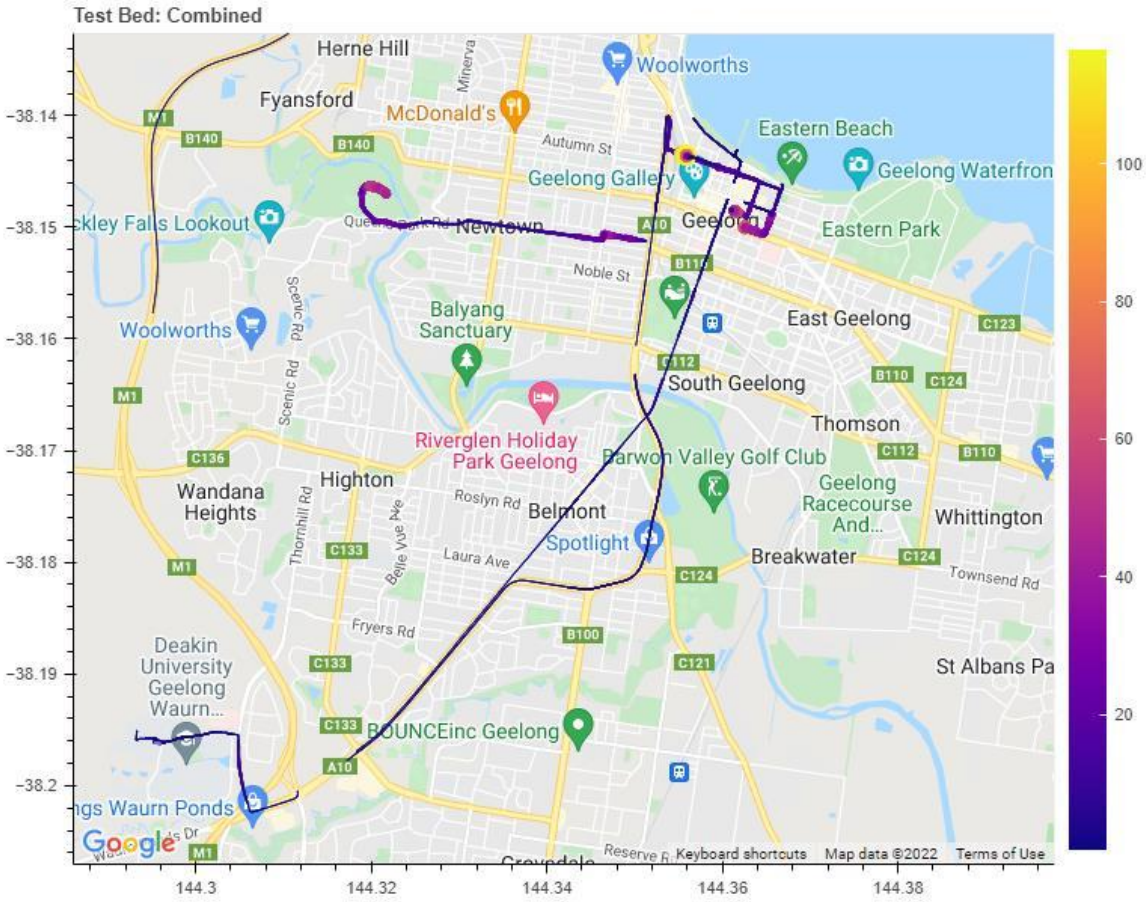
(a) (b)

3 Localisation system

The localisation error is defined as a standard deviation for parameters associated with the position such as longitude, latitude, altitude, and heading. The unit of localisation error is meters; showing the potential drifting from the actual position. A large localisation error in automated driving causes lane departure or driving in the wrong direction. Hence, it is important to understand the limits of the system to constrain the operational domain. iSCAN uses an IMU/GNSS onboard data fusion system. More details about the output attributes reported as the best positioning value can be found in section 7, ANNEX1.

The localisation error equals to $\sqrt{\sigma_{lat}^2 + \sigma_{long}^2}$, where σ_{lat} and σ_{long} are lateral and longitudinal standard deviation, respectively. Figure 3-1 shows the standard deviation error of the recorded data with the iSCAN system. The colour of the spline represents the localisation error in centimetres. This accumulative information will ultimately use to identify the places where the localisation error is good enough to drive an automated vehicle[1].

Figure 3-1 The standard deviation of IMU/GNSS localisation for a sample pilot data for (a) one trajectory (b) all the trajectories



The in-depth analysis of each road segment can be done by having the histogram of the selected road. Figure 3-2 presents the histogram for a single trajectory. For this specific trajectory, the location error is less than 18

[1] Usually the standard deviation less than 20cm is suitable for automated driving.

centimetres. Some other examples of trajectory with the corresponding localisation error are depicted in Section 8, ANNEX2.

Figure 3-2 A sample trajectory and the histogram of the standard deviation for localisation.

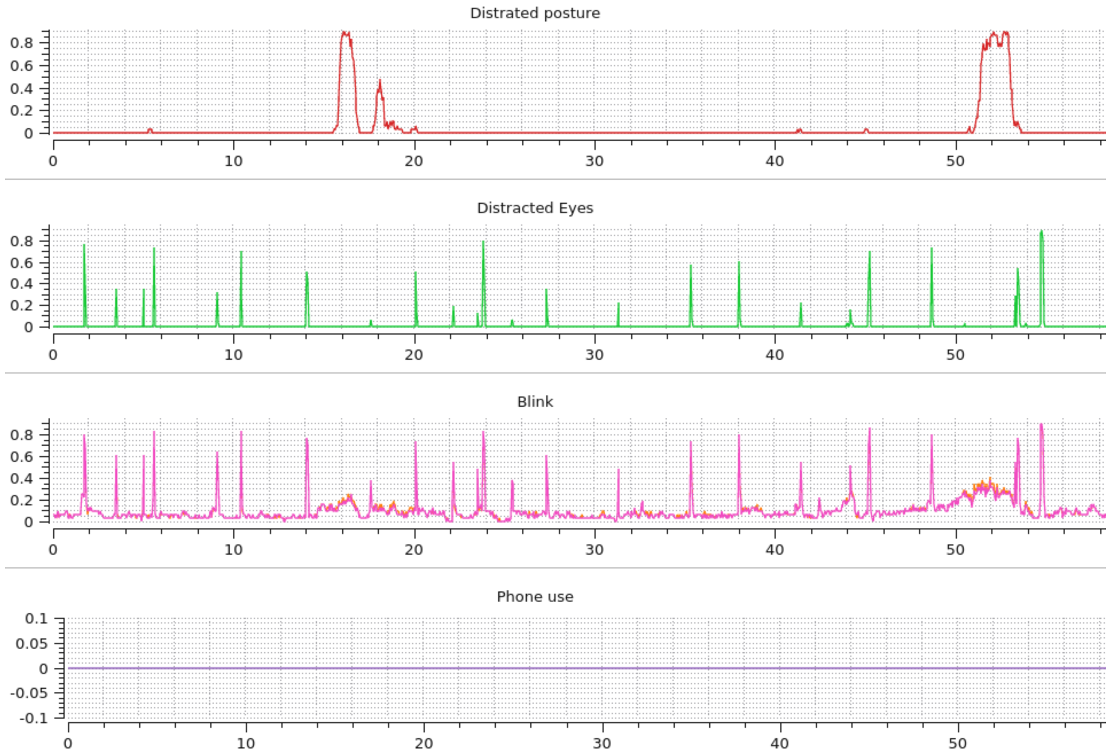
(a)

(b)

4 Driver monitoring system

Driver monitoring is a key component in semi-automated vehicles as the driver is still required to pay attention to the road and with less for the driver to do a secondary alert system to remind the driver to pay attention can be implemented. The iSCAN vehicle achieves this capability by installing a comma3 device that contains a driver monitoring camera. Using AI-run locally on the device, it has the capability to detect whether the driver has a distracted posture or distracted eyes, detects phone use and blinks. The device has a screen through which it can provide real-time visual feedback to the driver reminding them to keep their eyes on the road. The AI returns each parameter as a probability, as can be seen in Figure 4-1 with a couple of sample frames demonstrating its capability.

Figure 4-1 Driver monitoring probabilities over one minute sample

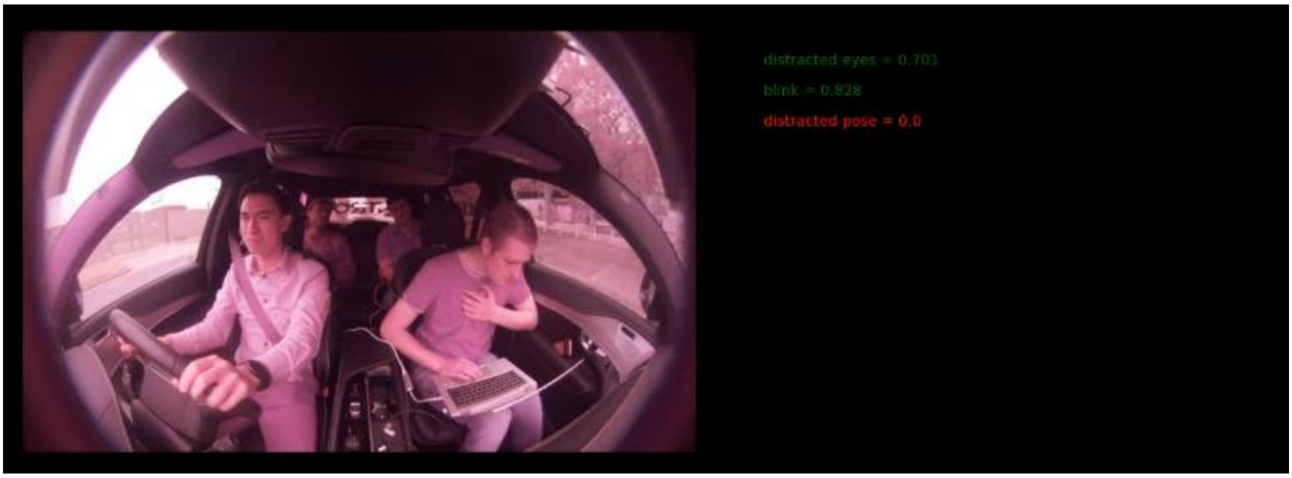


Along with driver monitoring the iSCAN Driving Monitoring System (DMS) can provide useful information including wheel and camera odometry for enhanced localisation, lane line detection, vehicle detection and more. Each message comes with its own timestamp which can be used in post-processing to synchronize with all other sensors, or a custom ROS-Node can be written to have the DMS timestamps recorded within the ROS framework each of the other sensors is written into.

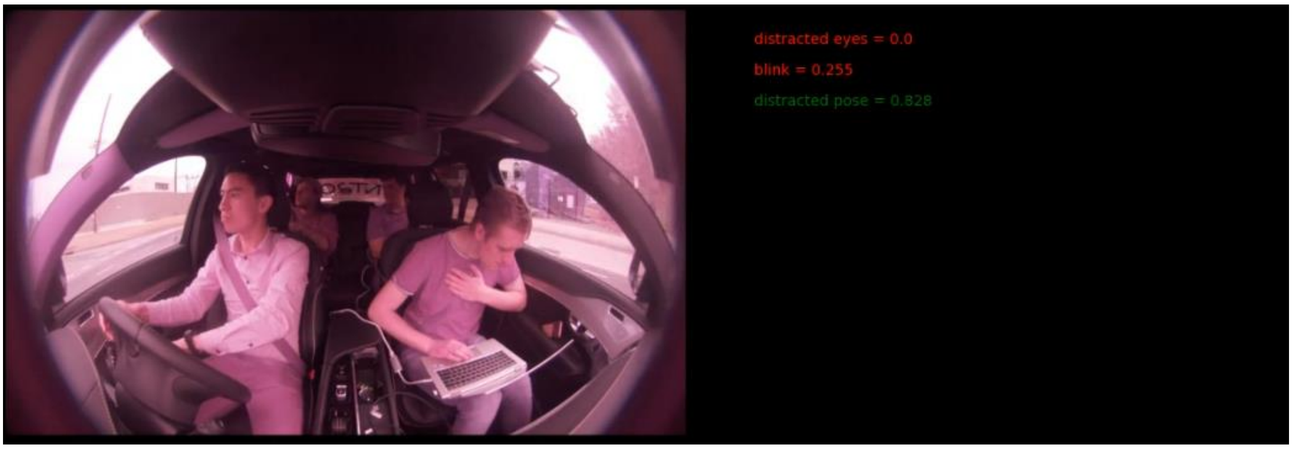
It is worth mentioning that the driving monitoring system is a part of the Advance Driving Assistive system (ADAS) integrated with the iSCAN vehicle. The device (Comma3) has level 2 automation capability if it can communicate with the Vehicle's CANBUS. The Volvo vehicle used in this project does not have compatibility with Comma3 so the vehicle cannot be driven in automated mode at this stage, however, if the sensor integration applies to a compatible vehicle[1] a level 2 automated driving is also applicable with current iSCAN sensor integration.

[1] Comma3 is compatible with other 150 vehicles https://comma.ai/vehicles

Figure 5.2 Driver monitoring camera and the driver states, a) Driver blink detection , and b) Distracted posture detected looking out side window



(a)



(b)

5 Learning and opportunities

The report shows the successful test of the hardware and software integration in iSCAB vehicle, The iSCAN project built the required knowledge base and expertise to develop different technical aspects such as perception, localisation, sensors' synchronisation and calibration. The iSCAN vehicle can be used to understand the interaction between different road users and the readiness of the infrastructure for the next generation of vehicles. The following points have been addressed by the team members as the learning and opportunities around iSCAN probe vehicles.

- **Flexible platform to integrate new sensors:** The iSCAN vehicle was designed in a way that adding new sensors do not require heavy modifications in the softwares. The new sensor is considered as an extra ROS-Node which can be separately defined with the iSCAN platform.

- **Perception**: Perception and Prediction are undeniably the most challenging part of automation. Additionally, an infallible perception and prediction system is the only way for the safe deployment of autonomous systems (Gremillion 2020). A perception system is developed in this project using the pre-trained model on a diverse dataset. However, an infallible system requires more targeted training and more than millions of miles of driving experience (real-world or custom-built simulation). The targeted training required training data that contains scenarios that are Australian specific – Melbourne hook turns, long road rails, huge mining equipment on low loaders etc. Additionally, the prediction on Australian specific wildlife like kangaroos.

- **Localisation**: The localisation accuracies can be spatially mapped to assess the performance of the localisation. Localisation is a difficult problem to solve in a geography like Australia. Our system has displayed different accuracy for localisation. However, it is necessary to spatially map localisation accuracies to discover the location where performance is poor. Weak localisation performance translates to hazardous spots for automation and henceforth necessary to investigate alternative location approaches such as simultaneous localisation and mapping (SLAM).

- **Synchronisation**: The current synchronisation is done on ROS timestamps. Since there is a lag between data logged by the sensor and then logged by ROS, the synchronisation can improve on the hardware level and post-processing level. In future, it is recommended to add hardware level synchronisation to the system to improve the data fusion applications.

- **Communication with the vehicle's sensor:** While Volvo served our purposes to develop iSCAN vehicle as a probe vehicle, we faced challenges to communicate with the vehicle to record data related to the wheel's speed, torque demand, odometer, fuel consumption, etc. Indeed, the Comma AI (i.e. ADAS system that has been used in iSCAN) is not compatible with the vehicle because the vehicle's data is not provided to the user by the manufacturer. CAN-BUS with OBDII connector is a common means of communication and most vehicle manufacturers have been used standard CAN protocol. Access to the CAN-BUS or using a compatible vehicle is recommended for future works especially if the target application is automated driving.

- **Automotive-grade camera**: Although the camera which has been used satisfied the current needs of iSCAN vehicle and provided more depth information on top of the camera footage, they are not the common cameras used in the automotive industry. It is recommended to use the automotive-grade cameras for the automated driving application so the data collected with iSCAN becomes comparable with other open-source datasets such as KITTI [1]or RobotCar[2] datasets. The changes will apply in the hardware level and the current software integration does not need a large modification.

- **Integration of V2X technology**: The iSCAN probe vehicle has been developed using open-source software integration. One of the future opportunities is to add an in-vehicle unit for Vehicle to Everything (V2X) communication, and eventually uses 5G technology for cooperative driving. This can be done by adding a Ros-Node for initialising the cooperative device.

---

[1] http://www.cvlibs.net/datasets/kitti/

[2] https://robotcar-dataset.robots.ox.ac.uk/

6 References

Bochkovskiy A, Wang C-Y, & Liao H-YM (2020) ‘YOLOv4: Optimal Speed and Accuracy of Object Detection’, *arXiv:2004.10934 [cs, eess]*, http://arxiv.org/abs/2004.10934, accessed 17 March 2022.

Bugnion P (2022) *gmaps: Google maps plugin for Jupyter notebooks*, https://github.com/pbugnion/gmaps, accessed 11 March 2022.

Chen X (2020) ‘Chapter 7 - Energy-based multisource localization’, in X Chen (ed)*Randomly Deployed Wireless Sensor Networks*Elsevier, doi:10.1016/B978-0-12-819624-3.00012-4.

Gremillion K (2020) *Perception Algorithms Are the Key to Autonomous Vehicles Safety | Ansys*, https://www.ansys.com/blog/perception-algorithms-autonomous-vehicles, accessed 24 March 2022.

He K, Zhang X, Ren S, & Sun J (2015) ‘Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition’, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9):1904–1916, doi:10.1109/TPAMI.2015.2389824.

Lin T-Y, Maire M, Belongie S, Bourdev L, Girshick R, Hays J, Perona P, Ramanan D, Zitnick CL, & Dollár P (2015) ‘Microsoft COCO: Common Objects in Context’, *arXiv:1405.0312 [cs]*, http://arxiv.org/abs/1405.0312, accessed 17 March 2022.

Liu S, Qi L, Qin H, Shi J, & Jia J (2018) ‘Path Aggregation Network for Instance Segmentation’Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, https://openaccess.thecvf.com/content_cvpr_2018/html/Liu_Path_Aggregation_Network_CVPR_2018_paper.html, accessed 17 March 2022.

Redmon J & Farhadi A (2018) ‘YOLOv3: An Incremental Improvement’, *arXiv:1804.02767 [cs]*, http://arxiv.org/abs/1804.02767, accessed 17 March 2022.

ROS: Home (2022), https://www.ros.org/, accessed 11 March 2022.

‘Unix time’ (2022) *Wikipedia*, https://en.wikipedia.org/w/index.php?title=Unix_time&oldid=1070146012, accessed 11 March 2022.

Ven BV de (2022) *Bokeh*, https://bokeh.org/, accessed 11 March 2022.

Wang C-Y, Liao H-YM, Wu Y-H, Chen P-Y, Hsieh J-W, & Yeh I-H (2020) ‘CSPNet: A New Backbone That Can Enhance Learning Capability of CNN’Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, https://openaccess.thecvf.com/content_CVPRW_2020/html/w28/Wang_CSPNet_A_New_Backbone_That_Can_Enhance_Learning_Capability_of_CVPRW_2020_paper.html, accessed 17 March 2022.

D. Huttenlocher, “Computer vision,” *Computer Sci. Handbook, Second Ed.*, pp. 43-1-43–23, 2004, doi: 10.4324/9780429042522-10.

7 ANNEX.1 The IMU/GNSS outputs

The Novotel-OEM7 provides different outputs including separated and fused data for localisation (http://wiki.ros.org/novatel_oem7_driver/position_generation). Table 7-1 shows the output attributes reported as the best positioning value.

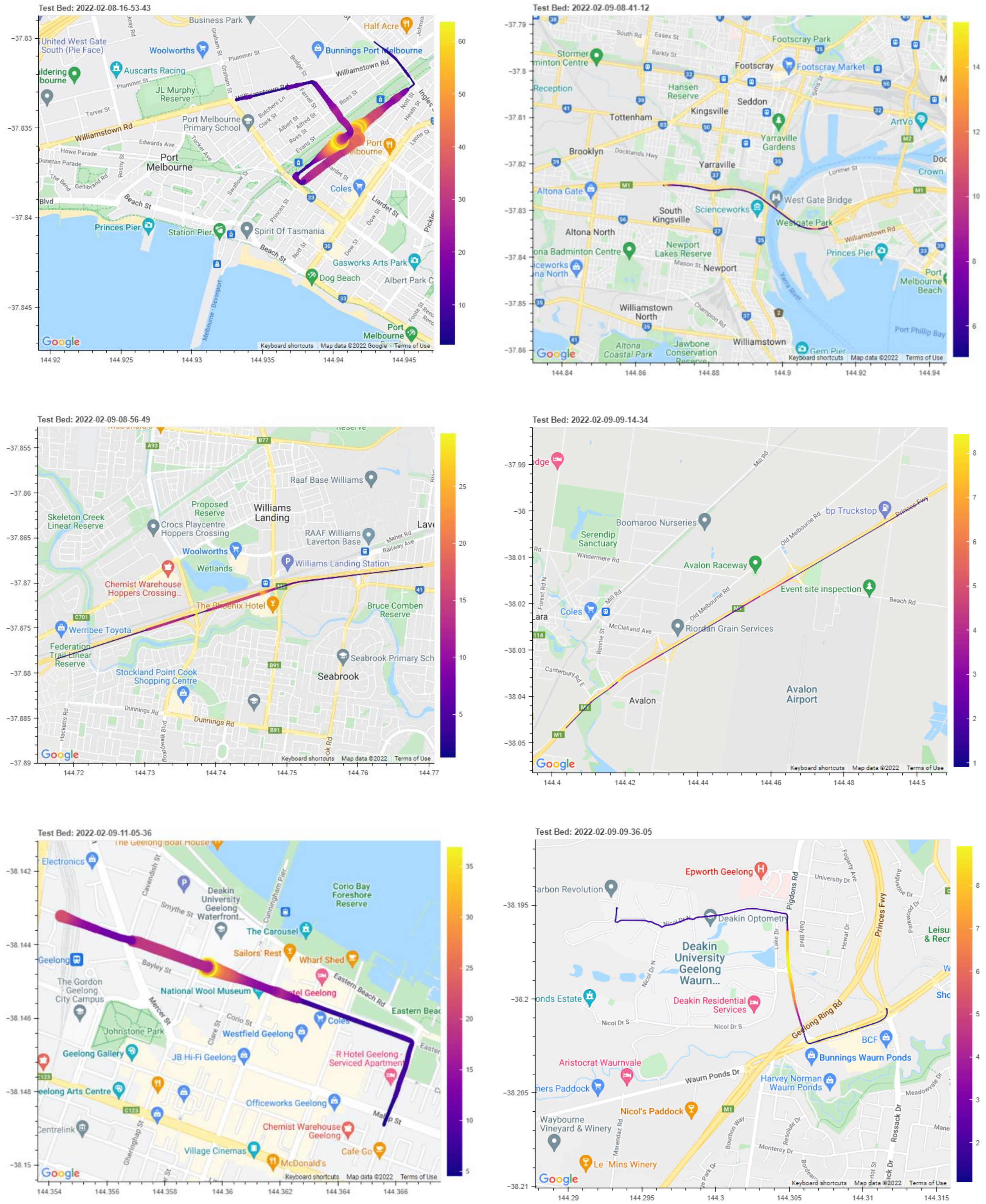
Table 7-1: Best position attributes reported by IMUGNSS solution

Field	Field type	Description	Format	Binary Bytes	Binary Offset
1	BESTPOS header	Log header. For information about the log headers, see ASCII, Abbreviated ASCII or Binary.		H	0
2	sol stat	Solution status, see Table: Solution Status	Enum	4	H
3	pos type	Position type, see Table: Position or Velocity Type	Enum	4	H+4
4	lat	Latitude (degrees)	Double	8	H+8
5	lon	Longitude (degrees)	Double	8	H+16
6	hgt	Height above mean sea level (metres)	Double	8	H+24
7	undulation	Undulation - the relationship between the geoid and the ellipsoid (m) of the chosen datum When using a datum other than WGS84, the undulation value also includes the vertical shift due to differences between the datum in use and WGS84.	Float	4	H+32
8	datum id#	Datum ID number 61 = WGS84 63 = USER	Enum	4	H+36
9	lat σ	Latitude standard deviation (m)	Float	4	H+40
10	lon σ	Longitude standard deviation (m)	Float	4	H+44
11	hgt σ	Height standard deviation (m)	Float	4	H+48

Field	Field type	Description	Format	Binary Bytes	Binary Offset
12	stn id	Base station ID	Char[4]	4	H+52
13	diff_age	Differential age in seconds	Float	4	H+56
14	sol_age	Solution age in seconds	Float	4	H+60
15	#SVs	Number of satellites tracked	Uchar	1	H+64
16	#solnSVs	Number of satellites used in solution	Uchar	1	H+65
17	#solnL1SVs	Number of satellites with L1/E1/B1 signals used in solution	Uchar	1	H+66
18	#solnMultiSVs	Number of satellites with multi-frequency signals used in solution	Uchar	1	H+67
19	Reserved		Hex	1	H+68
20	ext sol stat	Extended solution status (see Table: Extended Solution Status)	Hex	1	H+69
21	Galileo and BeiDou sig mask	Galileo and BeiDou signals used mask (see Table: Galileo and BeiDou Signal-Used Mask)	Hex	1	H+70
22	GPS and GLONASS sig mask	GPS and GLONASS signals used mask (see Table: GPS and GLONASS Signal-Used Mask)	Hex	1	H+71
23	xxxx	32-bit CRC (ASCII and Binary only)	Hex	4	H+72
24	[CR][LF]	Sentence terminator (ASCII only)	-	-	-

8 ANNEX.2 Localisation Plots

Figure 8-1 Sample of recorded trajectory with corresponding localisation error using iSCAN prob vehicle



CONTACT US

Amolika Sinha
Senior Professional
Next Generation Transport Systems Portfolio
Mobility Futures
E: amolika.sinha@arrb.com.au

Dr Sepehr G. Dehkordi
Senior Technology Leader
Next Generation Transport Systems Portfolio
Mobility Futures
E: sepehr.ghasemi@arrb.com.au

ARRB.COM.AU