

Design and Testing of a Heavy-Duty Platform for Autonomous Navigation in Kiwifruit Orchards

Mark Hedley Jones^{a,*}, Jamie Bell^{b,**}, Daniel Dredge^a, Matthew Seabright^a,
Alistair Scarfe^c, Mike Duke^a, Bruce MacDonald^b

^a*School of Engineering, University of Waikato, Hamilton, New Zealand*

^b*Faculty of Engineering, University of Auckland, Auckland, New Zealand*

^c*Robotics Plus Ltd, Neunham Innovation Park, Tauranga, New Zealand*

Abstract

Horticultural robots designed for in-field use usually require a means of transportation around orchards or farms. A common approach is to directly integrate a drive system – at the expense of increasing overall complexity. Alternatively, robots can be modularised and attached to general purpose platforms. General purpose platforms reported previously are designed to carry relatively light payloads orientated toward ground based crops. This paper presents a heavy-duty platform, capable of carrying modules of up to 1000 kg, beneath 1.4 m high pergola-style kiwifruit canopies. We discuss the design of the vehicle’s software, electrical system, and mechanical structure. Sensors suitable for autonomous navigation are evaluated in-orchard and tested with prototype algorithms. Row following trials show a repeatability of within ± 75 mm using only a multi-layer lidar. With a method of automatically turning between rows, the vehicle traversed over 10 km of orchard rows unassisted.

Keywords: Agricultural automation, autonomous navigation, sensor selection

1. Introduction

Short-term labour requirements within New Zealand’s kiwifruit industry peak twice a year corresponding with the pollination and harvesting of kiwi-

*markhedleyjones@gmail.com

**jamie977@gmail.com

Nomenclature

α	Slope angle (deg)
Δv	Change in velocity (m s^{-1})
ω	Angular velocity (rad s^{-1})
τ	Torque (N m)
B	Nominal tyre width (m)
C_{rr}	Rolling resistance coefficient (unitless)
E	Offset between a tyre and its axis of rotation (m)
F_{accel}	Force required to accelerate (N)
F_{grade}	Force due to a gradient (N)
F_{roll}	Force due to rolling resistance (N)
F_{total}	Total force (N)
G	Acceleration due to gravity (m s^{-2})
r	Wheel radius (m)
u	Friction coefficient (unitless)
W	Weight (N)
CAN	Controller Area Network
GLONASS	A GNSS service operated by the Russian Federation
GNSS	Global Navigation Satellite System
GPS	An GNSS service operated by the United States
IMU	Inertial Measurement Unit
PSA	Pseudomonas syringae pv. actinidiae (type of bacteria)
ROS	Robot Operating System
RTK-GPS	Realtime Kinematic GPS
SLAM	Simultaneous Localisation And Mapping (name of an algorithm)

fruit. The majority of employment during these peaks is filled by seasonal or casual workers (Timmins, 2009). As kiwifruit is the country's largest horticultural export by value (Statistics New Zealand, 2015), effective automation in this industry will promote economic growth. Development of an automated kiwifruit harvester with integrated drive system has previously been published (Scarfe, 2012). That work presented a platform designed to autonomously navigate through pergola-style kiwifruit orchards and had four integrated harvesting arms. Work presented here focuses on creating a platform that operates independently from orchard related tasks, such as fruit harvesting and pollination. Utilisation of a generalised base platform is increased by using it during harvesting *and* pollination seasons. Details of modules developed for use on the vehicle have been published separately (Williams et al., 2019b,c,a). Those modules have been built as part of a larger project focusing on automation in kiwifruit orchards.

Automated kiwifruit harvesting and pollination modules incorporate computer controlled manipulators and real-time machine-vision systems utilising computationally expensive neural-network based image processing. As a result, these systems are bulky and have geometric requirements dictated by sensors and the pergola-style growing system. They share the need for electrical power, air pressure, and a means of locomotion. However, they differ in the way they move when operating. The pollinating module moves at a fixed speed with minimum changes in angle, whereas the harvesting module advances a set distance between periods of being stationary. The platform must be able to self-drive in a way appropriate for each case.

While publishing details of BoniRob, a general purpose robotics platform for field use, the authors stated that "since the robot development already includes a high complexity, the application itself should be of comparably low complexity" (Ruckelshausen et al., 2009). By separating the development of the platform from other task-specific modules, the risk of a single part becoming overly complex is reduced.

The development of autonomous vehicles in agriculture is not new, but much of the literature relates to manned vehicles converted to drive-by-wire. Because the canopy of a pergola-style kiwifruit orchard can droop as low as 1.4 m to the ground under fruit loading, most standard vehicles are not suited to this environment. Many autonomous vehicles designed for use in orchards, such as vineyards, rely on Global Navigation Satellite Systems (GNSS) for guidance. However, the dense foliage of a kiwifruit canopy and the surrounding shelter-belts make receiving GNSS signals unreliable at best. Therefore,

the required vehicle must navigate through kiwifruit orchards whilst relying only on perception-based sensors. Additional requirements are to:

1. support a mass of 1000 kg,
2. have a maximum height of less than 1.4 m,
3. provide 8 kW of electrical power to modules,
4. turn between rows using existing headland areas,
5. include a bin-lifting mechanism for carrying fruit bins, and
6. provide a module mounting area no more than 400 mm from the ground.



Figure 1: The presented platform driving through a pergola-style kiwifruit orchard during winter months.

2. Related Work

The introduction of computers and digital camera technology during the 1980s sparked research into autonomous vehicles for agricultural use (Li et al., 2009). When publishing details of an autonomous vehicle in 1998, Tillett et al. cite difficulties dealing with variability in lighting and the environment as the reason no commercial vehicles were available at the time (Tillett

et al., 1998). Their vehicle combined wheel encoders, a compass, and accelerometers for odometry information. It also featured a camera based row guidance system. The system as a whole was capable of spraying individual plants whilst driving autonomously at 0.7 m s^{-1} (2.5 km h^{-1}). While their purpose built experimental vehicle proved capable of row following and targeted spraying, its design was not modular.

Four years later, two autonomous vehicles designed for weed mapping and control in open field crops were presented (Pedersen et al., 2002; Åstrand & Baerveldt, 2002). These platforms had simple chassis and drive systems as they were both at a prototype stage; neither were designed to carry modularised payloads. The first vehicle, presented by Åstrand & Baerveldt, featured: two-wheel steering, two-wheel drive, a camera based row guidance system, batteries, a combustion engine, and an air-compressor. While its appearance was basic, it contained most of the functionality required by our modularised fruit harvesting and pollination modules. The second unit, described by Pedersen et al., was four-wheel drive with two-wheel steering and used satellite navigation as its primary navigation input. It was battery powered only and lacked any sort of row guidance sensor or power generation unit. The authors found that row-crop based navigation using satellite navigation alone was not practical and proposed the integration of a row-guidance sensor in their next design. They also proposed a revised drive system with four-wheel steering.

Two years later, the revised design proposed by Pedersen et al. (2002) was presented by Bak & Jakobsen (2004). Its drive system was modularised with four identical drive/steering modules mounted to the chassis. This revised chassis featured a three-point suspension system, which ensured all four wheels stayed in contact with the ground. The system also incorporated the row-guidance sensor as proposed in earlier work, as well as a Real-Time Kinematic enabled GPS receiver (RTK-GPS), fibre optic gyroscope, compass, and wheel encoders. The authors noted that the control strategy for the four independently controlled wheels was “non-trivial”. While much more developed than the previous work of Pedersen et al. (2002), the platform was not designed to: carry heavy payloads, operate in the absence of satellite navigation, or power itself beyond its battery capacity.

In 2009, details of BoniRob were published by Ruckelshausen et al. (2009). Similar to the previous unit presented by Bak & Jakobsen (2004), it featured a gyroscope, RTK-GPS receiver, and four-wheel steering. However, it introduced the use of both single-plane and multi-layer laser range scanning,

known as lidar, for perception and row detection. A 2.8 kW petrol engine could also be mounted to the chassis, additional to its on-board batteries. It was capable of carrying a 150 kg payload in its dedicated module space. What made BoniRob particularly interesting was its ability to alter its track-width by actuating the legs to which its wheels were mounted. Like the robots before it, BoniRob was designed for use on open-field crops. During the previous year, some of these authors published details of a much simpler robot named ‘Weedy’ (Klose et al., 2008), also an open-field crop based sensing platform. BoniRob represents the first of the more general-purpose platforms designed to carry modularised payloads.

Most recently, Bawden et al. (2017) published details of their field-crop robot – Agbot II. For locomotion it uses two driven wheels in a differential drive configuration with castor wheels for support. It is battery powered and designed to autonomously return to a shipping container with a built-in solar powered charging station. The vehicle is made of two side modules bridged by a modular ‘implement unit’ containing the robotic systems. The side modules contain the drive system, whereas the centrepiece is designed to be specific to the application. Like the previously mentioned robots, its payload capacity (200 kg) is insufficient for our tasks and is orientated toward ground-based crops.

Of particular relevance, is the earlier work of Scarfe et al. on an autonomous kiwifruit picking robot (Scarfe et al., 2009; Scarfe, 2012). That work involved the creation of a hydraulically driven platform, with two-wheel steering and four-wheel drive. Four fruit-harvesting arms and a bin-lifting mechanism were directly integrated to the vehicle’s chassis. While that platform was designed to navigate through kiwifruit orchards autonomously, its ability to do so was not tested due to an outbreak of *Pseudomonas syringae* pv. *actinidiae* (PSA), which closed access to kiwifruit orchards. The platform had a petrol engine and made use of camera and lidar for row-guidance. It had sufficient carrying capacity for other roles, however it lacked modularity – restricting its use to kiwifruit harvesting.

With the exception of the platform presented by Scarfe (2012), all of the reviewed platforms were designed for use with open-field crops. None were designed for harvesting operations and therefore were not capable of carrying bins. Referring back to the statement from Ruckelshausen et al. (2009) that “the application itself should be of comparably low complexity”, one can see why research thus far has focused on simpler tasks such as inspection or weeding. However, once designs move past these applications it becomes

necessary to accommodate other shared requirements. A fork-lift mechanism is general enough that most orchard related tasks can benefit from it. For example, during harvesting it can hold a fruit collection bin. During a pollination season it can hold a tank of liquid pollen solution. The ability to pick up a standard pallet has broad applications in and around orchards too.

Reported use of GNSS systems indicate that they are not suitable for navigating row based crops on their own. With regards to the use of RTK based GNSS guidance, Slaughter et al. points out the trade-off of requiring an “unobstructed ‘view’ of the sky from all parts of the field” (Slaughter et al., 2008). Li et al. (2009) conclude that the use of either GPS and machine vision, or GPS and lidar will become a development trend. Based on the increased reception requirements, we discount the use of RTK based systems, but still consider the use of general purpose GNSS receivers as a navigation input.

3. Platform Design

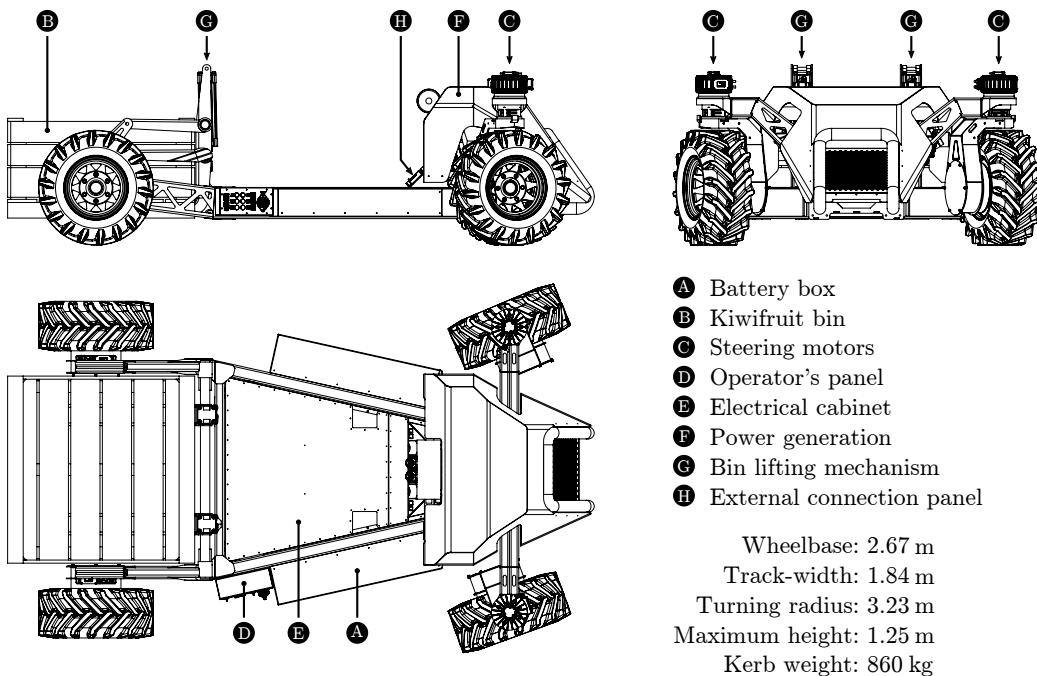


Figure 2: Profile drawings of the robotic platform with kiwifruit bin.

The vehicle's design is mostly influenced by the need to carry modularised robotic systems and fruit bins. Existing commercial platforms suitable for use in horticulture already exist, such as the Warthog from ClearPath Robotics, but the maximum payload, battery life, and vehicle geometry make them unsuitable for kiwifruit harvesting. The mass of robotic modules for pollination or harvesting can be as much as 600 kg and a bin of kiwifruit adds an additional 400 kg. The minimum canopy height in typical commercial orchards ranges from 1.4 m to 1.7 m, so the vehicle must also have a low profile. Modules carried by the platform require clearance from the canopy in addition to the height they occupy themselves. To maximise the space available to these modules the platform must be low-slung at the point they attach. Figure 2 illustrates the platform's design, with module area allocated between markers 'G' and 'H' in the side-view (top left). The top surface of the chassis in this region sits 360 mm above the ground.

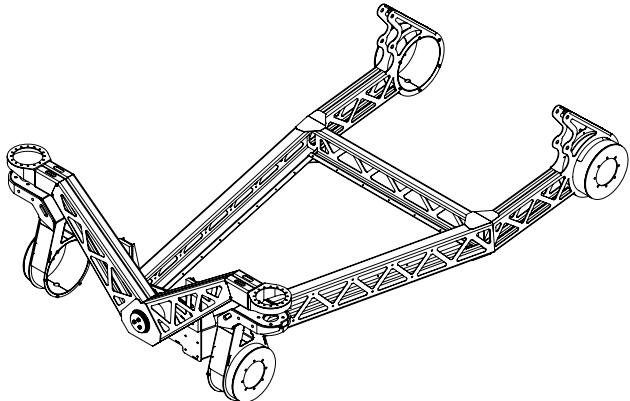


Figure 3: Drawing of the vehicle's chassis showing the front pivoting mechanism and steering linkages. The laser-cut and folded structure has a total mass of 190 kg.

The chassis is assembled from sections of 3 mm laser-cut and folded mild steel. The sections are welded together on jigs, also made from laser-cut and folded steel, before being powder-coated and assembled. Much of the folded chassis structure contains triangular cut-outs, reducing mass while having minimal impact on rigidity. Finite element analysis was used during the design phase to help identify areas needing to be strengthened and areas where material could be removed. This helped to ensure the platform met its target load capacity of 1000 kg, while the bare chassis weighs only 190 kg. A drawing of the bare chassis is shown in Figure 3.

Bin lifting forks occupy the area between the rear wheels. Fuel and compressed air tanks sit over the right-hand rear wheel, which can be seen in Figure 1. The bin lifter is actuated by two vertically mounted double-acting pneumatic cylinders (SMC CP96D100-320) which are controlled by a pneumatic valve block. Each cylinder is capable of exerting 4700 N at 600 kPa or 6300 N at 800 kPa.

3.1. Steering

The steering geometry is Ackermann based, with the front two wheels being actuated by brushless AC motors (Heinzmann PSM-G100). These motors can generate 7.32 N m of torque with a maximum angular velocity of 3000 rev/min and are rated at 2.3 kW. Their outputs are fed through fixed-ratio planetary gearboxes with a 64:1 reduction, increasing torque to 470 N m while reducing the maximum angular velocity to 47 rev/min.

Torque requirements for the steering motors are based on a static friction scenario with the vehicle loaded with a 1000 kg mass while sitting on concrete. This is described by the following equation:

$$\tau = Wu\sqrt{\frac{B^2}{8} + E^2} \quad (1)$$

where τ is the torque required to break static friction, W is the weight transmitted through each wheel, u is the coefficient of friction, B is the nominal width of the tyre, and E is the offset between the tyre's contact surface and its axis of rotation. The axis of rotation on the vehicle lies directly through the centre of the tyre, meaning $E = 0$. A value of 0.75 was used as the coefficient of friction as a best-guess representation of a tractor-grip tyre on dry concrete. The mass of the vehicle (800 kg), plus payload (1000 kg), and fuel (60 kg) adds to 1860 kg. Allowing for uneven weight distribution and a safety margin, the per-wheel mass supported is 500 kg, or a weight of 4900 N. The tyre width is 0.28 m. By combining these values, as per Equation 1, a torque of 388 N m is required to overcome static friction when actuating the steering wheels.

Actuating the steering wheels independently removes the need for mechanical linkages between them, allowing for more extreme steering angles and a simpler mechanical design. Both steered wheels have the freedom to rotate 330°, artificially limited by mechanical stops. At the tightest steering angle of 90°, the centre-point of the turn is located at the midpoint of the

rear wheels. The turning radius in this case should be equal to the distance between the front bumper and the rear wheels (3.18 m).

Implementing a four-wheel steering system would shift the pivot point to the vehicle's centre, roughly halving the turn radius, but this was deemed unnecessary. Headlands in kiwifruit orchards are sized for tractors with much larger turning radii than that of our platform. The use of a two-wheeled steering system removes the need to develop the “non-trivial” control strategies required by Bak & Jakobsen (2004). It also increases the usable area at the rear of the vehicle by removing the need for clearances around actuated wheels. A skid steer system was expected to cause ground damage to a level considered unacceptable to orchard owners.

The steering motors have incremental encoders, but no means of absolute positioning built-in. This means that the front-wheel angles must be aligned before use. A homing sequence at boot-up is used to find an absolute angle as a reference point for incremental rotation data. Inductive proximity sensors are used to detect the position of the wheels during this sequence.

3.2. Drive system

The vehicle features a three-point suspension system, similar to that used by Bak & Jakobsen (2004), to ensure all wheels remain in contact with the ground. It uses a pivoting front axle to do this and is depicted in Figure 3. As the operating speed for the vehicle is 1.39 m s^{-1} (5.0 km h^{-1}), the tyres alone were expected to provide sufficient shock absorption.

Performance requirements for the vehicle's traction system, whilst loaded, during up-hill acceleration, were calculated as follows:

$$F_{roll} = C_{rr} \times m \quad (2)$$

$$F_{grade} = m \times G \times \sin(\alpha) \quad (3)$$

$$F_{accel} = m \frac{\Delta v}{t} \quad (4)$$

$$F_{total} = F_{roll} + F_{grade} + F_{accel} \quad (5)$$

Where F_{roll} is the force due to rolling-resistance; F_{grade} is the grade (or incline) force; and F_{accel} is the force required for mass acceleration. A rolling-resistance coefficient (C_{rr}) of 0.04 was chosen as it represents the case of a pneumatic tyre on medium-hard soil (Robert Bosch GmbH, 2002). Other variables used are: a vehicle mass (m) of 1900 kg, slope angle (α) of 20° , velocity change (Δv) of 2.78 m s^{-1} (10 km h^{-1}), and an acceleration time (t)

of 6 s. Putting these values into Equations 2–5 gives a total force requirement of 7.99 kN. On a per-wheel basis this is 2.0 kN, or 729 N m when taking the wheel radius (r) of 0.365 m into account. Required traction power (P) is then calculated as follows:

$$\omega = 2\pi \times \frac{v}{2\pi r} = \frac{v}{r} \quad (6)$$

$$P = \tau\omega \quad (7)$$

where ω is the angular velocity of a wheel, v is the vehicle velocity, and τ is torque. At a velocity of 2.78 m s^{-1} (10 km h^{-1}), the calculations give a power requirement of 5.55 kW per wheel.

The selected motors are hub-mounted permanent-magnet brushless AC motors with integrated 40:1 fixed-radio planetary gearboxes (Heinzmann PSM-G120). Each motor is rated for 6.4 kW at 96 V with a maximum angular velocity of 3000 rev/min and torque of 20.4 N m. At the output of the gearbox the torque jumps to 816 N m while the angular velocity drops to 75 rev/min; giving the platform a top speed of 2.86 m s^{-1} (10.3 km h^{-1}).

In total there are seven brushless AC motors on the platform: four drive motors, two steering motors, and a motor used for electrical power generation. Each are connected to individual controllers (Sevcon Gen4 DC Size 4). These controllers are available in four input voltage options: 24–36V, 36–48V, 72V–80V, and 96V–110V. The six motors used for traction and steering are together capable of consuming 30.2 kW. With a 48 VDC bus this would equate to a current draw of 630 A. As 24 m of cabling is required to connect the motors and controllers to a common point on the vehicle, a 96 VDC bus was used to reduce the required gauge of that cable.

3.3. Power Distribution

The system's electrical bus connects the batteries and generator to motor controllers and on-board power converters. A series of heavy-duty contactors (TE Connectivity Kilovac LEV200) control each device's connection to the bus, as well as the bus's connection to the power source. Figure 4 illustrates the power-distribution on the platform.

Two battery modules attached to the sides of the chassis each house fifteen lithium-iron-phosphate (LiFePO₄) batteries connected in series. Together, the batteries (Winston/Thundersky WB-LYP90AHA) provide a nominal bus voltage of 96 V and a total electrical capacity of 8.64 kWh. The battery packs were manually ‘bottom-balanced’ before being installed and

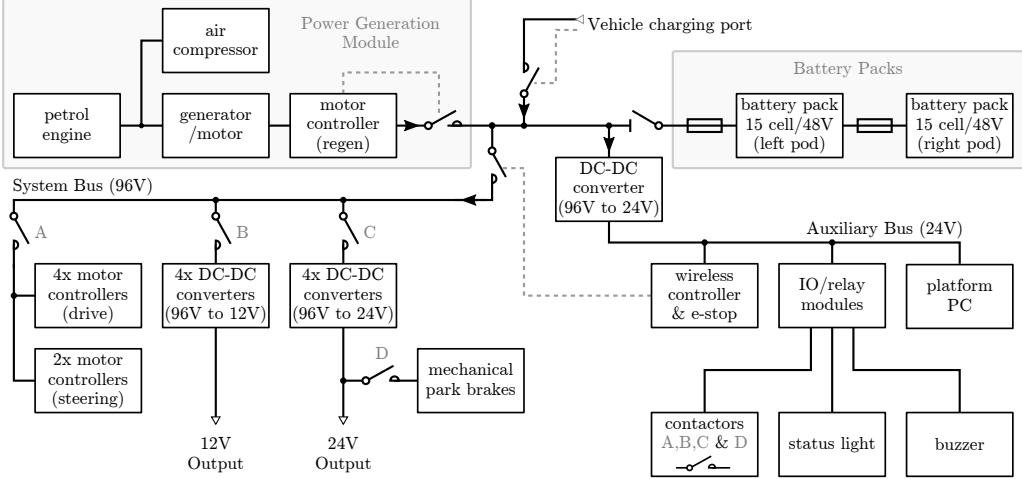


Figure 4: Power distribution system diagram. Dashed lines in grey indicate control lines to contactors.

no cell-level voltage monitoring is present. Maximum and minimum pack voltages were established by monitoring individual cells during charging and discharging. At the point that any individual cell exceeded a safe maximum/minimum threshold, the respective maximum/minimum pack voltage was recorded.

A hermetically sealed disconnect switch (Gigavac HBD41) isolates the batteries from the rest of the system. Once closed, an auxiliary 24 V bus becomes active that powers components required to bring the rest of the system on-line.

A power generation unit, comprised of a petrol engine (Honda GX-690), air compressor (Rotorcomp NK-1), and electrical generator, is housed at the front of the vehicle. Drive shafts of these units are connected via pulleys and a heavy-duty timing belt. The engine, compressor, and alternator are controlled and monitored by a micro-controller based control board. This board connects to the Platform PC via the system CAN-bus. The engine is capable of producing 16 kW, where up to 9.6 kW is converted to electrical power (limited in software) and 4.0 kW is converted to pneumatic power. The system maintains a pneumatic tank pressure of between 600 kPa and 800 kPa.

Electrical generation is done by a brushless AC motor/generator (Heinzmann PMSG-150) connected to the same model of motor controller used in

the drive system. The motor is a larger variant of those used for traction and steering, minus the gearbox. Its controller is configured only to have regenerative braking functionality, i.e., power could not be applied to the motor. This configuration allowed the system to control the rate of power generation by commanding brake-effort via its CAN interface. The controller provides voltage and current limits as well as the ability to reduce output as the batteries become charged. These settings provide all the functionality of a general purpose battery charger, making this a cost effective and versatile charging solution. Electrical energy from the power generation unit is fed to the batteries in a series-hybrid configuration. An charge plug has been fitted to allow charging of the batteries from an external source.

A fuel tank is fitted over the rear right-hand wheel (visible in Figure 1). It can hold 60l of petrol, allowing the vehicle to operate continuously for over 24 h. On-board DC-DC converters deliver 2.8 kW at 12 VDC, 3.8 kW at 24 VDC, and 3.5 kW at 240 VAC, simultaneously. A connection panel at the front of the module mounting area houses the weather-sealed plugs (Anderson SPEC PAC Series) through which these outputs are accessed.

3.4. Communications Architecture

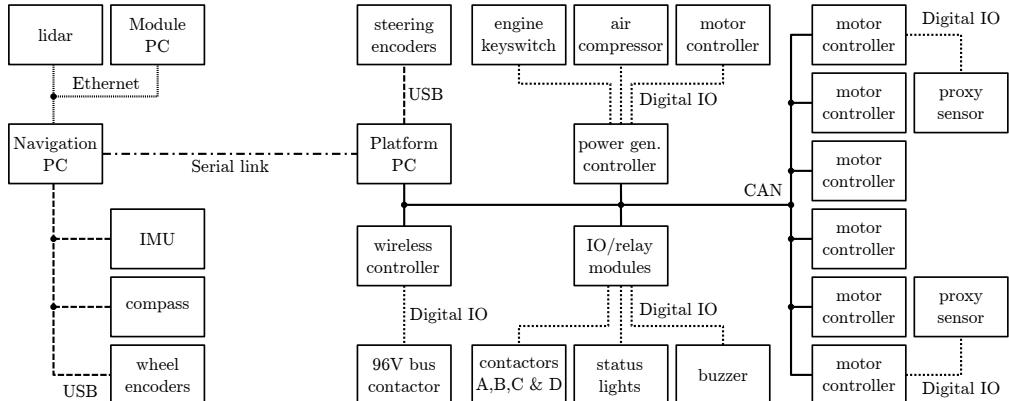


Figure 5: Communications level system diagram. Devices on right-hand side of the serial link are mechanically integrated into the vehicle, whereas those on the left are modular and can be removed.

The platform is centrally controlled by a general-purpose small-form-factor PC (Intel NUC) running Ubuntu 16.04 server edition, referred to as

the “Platform PC”. This computer communicates with most sub-systems via a CAN bus interfaced using a USB adaptor (IXXAT USB-to-CANV2).

A second computer, known as the “Navigation PC”, is responsible for higher level control of the vehicle. It is used to connect to navigation sensors, send drive commands to the platform, and perform processing tasks relevant to autonomous navigation. It too is a general-purpose PC running Ubuntu, but also contains two discrete graphics cards (Nvidia GTX 1080Ti). These graphics cards were used to accelerate neural network algorithms and some image processing functions. An Ethernet network connects this PC to the mounted payload modules, while a RS422 serial link is used to communicate with the Platform PC. Figure 5 illustrates this arrangement.

In addition to the drive commands generated by the Navigation PC, a wireless controller (HBC Radiomatic Eco) lets the operator issue drive commands via joystick. The controller’s receiver module contains relays that are directly controllable from the remote control. All inputs from this controller are also broadcast onto the CAN bus and read by software nodes on the Platform PC. The remote control has two joystick inputs, two selector switches, four buttons, and an emergency stop switch. The emergency stop switch is connected to the 96 V bus contactor via relay outputs from the receiver unit. If this switch is closed during operation, or the controller goes out of range, power to the bus is cut within 500 ms. This engages the mechanical park brakes, removes all tractive effort from the motors, and de-powers mounted modules.

The open-source Robotic Operating System (ROS) is used to facilitate communication between computers and software nodes running within each computer (Quigley et al. (2009)). Nodes written using this framework follow either a publish-subscribe or service-client pattern. To maximise code reusability, each device on the platform has its own ROS node dedicated to publishing device data or subscribing to generated device commands. Interface adapters, motor controllers, wireless controllers, lidar, and encoders are examples of devices on the platform with dedicated interface nodes. Nodes can be written in either C++ or Python and can be used simply to transform or perform calculations on data while passing it between other nodes. For instance, as shown in Figure 6, an ‘Ackermann kinematics’ node transforms a steering vector into individual wheel velocity and position/angle outputs. Among other things, ROS offers the ability to monitor and record all communication passing through it which can be replayed and examined later.

The manufacturer’s configuration of the motor controllers required them

to be interfaced using a combination of analogue and digital inputs. For example, the accelerator and steering inputs were required to be controlled by potentiometers actuated by the vehicle’s driver. However, the controllers also provide an option for a multi-motor vehicle configuration. In this configuration, the analogue inputs fed into a master controller are relayed to a second (slave) controller over a CAN interface. This interface is configured using a proprietary tool and is not intended for use other than between controllers configured with their software. However, by observing the communication protocol between a master and slave in operation, it was possible to implement a master node in software that runs on the Platform PC. With this, all motor controllers on the platform are programmed as slave devices. This allows them to accept drive commands via a CAN interface, allowing them to be directly controlled by ROS nodes.

Relay modules allow the Platform PC to toggle power to on-board power supplies, motor controllers, park-brakes, and lights. They also monitor the timing of synchronisation messages transmitted by the Platform PC onto the CAN bus. These synchronisation messages are configured to occur every 20 ms as an indication that the system is running as expected. Once a relay module detects an absence of these messages for 100 ms or longer it enters an error state. This causes the motor controllers and on-board power supplies to be shut-off and the park-brakes to be engaged. Synchronisation message monitoring is used as a fail-safe mechanism to ensure the system is promptly shut-down if the Platform PC fails.

The open-source simulation package Gazebo was used to simulate the vehicle’s steering geometry with input from a game-pad. This revealed issues that were resolved before implementation on the physical hardware. It also provided the opportunity to tune control parameters, such as steering sensitivity, while reducing the time to test.

4. Qualitative Navigation Sensor Evaluation

The choice of sensors incorporated into a vehicle determines which algorithmic approaches are available for navigation. Lidar, cameras, and GNSS receivers have been considered. Each sensor’s ability to capture relevant data is evaluated qualitatively by in-orchard trials.

Other sensors considered for inclusion are outlined in Table 1 along with their associated issues. Factors considered were the strengths and weaknesses in the context of orchard use, reported usage in literature, and availability

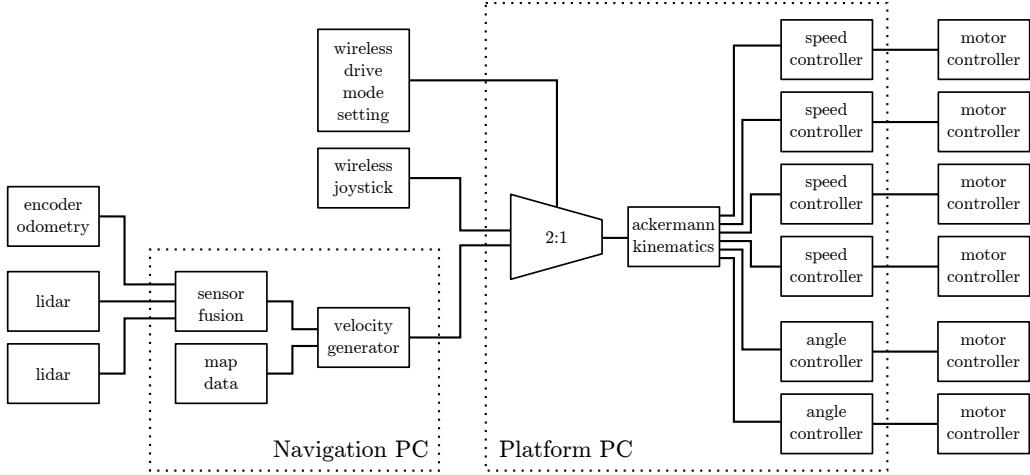


Figure 6: Simplified diagram showing connectivity between ROS nodes used for manual and autonomous platform control.

at a suitable price. A review of previous works highlighted both lidar and 2D cameras as offering high functionality for navigation and object detection. Time-of-flight cameras were a compelling option based on a cost-benefit analysis, especially if cheaper units worked outdoors in the presence of sunlight. Because localisation is such a key function, the performance of two GNSS receivers was also evaluated.

Sensor Type	Common Issues
GNSS receiver	Prone to signal loss from surrounding foliage
Inertial Measurement Unit	Error accumulation and thermal drift
Digital Compass	Prone to disturbance by nearby metallic structures
Encoder	Error accumulation
Lidar	Reduced visibility in fog and heavy rain
Time of Flight Camera	Reduced visibility in sunlight, fog and heavy rain
Camera	Reduced visibility in fog or direct sunlight, blurring
Thermal Camera	Reduced visibility in conditions of low thermal contrast

Table 1: Sensor types considered for inclusion on the platform.

As the drive motors have built-in wheel encoders, basic odometry data was already available. Encoders on driven wheels will give false readings if wheel slippage occurs so are not be used for odometry alone. However, the data provided can still be used to assist with mapping, localisation, and provide velocity feedback.

4.1. Camera

Three types of camera were tested: time-of-flight, 3D stereoscopic, and traditional 2D cameras. Smaller platforms (Clearpath Husky and Adept MobileRobots Pioneer P3-AT) were used to gather data used for evaluation. Cameras were mounted 0.8 m above the ground, roughly mid-way between the ground and the canopy, facing forward.

The time-of-flight camera was a Basler TOF640-20GM-850NM. It provides range, intensity, and confidence data at a resolution of 640 by 480 pixels. This specific model was chosen as it had previously proved useful when collecting depth data of kiwifruit canopies. During that time it had been operated under a range of lighting conditions and exhibited minimal occurrences of data loss. In those conditions the camera was mounted with its principal axis aligned vertically, pointing upwards to the canopy. However, subsequent testing with the camera mounted with its principal axis aligned horizontally revealed significant data loss in both sunny and overcast conditions. This is thought to be the result of two factors. The first is a lower reflectivity of objects in view of the camera when facing forwards, as opposed to facing up at a leafy canopy. The second is due to a dramatic increase in distance between the camera and the scene's subject matter. As the camera relies on active illumination of the scene, its ability to detect that illumination amongst ambient light will drop sharply with distance.

The 3D stereo camera tested was an Intel RealSense R200. It combines a stereo pair of infra-red cameras with a colour camera. Additionally, it features an infra-red projector as a means of adding texture to objects in its field of view to assist with stereo processing. The appealing characteristics of this sensor were its low cost and its claim of being long-range and able to work outdoors. However, in both overcast and sunny conditions it suffered from a *complete* loss of range data. This appeared to be the result of ambient light interfering with the infrared projector's signal.

Traditional, 2-dimensional, cameras trialled were the Basler Dart daA1600-60uc, Flir CM3-U3-13S2C-CS, and a Logitech C920 web-camera; sample images are shown in Figure 7. The Logitech C920 suffered from significant motion blur that, being a consumer grade web-camera, was not surprising. It also lacked the functionality of a hardware trigger and sent images with significant latency, measured at 150 ms. The Basler and Flir cameras both produced images of sufficient quality and featured hardware triggering. The Basler camera had a USB3 interface and an average image transfer time of 14 ms. The Flir camera had a USB2 interface and an average image transfer

time of 65 ms. The Basler offering was favoured for its later model image sensor, simpler software interface, and lower-latency.



Figure 7: Example images captured from trialled 2D cameras. Basler Dart daA1600-60uc (left), Flir CM3-U3-13S2C-CS (centre), Logitech C920 web-camera (right).

Both the time-of-flight and 3D stereoscopic camera systems were deemed unsuitable based on the occurrences of data loss. Images from the industrial 2D cameras (from Basler and Flir) were deemed suitable for object detection and classification. This was verified by processing the data using readily accessible detection algorithms such as convolutional neural networks. Using a pair of these 2D cameras it is also possible to build a stereoscopic pair. This provides the same functionality of the 3D stereoscopic camera from Intel, but without requiring the infra-red projector. Stereo pairs of industrial cameras have since proven useful on modularised harvesting and pollination modules for localising fruit and flowers, but were not tested for row following.

4.2. Lidar

Three lidar based sensors were evaluated, two single-plane and one multi-layer. The two single-plane lidar were the Hokuyo UTM-30LX and a SICK LMS111. The multi-layer lidar is a Velodyne VLP-16 which has 16 horizontal 360° layers separated over 15°. Data was collected from each sensor while driving through orchard rows with the units placed 0.8 m above ground level.

The intention was to use lidar as a means of detecting structure defining features of the orchard, such as posts, trunks and hedges. Detecting these features should allow for row boundary detection and general mapping and localisation. However, both single-plane lidar units produced clouds of unstructured data amongst the structured features; this is shown in Figure 8. The cause was the lidar's scan plane intercepting with the above canopy whilst driving over convex terrain. Similarly, this issue arose on concave

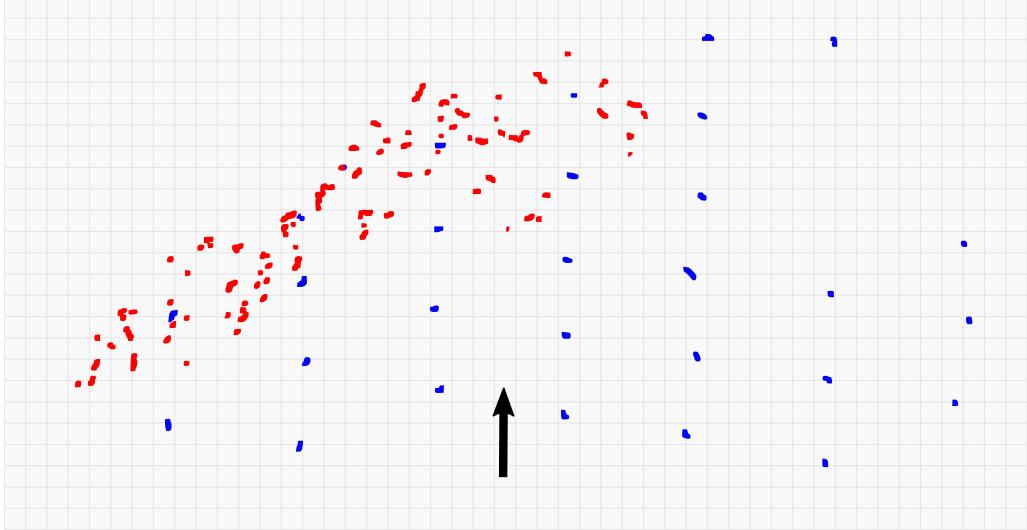


Figure 8: Birds-eye view of data captured from a single plane lidar showing non-structural points reflected by the canopy (indicated by red markers) and structural points from tree trunks and posts (blue markers). The arrow indicates the position and heading of the platform at the time of capture.

terrain when the plane intercepted with the ground. These situations are depicted in Figure 9.

The issue was reduced by the use of a multi-layer lidar and post-processing the scan data. Having sixteen layers available meant it was possible to select a scan layer that gives the most useful viewing range. Referring again to Figure 9, that would correspond to the dotted line above the horizontal (dashed) line which intercepts with a row defining feature (a tree trunk).

It was decided that a multi-layer lidar would be best suited for navigation due to its ability to see more distant features while driving on undulating ground. A single-plane lidar could still be used at short range as an independent channel of processing for redundancy or obstacle detection.

4.3. GNSS

Two GNSS receivers were evaluated: a Ublox Neo-M8N module and an OmniSTAR 5120VBS with AX0 series antenna. Both were connected to a single board computer (Beaglebone Black). The Ublox module was selected for its high sensitivity and internal low-noise amplifier. It was capable of receiving GPS, Galileo, GLONASS, and BeiDou GNSS signals concurrently.

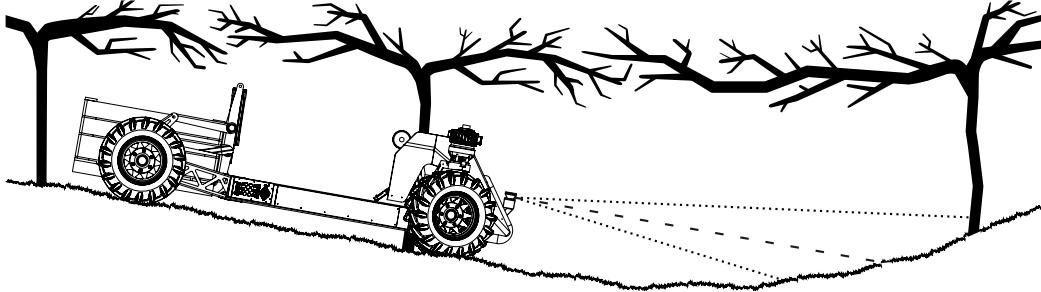


Figure 9: On concave slopes the lidar scan plane meets the ground instead of striking row-defining features. The dashed line shows a horizontal plane coming from the the lidar. Dotted lines represent the most upper and lower layers from the multi-layer lidar.

The OmniSTAR receiver was chosen for its external high-gain antenna (34 dB) which claims multi-path rejection. It was capable of receiving only GPS signals.

The testing procedure first involved planning a path through a single row of a kiwifruit orchard. The receivers were then tested separately over the course of approximately two hours by walking them along the planned path. Before testing, each unit was powered up and given 30 min to initialise in an open area near the kiwifruit orchard. During testing, each unit was walked slowly along the predetermined path with stops at each waypoint to provide time for a positional fix. The path was approximately 500 m in length and took approximately 15 min to complete, including stops at each waypoint. Way-points were spaced at intervals of 5.5 m along the row.

The path followed, together with coordinates collected from the receivers, are presented in Figure 10. It should be noted that data has been recorded for the round-trip so represents two passes along the path. It was noticed during testing that the signal quality lights on both receivers regularly indicated a loss of signal.

The Omnistar unit appears to track the approximate path well, but the data is sparse with regular loss of signal after entering the orchard. The Ublox unit collected more data, but was much less accurate. It may be possible to use a unit such as the Omnistar, which provided fewer but more accurate readings, as a sanity check for an approximate location within orchards. Overall, the units could not be relied on for localisation in this environment. These results indicate that GNSS receivers with similar performance to those trialled are unsuitable for use in kiwifruit orchards.

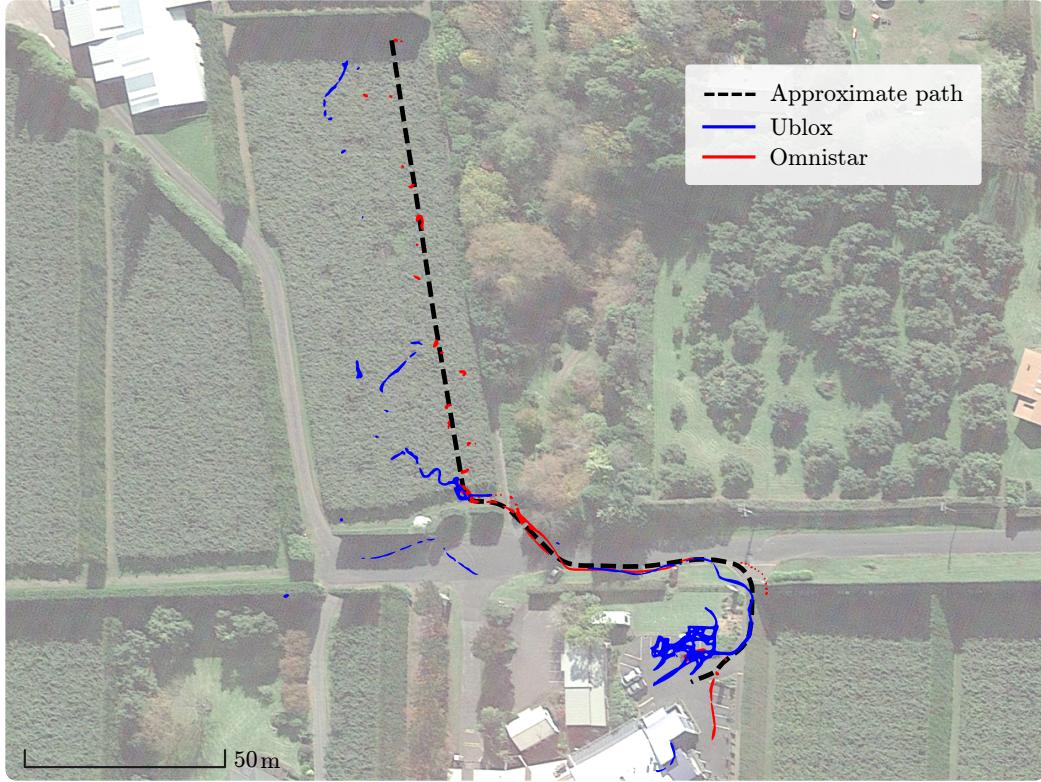


Figure 10: Aerial view of the path taken through the test orchard and the captured GPS data. Dashes, representing the approximate path, are spaced at intervals of 5.5 m.

5. Autonomous Block Traversal

Initially, two row following strategies were developed for block traversal. Details and results of each have been published separately. One method used a multi-layer lidar to detect the posts and trunks of the orchard and follow a centre-line between them (Bell et al., 2016). This involved segmenting points related to posts/trunks and then linking them together using a nearest neighbour approach. Because posts/trunks in the same row-line are nearer than those of the adjacent row, this approach resulted in the generation row boundary lines. Once each side of the row was determined, linear and angular offsets were calculated between the row's centre and the vehicle's pivot axis. Those offsets were then used to steer the vehicle so as to maintain the desired heading.

The second method used a single camera combined with a convolutional

neural network to segment areas within the captured image stream (Bell et al., 2017). These segmented areas were: traversable space, tree-lines, and row-ends. The area of the image marked as traversable space was used to calculate a centreline based on the area’s pixel-width in the image. That centreline was then used to generate a control vector to keep the vehicle centred with respect to the traversable area.

Both algorithms were developed on smaller, commercially available, platforms while the presented platform was being fabricated. A laptop (Dell E6410) with integrated graphics processor (Nvidia M5000M) was used on those test platforms to process sensor data and generate drive vectors. Both approaches produced paths that led to reproducible row following behaviour.

To determine when the vehicle was at the end of a row, the multi-layer lidar was used to detect the absence of canopy in a volume above the front and to the sides of the vehicle. The camera based method was unable to detect this end-of-row condition which is necessary for initiating the turn. It also lacked the ability to locate obstacles, which is critical for our target platform due to its size and power. Also, the lidar based approach required much less computational power to achieve similar performance. While it would be possible to combine the approaches, only the lidar based method was adapted for use on the target platform.

The multi-layer lidar was mounted horizontally above the front-right steering motor, visible in Figure 1 and in Figure 12. The smaller platforms used a skid-steer geometry, whereas our platform uses Ackermann steering geometry. When turning, a skid drive platform pivots along a lateral axis passing midway between the front and rear wheels. On the target platform however, that axis is aligned along what would be the rear axle. The software was modified to account for the change in pivoting axis as well as the new mounting location for the lidar.

Figure 11 presents a state diagram illustrating the steps taken when autonomously traversing an orchard block. The multi-layer lidar is used to detect whether the vehicle is *in_row* or *out_of_row* based on the detected presence of canopy. A row-end turn sees the platform execute a series of turn segments which have previously been tuned for the specific row number and turn direction. To start with, a template set of turn-segments is executed at each row’s end while under observation. If the vehicle gets too close to obstacles or nearby boundaries during execution, the operator would intervene before collision occurs.

A complete row-end turn can contain any number of ‘turn segments’. A

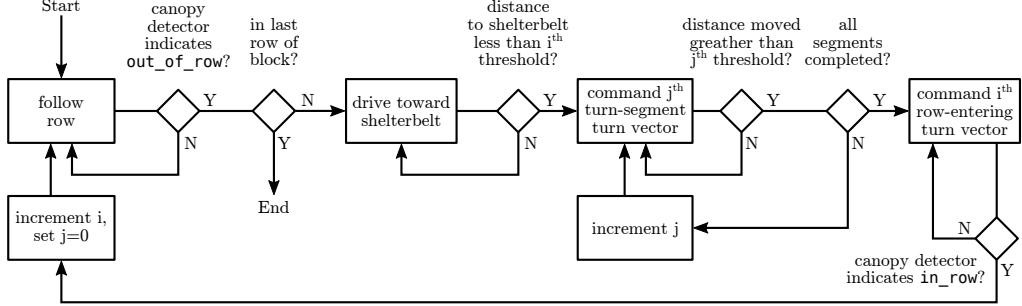


Figure 11: Flow-chart of the developed autonomous block traversal algorithm.

‘turn segment’ is simply a vector and an end-condition. Once the operator intervenes, he/she will tweak relevant parameters of the turn. This can be widening or tightening as well as lengthening or shortening the distance or angle of each segment. Finally, if an object is detected in the vehicle’s path during a turn, the steering angle is automatically increased so as to avoid the object, or the vehicle stops if the object is unavoidable. This happens independently from the parameters contained in the map. Figure 12 shows the platform performing a row-end turn while under autonomous control.

5.1. Pure Row-following

A third row-following algorithm was developed which was based on Bell et al. (2016), but with three key differences. The first was that the algorithm made no attempt to turn between rows or avoid obstacles; it simply followed a row until it detected the row’s end.

Secondly, point-cloud data was filtered based on each point’s height above the ground. This involved first filtering out points that were expected to represent the ground and using them to generate a representative 3D-surface. That surface was then used to filter the remaining points based on their height above the surface/ground; not so far as to potentially be part of the overhead canopy, but not so close that they may be weeds or related to the ground itself. Further processing checked the geometric relationship of the remaining points and made predictions about the location of posts/trunks. The process involved evaluating groupings of four posts/trunks to find those likely to represent adjacent pairs on either side of the row – forming a rectangle. These rectangles were then evaluated using a cost-function to sort for those having regular internal angles, correct orientation, and appropriate



Figure 12: A row end turn being performed autonomously by the platform. The multi-layer lidar (Velodyne VLP-16) is visible above the front-right steering motor.

side lengths. A similar process was then repeated to join the rectangles into lattices, representing a continuous section of orchard row. Such an arrangement can be seen in Figure 17 where a ladder-like structure is used to connect the posts/trunks in the generated map. Again using a cost-function, each lattice-section-proposal was evaluated based on the regularity of the areas of the rectangles they were made up of, the lattice’s overall linearity, and the number of rectangles it contained. The highest scoring lattice was then used as the best interpretation of the row’s geometry. This process was repeated for every data frame sent from the lidar (approximately 10 Hz). The final stage located the mid-point of the furthest post/trunk pair in front of the vehicle, which was used as a steering target. The position of the steering target was averaged over 10 samples to both stabilise its location and smooth its transition to a newly detected post/trunk pair.

The third alteration was that the detection of the *out_of_row* condition was calculated based the vehicle’s proximity to the steering target. Once the vehicle came within 2.5 m of the steering target, it was assumed that the vehicle had reached the end of the row. The value of 2.5 m was chosen

based on the amount of area at the end of the rows relative to the spacing of posts/trunks within the row.

6. Testing

6.1. Mass Loading

Structural integrity testing was carried out by mounting a 1100 kg mass to the vehicle's module area. No deflection of the vehicle's chassis structure was evident upon application of the test mass. Deflection of 1.5 mm was measured between the front pivot and the wheel supports. Static steering tests conducted on a dry concrete surface showed no reduction in ability to turn while loaded with the test mass. Dynamic tests involved three instances of stopping during a 10° descent at a speed of 10 km h^{-1} . During each test the vehicle came to a complete stop within 2.0 m.

6.2. Drive System

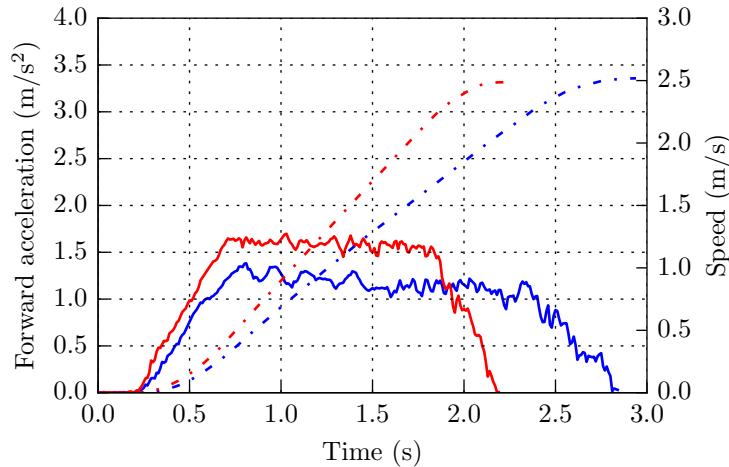


Figure 13: Acceleration of the platform on level ground (red) and up a 3.5° incline (blue).

Drive system testing saw the platform accelerate from a stand-still under manual control to its maximum speed on both level ground and whilst climbing 3.5° incline. The acceleration and inclination was measured using an Inertial Measurement Unit (IMU, LPMS-USBAL2). Figure 13 shows the

acceleration profile of the platform during both tests. The platform's weight during the test was estimated to have been (850 ± 50) kg in each case.

In both instances the vehicle reached a top speed of 2.5 m s^{-1} (9.0 km h^{-1}), 0.28 m s^{-1} short of the target speed. During the acceleration test on level ground, a peak power of between 684 W to 770 W , and torque of between 930 N m to 1046 N m was calculated per wheel. During the inclined acceleration test, a peak power of between 542 W to 610 W , and torque of between 779 N m to 876 N m was calculated per wheel. The torque calculations suggest the motors are developing their specified output of 816 N m . The motor controllers are configured to supply extra torque for short bursts, which could explain why the calculated torque on level-ground is higher than this value. The incline test began with the vehicle being held stationary using torque-control, which may have affected the controller's ability to produce the higher peak torque in this case. The lower than expected top speed in both cases suggest there are configuration issues with the motor controller's speed setting.

6.3. Turning Circle

Measurements of the vehicle's turning radius were performed at speeds of 1.39 m s^{-1} (5.0 km h^{-1}) and 2.78 m s^{-1} (10.0 km h^{-1}) on both dry tarmac and damp grassland. These speeds were calculated at the mid-point between the two front wheels. Having a wheelbase of 2.67 m , these speeds equate to angular velocities of 0.520 rad s^{-1} and 1.04 rad s^{-1} respectively.

In each test, a line was drawn on the ground in front of the vehicle to mark its starting position. The vehicle was then turned through an angle of $(180 \pm 10)^\circ$ under manual control with the steering angle set at 90° . The final angle was adjusted using odometry information to ensure the total angle was within $(180 \pm 3)^\circ$. A marker was then drawn on the ground at the front of the vehicle at this position. The distance between the markers was measured at 6.45 m for both surface types and both test speeds. This gives a turning radius of 3.23 m , 0.05 m wider than the estimate based on kinematic calculations. The authors put this discrepancy down to the accuracy of the angular calibration of the front wheels.

6.4. Bin Lifting

A pallet-mounted mass of 370 kg was used to test the bin lifter. The lifter's pneumatic valve block was manually activated until the pallet sat 250 mm above the ground. To raise the load, pneumatic pressure of 800 kPa

was applied to one port of each double acting cylinder while the other port was open to atmosphere. Lowering was done by opening both ports to atmosphere and allowing the load to descend under its own weight. This process was repeated five times. Additionally, the vehicle was driven 300 m whilst carrying the load, including an 3.5° incline for 30 m.

The lifting capacity of the mechanism was sufficient to raise the load to its target height. Whilst being actuated, signs of imbalance between the two cylinders was evident which resulted in shuddering. This shuddering was caused by excessive and unbalanced static-friction between both sides of the lifter's four-bar linkage mechanism, caused from over-tight sleeve bearings. The behaviour was also evident when hand actuating the lifter. Whilst driving, the load dropped by between 60 mm to 70 mm from its initial height. This drop, and the variation in resting position, is thought to be caused by the combination of static friction, driving related vibrations and the compression of air in the cylinders.

6.5. Turning Between Rows

Two orchard blocks (from different orchards) were used for row-turn testing. These blocks will be referred to as Block A and Block B. Block A was 1.15 km in total traversable length spread over 10 rows, while Block B was 670 m in total traversable length spread over 9 rows. After tuning the row-end turning manoeuvres, our platform navigated Block A consecutively 7 times without intervention. Figure 14 shows the number of interventions per traversal within Block A. A total of 19 traversals were used to tune the turns in this orchard. After tuning the row end-turns in Block B, it was navigated 3 times consecutively. Figure 15 shows the number of interventions per traversal whilst being tuned in Block B, with 10 traversals in total.

The weakness of the current navigation system is the need to tune the row-end turns manually for each site. The tuning required for the first orchard block amounted to eight traversals of the entire block. For the second block, seven complete traversals were required for tuning. This creates a significant resource overhead for deployment to new sites. If the turns are not sufficiently tuned, two types of failure occur. The most common case is that the vehicle turns between rows too tightly or not tightly enough and the object avoidance system is not sufficiently responsive to avoid a collision. All but four interventions were due to an imminent collision with a post, resulting from this situation. Three interventions during row-end turning were due to the platform trying to recommence row following before facing

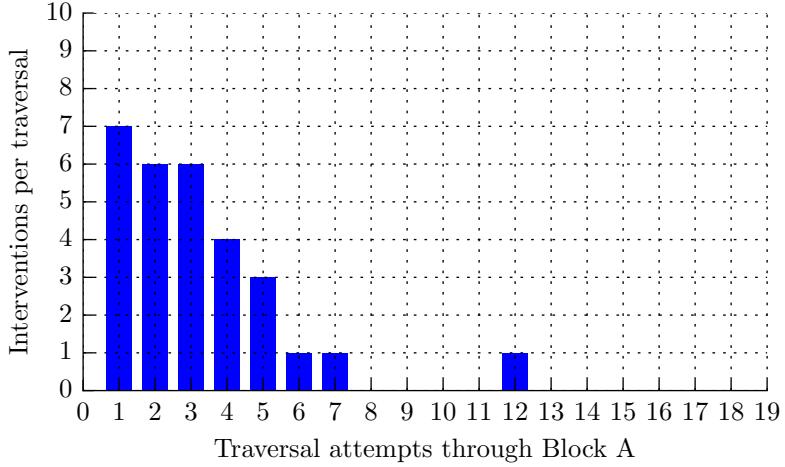


Figure 14: Number of interventions during tuning of row-end turns throughout block A.

the new target row. In this case the most feasible path for row following was detected along the headland area – instead of moving into the target row. One intervention, attempt 12 of block A, was caused by the canopy detection system triggering a row-end turn whilst still inside a row.

6.6. Pure Row-following

Row-following repeatability trials saw the platform self-drive through a single row, from approximately the same starting point, five times. Each trial was conducted at the vehicle’s target operating speed of 1.39 m s^{-1} (5 km h^{-1}). The modified version of Bell et al. (2016), as described in Section 5.1, was used for the row-following trial. At the beginning of each trial the navigation software was started with no prior information about the row.

To determine the vehicle’s trajectory for analysis, recordings of the lidar and IMU were taken and post-processed using a SLAM package off-line. The SLAM package used was Cartographer (version 1.0) that was integrated into a ROS package (cartographer_ros). It was used to calculate the vehicle’s trajectory and a 2D map of the surrounding environment (Hess et al., 2016). The spacial resolution of the SLAM map was 0.05 m/pixel. Wheel odometry provided by the drive motors was not used during the trials as signs of wheel slippage were apparent. Quick measurements conducted on grass showed an imbalance in angular velocity of approximately 5 % between the rear wheels.

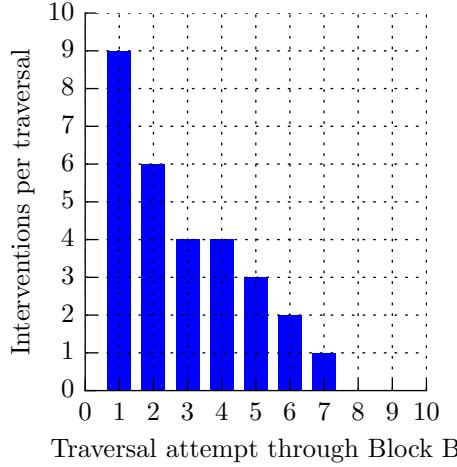


Figure 15: Number of interventions during tuning of row-end turns throughout block B.

Test number	Path length
1	106.66 m
2	103.70 m
3	104.13 m
4	104.47 m
5	104.37 m

Table 2: Total path length for each row-following test. The average length is 104.67 m and the range is 2.96 m (2.82 %).

The five trials and the generated SLAM map are presented together in Figure 16. While differences in start position are estimated to be less than 0.1 m, the trajectories show a spread of 1.69 m. We attribute this to a higher localisation uncertainty at the commencement of each trial. The total distance travelled in each trial is listed in Table 2.

Analysis of the tracking performance is particularly difficult as the row itself is not linear. Figure 17 shows the same five trajectories, but has been annotated to show approximate row boundaries. The vertical grey lines link the post/trunk pairs used to calculate row mid-points. Calculating the mid-point between each post/trunk-pair gives an idea of row's centre at forty locations along the row. Due to differences in the starting and end positions of each trial, each trajectory intersects 39 of these post/trunk-pair lines. The distance between each of these points and the five trajectories was measured

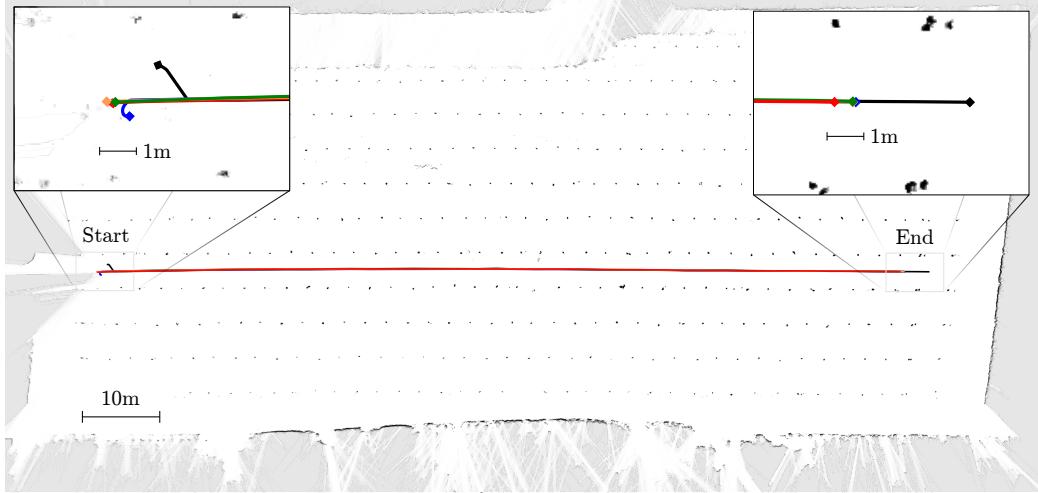


Figure 16: Trajectories recorded during five row-following repeatability trials. The trajectories are overlaid onto the generated SLAM map, which depicts the trial orchard block. Each colour represents one of five trials.

to quantify the deviation between each trajectory and the row’s centre. Figure 18 presents these centreline deviations along with error estimates arising from our measurement method. Sources of error were ± 1 pixel at each end of the post/trunk-pair lines as well as measurement rounding errors. In this analysis we make the assumption that the generated SLAM map was perfectly accurate. The maximum recorded displacement from the centreline was 326 mm, which occurred at the twenty-second post/trunk pair.

Figure 19 compares each trajectory to the average trajectory. The data presented in this figure makes no reference to the row’s centre, instead each trajectory is compared only to the average of the five trajectories. The distance between each trajectory and the average was calculated along the 39 post/trunk-pair lines shown in Figure 17. It shows that the worst case repeatability was less than ± 75 mm.

Finally, Figure 20 shows the final resting position of the vehicle after each trial. The path-following algorithm determines the end-of-row position based on its proximity to the heading target – placed between the furthest detected post/trunk pair in front of the vehicle. The end posts of each row are spaced slightly further than is usual inside the row. That extra spacing places the final post-pair at the boundary of the software-defined region used to detect post/trunk pairs. Stopping slightly earlier results in

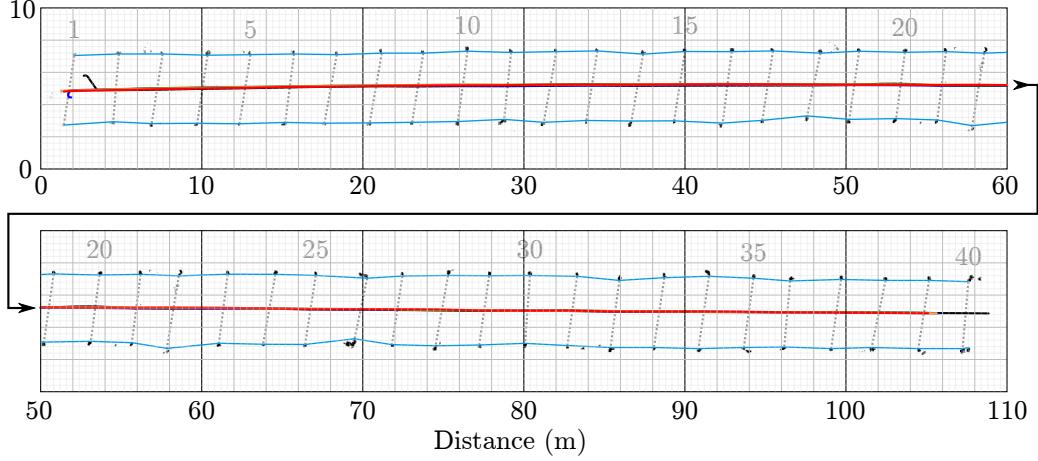


Figure 17: Cropped SLAM map with five row-following trials overlaid. Blue lines link posts and trunks along each boundary line. Grey dotted lines link post/trunk pairs, the mid-point of which define the row’s centre. Grey numbers indicate the post/trunk pair numbers relative to the start of the row.

the last post/trunk pair not being detected, as is the case in path groups A and B. Additionally, there is ambiguity in the position of the second-to-last post/trunk pair because vines have been planted in close proximity to the posts. Analysis of the recorded sensor data suggests sensitivity to this ambiguity to be the cause of separation between points A and B.

7. Discussion

The reported platform meets the requirements outlined in the introduction and has been trialled during three pollination and harvesting seasons. However, during those operations the vehicle was mostly operated under manual control because of the need to drive close to row boundaries. The width of the robotic modules meant it was necessary to perform two passes through each row in order to access the full canopy area.

Results from navigation tests indicate that multi-layer lidar with wheel-encoder feedback is sufficient for row-turning tasks. The method for turning between rows calls for further work. Row-end turning was a manual process that involved observing pre-calculated turns and manually adjusting the length or radius of path segments. Future work will focus on enabling the system to plan row-end turns based on perception based sensor data without the need for a pre-computed map. Observationally, detecting the row’s end

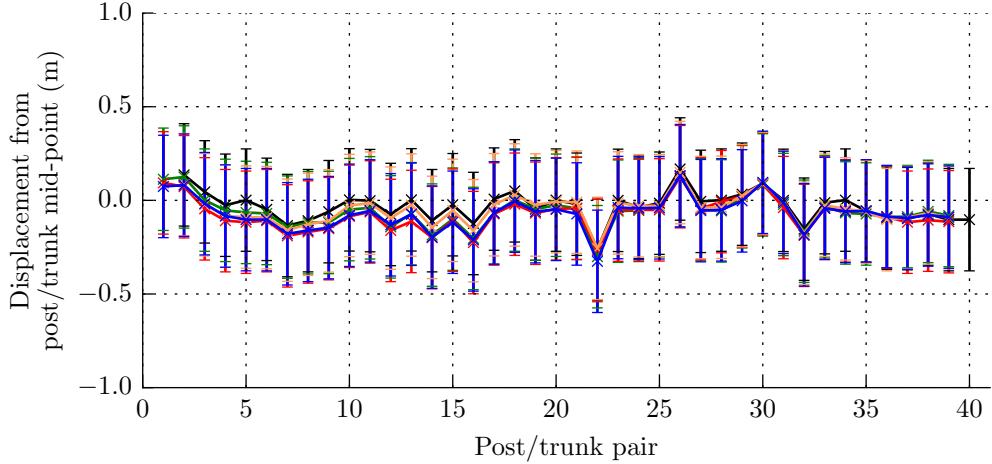


Figure 18: Vehicle displacement from calculated row centre-points during five row-following trials. Trace colours correspond to the paths presented in Figure 16. Error bars represent a ± 1 pixel error (± 0.274 m) from the rendered SLAM map, used to calculate the row’s mid-points.

based on the vehicle’s proximity to the last post-pair proved more reliable than detecting the presence/absence of canopy.

For pure row-following, without obstacle avoidance, a multi-layer lidar algorithm produced paths that were repeatable to within ± 75 mm. Our analysis showed a worst-case deviation of 326 mm from the row’s mid-point during row-following, however this figure appears to mostly be influenced by the non-linearity of the row itself. As Figure 18 shows, the worst case error for each trial occurred between post/trunk pair twenty-two. Matching this with the corresponding post/trunk pair in Figure 17 shows the placement of the right-hand post/trunk is likely to be a contributing factor. Our method of measuring deviation of the vehicle’s trajectory from the row’s centre calls for improvement.

The structure of the pergola-style kiwifruit orchard made generating odometry information directly from lidar scans reliable enough to replace wheel encoders and IMU based odometry. This odometry source was used by the pure-row-following algorithm presented in Section 5.1. It was generated by the SLAM package (Cartographer) by matching subsequent lidar frames. As it was based on visual information it was much less prone to drift than the wheel encoders or IMU, although this was not quantified.

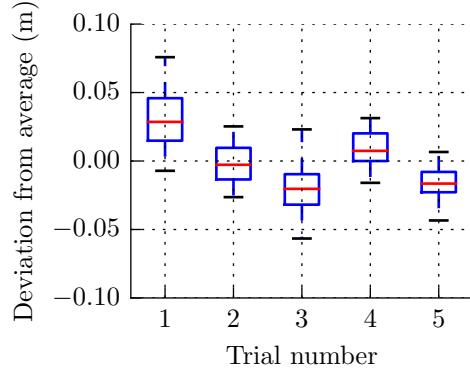


Figure 19: Repeatability analysis of the five row-following trials. Each column compares displacements of a single trajectory, from the row’s centre, to the average of the five trajectories. Whiskers represent maximum deviations in both the positive and negative directions.

The platform’s 96 V battery pack and electrical system introduced a electrical hazard that caused delays during development, assembly, and testing. The authors suggest a voltage of 48 V for similar developmental vehicles as it bears a reduced risk of injury from electrical shock. Inputs of 48 V are supported across a wider range of motors, motor controllers, and power converters, but cabling requirements are increased.

The series-hybrid electrical configuration allowed the vehicle to drive and provide power to subsystems without running the petrol engine. This was useful in testing scenarios, where people are in close proximity to the vehicle, as it eliminated exhaust fumes and reduced noise and vibration. However, robotic modules and the bin-lifting mechanism required pneumatic pressure to function. As the air-compressor was belt driven from the petrol engine it was necessary to frequently run the engine. An electric air-compressor would allow the system to run without the petrol engine for much longer periods.

The use of more general purpose platforms to test navigation algorithms enabled the navigation software to be developed in parallel with the physical hardware. Their smaller size eliminated the risk of serious injury and led to a speed-up in development and test cycles. It also meant that navigation testing could continue while the full sized platform was engaged in other activities.

An added benefit to the modularisation of the robotic harvesting and

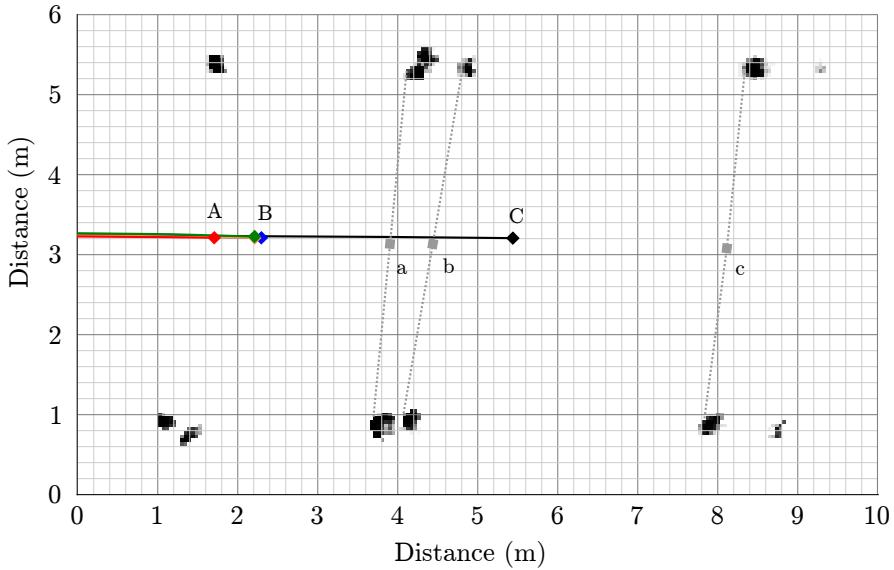


Figure 20: Graph showing row-following end positions relative to tracked row features. Grey dotted lines connect row feature pairs, with squares indicating their mid-points.

pollination systems was the ability to operate them independently from the vehicle. In the event that the vehicle required repair or maintenance work during a harvesting or pollination season, the modules could be mounted on a trailer and towed behind a quad-bike until the vehicle returned to service.

8. Conclusion

We present a platform designed specifically for autonomously transporting task-specific modules and kiwifruit bins through pergola-style kiwifruit orchards. The vehicle is capable of carrying over twice the mass of similar platforms reported previously. Calculations of the drive system's power and torque requirements, as well as the chassis' design, have been presented. A four-wheel-drive system with two individually-actuated steering wheels proved suitable for use in and around kiwifruit orchards. Subsequently, we deem the use of a four-wheel steering configuration in this environment to be unnecessary.

A variety of sensors deemed suitable for navigation were trialled in-orchard. Multi-layer lidar proved to be the most versatile sensor for orchard based navigation owing to its wide field-of-view and robust outputs. Row

following trials using an algorithm based only on multi-layer lidar was repeatable to within ± 75 mm over five trials. GNSS, time-of-flight, and infra-red stereoscopic cameras proved unsuitable as row guidance sensors. 2D cameras and neural network based processing proved capable of row following, but the lidar based method was more robust and less demanding of computation resources. Using a map of manually adjusted row-end turns, the platform has navigated over 10 km of orchard rows using only wheel-encoders and a single multi-layer lidar. A significant amount of work was required to tune the row-end-turns, which has implications for commercial deployment. Future work will focus on enabling the vehicle to plan its own row-end turns based on sensory data.

Acknowledgements

This research was supported by the New Zealand Ministry for Business, Innovation and Employment (MBIE) on contract UOAX1414. The authors acknowledge contributions from Phillip Ross, Gordon Neshausen, Josh Barnett, and Erin Simms who were involved with design and fabrication.

References

- Åstrand, B., & Baerveldt, A. J. (2002). An agricultural mobile robot with vision-based perception for mechanical weed control. *Autonomous Robots*, 13, 21–35.
- Bak, T., & Jakobsen, H. (2004). Agricultural Robotic Platform with Four Wheel Steering for Weed Detection. *Biosystems Engineering*, 87, 125–136.
- Bawden, O., Kulk, J., Russell, R., Mccool, C., English, A., Dayoub, F., Lehnert, C., & Perez, T. (2017). Robot for weed species plant-specific management. *Journal of Field Robotics*, (pp. 1–21).
- Bell, J., MacDonald, B. A., & Ahn, H. S. (2016). Row following in pergola structured orchards. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 640–645).
- Bell, J., Macdonald, B. A., & Ahn, H. S. (2017). Row Following in Pergola Structured Orchards by a Monocular Camera Using a Fully Convolutional Neural Network. In *Australasian Conference on Robotics and Automation (ACRA)* (pp. 133–140).

- Hess, W., Kohler, D., Rapp, H., & Andor, D. (2016). Real-time loop closure in 2D LIDAR SLAM. In *Proceedings - IEEE International Conference on Robotics and Automation* (pp. 1271–1278).
- Klose, R., Thiel, M., Ruckelshausen, a., & Marquering, J. (2008). Weedy – a sensor fusion based autonomous field robot for selective weed control. *Land. Technik 2008, 2045*, 167–172.
- Li, M., Imou, K., Wakabayashi, K., & Yokoyama, S. (2009). Review of research on agricultural vehicle autonomous guidance. *International Journal of Agricultural and Biological Engineering, 2*, 1–16.
- Pedersen, T. S., Nielsen, K. M., Andersen, P., & Nielsen, J. D. (2002). Development of an Autonomous Vehicle for Weed and Crop Registration. *International Conference on Agricultural Engineering AgEng2002, Budapest*.
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., & Ng, A. Y. (2009). ROS: an open-source Robot Operating System. In *ICRA workshop on open source software* (p. 5). Kobe volume 3.
- Robert Bosch GmbH (2002). *Electronic Automotive Handbook*. (1st ed.).
- Ruckelshausen, A., Biber, P., Dorna, M., Gremmes, H., Klose, R., Linz, A., Rahe, F., Resch, R., Thiel, M., Trautz, D., Weiss, U., Doma, M., & Rahne, R. (2009). BoniRob: an autonomous field robot platform for individual plant phenotyping. *Proceedings of Joint International Agricultural Conference (2009), 9*, 841–847.
- Scarfe, A. J. (2012). *Development of an Autonomous Kiwifruit Harvester*. Doctoral dissertation Massey University, Palmerston North, New Zealand.
- Scarfe, A. J., Flemmer, R. C., Bakker, H. H., & Flemmer, C. L. (2009). Development of an autonomous kiwifruit picking robot. In *Autonomous Robots and Agents, 2009. ICARA 2009. 4th International Conference on* (pp. 380–384). IEEE.
- Slaughter, D. C., Giles, D. K., & Downey, D. (2008). Autonomous robotic weed control systems: A review. *Computers and Electronics in Agriculture, 61*, 63–78.

- Statistics New Zealand (2015). Annual Fruit Exports Hit \$2 Billion for First Time. http://archive.stats.govt.nz/browse_for_stats/industry_sectors/imports_and_exports/OverseasMerchandiseTrade_MRJun15-x1.aspx. Accessed: 2018/01/18.
- Tillett, N. D., Hague, T., & Marchant, J. A. (1998). A Robotic System for Plant-Scale Husbandry. *Journal of Agricultural Engineering Research*, 69, 169–178.
- Timmins, J. (2009). Seasonal Employment Patterns in the Horticultural Industry. <http://archive.stats.govt.nz/~media/Statistics/browse-categories/income-work/employment-unemployment/leed/research-reports/seasonal-employment-patterns.pdf>. Accessed: 2018/01/18.
- Williams, H., Jones, M. H., Nejati, M., Penhall, N., Lim, J., Seabright, M., Bell, J., Ahn, H. S., Scarfe, A., Duke, M., & MacDonald, B. (2019a). Improvements to and large-scale evaluation of a robotic kiwifruit harvester. *Journal of Field Robotics*, (pp. 1–20).
- Williams, H., Nejati, M., Hussein, S., Penhall, N., Lim, J. Y., Jones, M. H., Bell, J., Ahn, H. S., Bradley, S., Schaare, P., Martinsen, P., Alomar, M., Patel, P., Seabright, M., Duke, M., Scarfe, A., & MacDonald, B. (2019b). Autonomous Pollination of Individual Kiwifruit Flowers: Toward a Robotic Kiwifruit Pollinator. *Journal of Field Robotics*, (pp. 1–17).
- Williams, H. A., Jones, M. H., Nejati, M., Seabright, M. J., Bell, J., Penhall, N. D., Barnett, J. J., Duke, M. D., Scarfe, A. J., Ahn, H. S., Lim, J. Y., & MacDonald, B. (2019c). Robotic Kiwifruit Harvesting using Machine Vision, Convolutional Neural Networks, and Robotic Arms. *Biosystems Engineering*, 181, 140–156.