Autonomous Navigation of an Agricultural Robot Using RTK GPS and Pixhawk

Ryan Moeller
Department of Mechanical Engineering
Idaho State University
Pocatello, ID USA
moelryan@isu.edu

Taher Deemyad
Department of Mechanical Engineering
Idaho State University
Pocatello, ID USA
deemtahe@isu.edu

Anish Sebastian
Department of Mechanical Engineering
Idaho State University
Pocatello, ID USA
sebaanis@isu.edu

***Abstract*— This paper discusses the design, implementation, and performance of an autonomous navigation system built for an agricultural robot. The onboard precision GPS system allows the robot to navigate to potato plants previously identified as infected with Potato Virus Y (PVY), in order to remove them from the field efficiently. This autonomous robot is in-line with emerging technologies in the field of precision agriculture. The navigation system is based on a Pixhawk microcontroller for ease of use and affordability. Because only sick plants must be removed, an RTK GPS module from Swift Navigation was used to maximize accuracy. The supporting components, both electrical and mechanical, are covered in detail. This includes communication equipment, motors, and an enclosure to protect the components from weather and terrain. This AGV will serve as a test platform for sensors and grasping mechanisms that will eventually be integrated on a larger, more advanced, final version.**

Keywords*— *Navigation, Obstacle Avoidance, Agricultural Robot, Autonomous Ground Vehicle, RTK GPS, Pixhawk

I. INTRODUCTION

Potato Virus Y (PVY) can devastate crops with up to 80% losses, depending on the specific type of potato [1]. While there are many methods currently being used to generate PVY resistant cultivars, the virus has quickly evolved in the last 10 years and continues to have a negative impact on farmers [2]. Microbiology attempts to produce cultivars with better PVY resistance, but mechanical removal of infected plants remains a full-proof, albeit currently inefficient method to prevent the spread of PVY once it is already present in a field.

Currently, mechanical removal consists of professionals identifying the infected plants and then manually removing them. This has two major drawbacks, first and foremost, identifying PVY correctly by sight is difficult. An experienced inspector may be accurate but is slow and the plants must be mature [9]. Because of this, remote sensing techniques are often employed [3]. Remote sensing uses cameras that can sense ranges of the electromagnetic spectrum that human eyes cannot. In hyperspectral remote sensing, the electromagnetic spectrum is split into hundreds of narrow "bands" of light, making hyperspectral imaging able to detect infections in plants even before symptoms occur [4], [5]. Hyperspectral cameras can be mounted on autonomous vehicles, making remote sensing a powerful technique. Remote sensing combined with modern-day Unmanned Air Vehicles (UAV) and Autonomous Ground Vehicles (AGV) allows crops of potatoes to be scanned. See Fig. 1 for how these two types of vehicles could work in tandem to achieve this.

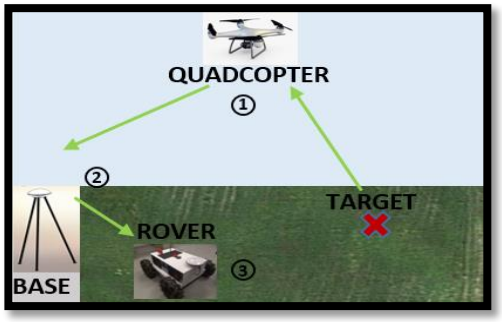

Fig. 1. UAV and AGV working in tandem. This figure illustrates the acquisition and delivery of a sick plant's GPS coordinates, labeled "target", using a quadcopter with hyperspectral imaging capability. The quadcopter then sends that coordinate to a rover that can remove the infected plant.

This would streamline the identification effort side, but the plant still has to be removed. This paper deals with designing a Global Positioning System (GPS) guided AGV that could carry a robotic arm to each identified plant and remove it by an automatic roguing mechanism. The roguing mechanism will be selected between various possible mechanical designs preferably with minimum actuation and weight [10], [11], [12]. Also, for the actuation system, soft robotic actuators [15] can be consider as a proper option.

Robots have been increasingly researched for agricultural applications, whether it be for fertilization, pesticide application, or inspection [6]. Robots can be superior to large machinery as they can be more selective where they spray, conserving resources. In addition, in some applications, it is more cost-efficient to send a robot than a manual operator [8]. Another way is using multi-robot exploration and task planning algorithm based on unknowingly distributed tasks over the field [13], [14].

In recent years, GPS has been a popular solution for autonomous robot navigation [7]. Being an agricultural robot for potato fields, obstructions such as buildings and trees are minimal. Obstructions such as these can be a downside to GPS. Poor weather can also be a factor, however, the bulk of the

growing season for potatoes is during summer months when the weather is generally calm.

The GPS network is just one of many in the Global Navigation Satellite System (GNSS). GNSS receivers work by measuring the time needed for a signal to reach the receiver from a satellite. Transferred signals traversing through the atmosphere may be slowed down or interfered with, causing an error. The average error of commercially implemented GPS is 2-4 m. The distance between two potato plants, in the field, is around 0.3 m (12 inches). This limited resolution would fall well short of the resolution required in the field. Therefore, to improve the accuracy of the navigation system for this AGV, a Real-Time Kinematic (RTK) system with an error of less than 0.1 m (3.93 inches) was utilized. An RTK system contains two main units; a base station unit and a rover unit. The base station unit is stationary and the rover unit is mounted on the AGV. The base station transmits position information in real-time to the rover, which helps the rover make corrections to its location, thereby achieving higher accuracy.

II. Navigation

For this robot, an RTK GPS system was combined with magnetometers to provide the robot with its position and also heading. A Pixhawk was used for the microprocessor, which calculates the error between the current position/heading and the desired position/heading and then adjusts wheel speed to correct itself. The Piksi Multi Evaluation Kit from Swift Navigation was chosen for the RTK GPS. The Piksi Multi has a powerful Xilinx Zynq 7020 with dual-core ARM Cortex processors and a 666 MHz clock speed. This allows it to perform RTK calculations quickly, obtaining an RTK fix in seconds. It is also relatively inexpensive as compared to other RTK GPS solutions. The base station and rover units each contain four major parts as shown in Table 1 & Fig. 2.

TABLE I. Piksi Multi Evaluation Kit main parts

Item #	Name	Quantity
1	Evaluation board	2
2	Lithium-ion battery	2
3	Free Wave radio modem	2
4	Survey grade GNSS antenna	2

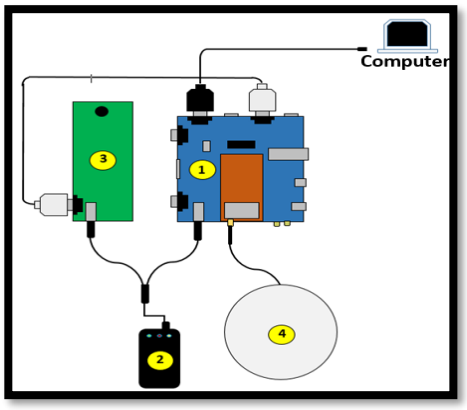

Fig. 2. Piksi Multi Evaluation Kit Main Parts – Both the base station and rover contain these four parts.

Table II has a comparison of similar systems and their most recent prices.

TABLE II. Comparison of RTK GPS Systems

Name	Cost
Piksi Multi Evalutation Kit	$ 2,295.00
Trimble R8s	$ 10,200.00
Spectra Precision SP80	$ 12,495.00
Geomax Zenith35 Pro	$ 9,995.00

A. *Evaluation of Components*

Before using the actual AGV, the components were tested using the radios that Swift Navigation provided. To set up the Piksi Multi in RTK mode, the base station and rover radios have to programmed to communicate with each other. Then, all components in Table 1 are wired to each other as shown in Fig. 2. The survey-grade GNSS antenna and Free Wave radio modem are connected to the evaluation board. The radio modem and evaluation board are powered by a lithium-ion battery. Finally, the evaluation board is connected with a USB cable to a computer. Swift Navigation provides free software called Swift Console to configure and interface with the Piksi Multi. 900 MHz or 2.4 GHz radios can be used during the evaluation of the GNSS receivers.

B. *Integration with Pixhawk*

The radios provided by Swift Navigation are only used during evaluation. The telemetry radios for the Pixhawk were used on the AGV so that RTK corrections could be sent over the same radio link that the mission planner shared with the Pixhawk. The mission planner had to retrieve the corrections from the Swift Console, a process outlined in Swift Navigation's documentation. For a brief summary of this process see Fig. 3.

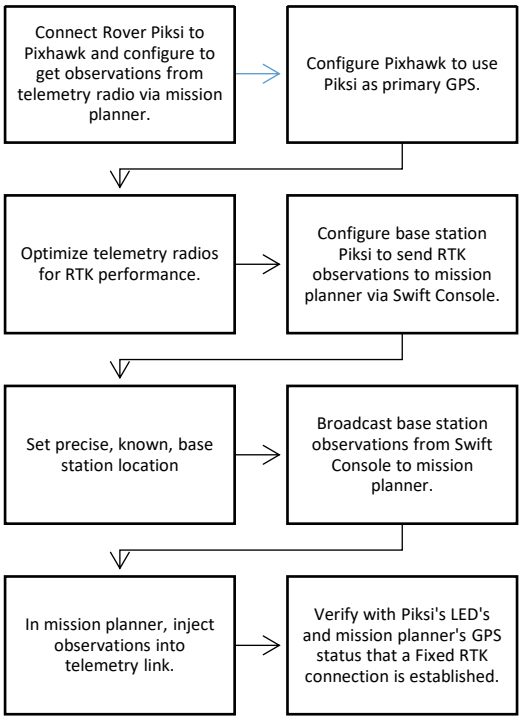

Fig. 3. Flowchart of Piksi & Mission Planner integration. The evaluation kit should be used beforehand to ensure that all components are working correctly.

Once the observations are sent to the Pixhawk, the Pixhawk can navigate autonomously with centimeter-level accuracy. Without RTK GPS, the Pixhawk performs to the accuracy level of whatever GPS module it is using.

The Pixhawk can be set to multiple modes. For this project, only three were used: "manual", "smart RTL", and "auto." Manual mode allows an RC transmitter to control the AGV with user input. More on how that was set up is covered in section III A. When in auto mode, navigation is done by the firmware loaded onto the PixHawk. In smart RTL mode, the robot will trace its path back to its starting location.

Waypoints are chosen in Mission Planner and sent to the Pixhawk via telemetry radios. When auto mode is turned on, the Pixhawk begins navigating from its current position to the first waypoint in a straight line. Once it gets within a programable radius from the waypoint, it moves on to the next one. When finished, it returns to its "home" location which is programmed with the mission planner.

In Fig. 4, "H" is for home location. Each waypoint has a number next to it to determine the order. Notice there is no waypoint number 2 on the map but there is one in the list. This is because a waypoint can be changed from a simple navigational goal to a trigger for an event via a drop-down menu. In this case, the event named DO_GRIPPER was chosen just as an example. The event triggered could be a camera shutter, grasping mechanism, etc. The event is only triggered when the AGV is within the programmed radius of the coordinate associated with the trigger. In our case, DO_GRIPPER lies halfway between waypoints 1 and 2.

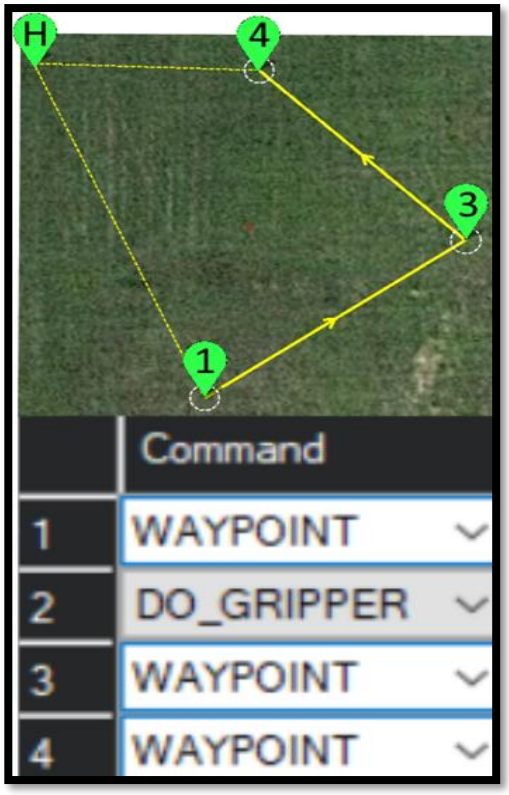

Fig. 4. Screenshots of the mission planner's navigation planning feature.

III. Rover Control System

A. *Electrical Components*

The control system for the RTK GPS revolves around the Pixhawk flight controller. The Pixhawk is an open-hardware solution for autonomous navigation and can be used for aerial as well as ground vehicles. The Pixhawk was paired with a Ground Control Station (GCS) called Ardupilot Mission Planner for affordable, customizable, and trusted performance. Along with the Pixhawk, a suite of electrical and mechanical hardware was used to have a manual as well as autonomous navigation (see Table III and Table IV). Although autonomous navigation is the goal, being able to control the robot manually provides convenience for positioning the robot for testing and also acts as a safety mechanism in case the autonomous navigation malfunctions and the robot gets lost or poses a danger to bystanders.

TABLE III. List of Electrical Connections

#	Component	Pixhawk Port Connected To	Other Connections to the Component
1	3S LiPo battery	Power	Piksi Multi Power
2	Pixhawk	n/a	
3	mRobotics SiK Telemetry Radio V2 915Mhz	Telem 1	
4	X4RSB FrSky receiver	Servo rail: ground, power, and signal of SBUS pin	
5	WASP-200 LRF	Telem 2	
6	Sabertooth 2x12 Motor Controller	Servo rail: Signal and ground of PWM output pins 1 and 3	Batteries to B- and B+.
7	(x2) 12 Volt 8 Ah Sealed Lead Acid Battery (SLA) - 0.250 Faston		Motor Controller
8	Buzzer and Safety Switch	Buzzer and Switch	
9	LiPo battery voltage alarm		3S LiPo battery
10	ReadytoSky GPS/Compass	GPS and I2C	
11	Piksi Multi-board	Serial 4/5 port	Battery (via voltage regulator), GNSS survey grade antenna
12	Survey grade GNSS antenna		Piksi Multi
13	FrSky Taranis Q X7 transmitter (not pictured)		Radio linked to X4RSB receiver

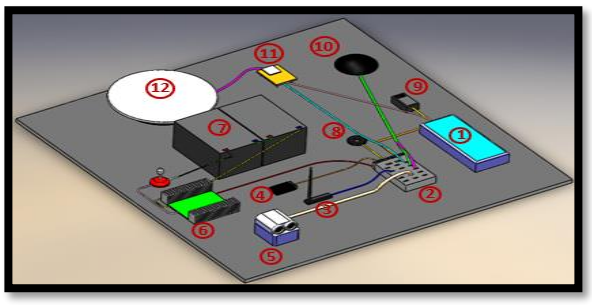

Fig. 5. Electrical connections. Everything except for the LIDAR is shown here.

A LiPo battery was chosen because of its compatibility with Pixhawk's provided power module and high current output. Everything on the AGV apart from the motors was powered with this battery.

The voltage alarm is an affordable and easy way to ensure that the LiPo battery does not go below its allowable voltage, preventing damage. The alarm can be programmed to trigger at different voltages, alerting the user when a return to base is needed.

The ReadytoSky GPS/Compass was used as a backup GPS so that the AGV could find its way home if the Piksi failed or glitched. It uses a NEO-8N GPS module from u-blox and has an advertised *maximum* accuracy of 0.6 meters.

Communication between the Pixhawk and Mission Planner was done through the SiK Telemetry Radios. These radios are designed specifically for Pixhawk/Mission Planner and have a range of 300 m.

An X4RSB FrSky receiver and FrSky Taranis Q X7 transmitter were used to send commands from the user to the robot. The X4RSB receiver has SBUS capability which was necessary to send multiple channels over a single wire that connects into the Pixhawk RCIN pin. The Q X7 is an inexpensive but highly capable transmitter. It is reliable and has 16 channels, two control sticks, and several switches all programmable in Mission Planner to do different tasks. One of these switches was programmed to switch the Pixhawk from manual to auto to Smart RTL mode. The channels for the two sticks on the transmitter were Aileron, Elevator, Throttle, and Rudder. Each of these corresponds to one of the two sticks in either vertical or horizontal movement. In the mixing menu, they were put in an AETR channel configuration. That and the switch used (SF, CH6) can be seen in Fig. 6.

Fig. 6. Configuration for channel mixing on the Taranis transmitter. The switch used channel 6.

In the mission planner, the parameters for the Pixhawk were also configured accordingly in Fig. 7.

Fig. 7. Mission Planner configuration. Roll = Aieleron. Pitch = Elevator. Yaw = Rudder.

Only roll and throttle were used by the Pixhawk and sent to the motor controller. Roll controlled the tank-style steering (left, right) and throttle controlled the speed of the wheels.

The motor controller is robust, and is able to supply two separate channels of 12 continuous amps each, has overcurrent and overheat shut down prevention, several input modes, synchronous regenerative technology (batteries are charged when going in reverse), and has a 5 Volt 1 Amp power output for peripheral devices if needed.

For the LIDAR A WASP 200-LRF was chosen because it is compatible with Pixhawk. It has a range of 0.2-300 meters and an accuracy of < 10 cm. Most importantly it uses a DF13 connector which is what makes it compatible with the Pixhawk ports

B. *Mechanical Components*

SuperDroid Robots was the source of most mechanical parts for the robot.

TABLE IV. LIST OF MECHANICAL COMPONENTS

Item	Description	Quantity
IG42 24VDC 078 RPM Gear Motor	24 Volt with planetary gears	4
ATR Wheel and Shaft Set Pair 8mm bore - 10-inch Traction Lug	Wheel and shaft with connecting hardware	4
Electric Power Hookup Kit	Wires, fuses, and switches	1
IG42 Motor Plate Hardware Kit		2
Motor Mount Plate - IG42		4
Electric Motor Hookup Kit		4

Calculations were done to ensure the motors were adequate for the application. The rated speed of the selected motor is 78 RPM. For the 10-inch diameter wheels used that is equivalent to 2.3 linear mph. While this is slow for a wheeled robot, the higher reduction ratio gives more torque for rough terrain so this rated speed was deemed sufficient. The rated torque was 0.05814 N·m (570 g·cm). With a 1:84 reduction ratio that is 4.88 N·m. To determine if this was enough torque the following formula was derived from a simple free body diagram. It solves for t, the time it takes to reach angular velocity V from a stopped position.

$$t = V \cdot \frac{I_{wheel} \cdot I_{motor}}{T_{motor} - Wr\sin(\theta)}$$

The moment of inertia of the motor was assumed to be 0. The moment of inertia of the wheel was calculated to be $\frac{1}{2}mr^2 = 0.018\ kg \cdot m^2$, V was set to the rated speed of 5.18 rad/s (2.3 mph) and r was the radius of the wheel. For $\theta = 0$, $t = 0.02$ seconds. See Fig. 8 for the results when the equation is solved for all values of theta (0° to 90°). An asymptote occurs at about 43.5°, indicating that's the maximum incline possible. Because potato farms have to be navigated by irrigation equipment and tractors, this high of a slope is rare.

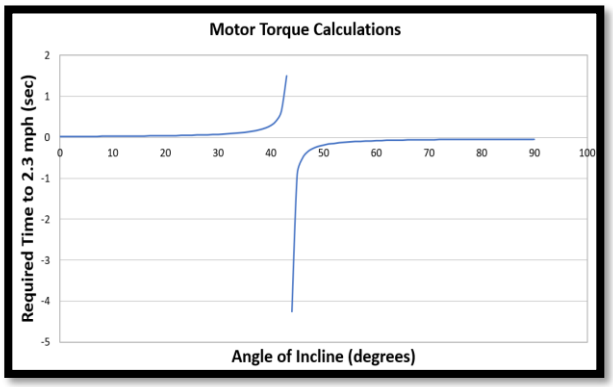

Fig. 8. Graph of motor torque calculation

IV. Obstacle Avoidance

There are not many obstacles in a potato field, and most of them are stationary. We determined that a simple range-finding LIDAR will suffice. A potential obstacle is the center pivot irrigation equipment, which moves at very slow speeds. Rocks, tractors, humans, and animals may also be encountered, but rarely.

The firmware onboard the Pixhawk can handle obstacle avoidance when a range-finding LIDAR is connected. Fig. 9 shows the parameters that configure the obstacle avoidance protocol known as "dodge" avoidance. Simply put, when the LIDAR detects an object that is too close, as defined by RNGFND_TRIGGER_CM, it stops, rotates RNGFND_TURN_ANGL degrees, and drives straight for RNGFND_TURN_TIME seconds and then resumes a direct course for the next waypoint.

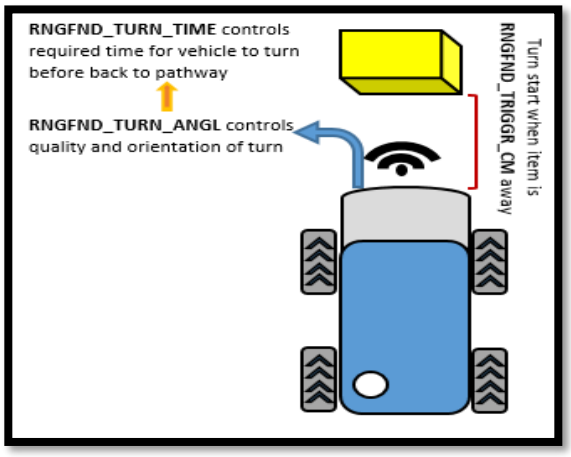

Fig. 9. A graphic explaining the parameters involved in setting up "dodge" obstacle avoidance on the Pixhawk using a rangefinding LIDAR

V. Enclosure & Bottom Plate

It was important to design an enclosure that protected the electrical components from mud and water and the elements are given its operating environment. Because this robot was a prototype, easy access to the components was also essential. It was designed first in SolidWorks and then acrylic sheets were laser cut and glued with Weldon #4 acrylic glue (Dichloromethane). Acrylic was used because it can be easily laser cut, allowing the enclosure to fit snugly onto the robot increasing protection from the elements. Acrylic is also readily available at hardware stores if quick repairs were needed.

The enclosure's joints were reinforced with high-temperature hot glue. When the acrylic sheets were cut out, holes were included in strategic positions for cables, antennas, etc. Once assembled, these holes were tested, and then the enclosure was spray painted white to lower heat transfer from the sun (Fig. 10).

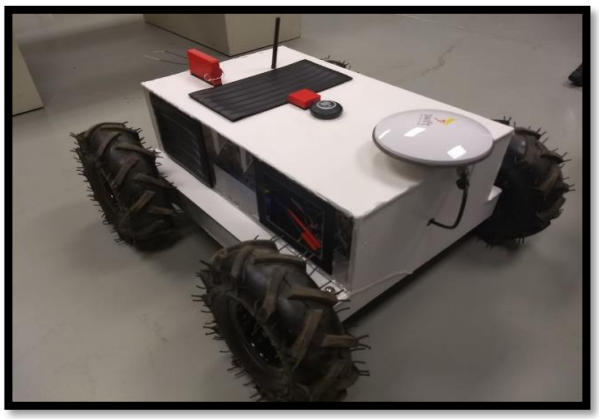

Fig. 10. Assembled and painted enclosure, placed on top of robot chassis.

The enclosure is held down by four bolts and can be lifted off with components still attached. The components are simply connected/disconnected through the easy access holes in the enclosure.

In addition to an enclosure, a bottom acrylic plate was laser cut for mounting the electrical components too. This could be removed easily and independently from the enclosure for debugging. For calibrating the magnetometers in the equipment, this bottom plate can stay bolted to the enclosure but lifted off the chassis.

VI. Results and Discussion

A. *Piksi Multi Accuracy*

The RTK GPS was true to its advertised accuracy (less than or equal to 10 centimeters). Using National Geodetic Marker sites, we were able to verify that the Piksi Multi is accurate to 5.6 centimeters using our setup.

There was significant difficulty finding a location with a GPS coordinate known accurately enough to verify the coordinates found by the Piksi. A couple of methods were considered. The easiest but most expensive is to survey a spot on the ground. This requires special equipment and training but can be done practically anywhere outside. The more difficult method is to find a geodetic marker, a previously surveyed position marked with a permanent landmark. The online National Geodetic Survey Data Explorer gives access to all such publicly recorded markers. However, many are located on privately held land and are difficult to access.

The marker used was at the Pocatello Regional Airport. Using the evaluation boards from Swift Navigation, the base

station Piksi was placed on one geodetic marker and the rover Piksi on another several hundred meters away. 50 GPS fixes were collected and converted into the UTM 12N projected coordinate system (for ease of calculation). Pythagorean's theorem was used to find the difference between the rover's estimated GPS location and the coordinates listed on the geodetic marker's datasheet. The average error was 5.6 centimeters. Because of difficulties related to the testing location, only this test was done. Airport security was required and the airport could only accommodate us for a short period of time.

Fig. 11. A geodetic marker

TABLE V. RESULTS FROM PIKSI MULTI ACCURACY TESTING

Test #	Accuracy (cm)	Base Station Test Location		Geodetic Permanent Identifier (PID)
1	5.6	42° 54' 47.03909" N	112° 35' 23.36463" W	AB8163

B. Autonomous Navigation Via Waypoints

The performance of the autonomous navigation was satisfactory. At times the AGV had problems finding its heading and would spin around until finding it again. This was a minor issue, though, and higher quality magnetometers or more careful calibration may fix this.

Accuracy during autonomous navigation wasn't tested using geodetic markers since accuracy had already been confirmed. However, there is a limitation related to the waypoint radius parameter described earlier that needs to be addressed. During testing, it was discovered that the lowest value that can be stored in the parameter is 1 meter, so as soon as the AGV is within 1 meter it would move on to the next waypoint. This could be solved by adding a trigger event so that a third party computer could handle the precise navigating within 1 meter of the infected potato plant.

C. Obstacle Avoidance

The WASP LIDAR listed in Table III was also tested. In this test, the SiK telemetry radios were bypassed with a USB cable. In this setup, the distance between the LIDAR and an obstacle placed in front of the AGV was within a few centimeters of the reported value in Mission Planner. However, obstacle avoidance was not tested with the telemetry connection. Future testing will be conducted using the telemetry connection.

VII. CONCLUSION

The AGV designed and built in this project has an extremely accurate RTK GPS, and combined with the Pixhawk navigational firmware is a suitable testing platform for an autonomous agricultural robot. The robot and components were relatively inexpensive but robust enough to handle typical terrains the robot will be operational in. Since it uses the Pixhawk it is extremely customizable as well. These features, along with obstacle avoidance in the future, make it extremely useful not just for this project but for many agricultural robot applications.

REFERENCES

[1] J.A. De Bokx, and H. Huttinga, 1981. Potato virus Y. Descriptions of plant viruses, no. 242. Commonw. Mycol. Inst./Assoc. Appl. Biol., Kew, England. Online www. dpvweb. net/dpv/showdpv. php.

[2] P. Nolte, J.L. Whitworth, M.K. Thornton, and C.S. McIntosh, 2004. Effect of seedborne Potato virus Y on performance of Russet Burbank, Russet Norkotah, and Shepody potato. Plant Disease, 88(3), pp.248-252.

[3] D. Krezhova, K. Velichkova, N. Petrov, and S. Maneva, (2017). The Effect Of Plant Diseases On Hyperspectral Leaf Reflectance And Biophysical Parameters. RAD Conference Proceedings. doi: 10.21175/radproc.2017.5.

[4] D. Krezhova, N. Petrov, and S. Maneva, 2012, October. Hyperspectral remote sensing applications for monitoring and stress detection in cultural plants: viral infections in tobacco plants. In Remote Sensing for Agriculture, Ecosystems, and Hydrology XIV (Vol. 8531, p. 85311H). International Society for Optics and Photonics.

[5] Z. Ni, Z. Liu, H. Huo, Z. H. Li, F. Nerry, Q. Wang, X. Li, "early Water Stress Detection Using Leaf-Level Measurements of Chlorophyll Fluoresecence and Temperature Data," Remote Sens., vol. 7, no. 3, pp. 3232-3249, Mar. 2015. DOI 10.3390/rs7030232.

[6] J. Bengochea-Guevara, J. Conesa-Muñoz, D. Andújar, and A. Ribeiro (2016). Merge Fuzzy Visual Servoing and GPS-Based Planning to Obtain a Proper Navigation Behavior for a Small Crop-Inspection Robot. Sensors, 16(3), 276. doi: 10.3390/s16030276.

[7] X. Gao, J. Li, L. Fan, Q. Zhou, K. Yin, J. Wang, and Z. Wang, (2018). Review of Wheeled Mobile Robots' Navigation Problems and Application Prospects in Agriculture. IEEE Access, 6, 49248–49268. doi: 10.1109/access.2018.2868848.

[8] S.M. Pedersen, S. Fountas, H. Have, and B.S. Blackmore. Agricultural robots—system analysis and economic feasibility. Precision Agric 7, 295–308 (2006) doi:10.1007/s11119-006-9014-9.

[9] G. Polder, P.M. Blok, H.A. de Villiers, J.M. van der Wolf, and J. Kamp, 2019. Potato Virus Y Detection in Seed Potatoes Using Deep Learning on Hyperspectral Images. Front. Plant Sci. 10:209. doi: 10.3389/fpls.2019.00209

[10] T. Deemyad, N. Hassanzadeh, and A. Perez-Gracia. Coupling mechanisms for multi-fingered robotic hands with skew axes. In Mechanism Design for Robotics (MEDER). Springer, 2018.

[11] T. Deemyad, O. Heidari, and A. Perez-Gracia. Singularity Design for RRSS Mechanisms. In USCToMM Symposium on Mechanical Systems and Robotics (MSR) Conferences. Springer, 2020.

[12] T. Deemyad. Design of Five-fingered Underactuated Hand for Two-position Tasks (Master's Thesis, Idaho State University), 2016.

[13] M. Dadvar, S. Moazami, H. R. Myler, and H. Zargarzadeh, "Multiagent task allocation in complementary teams: a hunter-and-gatherer approach," Complexity, vol. 2020, Article ID 1752571, 15 pages.

[14] M. Dadvar, S. Moazami, H. R. Myler, and H. Zargarzadeh, "Exploration and coordination of complementary multi-robot teams in a hunter and gatherer scenario," 2019, https://arxiv.org/abs/1912.07521.

[15] S. Habibian. "Analysis and Control of Fiber-Reinforced Elastomeric Enclosures (FREEs)." (Master's Thesis. 229, Bucknell University), 2019.