An Open Source Seeding Agri-Robot

Harry Rogers
School of Computer Science
University of Lincoln, United Kingdom
15623886@students.lincoln.ac.uk

Charles Fox
School of Computer Science
University of Lincoln, United Kingdom
chfox@lincoln.ac.uk

***Abstract*—To enable wider experimental uptake and build interest in autonomous precision agriculture, we present a low-cost, open source hardware and software platform for automated precision seed planting, for use by researchers and practical agri-hackers. The robot platform is about to plant seeds with higher within-row precision than conventional systems, and using high end Global Navigation Satellite System (GNSS) is able to store the geospatial location of each seed to accuracies sufficient to enable revisiting of plants during growth. It can also operate with lower cost GNSS using a swappable interface for other applications. We provide software up to the level of pose-to-pose driving using Dubins paths. The robot is based on a low-cost closed-source commercial off the self (COTS) vehicle, low-cost COTS seeder, and Raspberry Pi, with our open hardware design showing how to link them to each other and to the software. The hardware and software is being open-sourced at https://github.com/Harry-Rogers/PiCar as part of this publication.**

***Index Terms*—Agriculture, Raspberry Pi, GNSS, Mechatronics, Python, Raspbian, Open Source**

I. Introduction

Automated precision agriculture is becoming important around the world due to increased population and demand for food, declining fuel reserves, and climate change [1]. Currently there are many solutions to automate agriculture that involve harvesting crop, spraying pesticide, spraying weeds and planting seeds. There are many solutions like the IBEX robot that sprays weeds [2], the Rubion robot that picks strawberries[1], and Fendt's project MARS robot[2] that can monitor and accurately document precise planting of corn.

However, these are expensive, large-scale systems, which do not easily allow smaller scale innovators such as researchers, farmers in developing countries, and the maker community, to experiment and extend their designs. Innovation in both farming practice and engineering design often emerges bottom up from these communities. Therefore, this paper makes such a system available to the community.

Like most open source hardware (OSH), the system is based on off-the-shelf components which may not be OSH themselves. It combines low costs COTS vehicle[3], seed planter, and embedded computer and provides physical build instructions and software to run. It can plan and control an efficient point-to-point route including speeding up on long straights and slowing to control turns. Using a GNSS module it stores the Geographic Information System (GIS) coordinates of each seed planted so the system can return to it. This system is cheap enough to be reproduced at scale to create swarms that can complete many processes in a very small amount of time the total cost of the system is shown in table 1. Seed planting is chosen as a first task because even current agri-business systems such as large tractor driven drills are not able to plant seeds at this precision, with spacing between seeds in a wheat crop typically varying by several centimeters. The system is legal to operate autonomously under the guidelines given by [3].

[1]Rubion robot (Octinion; http://octinion.com/products/agricultural-robotics/rubion)

[2]Fendt's project MARS robot (Fendt; https://www.fendt.com/int/fendt-mars)

[3]COTS vehicle (Sunfounder Ltd; https://www.sunfounder.com/smart-video-car-kit-v2-0.html)

II. System Design

A. Open source Hardware

The main components consist of a Raspberry Pi 3 model B+, Robot HATS[3] a hat board that is on top of the Raspberry Pi, a PCA9865 [4] board for controlling servos, a TB6612 [5] board to control the DC motors and a camera for the system to see. Alongside this, a COTS seed dispenser [6] and GNSS module [7] have been added to the kit to complete the functionality for the project. Robot HATS sends power to the other boards as well as signals to turn the servos to steer or move the camera. PCA9865 sends signals to the servos to steer the car as well as pan or tilt the camera, it also sends signals to the TB6612 board to either turn the wheels or not. TB6612 sends signals to the DC motors depending on what is needed from the other boards. The seed dispenser is modified to be shorter therefore making it lighter it is attached with glue. Fig. 1 shows the built system. Table 1 shows the per-item cost, the system overall costs £206.96.

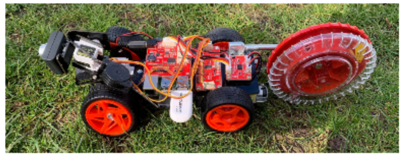

Fig. 1. Built system

TABLE I
PER-ITEM COST

Item	Category	Cost(GBP)
COTS Robot	Material	99.99
COTS Seed dispenser	Material	14.18
Raspberry Pi	Material	32.75
Power Adapter	Material	8.29
SD card	Material	5.29
Batteries	Material	16.99
Battery Charger	Material	11.99
GNSS Module	Material	12.48

B. Open Source Software

When turning the system on, the Raspberry Pi starts a script using the Django (https://www.djangoproject.com/) web framework so that an application on the user's device, such as a smart phone, can connect and operate the system. There are two main areas of software: movement and data.

1) Movement: Pose to pose path planning and control is provided, which can be used to plant seeds at regular intervals along one or more paths. Planning operates by using a Dubins path algorithm. There are three steps to completing this path: a starting turn, going straight, and ending turn. The Dubins [8] path is calculated by creating two arcs around the start and end point that are connected by the shortest straight line between them. The modification over the standard Dubins method is that during the straight, the controller speeds up to its maximum speed to reduce the time taken to be completed. During curves the controller slows down to its minimum speed to reduce skidding.

2) Data: The data collected from the GNSS module contains GIS coordinates of where seeds have been planted. The seeds' location is found by calculating the circumference of the dispenser and its location relative to the robot center. The open hole that dispenses the seed is at the top each time the system starts a new task. This is so that when half a rotation is completed the seed is dispensed. National Marine Electronics Association (NMEA) data collected from the GNSS module is processed and currently stored in CSV format though we plan to switch to a GIS database such as PostGIS.

III. Testing and Results

A. Design of experiment

An experiment was carried out to test whether or not the system is effective at planting seeds and recording their accurate GIS coordinates. In the experiment, the system must drive to a pose that is 6.8m away using a Dubins path. Along this path it must plant 20 seeds, at regular spacial intervals, and record their GIS coordinates. It is assumed that the system will drive over a cut trench and the seeds will then be covered after it completed its task. Whilst on the path it must also speed up when driving straight and slow down when turning, without affecting the seed spacing.

B. Results

All 20 seeds were planted. The system travelled 11.2m in total to get to the pose that was 6.8m away because of the Dubins Curve requirements. The planting rate was 1.1 seeds per second. Two seeds were planted at both of the first and second points due to traveling slowly. Seeds also once leaving the seed dispenser were moving at a rate that could cause them to roll or bounce off of the ground. Fig. 2 shows where each seed was planted on the Dubins path. Planted seed ground truth locations where then measured manually with a ruler and found to have a standard deviation of 139mm from their intended and logged locations. Minimum distance between seeds was on average 399mm apart from each other. The top speed of the system was 0.49m/s and the minimum speed of the system was 0.26m/s. The average speed the system was travelling at was 0.39m/s. The coordinates on average were 8.23m away from the actual location, before differential GNSS correction. The system was also able to do the path in reverse completing the path with the same speeds.

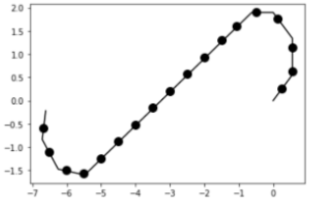

Fig. 2. Mapped seeds

IV. Conclusion

The observed seed planting accuracy of the system was 139mm. This was due to seeds bouncing off the ground as the trundle was moving very quickly on the straight distance therefore causing seeds to move from their original planting point. One way this system could make farming more efficient is if a swarm was created and the trundle size was reduced. This would mean that multiple rows could be completed at once. The system plants much less seeds per second than a large tractor system, but as it is a system it does not require to be paid by the hour as labourers would it is also open source so everyone can use it. However, some seeds at the earlier locations were not planted individually but were dispensed in groups in error, due to the low speed of the robot when setting off. Minimum distance between seeds was too large (399mm) for crops such as wheat, but could be reduced by using a smaller trundle or trundle gearing. The differential GNSS accuracy is such that driving the robot to revisit specific plants appears possible for seed spacings up to 139mm. The system can follow the exact path taken in reverse to do this. The results suggest that the system would be a useful test bed for research and hobbyist platforms and if scaled up to swarms could make seeding and other operations more efficient. As an open source hardware and software project, we welcome future contributions from the community.

References

[1] T. Duckett, S. Pearson, S. Blackmore and B. Grieve (2018) *Agricultural robotics: the future of robotic agriculture*. arXiv:1806.06762
[2] A. Binch and C. Fox (2017) *Controlled comparison of machine vision algorithms for Rumex and Urtica detection in grassland*. Computers and Electronics in Agriculture 140:123-138.
[3] S. Basu, A. Omotubora, M. Beeson and C. Fox (2020) *Legal framework for small autonomous agricultural robots*. AI & SOCIETY volume 35, pages113–134.
[4] Sunfounder Ltd; https://bit.ly/3401mRX
[5] Sunfounder Ltd; https://bit.ly/3dGy7Ih
[6] WOLF-Garten GmbH & Co.KG; https://bit.ly/2WXyGaU
[7] u-blox; https://www.u-blox.com/en/product/ubx-g7020-series
[8] Dubins, L.E. (July 1957). "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents". American Journal of Mathematics. 79 (3): 497–516.