UNIVERSITÄT HOHENHEIM

200 JAHRE 1818 2018

Fakultät Agrarwissenschaften
Institut für Agrartechnik
FG Verfahrenstechnik in der Pflanzenproduktion

Prof. Dr. Hans W. Griepentrog
Fachgebietsleiter

Dissertation

Perception for context awareness of agricultural robots

Submitted in fulfilment of the regulations to acquire the degree
"Doktor der Agrarwissenschaften"
(Dr. sc. agr. in Agricultural Sciences)

To the
Faculty of Agricultural Sciences

Submitted by: **Dipl.-Ing. David Reiser**

Born in: Freudenstadt, Deutschland

Year of publication: 2018

This thesis was accepted as a doctoral thesis (Dissertation) in fulfilment of the regulations to acquire the degree “Doktor der Agrarwissenschaften” (Dr.sc.agr. in Agricultural Sciences) by the Faculty of Agricultural Sciences at University of Hohenheim on 13.9.2018.

Date of oral examination: 10. December 2018

Examination Committee

Head of the Committee:	Prof. Dr. sc. agr. Marcus Rodehutscord
Supervisor and 1st Examiner:	Prof. Dr. sc. agr. Hans W. Griepentrog
2nd Examiner:	Prof. Dr. rer. nat. Arno Ruckelshausen
3rd Examiner:	Prof. Dr.-Ing. Stefan Böttinger

Self-publishing: David Reiser
Source of supply: Institute of Agricultural Engineering of
the University of Hohenheim
Garbenstraße 9, 70599 Stuttgart

Acknowledgements

The following thesis was written during my work at the Institute of Agricultural Engineering of the University of Hohenheim between 2014 and 2017. Whether mentioned or not, I would like to thank all those who have supported me along the way.

First, I want to thank my supervisor Prof. Hans W. Griepentrog. Thanks for the possibility to write this thesis, to always ask for advice when necessary and the easy and uncomplicated way of communication. Equally, I want to thank my second examiner Prof. Arno Ruckelshausen and my third examiner Prof. Stefan Böttinger for accepting revising my thesis.

Second I want to thank Karin and Helga for the support in administrative questions and the help to survive the jungle of bureaucracy. Thanks to Dimitris, Dietrich, Galib, Manuel, Miguel, Christian, Henry and Jörg for your support and nice discussions at lunchtime. It always helped to get a clear mind. Thanks to all the people from the workshop, you eased my life while I was struggling with mechanical problems and provided the necessary support. Thanks to Javier, Emir, Olim and Markus for your work. Equally, want to thank Dr. Nikolaus Merkt for allowing field experiments at the university vineyard.

Last but not least I want to thank my family, especially for the patience to listen to the endless story about “what paper I want to write and why this is so important for my thesis”. Thanks, Miri for your smiles and attention you brought to listen to my “hopefully not always-boring” work stories.

David Reiser

Stuttgart-Hohenheim, Dezember 2018

Contents

CHAPTER 1

Introduction

1.1 Problem description

The progress in agriculture can be indicated by the increase of productivity per worker involved. In advanced countries, the percentage of people working in agriculture dropped to 2.5%, compared to 75% during the late 1950s (Federico 2005). In Germany, the percentage dropped at the same time from 22.1% to 1.5% (Bundesamt 2015). This development was also described by Johnson (1997), who even expressed the prosperity of nations with the number of people working in agriculture. However, the world is changing and farming as an important production sector of a national economy has to adapt, to guarantee food security in future (Martin et al. 2013). New challenges for agriculture are ahead, like providing food for an increased world population (Gerland et al. 2014), mitigating climate change effects (Rosenzweig et al. 2014), securing water supply (Elliott et al. 2014), defeating soil erosion (Knijff et al. 2000) and encountering the wish of consumers for high-quality food and organic farming products (Reganold and Wachter 2016).

One part of the agricultural success story of the last century was due to mechanisation (Federico 2005). In the last decades, the trend went from small to big agricultural machines to increase productivity. The aim was to handle larger areas in less time with fewer people (Blackmore et al. 2006; Johnson 1997). This development made small machines impractical, as the entire farming process was optimized for large machines (King 2017). However, the size of the machines cannot increase forever, as they cause soil compactions, have to be transported to the fields, fit on public roads and have to have good fatigue live resistance (Paraforos et al. 2016). In future, new ways have to be found to improve agricultural productivity by using the advances of modern sensor technology and automation. Autonomous robots together with new sensor and computer technologies could lead to the next agricultural revolution and to improvements in food production (Bloem et al. 2014; King 2017).

In the last decades, the developments of precision and smart farming tried to deal with spatial variability on fields by variable rate applications (Stafford 2000). These technologies enabled a reduction of the farm management area down to a sub-field level. The next logical step would be to realize individual plant care with the help of sensor guidance (Dzinaj et al. 1998; Pedersen et al. 2006; Weiss and Biber 2011). Individual plant treatment could be realized by sensor guided implements or

with autonomous robots (Lee et al. 1999; Strothmann et al. 2017). If there are autonomous working robots, there will be no need for big and heavy machines, as they could work in swarms for 24 hours a day (Pedersen et al. 2006). These small, autonomous, in collaboration working machines, could be a solution for increasing productivity and food quality in future and would cause less soil compaction on the fields (Vougioukas 2012). With robotics, it could be possible to cultivate techniques back to agriculture which are right now discarded, because of unproductivity or high labour costs (van Ittersum and Rabbinge 1997). Even a collaboration between humans and machines could be imaginable (Cheein et al. 2015). Considering the aspects of food security, climate change and sustainable land management, there are many options for improvements, like organic fertilization, mechanical weeding, precision spraying or the goal of minimum soil disturbance (Branca et al. 2013). As the wealth of countries in Europe develops, it is more challenging for farmers to find workers for low-paid jobs like fruit pickers. Therefore, future has to deal with the automation of these tasks to keep agriculture in developed countries competitive.

The biggest challenge towards fully autonomous vehicles in agriculture is the high diversity in different outdoor environments. There are many tasks in agriculture that until now cannot be automated without adapting the environment to the behaviour of the robot, such as fruit picking, selective weeding or selective harvesting (Back et al. 2014). While robots are performing well in indoor environments, they are mainly unable to cope with the high diversity and uncertainty of our world (Shalal et al. 2013). Especially in outdoor environments, the performance of mobile robots is still far away from robust and reliable use (Back et al. 2014). One factor for this performance discrepancy in known environments, compared to unstructured and dynamic worlds, is the missing of context awareness (Bechar and Vigneault 2016a).

For sufficient performance, robots need to know where they are, what is around them and what are the objects of interests in the area (Brooks 1991). This could be also described with self-awareness and context awareness (Gorbenko et al. 2011, 2012; Lu 2014). Blackmore et al. (2006) describe self-awareness as the situation where a machine knows about its own processes and contexts. Robots or machines working in a known and static environment do not need to adapt, as long as the model for machine and environment is correct (Brooks 1991). The precision and repeatability of robots are supporting the robustness of static processes. However, if the environment and the system are changing in an unpredictable way, it is necessary to adapt the robot behaviour in a correct way to new inputs. The system must perceive new unknown situations in order to create knowledge from the given context (Brooks 1990). Processes and contexts should be combined in an intelligent system to perform robust and reasonable decisions (Brooks 1986). Bechar and Vigneault (2016) divide the

complexity of robotic domains into four different groups depending on the environment and the objects to handle (Table 1).

		Environment	
		Structured	Unstructured
Objects	Structured	Industrial domain	Military, space, underwater, mining domains
	Unstructured	Medical domain	Agricultural domain

Table 1: Description of the different domains in robotic disciplines (Bechar and Vigneault 2016b).

The uniqueness of agricultural robotics is the necessity to deal with unstructured objects in an unstructured environment. Therefore, we need a reliable sensor perception and context awareness for robots in the agricultural domain.

1.2 Sensor perception and knowledge creation

Perception is the first step to reach the goal of an autonomous system with reasonable self-awareness. With perception, machines can learn about their environment and the surrounding objects. When there is not a structured environment like in the industrial sector, perception is the most important part of reaching context awareness with autonomous machines. The more unstructured, the more knowledge is required for detection because more uncertainty disrupts the process (Bajcsy 1988).

Traditionally automation was using one sensor for controlling one actuator or function, which is not useful for an advanced robotic system. It is more convenient to use an information platform, gathering the sensor data available in a local cloud (Brooks 1986; Quigley et al. 2009). Brooks (1986) recommended this kind of architecture, as data from different sensors could be fused to different behaviour based tasks. This enables to create more robust, more complex and more intelligent systems, as different available sensors could confirm the gathered information. In addition, a malfunction of one sensor does not mean that the task could not be fulfilled. Other sensors could close the gap and bring in the necessary knowledge for the process (Brooks 1986; Dzinaj et al. 1998; Fender et al. 2006).

Before relevant knowledge could be created out of perception, it is necessary to create a sensor able to transform attributes into electrical signals. A smart sensor would be able to convert this data to usable information. Used in the right context of an intelligent system, information could be transformed into knowledge, which is the basis for a decision. The knowledge gained by an intelligent system can differ between each information created and the context. In addition, the complexity of the decision-making can have high differences. As long as the assumed models and the sensor system

are correct, the created information could be converted to knowledge, which can be the reason for a decision of e.g. a mobile autonomous system (Griepentrog 2017) (Figure 1).

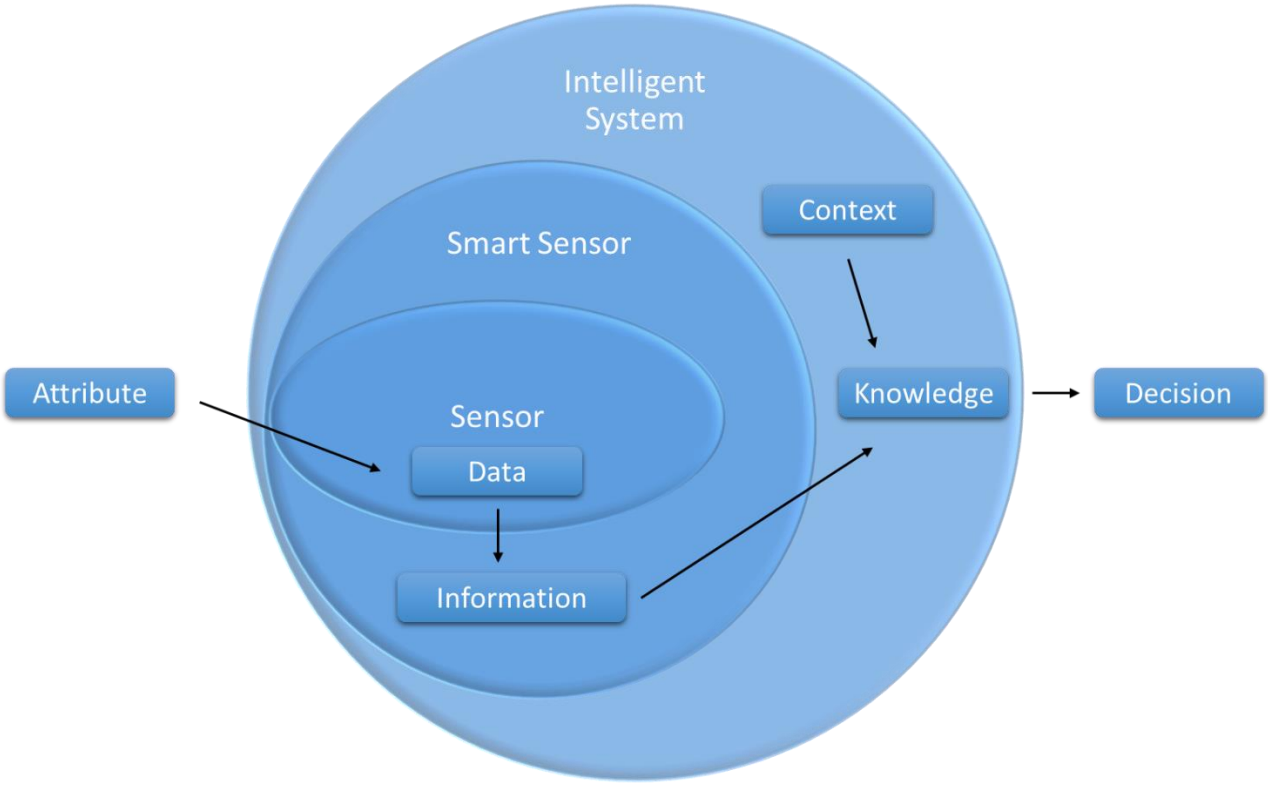

Figure 1: Relation between an attribute and the decision-making of an intelligent system (Griepentrog 2017).

The decision-making of an autonomous robot could be as follows: The robot system wants to know where it is. For the definition of the actual position, a Global Navigation Satellite System (GNSS) could be used. The sensor measures the signals sent by satellites and creates data. The information (position) is created out of the precise knowledge of the position of the satellites (orbits) and the time difference between the received sensor signals by trilateration. Together with the context, like the modelling of the earth and precise maps, a decision could be made where the robot should go next. The position of an autonomous system could be gained by an absolute or relative coordinate system. An absolute coordinate system defines the position in a fixed frame of reference, a relative system could take the coordinates relative to e.g. a moving vehicle (Griepentrog et al. 2006a). As long as the context fits with the known model (e.g. map or crop line), the results are trustable. This is sufficient for automated steering or navigation on agricultural fields, as there are static field borders and the crop rows and patterns are not changing after seeding (Griepentrog et al. 2006b). However, maps and models have to be created beforehand, what makes this technique uninteresting for direct environment interaction under changing and unknown conditions. The decision-making is based on static

information, not able to adjust to an always-changing world. The question is how we could connect models with the changing of the environment and how to use this information.

One way to adapt to changes in the environment could be a Wireless Sensor Network, sending information about the surrounding of one location. However, these sensors are generally fixed to one place, just able to create information at the actual position where they were mounted.

A different way is the use of tactile or visual sensors, mounted at a mobile robot, to get online information about the environment. Here the robot could lead the sensors to the position of interest to create the data. For good reference, a precise position system could help to set the information into the right context. This requires a robust object detection, what is dependent on a robust software classification.

Standard robot navigation does not distinguish between different objects. Out of the context (sensor attachment), it just gets decided if there is an obstacle or not (Barawid et al. 2007). Crop row detection is a basic function in agriculture for automating robot systems or to create driver assistant systems (Hague et al. 1997; S. Hiremath et al. 2014; G. Jiang et al. 2010a). Autonomous navigation could be performed like this, as long as the environment is controllable and could be modelled. However, this is not sufficient when objects are changing, or the behaviour of the robot have to be adjusted to the behaviour of the objects in front. This is a well-known problem in the research area of human-robot interaction (Gorbenko et al. 2012; Kruse et al. 2013; Luber et al. 2012).

1.3 The uncertainty of robot and sensory perception in agriculture

In outdoor robotics, many parameters could affect the result of sensory perception and the resulting decision made, like sunlight, uneven terrain or changing weather conditions (Vázquez-Arellano et al. 2016). However, as long as the accuracy of the sensor and the correct context are known, errors could be detected and analysed by an intelligent system. For a smart planning and task execution of a robot, the interaction with the environment is necessary (Xu and Van Brussel 1997). On a field, the objects are changing as plants are growing and even the surroundings like weather and soil conditions can change (Bechar and Vigneault 2016b). This makes it challenging to program algorithms, able to perform under all possible circumstances. The challenges differ strongly between the crop, the season and side specific parameters like the soil composition. Therefore, to navigate in the winter season in an orchard with cleanly defined borders is easier than to run a machine autonomously without calibrating in crop rows in different growth stages. Especially the change of appearance of plants makes it hard for localizing the same object with sensors at different growth periods.

Classical object recognition algorithms are challenged by the diversity of plants during growth stages and their structure (Bechar and Vigneault 2016b). One advantage of agriculture is the semi-structured

environment. This could help to find a better way for classification of plants, as they are planted in ordered structures. This could help to reach a higher percentage of robot autonomy.

1.4 Levels of robot autonomy

Autonomy could be defined by the percentage of human interaction required. A fully autonomous system would be a machine, with no human interaction or goal settings. Bechar and Vigneault (2016b) appoint to the Pareto principle for robot autonomy on agricultural fields. They claim that 80% of tasks are easy to automate, but the remaining 20% are hard to achieve (Stentz et al. 2002). The farmer is an expert in his field, so why should we aim to replace him and do not use his expertise? We could try to adapt the knowledge of the farmer to the machine by supervised learning. Therefore, it could be easier to gain a useful level of autonomy with a commercial use.

The advantages of human-machine (or robot) interaction are huge. The robots could learn a task from a worker and do the task as long as necessary as robots do not need breaks (Anzai 1993; Cheein et al. 2015). The lacking part about modern robots is still the same like three decades ago (Brooks 1990). They are still a long way from making reasonable and robust decisions in fast-changing environments, which is also reflected in the performance of harvesting robots of the last three decades (Back et al. 2014). Until this state is not improved, the goals have to be defined by humans, assisted by machines. However, with the actual state of the art of robotic performance and artificial intelligence, it is quite questionable, if the results of autonomous interacting robots would be better than when a human operator decides the goals.

One of the most interesting parts for automation in agriculture is navigation (Stentz et al. 2002). In a deterministic system, the path has to be created in all details by a user to create a point-to-point navigation. The details of the work plan have to be created beforehand, so these systems have a small percentage of autonomy (Griepentrog et al. 2006a). If the planning and the environment do not fit, the system will fail. A reactive system, able to react in real-time to environmental changes, would be much more robust. The combination of both variants in hybrid systems seems necessary to enable robust autonomous navigation.

For high levels of autonomy, machines could use special architectures to deal with the differences in the environment. Strube (1998) described a three-layer architecture for action control including physiological (reflexes), associative (stimulus-response) and deliberative regulation (goal management). The first layer responses (the reflexes) have to be as fast as possible, but there is more time for associative layer decisions. The last layer, the deliberative regulation, is in general not time critical, as global goal planning does not have to be changed in real time when the other two layers enable the system to deal with local uncertainties.

Another way for robot control under different circumstances could be to define different programmes, which are activating or changing parameters if they are needed. When the context is known, it is much easier to adapt algorithms to work robustly. One practical example is the so-called “mode changer”, defining different operational tasks for the programming, like one program for line following and one for headland turning (Griepentrog et al. 2006a; S. Vougioukas et al. 2004). This allows a system to be more flexible and user-friendly, which would be the first way to get semi-autonomous machines up and running.

1.5 Aim and objectives

This thesis aims at increasing context awareness for robots in agriculture through different approaches of environmental perceptions. Therefore, different strategies to increase context awareness with perception are addressed and applied to mobile robots in agriculture:

- Enable objects to communicate with the robot
- Control the environment to detect unstructured objects
- Model changes to predict algorithm behaviour
- Detect objects before interacting with them
- Use algorithms which are not affected by changes in the morphology or environmental parameters

These points lead to the five objectives of the thesis:

1. Set up an application for static local sensor communication with a mobile vehicle
2. Use sonar sensors to detect unstructured objects in a controlled environment
3. Find a way to describe the influence of growth stage on algorithm outcomes
4. Use the gained sensor information to detect single plants
5. Improve the robustness of algorithms under noisy condition

1.6 Appended papers

The dissertation is based on the following five papers:

A. Reiser, D., Paraforos, D.S., Khan, M.T., Griepentrog, H.W., Vázquez Arellano, M., (2017). Autonomous field navigation, data acquisition and node location in wireless sensor networks. Precision Agriculture, 18 (3), 279-292. doi:10.1007/s11119-016-9477-2.

B. Reiser, D., Martín-López, J., Memic, E., Vázquez-Arellano, M., Brandner, S., Griepentrog, H. W., (2017). 3D Imaging with a Sonar Sensor and an Automated 3-Axes Frame for Selective

Spraying in Controlled Conditions. Journal of Imaging, 3(1):9. doi:10.3390/jimaging3010009.

C. Reiser, D., Miguel, G., Arellano, M.V., Griepentrog, H.W., Paraforos, D.S., (2016). Crop row detection in maize for developing navigation algorithms under changing plant growth stages. In: Advances in Intelligent Systems and Computing. doi:10.1007/978-3-319-27146-0_29.

D. Reiser, D., Arellano, M.V., Izard, M.G., Griepentrog, H.W., Paraforos, D.S., (2018). Iterative Individual Plant Clustering in Maize with Assembled 2D LiDAR Data. Computers in Industry. doi:10.1016/j.compind.2018.03.023.

E. Reiser, D., Vázquez-Arellano, M., Izard, M.G., Paraforos, D.S., Sharipov, G., Griepentrog, H.W., (2017). Clustering of Laser Scanner Perception Points of Maize Plants, In: Advances in Animal Biosciences: Precision Agriculture (ECPA) 2017,8:2. doi:10.1017/S20404700111X.

To show how perception and localisation could be done with machine-to-machine sensor communication, Paper A introduces a method for the data acquisition and localization in wireless sensor networks and shows how this could be used for precision agriculture.

Paper B focused on the detection and perception of objects in 3D point cloud representations, which could be gained with different sensor types and examining the possibility of sonar sensors for selective spraying in a controlled environment.

The next Paper C points on the experiment planning for describing plant growth changes in maize and how this is affecting the outcome of algorithm results.

The gained data from different growth stages is used in Paper D to create 3D point clouds with a 2D LiDAR scanner and to use this point clouds for single plant detection.

The final Paper E introduces an algorithm for separating and clustering point clouds with the help of a previous known plant position. This method is particularly insensitive to noisy conditions.

References

Anzai, Y. (1993). Human-robot-computer interaction: a new paradigm of research in robotics. Advanced Robotics, 8(4), 357–369. doi:10.1163/156855394X00158

Back, W. C., van Henten, E. J., & Edan, Y. (2014). Harvesting Robots for High-value Crops: State-of-the-art Review and Challenges Ahead. Journal of Field Robotics, 31(6), 888–911. doi:10.1002/rob

Bajcsy, R. (1988). Active perception. In Proceedings of the IEEE (Vol. 76, pp. 966–1005). doi:10.1109/5.5968

Barawid, O. C., Mizushima, A., Ishii, K., & Noguchi, N. (2007). Development of an Autonomous Navigation System using a Two-dimensional Laser Scanner in an Orchard Application. Biosystems Engineering, 96(2), 139–149. doi:10.1016/j.biosystemseng.2006.10.012

Bechar, A., & Vigneault, C. (2016a). Agricultural robots for field operations Part 2: Operations and systems. Biosystems Engineering, 149, 94–111. doi:10.1016/j.biosystemseng.2016.06.014

Bechar, A., & Vigneault, C. (2016b). Agricultural robots for field operations : Concepts and components. Biosystems Engineering, 149, 94–111. doi:10.1016/j.biosystemseng.2016.06.014

Blackmore, B. S., Griepentrog, H. W., & Fountas, S. (2006). Autonomous Systems for European Agriculture. In Automation Technology for Off-Road Equipment. Bonn.

Bloem, J., Doorn, M. van, Duivestein, S., Excoffier, D., Maas, R., & Ommeren, E. van. (2014). The Fourth Industrial Revolution Things to Tighten the Link Between it and ot. VINT research report, 1–39.

Branca, G., Lipper, L., McCarthy, N., & Jolejole, M. C. (2013). Food security, climate change, and sustainable land management. A review. Agronomy for Sustainable Development, 33(4), 635–650. doi:10.1007/s13593-013-0133-1

Brooks, R. (1986). A robust layered control system for a mobile robot. IEEE Journal on Robotics and Automation, 2(1), 14–23. doi:10.1007/s10462-012-9368-5

Brooks, R. (1990). Elephants Don't Play Chess. Robotics and Autonomous Systems, 6(1–2), 3–15. doi:10.1016/S0921-8890(05)80025-9

Brooks, R. (1991). Intelligence without Reason. Artificial Intelligence, 47(1–3), 139–159. doi:10.1007/BF01538672

Bundesamt, S. (2015). Landwirtschaft in Deutschland. Statista. https://de.statista.com/statistik/studie/id/6455/dokument/landwirtschaft-statista-dossier/. Accessed 24 May 2017

Cheein, F. A., Herrera, D., Gimenez, J., Carelli, R., Torres-Torriti, M., Rosell-Polo, J. R., et al. (2015). Human-robot interaction in precision agriculture: Sharing the workspace with service units. In 2015 IEEE International Conference on Industrial Technology (ICIT) (pp. 289–295). doi:10.1109/ICIT.2015.7125113

Dzinaj, T., Hörstkamp, S., Linz, A., Ruckelshausen, A., Böttger, O., Kemper, M., et al. (1998). Multi-Sensor-System zur Unterscheidung von Nutzpflanzen und Beikräutern. Zeitschrift für Pflanzenkrankheiten und Pflanzenschutz, XVI, 233–242.

Elliott, J., Deryng, D., Müller, C., Frieler, K., Konzmann, M., Gerten, D., et al. (2014). Constraints and potentials of future irrigation water availability on agricultural production under climate change. In Proceedings of the National Academy of Sciences of the United States of America (Vol. 111, pp. 3239–44). doi:10.1073/pnas.1222474110

Fender, F., Hanneken, M., Stroth, S. I. der, Kielhorn, A., Linz, A., & Ruckelshausen, A. (2006). Sensor fusion meets GPS: individual plant Detection. In Proceedings of CIGR Eur-AgEng/VDI-MEG (pp. 2–6). https://my.hs-osnabrueck.de/ecs/fileadmin/groups/156/Veroeffentlichungen/2006-Bonn-GPS.pdf

Gerland, P., Raftery, A. E., Ševcíková, H., Li, N., Gu, D., Alkema, L., et al. (2014). World Population Stabilization Unlikely This Century. Science, 346(6206), 234–237. doi:10.1126/science.1257469.World

Gorbenko, A., Popov, V., & Sheka, A. (2011). Robot self-awareness: Temporal relation based data mining. Engineering Letters, 19(3), 169–178.

Gorbenko, A., Popov, V., & Sheka, A. (2012). Robot Self-Awareness: Exploration of Internal States. Applied Mathematical Sciences, 6(14), 675–688.

Griepentrog, H. W. (2017). Mess- und Regelungstechnik Agrartechnische Systeme - Definitionen. Teaching Handouts, Universität Hohenheim.

Griepentrog, H. W., Blackmore, S., & Vougioukas, S. G. (2006a). Positioning and Navigation. In A. Munack (Ed.), CIGR Handbook of Agricultural Engineering (pp. 195–204). American Society of Agricultural and Biological Engineers.

Griepentrog, H. W., Nørremark, M., & Nielsen, J. (2006b). Autonomous intra-row rotor weeding based on GPS. In CIGR World Congress Agricultural Engineering for a Better World (pp. 2–6). Bonn.

Hague, T., Marchant, J. a., & Tillett, N. D. (1997). Autonomous robot navigation for precision horticulture. In Proceedings of International Conference on Robotics and Automation (Vol. 3). doi:10.1109/ROBOT.1997.619062

Hiremath, S., van Evert, F. K., Braak, C. Ter, Stein, A., & van der Heijden, G. (2014). Image-based particle filtering for navigation in a semi-structured agricultural environment. Biosystems Engineering, 121, 85–95. doi:10.1016/j.biosystemseng.2014.02.010

Jiang, G., Zhao, C., & Si, Y. (2010). A machine vision based crop rows detection for agricultural robots. In Proceedings of the 2010 International Conference on Wavelet Analysis and Pattern Recognition (Vol. 11, pp. 114–118). doi:10.1109/ICRA.2014.6907296

Johnson, D. G. (1997). Agriculture and the Wealth of Nations. In Papers and Proceedings of the Hundred and Fourth Annual Meeting of the American Economic Association (Vol. 87, pp. 1–12).

King, A. (2017). The future of agriculture. Nature, 544.

Knijff, J. M. van der, Jones, R. J. A., & Montanarella, L. (2000). Soil Erosion Risk Assessment in Europe. European Soil Bureau, Joint Research Center of the European Commission.

Kruse, T., Pandey, A. K., Alami, R., & Kirsch, A. (2013). Human-aware robot navigation: A survey. Robotics and Autonomous Systems, 61(12), 1726–1743. doi:10.1016/j.robot.2013.05.007

Lee, W. S., Slaughter, D. C., & Giles, D. K. (1999). Robotic weed control system for tomatoes. Precision Agriculture, 1, 95–113.

Lu, D. V. (2014). Contextualized Robot Navigation. Washington University in St. Louis.

Luber, M., Spinello, L., Silva, J., & Arras, K. O. (2012). Socially-aware robot navigation: A learning approach. In IEEE International Conference on Intelligent Robots and Systems (pp. 902–907). doi:10.1109/IROS.2012.6385716

Martin, G., Martin-Clouaire, R., & Duru, M. (2013). Farming system design to feed the changing world. A review. Agronomy for Sustainable Development, 33(1), 131–149. doi:10.1007/s13593-011-0075-4

Paraforos, D. S., Griepentrog, H. W., & Vougioukas, S. G. (2016). Country road and field surface profiles acquisition, modelling and synthetic realisation for evaluating fatigue life of agricultural machinery. Journal of Terramechanics, 63, 1–12. doi:10.1016/j.jterra.2015.10.001

Pedersen, S. M., Fountas, S., Have, H., & Blackmore, B. S. (2006). Agricultural robots—system analysis and economic feasibility. Precision Agriculture, 7(4), 295–308. doi:10.1007/s11119-006-9014-9

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., et al. (2009). ROS: an open-source Robot Operating System. In ICRA workshop on open source software (Vol. 3, p. 5).

Reganold, J. P., & Wachter, J. M. (2016). Organic agriculture in the twenty-first century. Nature Plants, 2, 1–8. doi:10.1038/NPLANTS.2015.221

Rosenzweig, C., Elliott, J., Deryng, D., Ruane, A. C., Müller, C., Arneth, A., et al. (2014). Assessing agricultural risks of climate change in the 21st century in a global gridded crop model intercomparison. In Proceedings of the National Academy of Sciences of the United States of America (Vol. 111, pp. 3268–73). doi:10.1073/pnas.1222463110

Shalal, N., Low, T., McCarthy, C., & Hancock, N. (2013). A Review of Autonomous Navigation Systems in Agricultural Environments. In 2013 Society for Engineering in Agriculture Conference: Innovative Agricultural Technologies for a Sustainable Future. (p. 10). Engineers Australia.

Stafford, J. V. (2000). Implementing precision agriculture in the 21st century. Journal of Agricultural Engineering Research, 76(3), 267–275. doi:10.1006/jaer.2000.0577

Stentz, A., Dima, C., Wellington, C., Herman, H., & Stager, D. (2002). A System for Semi-Autonomous Tractor Operations. Autonomous Robots, 13(1), 87–104.

Strothmann, W., Ruckelshausen, A., Hertzberg, J., Scholz, C., & Langsenkamp, F. (2017). Plant classification with In-Field-Labeling for crop/weed discrimination using spectral features and 3D surface features from a multi-wavelength laser line profile system. Computers and Electronics in Agriculture, 134, 79–93. doi:10.1016/j.compag.2017.01.003

Strube, G. (1998). Modelling Motivation and Action Control in Cognitive Systems. Mind Modelling, 111–130.

van Ittersum, M. K., & Rabbinge, R. (1997). Concepts in production ecology for analysis and quantification of agricultural input-output combinations. Field Crops Research, 52, 197–208.

Vázquez-Arellano, M., Griepentrog, H. W., Reiser, D., & Paraforos, D. S. (2016). 3-D Imaging Systems for Agricultural Applications - A Review. Sensors, 16(618), 24. doi:10.3390/s16050618

Vougioukas, S., Fountas, S., Blackmore, B. S., & Tang, L. (2004). Navigation Task in Agricultural Robotics. In International conference on information systems and innovative technologies in agriculture, food and environment. Thessaloniki, Greece (pp. 55–64).

Vougioukas, S. G. (2012). A distributed control framework for motion coordination of teams of autonomous agricultural vehicles. Biosystems Engineering, 113(3), 284–297. doi:10.1016/j.biosystemseng.2012.08.013

Weiss, U., & Biber, P. (2011). Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. Robotics and Autonomous Systems, 59(5), 265–273. doi:10.1016/j.robot.2011.02.011

Xu, H., & Van Brussel, H. (1997). A behaviour-based blackboard architecture for reactive and efficient task execution of an autonomous robot. Robotics and Autonomous Systems, 22(2), 115–132. doi:10.1016/S0921-8890(97)00035-3

CHAPTER 2

Paper A

Autonomous Field Navigation, Data Acquisition and Node Location in Wireless Sensor Networks[1]

David Reiser, Dimitris S. Paraforos, Muhamad T. Khan, Hans W. Griepentrog and Manuel Vázquez-Arellano

Abstract

To overcome the limited transmission range of spatially separated nodes of a wireless sensor network (WSN), a small 4-wheel autonomous robot assembled the data from nodes distributed in a vineyard. First, the robot followed a predefined way-point route between the grapevine rows, in order to evaluate the sensor node locations by their received signal strength indication (RSSI). Then, the recorded and geo-referenced RSSI data were analysed and mapped. By using the evaluated node positions, an optimised second route was generated. While navigating, a laser scanner was used for obstacle detection and avoidance. Path planning with known positions of the nodes reduced the driving time by 15 times compared with the first run, because the hybrid control system used was capable of navigating within the plantation even perpendicular to the row structures. For locating the nodes, results based on trilateration were compared with the values of an attached differential global navigation satellite system (DGNSS). The results showed that it is possible to locate and geo-reference the sensor nodes with a robot, even without any prior knowledge about their absolute position. The best achieved location showed a deviation with DGNSS of 1.2 m and with RSSI trilateration of 0.6 m compared to the actual position.

Keywords: Spatial RSSI variation, WSN, hybrid control, Vineyard navigation, Trilateration

[1] The publication of Chapter 2 was published with consent of the Springer International Publishing. The original publication was in: Precision Agriculture (2017), 18 (3), 279-292. It can be found under the following link: http://doi.org/10.1007/s11119-016-9477-2

2.1 Introduction

Today's precision farming is dependent on reliable information about the production process and its environmental and physico-chemical parameters. Wireless sensor networks (WSN) in connection with sensors can provide a wide variety of ambient conditions such as canopy, temperature and soil moisture. Especially the low-cost and low-power consumption makes WSN useful for tracking and monitoring in diverse areas (Yick et al. 2008). Typical applications for sensor networks are environmental monitoring, precision agriculture, machine and process control, building and facility automation as well as traceability systems (Wang et al. 2006).

In precision farming, many of these applications need to cover large areas with geo-referenced data sets (Camilli et al. 2007). However, as long as the sensor nodes and long-lasting batteries for the devices are still expensive, it is important to maintain cost and power consumption as low as possible (Anisi et al. 2015).

The geo-referencing of the nodes could be done manually, or by analysing the received signals to autonomously locate the nodes. The principles for locating the nodes autonomously, vary from trilateration, triangulation and scene matching (Elnahrawy et al. 2004; Papamanthou et al. 2008). However, non-line-of-sight conditions between the sensor nodes impose errors on the distance estimation (Alsindi et al. 2009). Elnahrawy et al. (2004) investigated the limits of locating nodes using signal strength in a static network and expected a median error of roughly 3 m in indoor environments. These results match with other researches using distance-based self-location of the sensor nodes in a WSN with RSSI values in different environments (Alippi & Vanini, 2006; Wang et al., 2011).

In agricultural areas with a high density of trees and bushes, WSN signals can be affected by several error sources, which makes autonomous node location and data transmission a challenging task. When the network settings and node positions are not well formed, signal-fading and attenuation losses could influence the performance and cause data loss (Jakes 1974). Therefore, it is important to keep the transmission range short, as the data loss rate can be decreased with the reduction of perturbations and radio interference (Bhadauria et al. 2011). Additionally, there is always a balancing between reliable communication among nodes and the need for spatial sampling, as well as keeping the costs acceptable (Vougioukas et al. 2013). Especially for large areas, proper planning is necessary for keeping the whole area under observation (López et al. 2011). Even in high density WSN, connectivity can suffer in some areas (Vecchio and López-Valcarce 2014). However, the spatial density of WSN-nodes is low due to the investment and set up costs, or because of the need to capture site-specific heterogeneity.

The task to collect data from all WSN nodes within a plantation or a field could be performed by an autonomous platform. The robot could be used as a mobile receiver to overcome the transmission gaps within a WSN and attenuation and fading could also be minimized. This was investigated by Bhadauria et al. (2011) who performed data acquisition of spatially dispersed wireless devices, but with previously known sensor node positions. Sichitiu and Ramadurai (2004) presented a beacon-based location technique for sensor nodes, based on a mobile device, that was aware of its own position. Caballero et al. (2008) used a mobile robot to calibrate the WSN positions and analysed the received signal strength indication (RSSI) values for the node location. However, the authors evaluated the results in open fields with line-of-sight between the nodes, which makes it hard to compare with real agricultural environments.

Considering the economic factor, Vougioukas et al. (2013) assumed that the additional costs of a WSN can be better justified for high-value crops, e.g. in orchards and vineyards, where the use of a robotic system is recommended. The robot should be as small as possible, because it fits better to the plantation structures than conventional big machinery (Griepentrog et al. 2013).

One application could be that the sensor nodes are placed by the farmer just at the points of interest and the robot takes care of data acquisition. As the robot would be able to drive close to the sensor nodes, transmission costs and data loss rate of the nodes could be minimized. If the autonomous system could geo-reference and map the transmitted sensor data, the analysis could be done automatically. Also network installation time could be minimised, as manual geo-referencing would not be necessary. Beside mapping the values, the node location could help to shorten the driving time of the robot, which increases the working radius of an autonomous machine.

The objective of this work was to evaluate the use of a robot for data acquisition and data location for spatially separated WSN in agricultural environments. An autonomous data acquisition system based on a small field robot should collect data from distributed sensor nodes within a vineyard. Furthermore, RSSI values of all wireless transmitter nodes should be investigated, to ease and optimise the robot navigation in space and time. The collected RSSI values can be used for geo-referencing the nodes with trilateration. The positions found can be compared with the locations gained from a DGNSS-based positioning system.

2.2 Materials and methods

2.2.1 Wireless Sensor Network (WSN)

The network was built up by the wireless development tool eZ430-RF2500 (Texas Instruments, Dallas, USA). The eZ430-RF2500 consists of two core components: an MSP430F2274

microcontroller and a CC2500 2.4-GHz ISM (Industrial, Scientific and Medical) frequency band wireless transceiver. The eZ430-RF2500T target board was used for the transmitter nodes to realise a stand-alone system (see Figure 1a). The package provided an USB debugging interface which enabled the eZ430-RF2500 to transmit and receive data from a computer via a serial interface (see Figure 1b). This part was used as the receiver node. The development tool eZ430-RF2500 was chosen because of the easy use of the tools and available demo code.

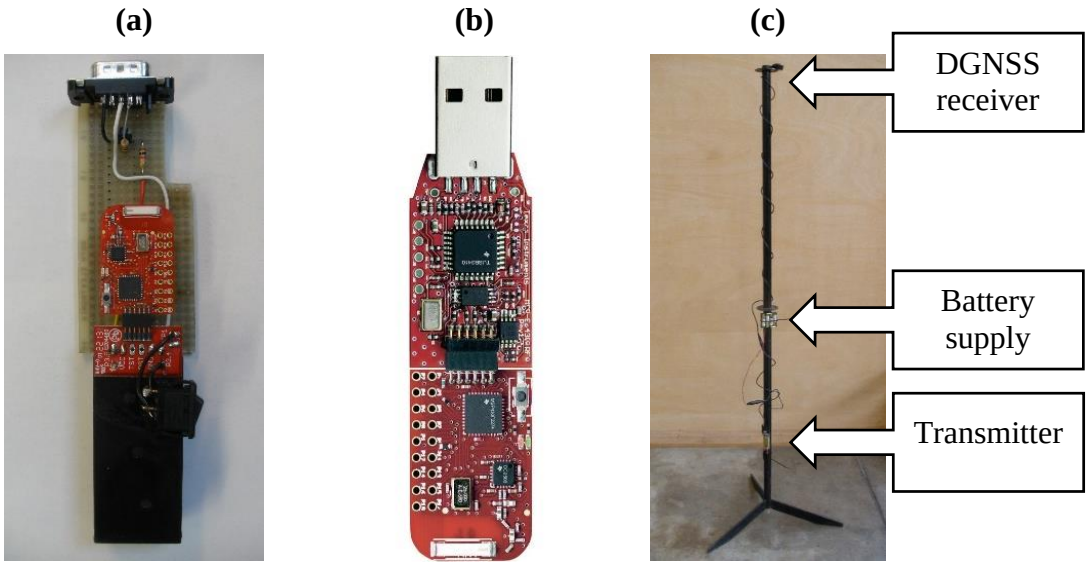

Figure 1: (a) Standalone transmitter node, (b) USB-debugger as receiver and (c) the pole used for the transmitter nodes with a DGNSS receiver and power supply.

The receiver was synchronised with the robot operating system. Four transmitters were used. Each one was mounted on a pole (see Figure 1c), at a height of 0.45 m, having the same height as the receiver on the robot platform. The target boards had a battery power supply and a serial RS232 interface. The serial interface was used to connect with a DGNSS receiver placed at the top of the pole at 1.90 m (see Figure 1c). For DGNSS, the Navilock NL-603P (Navilock, Berlin, Germany) receiver was used, which had a built-in high sensitivity antenna (-162 dBm) and also provided low cost setup. Every transmitter was assigned a unique identification number (ID), in order to be able to filter the received data for every node individually. As soon as a connection was established between transmitter and receiver, the ID and current DGNSS signal was transmitted.

2.2.2 Autonomous outdoor robot

A small 4-wheel autonomous robot called TALOS, manufactured by the Institute of Agricultural Engineering at the University of Hohenheim, Germany, with differential steering was used to move the receiver node of the WSN (see Figure 2a). The size of the robot platform is 400x500x600 mm. The robot was equipped with four separate driving motors with a total of 200 W, and a weight of

40 kg. The average driving speed during the experiment was 0.8 ms^{-1}, with a maximum torque of 2.9 Nm per motor.

The robot system was equipped with an inertial measurement unit (IMU) VN-100 (VectorNav, Dallas, USA), two LMS111 two-dimension (2D) Light Detection and Ranging (LIDAR) laser scanners (Sick, Waldkirch, Germany) at the back and the front. To provide geo-reference, an AgGPS 542 real time kinematic (RTK) (Trimble, Sunnyvale, USA) was used. The robot was controlled by an embedded computer, equipped with i3-Quadcore processor with 3.3 GHz, 4 GB RAM and SSD hard drive. For energy supply, batteries of 12 V/48 Ah capacity were provided. This gave an operating time around 4-5 h, depending on the required motor torque, task and additional weight of equipment placed on the robot platform. The operating system was Ubuntu 14.04 and used the robot operating system (ROS) (see Figure 2b) middleware for the navigation algorithms. ROS uses a combination of C++ and Python programming languages. Some parts of the open-source software code of Frobomind (Jensen et al. 2014) were used and customized to implement the deterministic path planning.

To follow an optimized route, the robot should be small enough to drive even perpendicular to the row structures. To be able to navigate in the chosen terrain, a hybrid control system was used, combining deterministic and reactive path planning. While the robot was carrying the receiver node of the WSN, the position system of the robot was used to locate this mobile beacon and synchronize the received data from the sender nodes.

(a) **(b)**

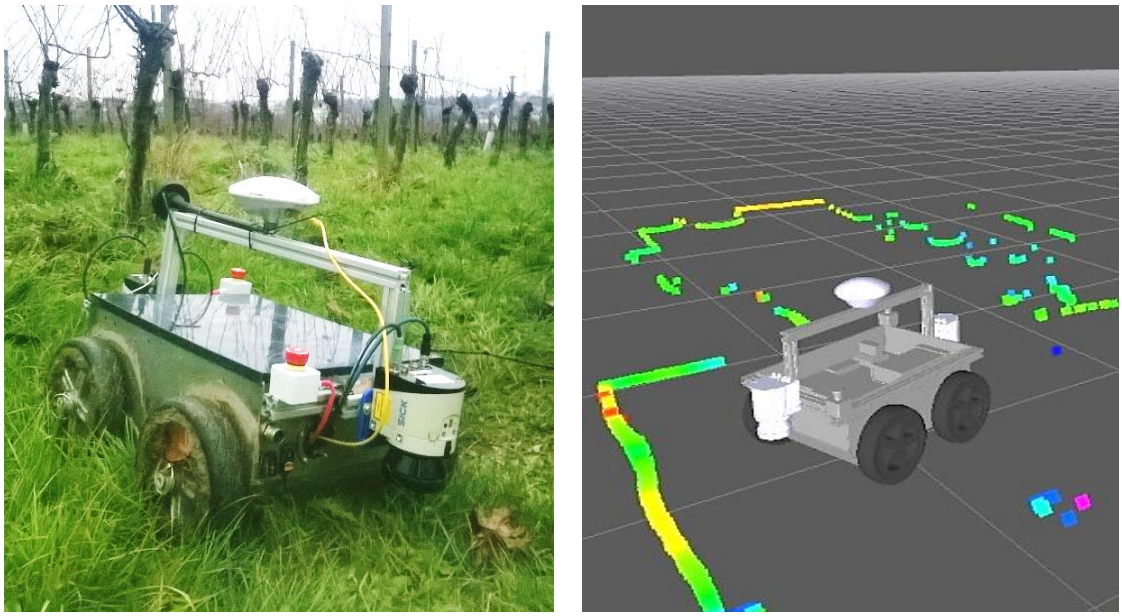

Figure 2: (a) The robot system TALOS and (b) a screenshot of the visualizing software of the robot.

2.2.3 Hybrid robot control

The information of the RTK-GNSS receiver was used for absolute positioning of the robot and for estimating the next goal point of the deterministic path planning. The orientation of the robot was provided by the on-board IMU. The initial orientation of the IMU was determined by using the driving direction obtained from the RTK-GNSS positions.

The robot position and the navigation goal points were defined as separate co-ordinate systems. The distance could be evaluated by receiving the transform between the two frames; this was done by using quaternions. The motor speed was controlled by a simple PID control using the error between the robot position and the direct line to the next goal point. For avoiding obstacles, the deterministic control command was replaced by an obstacle avoidance control (using the two laser scanners). This control layer was activated when an obstacle was detected. When an obstacle blocked the path, the avoidance control defined a new local navigation goal point for the robot, taking the local environment into account. This allowed the robot to drive around the trunks and other small sized obstacles that could be found in the vineyard.

For the first deterministic path following, no obstacle avoidance was needed and the path following was mainly influenced by the surface of the vineyard. For the second task, the hybrid robot control was necessary because it was required to drive perpendicularly through the rows. There, the navigation goal points were set exactly at the estimated node positions. This position was evaluated from the received data of the first task.

2.2.4 Data acquisition and processing

The position of the robot, provided by the RTK-GNSS receiver and the transmitter data, were collected by an external laptop running Windows 8.1 mounted on the robot. The RTK-GNSS data were transmitted via TCP/IP with a rate of 5 Hz. The RSSI value was calculated by the receiver whenever a transmitter was in range and new data were acquired. For acquiring and storing data from the RTK-GNSS and the receiver, software was developed in Microsoft Visual Studio C# 2010. The software developed followed a multi-thread architecture where a unique thread was dynamically created for every connected sensor, allowing parallel data acquisition. As soon as raw sensor data were acquired by each thread, a time stamp with one millisecond resolution was added to the character string. The created character string was stored in a text file (.txt) for post-processing. Despite the asynchronous communication that resulted in data acquisition at different instances, the time stamp eased post-processing. Since the acquired data were measured at different moments, a piecewise cubic interpolation method was used to calculate the values at desired time instances.

For precise RTK-GNSS measurements, a base station was placed at the side of the vineyard with the antenna at a height of 1.20 m. The positional geodetic datum was WGS84. The GNSS antenna was

placed on the robot at the horizontal centre of rotation at a height of 0.6 m. For navigation and integration of the robot movement, the position datum had been internally transformed to the universal transverse mercator (UTM) co-ordinate system in terms of east [m], north [m] and altitude [m].

The robot absolute positions and the evaluated RSSI values were first used to map the RSSI data. As a second step, this geo-referenced RSSI data were used to estimate the node position by trilateration. Since the devices had an on-board DGNSS receiver, the estimated node position could be directly compared with the transmitted node position. To locate the nodes only by their RSSI value using trilateration, the absolute position of the receiver, placed on the robot, and the approximate distance of the transmitter were needed.

The behaviour for the path loss (PL) in free space of a transmitted signal can be described by the Friis equation (Friis 1946). As the RSSI value is dependent on PL, a direct correlation between distance and RSSI can be expected. As in vineyards, the signals may experience attenuation and so a propagation model for signal attenuation prediction was necessary. With a given dataset, this could be also derived empirically (Yang et al. 2015). Thus, a best-fit curve can help to directly estimate the actual distance out of the RSSI value. The following equation was used for the approximate distance determination:

$$d = A * \frac{RSSI^B}{RSSI_{1m}} + C \tag{1}$$

Where $RSSI_{1m}$ is the RSSI value at 1 m distance, RSSI is the actual RSSI value, A, B and C are constants. To ease the navigation and location problem, the 2D distance was considered, assuming that the receiver and the transmitters were at the same height (~0.5 m). Using three beacon positions P_1, P_2 and P_3 and their resulting receiver-transmitter distance d_1, d_2 and d_3, the trilateration for the sensor node position was calculated by the following equation:

$$x = \frac{{d_1}^2 - {d_2}^2 + (P_{2,x} - P_{1,x})^2}{2(P_{2,x} - P_{1,x})} \tag{2}$$

$$y = \frac{{d_1}^2 - {d_3}^2 + (P_{3,y} - P_{1,y})^2 + (P_{2,x} - P_{1,x})^2}{2(P_{2,x} - P_{1,x})} - \frac{(P_{3,y} - P_{1,y})}{(P_{2,x} - P_{1,x})} x \tag{3}$$

The received beacons of the sensor nodes could continuously be used to update the estimated node positions by solving the mean value of all trilateration results.

2.2.5 Field experiments

The first tests were made to determine the maximum transmitter range of the wireless devices under ideal conditions. One pole (see Figure 1c) was placed in an obstacle free courtyard and the robot with a receiver drove towards the pole at a speed of 0.3 ms^{-1}. The maximal transmission radius was manually obtained with a measuring tape.

To investigate the influence of leaves on the RSSI value, a second test was conducted, using an artificial canopy wall. The wall length was 5 m and the height 2.5 m. The trunks were old vine trunks and the leaves were plastic leaves. The whole construction was held by a metal frame and two wires at a height of 1.5 and 0.8 m. In order to simulate the water content of leaves, the canopy wall was moistened. As electromagnetic waves can also spread underwater, it was assumed that the influence of moisture will not block, but just attenuate the signals (Park et al. 2015).

For the field tests, a local vineyard (48.710115N, 9.212913E) in Stuttgart-Hohenheim (Germany) was used, with an almost flat surface. An area of 85 x 60 m, with a total of 33 grapevine rows was explored by the robot. As a common vine training system in Germany, the grapevine rows were stabilized by poles and two horizontal wires at an average height of 1.50 m and 0.8 m (see Figure 2a). Grass with some vehicle tracks and molehills covered the navigation terrain. The four transmitter poles (see Figure 1c) were placed in a square with a sufficient distance to assure that there were no signal overlays. This minimal distance was determined in a transmitter range test before going to the vineyard.

The initial navigation route was defined by using absolute way-points to guide the robot through all rows in the transmission range of the installed wireless devices. The robot used the onboard RTK-GNSS for path tracking. As soon as the wireless receiver at the robot established a connection to one transmitter of the WSN, the sensor data were transferred. The final navigation test used the positional points with the highest RSSI value of every node, to plan an optimised route for the robot to serve all devices.

In Figure 3, the in-field route followed, in UTM co-ordinates, and the positions of the wireless transmitters, provided by the DGNSS receivers, are shown. The UTM positions of established connections between the receiver and the transmitter nodes are also illustrated. In order to examine whether the height of the wireless receiver at the robot platform had an impact on the perception, in the second part of the field experiments (right side of Figure 3), the wireless receiver was placed at a height of 0.7 m compared to the first part (left side of Figure 3) where it was at 0.45 m from the ground.

For evaluating the path loss rate of the RSSI values, the point of the highest RSSI value was used as ground truth for the node position. The transmission distance was calculated from the UTM receiver positions.

2.3 Results and discussion

The testing in a courtyard before going to the vineyard, to investigate the transmitter signal strength, showed a reliable communication up to 26 m at a receiver height of 0.5 m. When using a simulated grapevine canopy of artificial leaves, the signal was mainly interrupted by the trunks. That led to signal fluctuation in the test area. Based on the courtyard testing, the distance between the nodes was set to 30 m in the vineyard. As seen in Figure 3, the devices had been arranged in an almost square configuration.

As soon as the robot reached the end of the field, the points with the highest RSSI value of every single transmitter node were evaluated. These points were considered as transmitter positions for the hybrid controlled navigation.

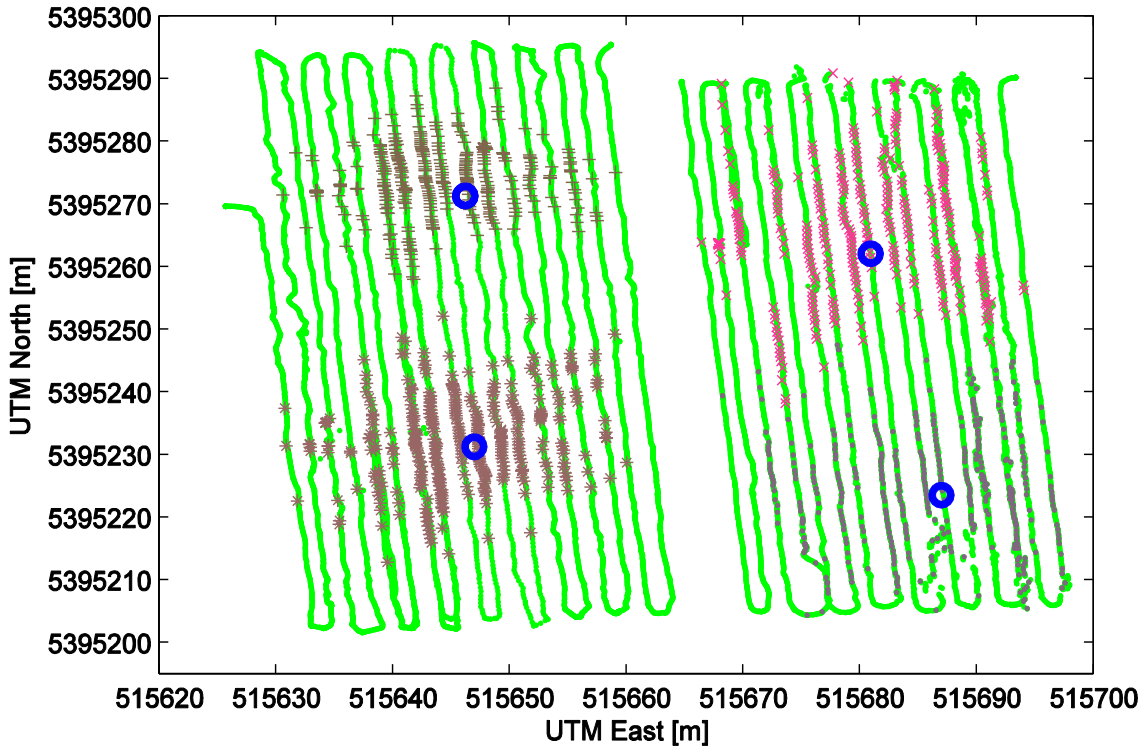

Figure 3: RTK-GNSS route followed by the robot (-, green), mean DGNSS positions of the wireless transmitters (o, blue) and positions with established connection between the wireless receiver and transmitter for the nodes 1 (+), 2 (*), 3 (.) and 4 (x).

This hybrid control system was responsible for navigating the robot, from the end of the field directly to all four evaluated transmitter positions, for collecting the data and then to the starting point. The

route followed and the spatial variability map of the RSSI values, using kriging interpolation (Kravchenko and Bullock 1999), are shown in Figure 4.

The distance travelled along the deterministic path through all rows in the tests (see Figure 3) was around 3000 m, while the hybrid controlled path just needed 200 m of distance travelled to gain a connection to all transmitter nodes. With an average speed of 0.8 ms^{-1}, 3750 s were required for the deterministic path and around 250 s for the hybrid control architecture. This results in a faster performance with a factor of around 15.

When using the DGNSS which was connected to each node, the location algorithm would just need one received signal from each node to locate it. This would make the deterministic path following through all rows unnecessary. It is apparent that the positions of the highest RSSI values and the evaluated positions from the DGNSS receivers do not match. Nevertheless, all four positions from the DGNSS receivers lay in an area where the RSSI value was higher than 44%. This value was considered sufficient to establish a stable data transmission between transmitter and receiver.

The maximum difference between the point of the highest RSSI value and the received DGNSS position was 3.7 m (see Table 1). That led to the assumption, that an absolute geo-referencing precision around 4 m is achievable with just using DGNSS signals. The distance change per percentage of the RSSI value was for all nodes in the range of one meter, with a minimum for node 4 and a maximum at node 1 (see Table 1). The mean standard deviation for all distances of the received values was always higher than 4 m. This makes the RSSI value as a single signal not usable for accurate node location. Interestingly, the maximal and minimal values for the RSSI of all nodes just differed by around 2 %. Consequently, it can be assumed that the changes of the antenna gain of receiver and transmitter signal have a minimal influence on the path loss.

Table 1: Specific parameters of the received signal and the transmitted DGNSS signal.

Node no.	Mean standard deviation [m]	Distance per RSSI [m/%]	RSSI max [%]	RSSI min [%]	Distance max [m]	Distance RSSI max to DGNSS [m]
1	5.0880	1.4286	56	32	16.8	3.7005
2	4.4002	1.0427	57	35	21.1	1.2112
3	5.5079	0.9600	57	33	25.0	3.1384
4	4.1440	0.9058	58	32	27.6	2.4029

From Figure 4, it can be concluded that the RSSI values were dependent on the distance between the transmitter and receiver, as they decrease radially outward to the centre point when there was a received signal. As the standard deviation of the signal is high, a better result could be expected when

the values are averaged over time. This is possible as the node location for most applications is not time-critical. In Figure 5, the RSSI values were averaged every meter.

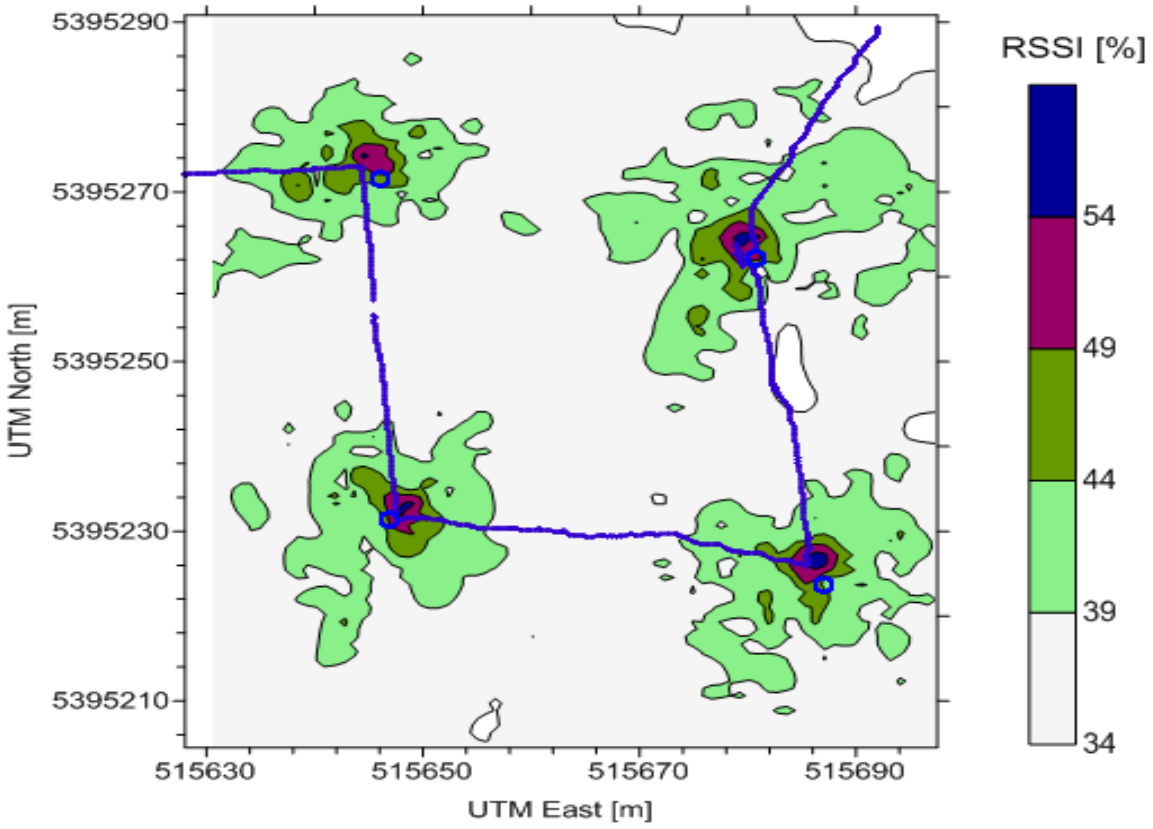

Figure 4: RSSI variability map of the received information in the vineyard together with the transmitter positions (o), and the route that the robot followed based on hybrid control navigation (·).

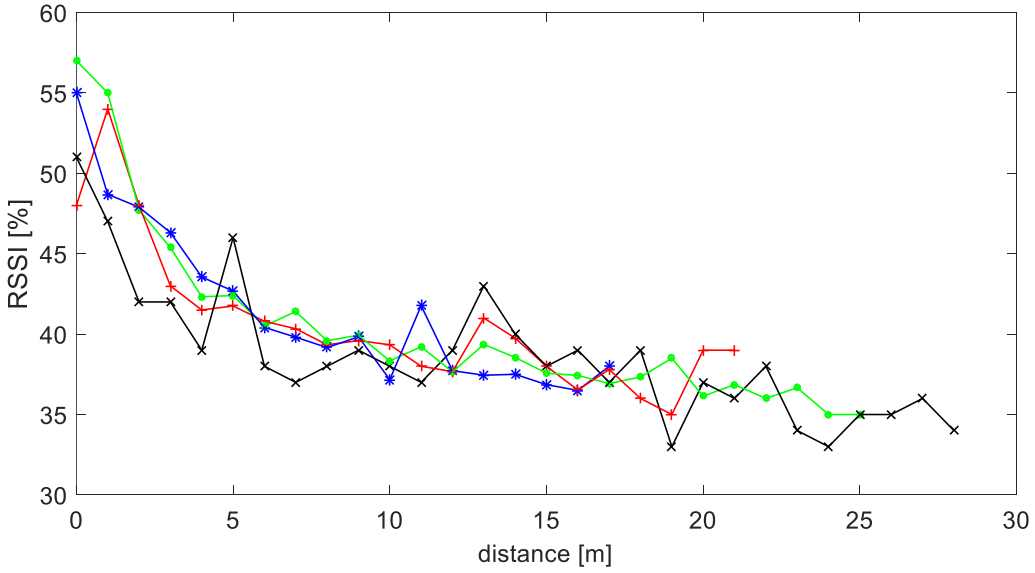

Figure 5: Mean values for node 1 (*), node 2 (+), node 3 (.) and node 4 (x), rounded for every meter.

2.3.1 Node location based on RSSI values

When comparing the values of Figure 5, it can be seen that the maximum transmission range of nodes 3 and 4 was higher compared to that of numbers 1 and 2. This could happen because the receiver height was changed in this part from 0.5 m to 0.7 m. When fitting Equation 1 to the data set of Figure 4, using the Matlab R2015b Curve Fitting Tool (MathWorks, Natick, MA, USA) with the Levenberg-Marquardt method, the following parameters for A, B and C were derived (see Table 2):

Table 2: Best-fit equation parameters for the distance evaluation of the transmitter nodes of Equation 1.

node no.	A	B	C
1	2.196	-5.306	-2.164
2	0.7913	-8.402	-0.4043
3	1.408	-6.87	-1.491
4	10.74	-2.656	-13.43

The constants of Table 2 together with Equations 2 and 3, give the option to analyse the node location by the RSSI value. When the robot is not passing close to nodes in a WSN, it cannot be guaranteed that the highest RSSI value is also corresponding to the node position. The following evaluation assumed that the robot was not passing areas with an RSSI value higher than 49 %. This corresponds approximately to a distance of 3 m to the sensor. To guarantee good data transmission, a minimal RSSI value of 39 % was assumed. So for the trilateration only, the areas with RSSI between 39 % and 49 % were considered. For solving the trilateration, three points in the dataset were randomly picked. This was done for the dataset in the range between 1 and 500 times. Out of three points, a resulting node position was evaluated and compared with the other resulting points. Out of the resulting node positions, the mean value was composed and the difference to the node position was evaluated. The results are shown in the following graph for all 4 nodes (see Figure 6). As can be seen in Figure 6 and the values in Table 3, the best fit was possible for node 2, with a mean distance to the assumed node position of 0.5992 m. For all nodes, the results stabilised after around 100 selected random points for the trilateration.

Table 3: Mean value of all calculated node positions with trilateration.

Node no.	Mean distance [m]	Standard deviation [m]	RMSE [m]
1	1.4942	0.4855	0.4850
2	0.5992	0.6789	0.6782
3	2.2021	0.6207	0.6201
4	2.1818	0.8061	0.8053

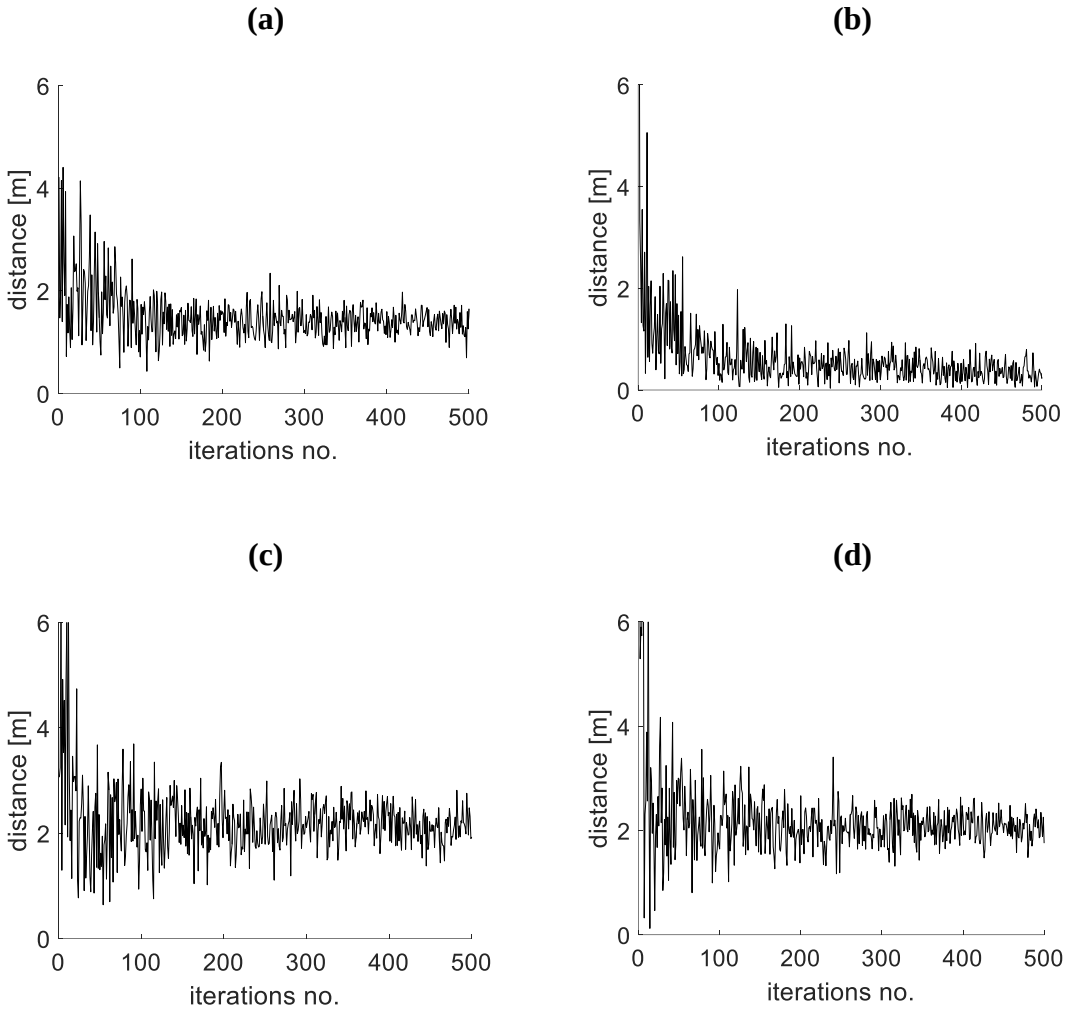

Figure 6: Distance between trilateration mean value and the highest RSSI value of the nodes 1 (a), 2 (b), 3 (c) and 4 (d).

The spatial distribution for 1000 randomly selected points for every node can be seen in Figure 7. There, it can be seen that the node estimation based on trilateration, had a high variance, but it also shows a normal circular distribution around the node. Therefore, the best results can be expected when sample points all around the node can be collected. This could help to locate the node position more precisely and have the option for even better node location based on particle or kalman filters

(Caballero et al. 2008; Papamanthou et al. 2008). Also, using another path loss model such as the Hata Urban Model could help to increase precision (Chrysikos and Kotsopoulos 2013).

Even without additional filters or path loss models, the results are already more precise than the attached DGNSS. This gives the option to skip this additional device and just use the trilateration technique together with a precise location system on the robot, to geo-reference the nodes. The maximum RSSI value used for the trilateration was 49 %, which correlates with a distance of approximately 3 m (see Figure 4). Passing the nodes at this distance should be sufficient to evaluate the precise position. With a row width of 1.5 m, as in the vineyard used, precise node positioning would even be possible if the robot would always skip 3 rows.

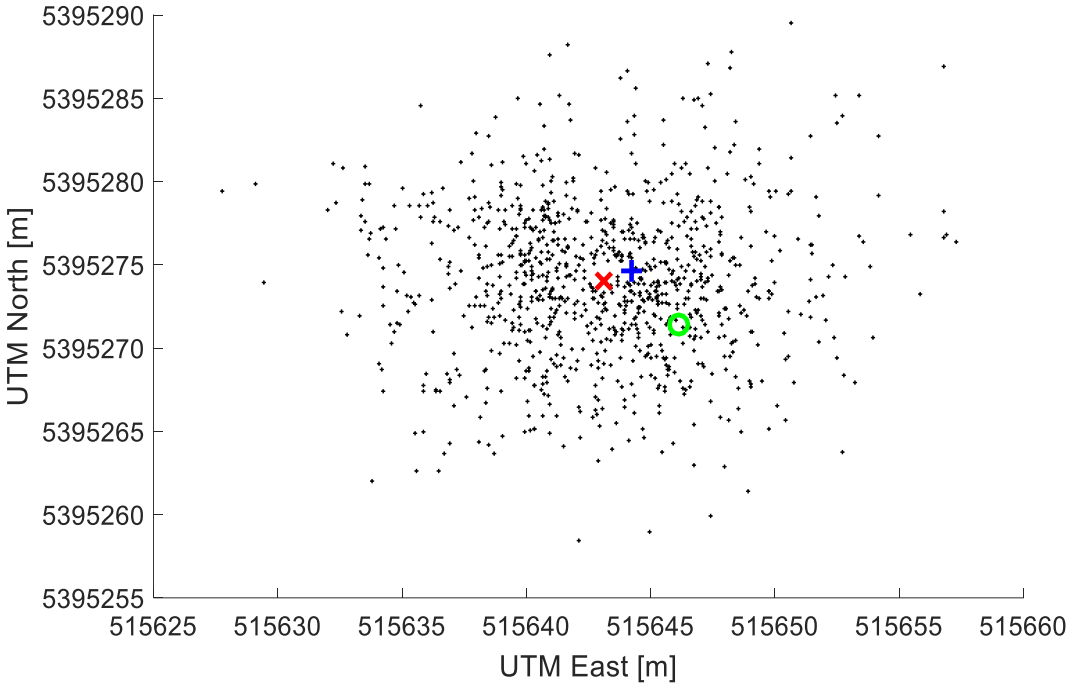

Figure 7: Spatial distribution of 1000 randomly picked trilateration results for the RSSI value of estimated node 1 position (+), DGNSS position (o) and estimated trilateration position (x).

Alippi and Vanini (2006) achieved the best localisation precision with 7 fixed known anchors with an average error of 2.3 m which is similar to the current results (see Table 3). When comparing the results with the theoretical error developments shown by Savvides et al. (2005), the RMS location error reached around 1 m for more than 60 nodes in the network. This means, that the results achieved are at least as precise as would be achievable with a highly dense WSN with more than 60 neighbour nodes. The reason for these good results could be due to the use of the robot RTK-GNSS system, which makes a highly precise receiver location possible.

Apart from the RS232 interface, the transmitters were also equipped with analogue and digital inputs. This gives the capability to connect many sensors in order to provide information that is essential for performing agricultural tasks, related to precision agriculture. Soil moisture or soil temperature sensors could be connected to the transmitter nodes. Then, the robot can collect and geo-reference the data via trilateration. This would be possible even when some of the sensors were removed, or had been moved to another place since the last data acquisition. The short transmission distance to the robot allows production of more cost and power efficient devices that are capable of lasting a long time in the field without any additional maintenance service. Whenever additional data is required, the navigation algorithm of the robot and the evaluated transmitter positions could be restarted to follow the route without any additional information. A system capable of recharging on its own would be able to collect data from a large field without any supervision. All this could help to make possible an easy and completely automated WSN node setup in a field or vineyard with low cost sensors.

In the future, it has to be evaluated whether the skipping of several rows for collecting the RSSI values for trilateration would be sufficient to evaluate the transmitter positions with appropriate accuracy. Also the trilateration behaviour under real path following in the vineyard and at different seasons has to be evaluated. Another option for optimizing the process of data collection could be the use of drones, because they do not have to take care about obstacles, when operating at a certain height. However, it should be investigated how the canopy will affect the signalling when drones do the data acquisition. When collecting the data from ground based wireless antennae, the flight path of the drones should be denser than the path of a ground vehicle, because the flight height will influence the minimal possible receiver distance to the nodes. Also, it should be taken into account, that the communication signals of the drones and the WSN could overlay, causing navigation problems.

Another big advantage of data collection by driving to the sensor with a robot is that the signals could not be easily disturbed by interfering transmitters from outside sources. So short range data transmission could help to keep the surveillance of the environment running even under harsh conditions.

2.4 Conclusions

The results of the field tests showed good performance in detecting and servicing all transmitter nodes in the vineyard without any prior knowledge of their absolute position. Therefore, data collection and transmission can be completed by a fully autonomous operation. The second run did a minimized total distance because not all the rows (tree or crop) needed to be passed by the field robot. Furthermore, due to the hybrid control architecture, the robot could react to environmental changes and avoid obstacles. This updated route minimized the running time of the robot and made the overall

operation more cost-effective. In the present work, a DGNSS was connected to the wireless transmitters and the information was acquired by the receiver on the robot. Also, the use of trilateration for the location of the nodes was investigated. Trilateration showed better location results than the one provided by DGNSS data at all nodes. Therefore, it could be assumed that, with the use of RTK-GNSS on the robot and RSSI trilateration, the sensor nodes could be precisely geo-referenced by just passing the nodes.

Acknowledgements

The project was conducted at the Max-Eyth Endowed Chair (Instrumentation & Test Engineering) at Hohenheim University (Stuttgart, Germany), which is partly grant funded by the Deutsche Landwirtschafts-Gesellschaft e.V. (DLG).

References

Alippi, C., & Vanini, G. (2006). A RSSI-based and calibrated centralized localization technique for wireless sensor networks. In: Proceedings - Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops, PerCom Workshops 2006, pp 301–305. doi:10.1109/PERCOMW.2006.13

Alsindi, N., Duan, C., Zhang, J., & Tsuboi, T. (2009). NLOS Channel Identification and Mitigation in Ultra Wideband ToA-Based Wireless Sensor Networks. In: 6th Workshop on Positioning, Navigation and Communication (WPNC 2009), Hannover, pp 59–66. doi: 10.1109/WPNC.2009.4907804

Anisi, M. H., Abdul-Salaam, G., & Abdullah, A. H. (2015). A survey of wireless sensor network approaches and their energy consumption for monitoring farm fields in precision agriculture. Precision Agriculture, 16(2), 216–238. doi:10.1007/s11119-014-9371-8

Bhadauria, D., Tekdas, O., & Isler, V. (2011). Robotic data mules for collecting data over sparse sensor fields. Journal of Field Robotics, 28(3), 388–404. doi:10.1002/rob.20384

Caballero, F., Merino, L., Gil, P., Maza, I., & Ollero, a. (2008). A probabilistic framework for entire WSN localization using a mobile robot. Robotics and Autonomous Systems, 56(10), 798–806. doi:10.1016/j.robot.2008.06.003

Camilli, A., Cugnasca, C. E., Saraiva, A. M., Hirakawa, A. R., & Corrêa, P. L. P. (2007). From wireless sensors to field mapping: Anatomy of an application for precision agriculture. Computers and Electronics in Agriculture, 58(1), 25–36. doi:10.1016/j.compag.2007.01.019

Chrysikos, T., & Kotsopoulos, S. (2013). Site-specific Validation of Path Loss Models and Large-scale Fading Characterization for a Complex Urban Propagation Topology at 2.4 GHz. In: Proceedings of the Interntational MultiConference of Engineers and Computer Scientists 2013 Vol II, IMECS 2013, Hong Kong, pp 585–590.

Elnahrawy, E., Li, X., & Martin, R. P. (2004). The Limits of Localization Using Signal Strength : A Comparative Study. In: First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON, pp 406–414. doi:10.1109/SAHCN.2004.1381942

Friis, H. T. (1946). A Note on a Simple Transmission Formula. In: Proceedings of the IRE, 34(5), 254–256. doi:10.1109/JRPROC.1946.234568

Griepentrog, H. W., Dühring Jaeger, C. L., & Paraforos, D. S. (2013). Robots for Field Operations with Comprehensive Multilayer Control. KI - Künstliche Intelligenz, 27(4), 325–333. doi:10.1007/s13218-013-0266-z

Jakes, W. C. (1974). Microwave mobile Communications. (William Perkins, Ed.). New York: NY: Wiley Interscience.

Jensen, K., Larsen, M., Nielsen, S., Larsen, L., Olsen, K., & Jørgensen, R. (2014). Towards an Open Software Platform for Field Robots in Precision Agriculture. Robotics, 3(2), 207–234. doi:10.3390/robotics3020207

Kravchenko, A., & Bullock, D. G. (1999). A Comparative Study of Interpolation Methods for Mapping Soil Properties. Agronomy Journal 91, 393–400. doi:10.2134/agronj1999.00021962009100030007x

López, J. A., Garcia-Sanchez, A.-J., Soto, F., Iborra, A., Garcia-Sanchez, F., & Garcia-Haro, J. (2011). Design and validation of a wireless sensor network architecture for precision horticulture applications. Precision Agriculture, 12(2), 280–295. doi:10.1007/s11119-010-9178-1

Papamanthou, C., Preparata, F. P., & Tamassia, R. (2008). Algorithms for Location Estimation Based on RSSI Sampling. In: Algorithmic Aspects of Wireless Sensor Networks, Springer Berlin Heidelberg, pp 72–86. doi: 10.1007/978-3-540-92862-1_7

Park, D., Kwak, K., Kim, J., & Chung, W. K. (2015). Underwater Sensor Network using Received Signal Strength of Electromagnetic Waves. In: IEEE International Conference on Intelligent Robots and Systems, Hamburg, pp 1052–1057. doi: 10.1109/IROS.2015.7353500

Savvides, A., Garber, W. L., Moses, R. L., Member, S., Srivastava, M. B., & Member, S. (2005). An Analysis of Error Inducing Parameters in Multihop Sensor Node Localization, IEEE Transactions on Mobile Computing 4(6), 567–577. doi: 10.1109/TMC.2005.78

Sichitiu, M. L., & Ramadurai, V. (2004). Localization of Wireless Sensor Networks with a Mobile Beacon. In: IEEE International Conference on Mobile Ad-hoc and Sensor Systems, pp 174–183. doi:10.1109/MAHSS.2004.1392104

Vecchio, M., & López-Valcarce, R. (2014). Improving area coverage of wireless sensor networks via controllable mobile nodes: A greedy approach. Journal of Network and Computer Applications, 48, 1–13. doi:10.1016/j.jnca.2014.10.007

Vougioukas, S., Anastassiu, H. T., Regen, C., & Zude, M. (2013). Influence of foliage on radio path losses (PLs) for wireless sensor network (WSN) planning in orchards. Biosystems Engineering, 114(4), 454–465. doi:10.1016/j.biosystemseng.2012.08.011

Wang, N., Zhang, N., & Wang, M. (2006). Wireless sensors in agriculture and food industry—Recent development and future perspective. Computers and Electronics in Agriculture, 50(1), 1–14. doi:10.1016/j.compag.2005.09.003

Wang, X., Yuan, S., Laur, R., & Lang, W. (2011). Dynamic localization based on spatial reasoning with RSSI in wireless sensor networks for transport logistics. Sensors and Actuators A: Physical, 171(2), 421–428. doi:10.1016/j.sna.2011.08.015

Yang, X. G. X., Ming, M. C., & Wang, Y. (2015). A model with leaf area index and apple size parameters for 2.4 GHz radio propagation in apple orchards. Precision Agriculture, 16, 180–200. doi:10.1007/s11119-014-9369-2

Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. Computer Networks, 52(12), 2292–2330. doi:10.1016/j.comnet.2008.04.002

CHAPTER 3

Paper B

3D Imaging with a Sonar Sensor and an Automated 3-Axes Frame for Selective Spraying in Controlled Conditions[2]

David Reiser, Javier M. Martín-López, Emir Memic, Manuel Vázquez-Arellano, Steffen Brandner and Hans W. Griepentrog

Abstract

Autonomous selective spraying could be a way for agriculture to reduce production costs, saving resources, protecting the environment and helping to fulfil specific pesticides regulations. The objective of this paper was to investigate the use of a low-cost sonar sensor for autonomous selective spraying of single plants. For that, a belt driven autonomous robot was used, with an attached 3-axes frame with 3 degrees of freedom. In the tool centre point (TCP) of the 3-axes frame, a sonar sensor and a spray valve were attached to create a point cloud representation of the surface, detect plants in the area and perform selective spraying. The autonomous robot was tested on replicates of artificial crop plants. The location of each plant was identified out of the acquired point cloud with the help of Euclidian Clustering. The gained plant positions were spatially transformed from the coordinates of the sonar sensor to the valve location to determine the exact irrigation points. The results showed that the robot was able to automatically detect the position of each plant with an accuracy of 2.7 cm and could spray on these selected points. This selective spraying reduced the used liquid by 72%, when comparing it to a conventional spraying method in the same conditions.

Keywords: Selective spraying, single plant, agricultural robot, sonar, ultrasonic, point cloud, 3D imaging

[2] The publication of Chapter 3 is done with the consent of the MDPI Publishing. The original publication was in: Journal of Imaging 2017, 3(1):9. It can be found under the following link: http://doi.org/10.3390/jimaging3010009

3.1 Introduction

Pesticides are one of the main contamination factors of water sources and ecosystems in the world (El-Gawad 2016). In the European Union for example, a rate of 300 000 tons of pesticides were applied each year. This lead to a water pollution that exceeded the threshold set by the regulation commission of the European Union (N. 546/2011) (Gaillard et al. 2016). However, the excess of weeds increases the cost of the cultural practices, destroy crops, modify the effectiveness of agricultural equipment and decrease the fertility of soils (Kira et al. 2016; Oerke 2006). This leads to the assumption that the productivity of conventional farming is dependent on chemicals (Oerke 2006). Although the use of pesticides, fungicides and fertilizers contributes to improving the quality and productivity of crops in agriculture, significant problems were reported when the chemicals are applied uniformly over the fields. Typical related problems are the negative effects on the environment (Doulia et al. 2016; El-Gawad 2016), herbicide-resistant weeds (Heap 2014) and human and animal health issues (Alves et al. 2016). A decrease of the distributed chemicals, would not only result in a decrease in the contamination of ecosystems (Solanelles et al. 2006), but also in a reduction of costs and an increase of crop production efficiency (Chang et al. 2016). Therefore, one goal for future farming should be to apply chemicals just at the place where they are required (Gonzalez-de-Soto et al. 2016). Instead of applying fixed amounts of pesticides over the field, each plant could be detected individually and treated when needed (Peteinatos et al. 2014). Modern technology should already be able to reduce the amount of the application, working time and environmental damage (Oberti et al. 2015; Peteinatos et al. 2014). However, this requires robust and reliable sensors, software processing and application systems (Blackmore et al. 2006; Slaughter et al. 2008).

The technologies for the automatic detection and removal of weeds progressed significantly in the last decades. This was mainly because of the use of new sensor types and improvements in crop management and control treatments with herbicides, pesticides or fungicides (Gonzalez-de-Soto et al. 2016; Kunz et al. 2016; Lee et al. 1999; Peteinatos et al. 2014). Many investigations have focused on the use of image analysis techniques, stereo photogrammetry, spectral cameras, time-of-flight-cameras, structured light sensors and light detection and ranging (LiDAR) laser sensors (Backman et al. 2012; Garrido et al. 2015; Gil et al. 2007; Kira et al. 2016; Woods and Christian 2016). Typically the sensors were used on tractor implements to recognize the weeds and to use this information for selective spraying (Berge et al. 2012), or on robots, to automate the whole navigation and application process of selective spraying (Oberti et al. 2015).

However, most of the mentioned sensor principles are limited by the high costs and/or complexity to acquire and to process the information (Bietresato et al. 2016). This brings difficulty for the farmer to implement this technology, as agricultural machinery should have a good cost-benefit rate.

Since the new Microsoft Kinect v2 sensor was introduced to the consumer market, it was also used in agricultural research for environment perception (Andújar et al. 2016; Y. Jiang et al. 2016; Kusumam et al. 2016; Vázquez-Arellano et al. 2016). This time-of-flight based sensor system is quite promising, but is still highly affected by sunlight and requires high computational resources to process the data (Vázquez-Arellano et al. 2016).

Using sonar sensors for selective spraying was already part of several research papers. A research was conducted comparing LiDAR systems with ultrasonic sensors to estimate the vegetation. The results indicated a good correlation between the estimation made by LiDAR and ultrasonic sensors (Tumbo et al. 2002). The first sprayer with automatic control was tested 1987 (Giles et al. 1987). This system used ultrasonic sensors to measure the distance to the foliage. The impact of the volume reduction of liquid was between 28-35% and 36-52%. It was confirmed that spraying systems can be controlled more precisely using ultrasonic and optical sensors to open and close individual nozzles by recognizing the presence or absence of plants (Doruchowski and Holownicki 2000). Also two ultrasonic transducers with solenoid valve control were developed, finding savings of 65% and 30% respectively of the liquid spray (Solanelles et al. 2006). Even a ultrasonic low-cost spraying system was developed and tested in a wild blueberry field to reduce initial costs (Swain et al. 2009). Additionally, ultrasonic sensors have showed very good results to determine plant height (Zaman et al. 2005).

In maize, the weed height is in general lower than the crop height, therefore a correct height estimation would be enough to distinguish between weed and crop (Peteinatos et al. 2014). A research concluded that in a simplified system of semi-automatic spraying, the most important parameters are the plant height and the planting density (Walklate et al. 2006). Therefore, just the plant height information would allow already a significant reduction in spray volume, while maintaining the coverage rates and the penetration similar to conventional spraying methods (Gil et al. 2007).

The most described selective spraying methods performed a 1D control with the sonar sensors, using the distance measurement to activate the nozzles, but not taking the 3D position of the sensors into account. When this information is known, it is possible to even create out of the 1D information of a sonar sensor a 3D image representation of the environment. This point clouds are useful representations, allowing to estimate the shape, position and size of specific objects. They are commonly used for autonomous navigation and mapping (Reiser, Garrido, et al. 2016; Reiser, Paraforos, et al. 2016; Zlot and Bosse 2014), plant detection (Weiss and Biber 2011), harvesting (Back et al. 2014) and phenotyping (Dornbusch et al. 2007; Garrido et al. 2015). However, the acquisition of point clouds is mainly performed with expensive equipment like LiDAR, stereo cameras, time-of-flight-cameras or structured-light sensors.

All these vision sensors are light sensitive and thus inherently flawed for agricultural applications, bringing more challenges for the software algorithms (Vázquez-Arellano et al. 2016). This is not the case for sonar sensors. Therefore, a low-cost 1D sonar sensor mounted on a 3D positioning system could replace visual sensors. Combined with adequate algorithms, the acquired data could be used to detect single plant positions.

The main objective of this paper is to describe and evaluate the capability of a low-cost system for 3D point cloud generation and for the use of single plant detection and treatment. For that, an ultrasonic sensor with a cost of around 2 € was used. Compared to other possible distance sensors, it is one of the cheapest solutions available on the market. For other low cost sensors such as a 1D-LiDAR a price of approximately 80 € and for a cheap 3D camera (e.g. Microsoft Kinect v2) around 200 € have to be considered. The sonar sensor was attached to a 3D axis frame, which in turn was mounted on a mobile robot. The ultrasonic sensor should be used to detect plants in the working area of the 3D frame and to perform a selective plant spraying. To extend the working area of the system, a mobile robot platform was used to move the autonomous system to the next area after the selective spraying was performed.

The paper is ordered as follows: The second chapter describes the whole setup of sensors and software tools. This includes the mobile robot, 3D frame and the calibration setup. In addition, the experiment and the data processing are described in detail. The third chapter present the results, combined with the discussion. This is followed by the conclusion.

3.2 Materials and methods

3.2.1 Hardware and sensor setup

The developed automatic system was composed by four parts: A robotic vehicle, a movable frame with three degrees of freedom a distance-sensor and a precision spraying system. The frame mounting point for the sensor was assumed as the TCP. The working space of the frame was 1.1 m x 1.4 m x 1.0 m (*x, y, z* respectively). The whole system was mounted on a field robot platform called Phoenix (developed at Hohenheim University). The robot platform is able to drive 5 km/h up to a 30 degree slope and with an additional payload of 200 kg. The vehicle weights around 450 kg. The system is driven by two electrical motors with a total power of 7 kW. The system was powered by four built-in batteries, which could provide an operation time of approximately 8-10 hours, depending on driven speed, elevation and workload. The navigation computer of the robot was a Lenovo ThinkPad with i5 processor, 4 GB RAM and 320 GB disk space.

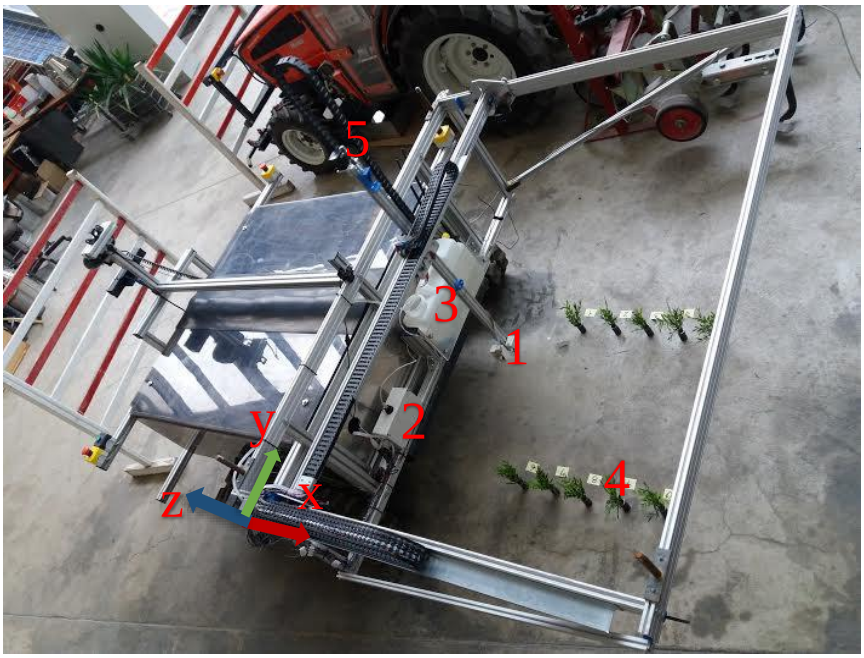

Figure 1: Description of the used automatic spraying system with all the hardware components mounted (1. TCP, 2. CPU + motor controllers, 3. water tank, 4. plants, 5. calibration point).

As distance sensor, a low-cost ultrasonic sensor HC-SR04 (Cytron Technologies, Johor, Malaysia) was mounted at the TCP of the frame, pointing downwards, perpendicular to the surface. The sensor had a measuring range from 2 cm to 400 cm and an effective angle measurement of 15 degree. The measuring rate was limited to 10 Hz to avoid signal overlapping. The precision spraying system used a DC-Pump (Barwig GmbH, Germany) with a maximum supply of 22 l/min and maximum pressure of 1.4 bar. Water was used to emulate the pesticide/herbicide application, which was flowing through a plastic hose with a diameter of 10 mm that ran along the frame to the TCP. The movement of the frame axes was done through bipolar stepper motors Nema 17 (Osmtec, Ningbo, China). The stepper motors (2 for the *x*-axis, 1 for the *y*-axis, and 1 for the *z*-axis) had a step angle of 1.8 degree (200 steps/revolution), a holding torque of 0.59 Nm, a step accuracy of 5% and a maximum travel speed for each axis of 0.2 m/s. The frame motors and sensors were controlled with two low-cost Raspberry Pi 2, Model B computers (Raspberry Pi Foundation, Caldecote, United Kingdom). To control the four stepper motors and the DC-Pump, three motor controllers Gertbot V 2.4 (Fen Logic Limited, Cambridge, United Kingdom) were connected to one Raspberry Pi. The sonar sensor was attached to the GPIOs of a second Raspberry Pi. The second Raspberry Pi was used because the attached Gertbots blocked the access to the Raspberry Pi GPIOs. So using a second Raspberry Pi allowed an easier and faster implementation of the sonar sensor to the 3D frame. The whole setup of the robot with frame and plants is shown in Figure 1.

3.2.2 Software setup

The control software for the robot system was programmed for the ROS-Indigo middleware. This included the driving and navigation software of the mobile robot, as well as the control software for the frame, data acquisition of the sensors, plant detection algorithms and the precision spraying device. For visualization of the processes and acquired data, the built-in ROS 3D visualization tool "rviz" and the survey tool "rqt" were used. The driver software for the 3D frame, the pump and the motors were programmed directly on one of the Raspberry Pi computers. The sonar driver was programmed on the second Raspberry Pi. All computers were connected to the local robot network with Ethernet cables. For synchronization of all included systems and sensors of the robot and to collect the datasets, the ROS environment was installed on all systems and was used to define a synchronized time for the whole robot setup. The higher-level software was programmed on the robot navigation computer, for defining goal points and assessing the acquired data. For the point cloud assessment, parts of the PCL-Library were used (Rusu and Cousins 2011). The Raspberry Pi operating system was Raspbian Jessie 4.1 and on the navigation computer Ubuntu 14.04. All parts of the software were programmed in a combination of C/C++ and Python programming languages. For post processing of the recorded point clouds Matlab R2015b (MathWorks, Natick, MA, USA) was used.

3.2.3 Calibration and system test

For calibrating the frame, the relative position of the 3-axis frame was measured by a highly precise total station (SPS930, Trimble, Sunnyvale, USA), together with a MT1000 tracking prism. In order to do that, the prism was attached to the highest point of the *z*-axis on the frame over the sonar sensor (see Figure 1). After that, the frame was moved to seven different positions inside the workspace and the relative total station output was evaluated. The measured points were spatially separated and covered the whole workspace. The total station output was compared with the moved steps of the motors and were used to define the calibration parameters of the software.

For calibrating the sonar sensor, measurements on a flat surface of a concrete floor were done and the results were compared with the movement of the *z*-axis. The TCP was moved downwards by the software to the relative distances of 10, 20 and 30 cm. This test was performed three times per height at different positions. All calibrations were referred to the frame coordinate system. The results were evaluated with the mean value ($\bar{d}$) (1), standard deviation (std_{dev})(2) and root mean square error (RMSE)(3). d_i describes the distance output of the sensor, N the number of tests and d_r the real value (ground truth).

$$\bar{d} = \frac{1}{N}\sum_{i=1}^{N} d_i \quad (1)$$

$$std_{dev} = \sqrt{\frac{1}{N} * \sum_{i=1}^{N} (d_i - \bar{d})^2} \quad (2)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (d_i - d_r)^2} \quad (3)$$

The values reached by the sonar system were compared to the theoretical reachable accuracy of a stereo vision system. Stereo vision systems are quite common sensing systems for automated outdoor vehicles, making them reasonably comparable with the sonar sensor system (Rovira-Más et al. 2010). As the accuracy of camera systems is highly affected by the object distance to the sensor position, this effect increases when using a stereo camera for 3D point cloud generation. Stereo cameras need textures in the images to find matching pixels in the two camera images, what brings lack of distance measurements since matching is not achieved. However, if a perfect stereo camera is considered, the depth accuracy ΔZ could be described with (Pajares et al. 2016):

$$\Delta Z = \frac{B}{2} tan\left(\tan^{-1}\left(\frac{2Z}{B}\right) + \theta\right) - Z \quad (4)$$

With B as baseline (distance between the two camera systems), Z as the object distance, and θ as the pixel expanding angle. θ could be described as:

$$\theta = \arctan\left(\frac{p}{f}\right) \quad (5)$$

With p as the pixel size and f as the focal length of the camera (Pajares et al. 2016). For comparing the two sensor principles, two different cases were considered. First, replace the sonar sensor with a stereo camera and second, replace the sonar sensor and the 3D frame by one fixed camera, spotting the workspace of 1.3 x 0.9 m. As camera parameters the values provided by Pajares et al. were used (Pajares et al. 2016).

3.2.4 Experiment description

The experiment was realized at the University of Hohenheim (Germany) and used artificial crop plants with a height of 20 cm. To simulate the spatial separation of corn crop, the positions of the plants consisted of 2 crop rows with 5 plants each, following the seeding density for maize used by (Reckleben 2011), with 0.75 m between row and 0.14 m between plants. The ground truth position of each plant was measured using a pendulum attached to the TCP that was moved over the top of

the plants. The software based TCP position was saved as plant reference. Just the (x, y) coordinates were considered (see Figure 1).

To automatically detect the plants with the sonar sensor, first a scan of the whole working area was performed. The scanned route was defined to drive lineally on the field along the frame with a distance between each scanned line of 2 cm in the x-axis. The representation of a typical scan is shown in Figure 2a, where the green line is showing the planed path for the ground scanning. In addition, some tests with 2x1 cm and 2x3 cm were performed. However, the 2x2 cm resolution was used because it achieved the best results in the smallest amount of time.

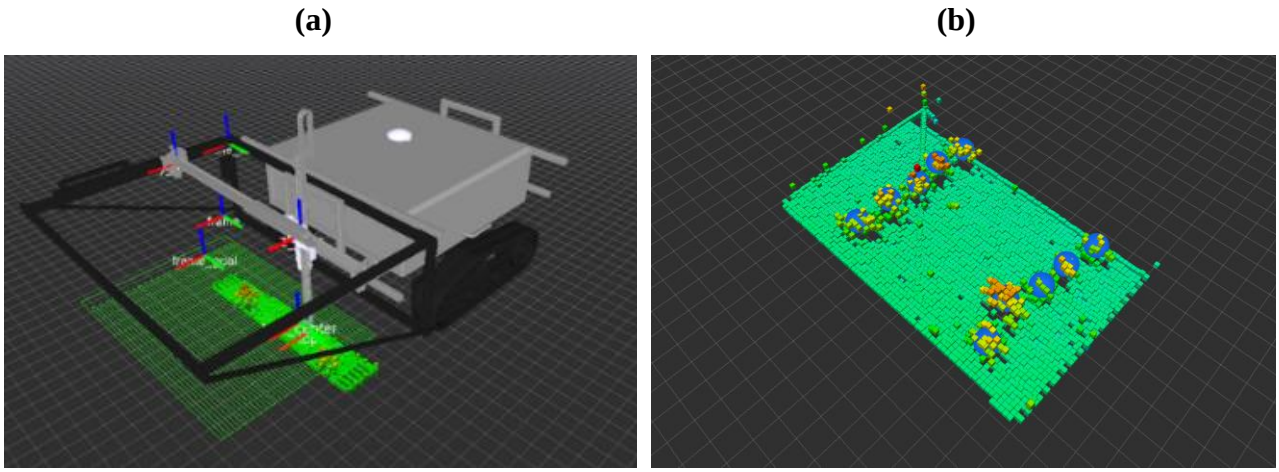

Figure 2: Visualization of the robot system (a) and the acquired point cloud in ROS-rviz, with the estimated plant positions by the software algorithm (blue dots) (b).

3.2.5 Point cloud assembling and processing

For each value received from the sensor, one point of the point cloud was estimated. The programmed software used different coordinate systems for each axis of the frame and was setting one coordinate system at the sensor position. The position of each axis coordinate system was estimated by the amount of steps sent to the stepper motors from the motor controllers. To estimate then the precise position of a received sensor value, the transform between the sensor coordinate system and the reference coordinate system was solved. This was possible, as the time stamp between all used controllers and sensors was synchronized. For the precise position estimation, the values were linearly interpolated. This procedure allowed to create a spatial 3D point cloud out of the one dimensional measurement of the sensor. To get rid of double measurements, all points in a grid with an edge length of 2 cm were filtered and summarized to one point. This led to a 3D image with a pixel size of 2x2 cm.

This 3D image was then processed to obtain the single plant positions in the considered area. First all points belonging to the ground plane were removed, using a basic RANSAC plane algorithm (Fischler and Bolles 1981). The defined parameters were a maximum of 1000 iterations, sigma of 0.005 m and

a distance threshold of 0.045 m. To get rid of noise, the remaining points were filtered, using a radius outlier filter (PCL 1.7.0, RadiusOutlierRemoval class). Points were considered as noise, when inside a radius of 0.08 m around the point less than four additional points were found. For separating the resulting points, a k-d-tree clustering was used (PCL 1.7.0, EuclideanClusterExtraction class) assuming that the plants were separated by a gap of minimum of 0.04 m. Of each single point cloud cluster, the 3D centroid c was evaluated and assumed as the resulting plant position. With N as number of points p in the cluster (Equation 4).

$$c = \frac{1}{N}\sum_{i=1}^{n} p_i \tag{6}$$

The resulting plant positions of the algorithm were compared with the ground truth to obtain the precision. The assembled point cloud and the detected plant positions could be visualized in real-time with the help of the ROS visualization tool "rviz" (see Figure 2b).

3.2.6 Precision spraying

The plant coordinates were extracted automatically from the point cloud, and subsequently could be used for the implementation of the automatic spraying algorithm. The coordinates were sent as goals to the frame motor driver board, and the programmed goal manager activated the pump as soon as the goal was reached. Because of the linear shift between sonar sensor and pump exit point, a static translation was applied to the plant positions with 3 cm *x*-axis and 4 cm *y*-axis. Two spraying configurations were applied for the comparison of the saved liquid. The first method was a continuous application along the plant rows (conventional spraying). The second method applied the liquid spray at the estimated plant position (selective spraying). The pump was turned on, for the first method, when the first plant position was reached in a distance range of +/- 2 cm, while for the second, the pump was turned on when the estimated plant position was reached with a precision of +/- 1 mm. This procedure was performed with three replicates, in which the amount of liquid applied in each test was quantified with small canisters with a diameter of 35 mm, placed under the application points. In total 17 canisters were placed in a row, covering a distance of 0.56 m. For both methods 42 ml were used. At the conventional method, the amount of water was applied over the whole line, while at the selective method the frame was moving to the plant poses and was applying the water (14 ml per plant). The three plant poses were assumed to be at the *x*-axis positions 0.035 m, 0.175 m and 0.315 m, separated by a spacing of 14 cm.

3.3 Results and discussion

After the calibration of the TCP with the help of the total station, the measured positions of the frame showed results of +/- 1 mm deviation between the software estimated positions and the total station.

This precision was achievable for two relative points. The system was able to recalibrate itself by driving once again to the end stop position at the edges of the working range. This allowed relocating the same position with the same precision inside the working range of the frame, as long as the mobile robot system was not moving. As soon as the system was starting and stopping for several times without reaching an end stop, the position error accumulated. This was caused by the mechanical inertia while stopping the system, as a consequence generating a slight slipping of the axis. For testing the limits of the system, the 7 points were approached, without returning to the next end stop. The details of the performed test points are described in the following Table 1. Because of the accumulation of the errors, the precision for the *y*-axis got an absolute precision deviation of 4 mm. The highest measurement error was 0.32%, compared to the distance. It could be assumed, that the system is able to collect spatial referenced sensor data in the millimetre range. This level of precision was obtained thanks to the controllable discrete angular displacement engines, whose high resolution and precise steps played a significant role in the overall accuracy of the positioning system.

Table 1: List of the performed calibration points with results.

No.	**Total Station measurement [m]**		**Software position [m]**	
	x	**y**	**x**	**y**
1	0.000	0.000	0.000	0.000
2	0.000	1.001	0.000	1.000
3	0.001	1.459	0.000	1.460
4	0.499	0.496	0.500	0.500
5	0.500	1.456	0.500	1.460
6	1.000	0.005	1.000	0.000
7	1.000	1.455	1.000	1.460

Regarding the calibration of the ultrasonic sensor, the height detected by the ultrasonic sensor and vertical movement of the TCP was compared. It indicated a variable precision of the system depending on the height of the sensor. This precision decreased as the distance was reduced to the measured object, with a RMSE of 5.5 mm for 10 cm, going down to 0.2 mm at a height above 30 cm (see Table 2). The maximum standard deviation was 4.1 mm and the minimal at 30 cm of 0.2 mm. For this reason, the detection of plants (with an average height of 20 cm) was performed at a height of 40 cm from the ground, thus avoiding an increase in measurement error.

For assessing the accuracy of a theoretical stereo vision system as a possible replacement for the sonar sensor, a baseline of 30 cm, a focal length of 10 mm and a camera pixel size of 5 µm as described by

Pajares et al. were assumed. When solving the Equations 4 and 5 with these parameters and a variable object distance, the theoretical depth accuracy could be described with the following graph (see Figure 3). To compare the values, the absolute values were also included in the following Table 2 for the estimated height poses of 10, 20 and 30 cm.

Table 2: Calibration results of the sonar sensor compared with the accuracy of a theoretical stereo camera.

Estimated pose [m]	**Measured sonar value [m]**			**Stereo camera [m]**
	$\bar{d}$	std_{dev}	**RMSE**	**accuracy**
0.1000	0.1037	0.0041	0.0055	0.00011
0.2000	0.2037	0.0019	0.0041	0.00021
0.3000	0.3001	0.0002	0.0002	0.00038

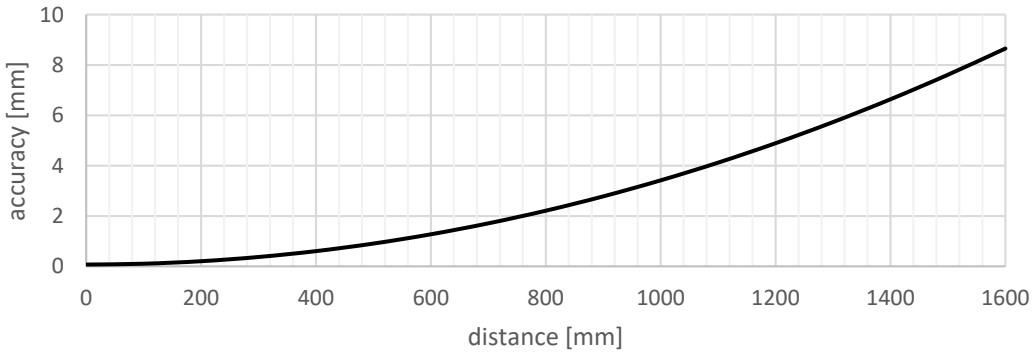

Figure 3: Development of the accuracy of a stereo vision system, with the assumed parameters B=300 mm, f=10 mm and p=5 µm (see Equations 4, 5).

It is visible, that a 3D visual system accuracy is highly dependent to the object distance. Therefore, it could be seen, that the sonar sensor is less precise than the stereo system at short distances (10 cm), but perform already better when the object is placed at 30 cm distance. This means, that the advantage of a stereo camera is not laying at the depth accuracy, compared to a sonar sensor, but just by the spatial resolution. Instead of mounting the camera at the TCP, the stereo camera could also be mounted in a fixed position to monitor the complete workspace of 1.3 x 0.9 m at once. By using a typical camera angle of view of 50 degrees, this would lead to a necessary mounting height of at least 1.39 m, what would guide to an accuracy of 6.58 mm.

This would increase the error compared to the used low-cost sonar system. As the accuracy of stereo cameras are effected by sunlight, light reflections, lack of texture or light (night) and the used

matching algorithm (Hirschmüller 2008), this accuracy could get worse under real field conditions, what would not be the case with sonar sensors.

To increase the accuracy of the system and to minimize the influence of the accumulation of path errors (see Table 1), the sonar scans of the 3D frame were separated in 10 small segments, of 10 cm x 140 cm. After each segment, the frame was driven to one end stop. The total time for each scan was 10 minutes. The unfiltered representation of the experimental area contained a total of 3801, 4383 and 4469 points for the 3 performed scans. After filtering the scan area to one point for each grid of 2x2 cm, the resulting points for the 3 scans were 2931, 2906 and 2907 points. The results of the point clouds are shown in Figure 4. Only points in the *x*-axis and *y*-axis greater than 0.1 m were considered. The 3D point representation is shown on the left side. The right picture depicts the same data as surface reconstruction. The value of every grid was extrapolated by smoothing the values with the given point cloud dataset, using a Delaunay triangulation principle with the use of the Matlab function gridfit (Errico 2016). The showed graphs were solved by setting a grid size of 1 cm over the considered area, defining a smoothing factor of 2 for the dataset.

Out of the generated 3D point clouds, it is possible to observe the plant positions represented by the higher altitudes. Around the centre of the plant, the height was forming the maximum, with a balloon like shape. When reconstructing a surface out of the 3D point cloud (see Figure 4), the peaks are strongly correlated with the real plant position, that leads to the assumption that the peaks are also the real plant centre positions. However, the reflections of leaves caused points to overlap, making the clusters of the single plants harder to separate. In addition, some plants were detected better than others, where the morphology of each plant could be a cause. The difference in height between plants was in some cases more than 10 cm. However, there was always a detectable peak for every single plant.

The applied algorithm for the single plant detection needed a clear spacing between the points to cluster the plants. In this described case, the distance between the plants was set to 0.14 m. This distance was sufficient to estimate a clear spacing between the plants, what lead to an automatic performed plant detection rate of 100% in all performed scans.

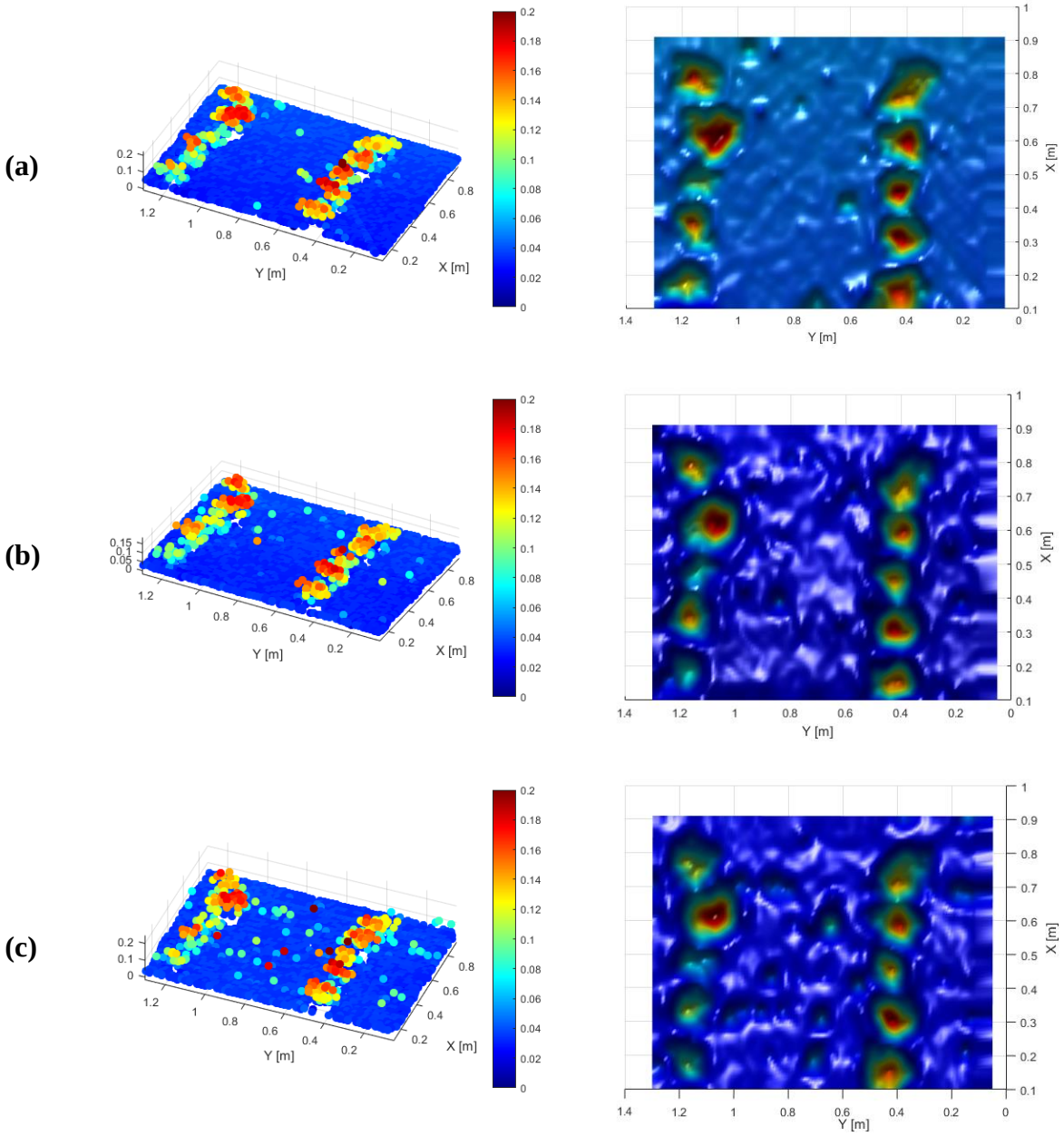

Figure 4: Visualization of the received 3D point clouds and surface reconstructions for the three scans (a), (b) and (c). All readings are in [m].

Obtaining the coordinates of each plant was done in real time by the described algorithm. The 3D centroid of each cluster was assumed as plant position. However, the actual centre of the plant varied depending on the orientation of the leaves, their density, size and height. This caused that the centre of the cluster did not match exactly the real plant origin, but represented the centre of the plant's leaf area, what brings advantages for autonomous spraying, as the leaves are normally the areas of interest. Figure 5 shows a comparison between the pendulums measured coordinates of the plant centres and the coordinates obtained from the point clouds. The average distance of the plants to the real position was 2.4 cm in the (x, y) coordinates with a RMSE of 2.7 cm. This gives the system a good precision

considering the influence of the leaf area on the determination of the location of the plants and the systems TCP movement.

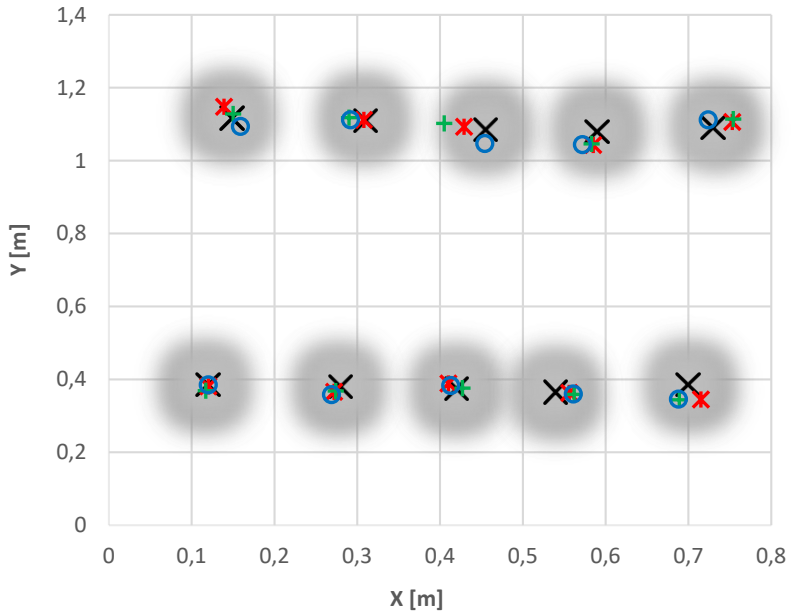

Figure 5: Estimated plant poses in *x* and *y* with the ground truth (x) and the results for the three scans (*, o, +).

Therefore, it was assumed, that the system is capable of obtaining the location of plants without prior measurements of their actual location, when the following conditions are fulfilled:

- The ground surface must have a planar shape, that it can be detected with a RANSAC plane-fitting algorithm. The soil irregularities must be smaller than the height of the plants.
- The plant leaves should not cover the area between the plants, so that height differences at the plant gaps are detectable.
- No other sonar sources are interfering with the sensor system.
- The plant height is smaller than approximately 0.5 m, since the TCP or the mobile robot could touch the leaves, producing wrong measurements or damaging the plants.

As the sensor system is not affected by sunlight, it could be assumed that the sensor detection will also perform well under outdoor field conditions, when the monitored environment is similar to the tested experiment, with a planar like soil surface and spatially separated plants. One advantage of the used sonar sensor is, compared to other sensor types, that inside the measurement cone, the highest value is reflected back. Therefore, since the closest point is reflected, a low image resolution can be

generated without missing the high peaks of a region. However, the spatial resolution of the sensor hardly allows the detection of small plant details like single leaves. On the other hand, with a LiDAR system, it would be easy to miss the highest spot of a plant, when the measurement resolution is too low. Consequently, sonar sensors could help to create robust 3D imaging methods for precision spraying and plant height estimation.

However, when using the sensor system for outdoor field testing, the main influence factor on the results is assumed to be the change of the surface structure. Especially the roughness of the soil could cause measurement errors. As sonar sensors are not able to detect flat surfaces within higher inclination to the sensor, this could cause measurement errors in rough terrain. Same problem could occur when plants form sharp edges. Wind could change the surface while scanning, leading to moving plant leaves which could result in wrong aligned plant positions or measurement noise. The measurement quality will not change with sonar sensors under changing weather conditions in general, like it is typical with vision based sensor systems, so that shading of the sensors must not be considered.

As previously mentioned, it was found that planting distances significantly affect the formation of clusters. Close proximity between plants produced that the corresponding plant points could not be separated and two plants were considered as one. In the described experiment, the plant spacing was set to 14 cm, which was adequate to identify all the plants without causing overlaps or unifications of clusters. As long as the plants are showing a height difference between their gaps and do not completely overlap, the plants can then be detected and clustered.

For comparing selective to conventional spraying, in total six tests were performed. The measured values were averaged for every canister with the spacing of 0.035 m and the results were compared. The results are shown in the following Figure 6.

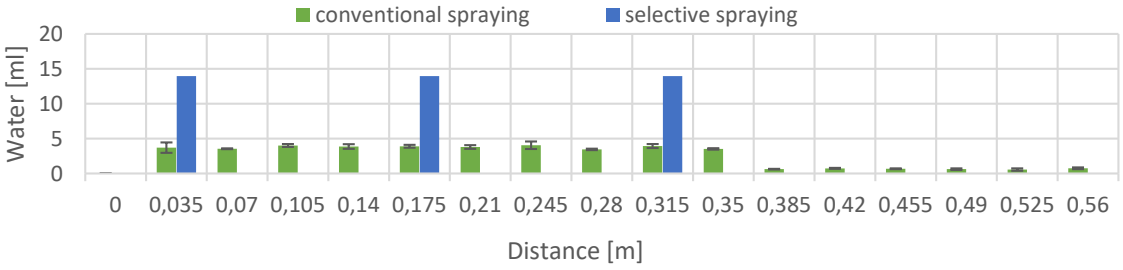

Figure 6: Average distribution of the conventional spraying compared to the selective spraying with two standard deviations.

The standard deviations of the measured liquid content of the performed experiments were between 0.754 ml and 0.042 ml. The selective method applied all water spray precisely on the single plants,

without any loss in the canisters of 3.5 cm diameter. The results of the continuous spraying showed a constant distribution over the measured area. The water distribution, showed a continuous application of approximately 4 ml in the first 0.35 m of the path. However, the value was decreased to values close to 0.8 ml in the rest of the way. This phenomenon was generated because the pump did not provide a continuous pressure, or a shut off valve to keep the liquid inside the hose. In the selective application, no liquid was lost, as the TCP was not moving while the pump was working.

When comparing the amount of water applied at the plant pose canister, the selective spraying method applied in average 3.6 times more water to the plants than the conventional method. With the conventional method, a total of 151.2 ml must be applied so that the plant obtains the same amount of water. This would represent a saving of 109.2 ml or 72% of the conventional spraying method.

Similar values were reported by (Gil et al. 2007; Gonzalez-de-Soto et al. 2016; Maghsoudi et al. 2015; Oberti et al. 2015), who gave a pesticide reduction of 34.5%, 58%, 64% and 66% respectively, when compared to a conventional homogeneous spraying. The result of this tests confirmed that an inexpensive precision system could confer significant reductions in the use of herbicides.

The main disadvantage of the proposed method is the low in speed. As the actual scanning time of the workspace needs around 10 minutes for 10 plants, a performance of 1 minute per plant would lead at a density of 9 plants per square meter (Reckleben 2011) to a performance of approximately 166 hours per hectare.

This means, that for real-time applications, scanning with a sonar sensor at the TCP is not promising. Much faster would be a fixed 2D array of sonar sensors spaced within a few centimetres, using the robot movement to create a spatial 3D image. When this is combined with a movable spray nozzle per row, the speed could be adjusted depending on the needed sonar sensor grid size. With a grid size of 2 cm and the used 10 Hz of the sonar sensor a driving speed of sonar sensor, the system could cover 1.3 m² in 5 seconds, what leads to approximately 10 hours per hectare. With a robotic system working 24 hours a day, this could already have a good cost benefit ratio. The 3D image obtained by the system could even be used to spray according to the measured canopy volume of the 3D image to apply the liquid more effectively. In addition, the use of a magnet valve would help to apply the liquid more precisely to the plants.

Future work should also include research in real outdoor conditions, to define the minimal necessary grid size of the sensor readings in order to be fast but without missing useful information. To understand the advantages of different sensor systems, it could also help to compare the differences between the use of a LiDAR sensor and a sonar sensor, comparing detection rate, precision and possible speed for detecting single plants. For more robust plant detection, other clustering algorithms like a k-means or graph-cut based principles could be investigated.

3.4 Conclusions

An autonomous selective spraying robot was developed and tested at the University of Hohenheim. This system was composed by a mobile robot, with a mounted frame with three degrees of freedom. An ultrasonic sensor was mounted at the frame TCP to scan the surface with the aim of determining the height of the objects inside the working range. The precision of the horizontal movement of the TCP was determined with +/- 1 mm. The vertical precision depended on the height measurement of the ultrasonic sensor. The ultrasonic sensor RMSE was between 5.5 mm to 0.2 mm, depending on the distances between the objects and the sensor. The 3D position of the TCP was fused with the sonar sensor value to generate a 3D point cloud.

Three scans were performed, finding on average a total of 4127 points per scan. This point cloud allowed differing plants present on the scan area. To acquire the exact plant positions, the ground points were removed by a RANSAC plane-fitting algorithm. The resulting points were separated with a Euclidian distance clustering. The centroid of each cluster represented the estimated plant position. The estimation of the plant coordinates demonstrated an RMSE of 2.7 cm, allowing a real-time location and spraying of the position by an attached control valve at the TCP. This would be sufficient to detect single plant structures and allow single plant treatment. The comparative tests between the selective spraying method and the conventional spraying in this investigation showed that with the reduction of the amount of the spraying operation in places where it is not necessary, it could provide savings of about 72%.

The sonar sensor allows variable grid sizes and is not affected by sunlight, leading to the assumption that it could perform well in outdoor conditions. From the economic and environmental point of view, the designed system could provide ideas for low-cost precision spraying by reducing soil and groundwater contamination. Therefore, it could decrease the use of herbicides, workers and working time.

Acknowledgement

The project was conducted at the Max-Eyth Endowed Chair (Instrumentation & Test Engineering) at Hohenheim University (Stuttgart, Germany), which is partly grant funded by the Deutsche Landwirtschafts-Gesellschaft e.V. (DLG).

Author contributions

The experiment planning and the programming of the software was done by the first and the third author. The writing was performed by the first and the second author. The experiments were conducted by the first, second and third author. First, second and third author analyzed and processed

the data. The mechanical development of the frame and design was performed by the fourth and fifth author. The whole experiment was supervised by the sixth author.

Conflicts of Interest: The authors declare no conflict of interest.

References

Alves, J., Maria, J., Ferreira, S., Talamini, V., Fátima, J. De, Medianeira, T., et al. (2016). Determination of pesticides in coconut (Cocos nucifera Linn.) water and pulp using modified QuEChERS and LC–MS/MS. Food Chemistry, 213, 616–624. doi:10.1016/j.foodchem.2016.06.114

Andújar, D., Dorado, J., Fernández-Quintanilla, C., & Ribeiro, A. (2016). An approach to the use of depth cameras for weed volume estimation. Sensors (Switzerland), 16(7), 1–11. doi:10.3390/s16070972

Back, W. C., van Henten, E. J., & Edan, Y. (2014). Harvesting Robots for High-value Crops: State-of-the-art Review and Challenges Ahead. Journal of Field Robotics, 31(6), 888–911. doi:10.1002/rob

Backman, J., Oksanen, T., & Visala, A. (2012). Navigation system for agricultural machines: Nonlinear Model Predictive path tracking. Computers and Electronics in Agriculture, 82, 32–43. doi:10.1016/j.compag.2011.12.009

Berge, T. W., Goldberg, S., Kaspersen, K., & Netland, J. (2012). Towards machine vision based site-specific weed management in cereals. Computers and Electronics in Agriculture, 81, 79–86. doi:10.1016/j.compag.2011.11.004

Bietresato, M., Carabin, G., Vidoni, R., Gasparetto, A., & Mazzetto, F. (2016). Evaluation of a LiDAR-based 3D-stereoscopic vision system for crop-monitoring applications. Computers and Electronics in Agriculture, 124, 1–13. doi:10.1016/j.compag.2016.03.017

Blackmore, B. S., Griepentrog, H. W., & Fountas, S. (2006). Autonomous Systems for European Agriculture. In Automation Technology for Off-Road Equipment. Bonn.

Chang, J., Li, H., Hou, T., & Li, F. (2016). Paper-based fluorescent sensor for rapid naked-eye detection of acetylcholinesterase activity and organophosphorus pesticides with high sensitivity and selectivity. Biosensors and Bioelectronic, 86, 971–977. doi:10.1016/j.bios.2016.07.022

Dornbusch, T., Wernecke, P., & Diepenbrock, W. (2007). A method to extract morphological traits of plant organs from 3D point clouds as a database for an architectural plant model. Ecological Modelling, 200(1–2), 119–129. doi:10.1016/j.ecolmodel.2006.07.028

Doruchowski, G., & Holownicki, R. (2000). Environmentally friendly spray techniques for tree crops. Crop Protection, 19, 617–622.

Doulia, D. S., Anagnos, E. K., Liapis, K. S., & Klimentzos, D. A. (2016). Removal of pesticides from white and red wines by microfiltration. Journal of Hazardous Materials, 317, 135–146. doi:10.1016/j.jhazmat.2016.05.054

El-Gawad, H. A. (2016). Validation method of organochlorine pesticides residues in water using gas chromatography – quadruple mass. Water Science, 30(2), 96–107. doi:10.1016/j.wsj.2016.10.001

Errico, J. D. (n.d.). Matlab gridfit function. https://de.mathworks.com/matlabcentral/fileexchange/8998-surface-fitting-using-gridfit. Accessed 23 December 2016

Fischler, M. A., & Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM, 24, 381–395. doi:10.1145/358669.358692

Gaillard, J., Thomas, M., Iuretig, A., Pallez, C., Feidt, C., Dauchy, X., & Banas, D. (2016). Barrage fishponds: Reduction of pesticide concentration peaks and associated risk of adverse ecological effects in headwater streams. Journal of Environmental Management, 169, 261–271. doi:10.1016/j.jenvman.2015.12.035

Garrido, M., Paraforos, D., Reiser, D., Vázquez Arellano, M., Griepentrog, H., & Valero, C. (2015). 3D Maize Plant Reconstruction Based on Georeferenced Overlapping LiDAR Point Clouds. Remote Sensing, 7(12), 17077–17096. doi:10.3390/rs71215870

Gil, E., Escolà, A., Rosell, J. R., Planas, S., & Val, L. (2007). Variable rate application of Plant Protection Products in vineyard using ultrasonic sensors. Crop Protection, 26(8), 1287–1297. doi:10.1016/j.cropro.2006.11.003

Giles, D. K., Delwiche, M. J., & Dodd, R. B. (1987). Control of orchard spraying based on electronic sensing of target characteristics. Transactions of the ASABE, 30, 1624–1630. doi:10.13031/2013.30614

Gonzalez-de-Soto, M., Emmi, L., Perez-Ruiz, M., Aguera, J., & Gonzalez-de-Santos, P. (2016). Autonomous systems for precise spraying-Evaluation of a robotised patch sprayer. Biosystems Engineering, 6. doi:10.1016/j.biosystemseng.2015.12.018

Heap, I. (2014). Global perspective of herbicide-resistant weeds. Pest Management Science, 70(9), 1306–1315. doi:10.1002/ps.3696

Hirschmüller, H. (2008). Stereo processing by semiglobal matching and mutual information. IEEE Transactions on Pattern Analysis and Machine Intelligence, 30(2), 328–341. doi:10.1109/TPAMI.2007.1166

Jiang, Y., Li, C., & Paterson, A. H. (2016). High throughput phenotyping of cotton plant height using depth images under field conditions. Computers and Electronics in Agriculture, 130, 57–68. doi:10.1016/j.compag.2016.09.017

Kira, O., Linker, R., & Dubowski, Y. (2016). Estimating drift of airborne pesticides during orchard spraying using active Open Path FTIR. Atmospheric Environment, 142, 264–270. doi:10.1016/j.atmosenv.2016.07.056

Kunz, C., Sturm, D. J., Peteinatos, G. G., & Gerhards, R. (2016). Weed suppression of Living Mulch in Sugar Beets. Gesunde Pflanzen, 1–10. doi:10.1007/s10343-016-0370-8

Kusumam, K., Kranjík, T., Pearson, S., Cielniak, G., & Duckett, T. (2016). Can You Pick a Broccoli ? 3D-Vision Based Detection and Localisation of Broccoli Heads in the Field. In IEEE International Conference on Intelligent Robots and Systems (pp. 1–6). doi:10.1109/IROS.2016.7759121

Lee, W. S., Slaughter, D. C., & Giles, D. K. (1999). Robotic weed control system for tomatoes. Precision Agriculture, 1, 95–113.

Maghsoudi, H., Minaei, S., Ghobadian, B., & Masoudi, H. (2015). Ultrasonic sensing of pistachio canopy for low-volume precision spraying. Computers and Electronics in Agriculture, 112, 149–160. doi:10.1016/j.compag.2014.12.015

Oberti, R., Marchi, M., Tirelli, P., Calcante, A., Iriti, M., Tona, E., et al. (2015). Selective spraying of grapevines for disease control using a modular agricultural robot. Biosystems Engineering, 6. doi:10.1016/j.biosystemseng.2015.12.004

Oerke, E.-C. (2006). Crop losses to pests. The Journal of Agricultural Science, 144(1), 31. doi:10.1017/S0021859605005708

Pajares, G., García-Santillán, I., Campos, Y., Montalvo, M., Guerrero, J., Emmi, L., et al. (2016). Machine-Vision Systems Selection for Agricultural Vehicles: A Guide. Journal of Imaging, 2(4), 34. doi:10.3390/jimaging2040034

Peteinatos, G. G., Weis, M., Andújar, D., Rueda Ayala, V., & Gerhards, R. (2014). Potential use of ground-based sensor technologies for weed detection. Pest Management Science, 70(October), 190–199. doi:10.1002/ps.3677

Reckleben, Y. (2011). Cultivation of maize - which sowing row distance is needed? Landtechnik, 66(5), 370–372.

Reiser, D., Izard, M. G., Arellano, M. V., Griepentrog, H. W., & Paraforos, D. S. (2016). Crop Row Detection in Maize for Developing Navigation Algorithms under Changing Plant Growth Stages. In Advances in Intelligent Systems and Computing (Vol. 417, pp. 371–382). Lisbon: Springer. doi:10.1007/978-3-319-27146-0_29

Reiser, D., Paraforos, D. S., Khan, M. T., Griepentrog, H. W., & Vázquez Arellano, M. (2016). Autonomous field navigation, data acquisition and node location in wireless sensor networks. Precision Agriculture, 1. doi:10.1007/s11119-016-9477-2

Rovira-Más, F., Wang, Q., & Zhang, Q. (2010). Design parameters for adjusting the visual field of binocular stereo cameras. Biosystems Engineering, 105(1), 59–70. doi:10.1016/j.biosystemseng.2009.09.013

Rusu, R. B., & Cousins, S. (2011). 3D is here: point cloud library. IEEE International Conference on Robotics and Automation, 1–4. doi:10.1109/ICRA.2011.5980567

Slaughter, D. C., Giles, D. K., & Downey, D. (2008). Autonomous robotic weed control systems: A review. Computers and Electronics in Agriculture, 61(1), 63–78. doi:10.1016/j.compag.2007.05.008

Solanelles, F., Escolà, A., Planas, S., Rosell, J. R., Camp, F., & Gràcia, F. (2006). An Electronic Control System for Pesticide Application Proportional to the Canopy Width of Tree Crops. Biosystems Engineering, 95(4), 473–481. doi:10.1016/j.biosystemseng.2006.08.004

Swain, K. C., Zaman, Q. U. Z., Schumann, A. W., & Percival, D. C. (2009). Detecting weed and bare-spot in wild blueberry using ultrasonic sensor technology. American Society of Agricultural and Biological Engineers Annual International Meeting 2009, 8(9), 5412–5419. doi:doi:10.13031/2013.27281

Tumbo, S. D., Salyani, M., Whitney, J. D., Wheaton, T. A., & Miller, W. M. (2002). Investigation of Laser and Ultrasonic Ranging Sensors for Measurements of Citrus Canopy Volume. Applied Engineering in Agriculture, 18(3), 367–372.

Vázquez-Arellano, M., Griepentrog, H. W., Reiser, D., & Paraforos, D. S. (2016). 3-D Imaging Systems for Agricultural Applications - A Review. Sensors, 16(618), 24. doi:10.3390/s16050618

Walklate, P. J., Cross, J. V, Richardson, G. M., & Baker, D. E. (2006). Optimising the adjustment of label-recommended dose rate for orchard spraying, 1–7. doi:10.1016/j.cropro.2006.02.011

Weiss, U., & Biber, P. (2011). Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. Robotics and Autonomous Systems, 59(5), 265–273. doi:10.1016/j.robot.2011.02.011

Woods, J., & Christian, J. (2016). Glidar: An OpenGL-based, Real-Time, and Open Source 3D Sensor Simulator for Testing Computer Vision Algorithms. Journal of Imaging, 2(1), 5. doi:10.3390/jimaging2010005

Zaman, Q. U., Schumann, A. W., & Miller, W. M. (2005). Variable rate nitrogen application in Florida citrus based on ultrasonically-sensed tree size. Applied Engineering in Agriculture, 21(3), 331–335.

Zlot, R., & Bosse, M. (2014). Efficient Large-scale Three-dimensional Mobile Mapping for Underground Mines. Journal of Field Robotics, 31(5), 758–779. doi:10.1002/rob.21504

CHAPTER 4

Paper C

Crop Row Detection in Maize for Developing Navigation Algorithms under Changing Plant Growth Stages[3]

David Reiser, Miguel Garrido-Izard, Manuel Vázquez-Arellano, Hans W. Griepentrog and Dimitris S. Paraforos

Abstract

To develop robust algorithms for agricultural navigation, different growth stages of the plants have to be considered. For fast validation and repeatable testing of algorithms, a dataset was recorded by a 4 wheeled robot, equipped with a frame of different sensors and was guided through maize rows. The robot position was simultaneously tracked by a total station, to get precise reference of the sensor data. The plant position and parameters were measured for comparing the sensor values. A horizontal laser scanner and corresponding total station data was recorded for 7 times over a period of 6 weeks. It was used to check the performance of a common RANSAC row algorithm. Results showed the best heading detection at a mean growth height of 0.268 m.

Keywords: ground-truth, reference, algorithms, RANSAC, total station, LIDAR, plant position, growth status, row navigation

4.1 Introduction

Autonomous robots can have a key role in increasing sustainability and resource efficiency in food production for future world population (English et al. 2014). Therefore, the navigation must be planed precisely and be robust enough to deal with the changing conditions on a field. But this requires, that the machines know where the crop plants are and that they don´t get destroyed by the vehicle. As

[3] The publication of Chapter 4 is done with the consent of the Springer International Publishing. The original publication was in *Advances in Robotics, Volume 1 Robot 2015: Second Iberian Robotics Conference*. It can be found under the following link: http://doi.org/10.1007/978-3-319-27146-0_29

most of the current crops are planted in row structures, detecting these rows is one of the basic needs for the autonomous navigation of robots in semi-structured agricultural environments. Many researches had been conducted on detecting this line structures by camera images ((Marchant and Brivot 1995),(G. Jiang et al. 2010b)), light detection and ranging (LIDAR) laser scanner data ((Hansen et al. 2010),(Barawid et al. 2007),(S. A. Hiremath et al. 2014)), or other types of sensors. Nevertheless, precise line detection, relying on noisy sensor data, is still a challenging task for a computer algorithm due to the inherent uncertainty in the environment (S. A. Hiremath et al. 2014). Humans can detect objects and shapes because of experience rather than a formal mathematical definition, like a computer algorithm does (Papari and Petkov 2011). The environment has a countless number of variables influencing the sensors, making it hard to get the right information out of the values (Russell and Norvig 1995). Aside from that, plants on the field are changing their shape rapidly, making object recognition even more challenging. First the plants are growing and, second, the conditions are changing. Therefore, there is a necessity for calibration of the algorithms before the robot is able to perform the task autonomously (S. A. Hiremath et al. 2014).

In addition, weather and lighting conditions can already produce big changes in the results. This is especially problematic for image analysis, where alternate and discontinuous luminance usual affects the outcome (Papari and Petkov 2011).

To deal with these uncertainties, researchers have used simulated datasets (Weiss and Biber 2011), artificial plants ((Weiss and Biber 2011), (Bochtis et al. 2015)) or recorded datasets ((English et al. 2014), (G. Jiang et al. 2010b), (S. A. Hiremath et al. 2014)) to evaluate their algorithms. Since a simulation is always an approximate model of the environment, it will never cover all possibilities (Russell and Norvig 1995). When recording data, the question is of how to refer to the algorithm performance. One option is to set the crop row manually (S. A. Hiremath et al. 2014). The precise sensor value recording of the same plants over different growth stages, can be a good way for the later evaluation of navigation algorithms. In order to understand how algorithms behave under changing conditions in a field, it is necessary to know the pictured objects and how the sensors react on them. Therefore, it is important to know the correct plant position and parameters. In order to achieve that, the plant parameters must be mapped and referenced in every new test.

The aim of this paper is to show how algorithm analysing could be improved using precise referenced sensor data, especially when the same plants can be investigated with the same sensors over different growth stages. For that purpose, the data set of a horizontal LIDAR is used. With the help of a highly accurate total station, all sensor data sets can be converted into the same reference frame, in order to obtain comparable results. This approach is tested by the performance evaluation of a common random sample consensus (RANSAC) line fitting algorithm (Fischler and Bolles 1981). The

RANSAC algorithm has the advantage of being fast and robust against outliers, resulting in advanced performance when dealing with noisy sensor data. Therefore, the performance at different growth stages can be precisely evaluated, by using the same reference.

4.2 Materials and methods

4.2.1 Hardware and sensors

A small 4-wheel autonomous robot with differential steering was used to move the sensors with manual control through the crop rows (see Figure 1). The size of the robot platform was 500 x 600 x 1100 mm. The weight of the robot is 50 kg and it is equipped with four motors with a total power of 200 W. Maximum driving speed is 0.8 m/s and the maximum static motor torque is 4 x 2,9 Nm. The robot system is equipped with wheel encoders, a VN-100 Inertial Measurement Unit (IMU) (VectorNav, Dallas, USA) and a LMS111 2D-LIDAR laser scanner (SICK, Waldkirch, Germany). The laser scanner was mounted horizontally at the front of the robot at a height of 0.2 m above the ground level. All other mounted sensors had not been used in this paper.

For evaluating the precise position of the robot, the SPS930 Universal Total Station (Trimble, Sunnyvale, USA) was utilized. The total station was tracking a Trimble MT900 Machine Target Prism, which was mounted on top of the robot at a height of 1.07 m in order to guarantee always line of sight to the total station (see Figure 1).

The robot is controlled by an embedded computer, equipped with i3-Quadcore processor with 3.3 GHz, 4 GB RAM and SSD Hard drive. For energy supply, two 12V/48Ah batteries are providing an operating time of around 4-5 h, depending on the load torque, task and additional weight of equipment placed on the robot platform. The total station data was sent to a Yuma 2 Rugged Tablet Computer (Trimble, Sunnyvale, USA); this tablet is equipped with an Intel Atom CPU N2600 dual-core processor with 1.6 GHz, 4 GB RAM, SSD Hard drive, and a self-sufficient battery. Connectivity to the SPS930 total station is provided by an internal 100mW radio antenna at the 2.4 GHz (IEEE 802.11) range. The Yuma 2 Rugged Tablet Computer was connected to the robot computer via serial RS232 interface for continuous data exchange.

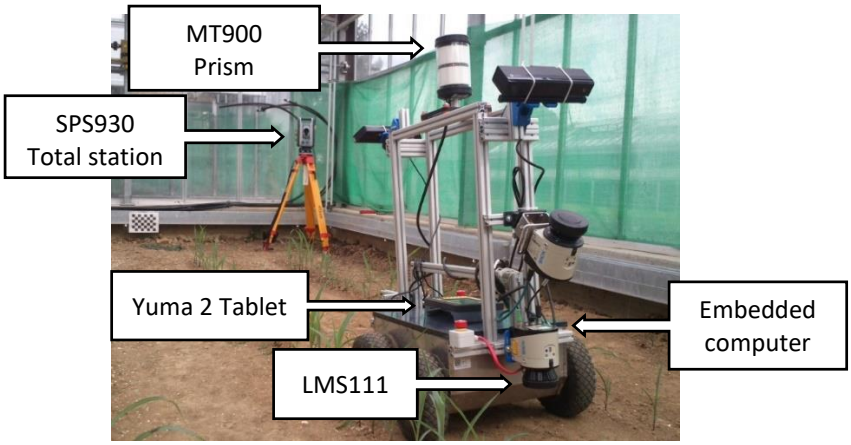

Figure 1: Robot platform, equipped with the sensors and the reference prism.

4.2.2 Software

The robot computer runs by Ubuntu 14.04 and use the Robot Operating System (ROS-Indigo) middleware for the data recording. The software components had been programmed in a combination of C++ and Python programming languages.

The Trimble Yuma 2 Tablet was running under Windows 7 Professional and executed the Trimble SCS900 Site Controller Software Version 3.4.0. The Trimble software includes an easy-to-use graphical interface for fast calibration and point measurement. It also has the option to export the actual prism position via serial RS232 interface. The tablet was placed on the robot and the serial output was directly used by the ROS system to refer the robot position to the total station coordinate frame. The prism position data was time stamped, according to the computer system time, together with the sensor data. The data flow diagram can be seen in Figure 2.

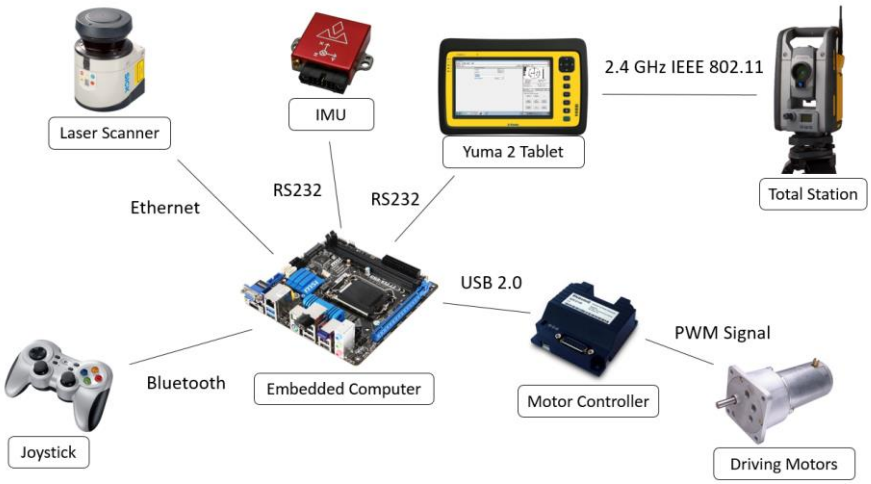

Figure 2: Data flow diagram of the robot sensor setup.

4.2.3 Referencing and data acquisition

To create a relative coordinate frame in the greenhouse, five fixed attachments for the MT1000 Prism had been placed on predefined positions. The accurate position of these five points could be located by just screwing the prism on the attachments for every new field test. The absolute point distances were stored during the first setup. Before every subsequent data acquisition, these points were measured once again by the total station in order to recalibrate the system to the first total station position with the original Cartesian coordinate frame. After every test, the inaccuracy in the static measurement was evaluated by reassessing each of these fixed points. The shift between the first reference points and the actual measurement was evaluated by the Trimble SPS software and was in all tests below 4 mm for all three dimensions. The total station was always placed almost at the same position inside the greenhouse, which lies around the zero point of the coordinate frame.

After plant emergence, the stem position was measured with the aid of pendulum hanging from a tripod; the MT1000 Prism was attached at the centre of the tripod. It was assumed that the centre of each plant stem's position will not change during the period of growth. Consequently, each of these points was used as the reference position of each individual plant.

Due to the robot rigid body frame that was carrying the sensors, a static transformation between the prism and the sensor positions was performed. The three-dimensional orientation of the robot in space was evaluated by the IMU, which was placed at the centre of the robot and on the same axis as the prism. As the orientation of the prism could not be evaluated by the total station, the position of the prism was fused with the IMU data to transform the laser scans to the greenhouse frame.

This procedure allowed to directly transform the recorded sensor data into the same coordinate frame, and even to assign them to single plant positions. In Figure 3a the test environment with the moving robot is presented and in Figure 3b the corresponding sensor value visualization of all attached sensors in the ROS environment is illustrated. The blue points correspond to the horizontal laser scanner data. The sensor files were separated at a size of 4 GB. The timestamp was according to the robot embedded computer system time, with a resolution of one millisecond. The LIDAR data was collected with an average of 25 Hz and a resolution of 0.5 degree. The total station updated the data with 15 Hz. The IMU data was transmitted with 40 Hz. Linear interpolation was used to fuse robot position and the sensor data before transforming it to the global coordinates of the greenhouse frame.

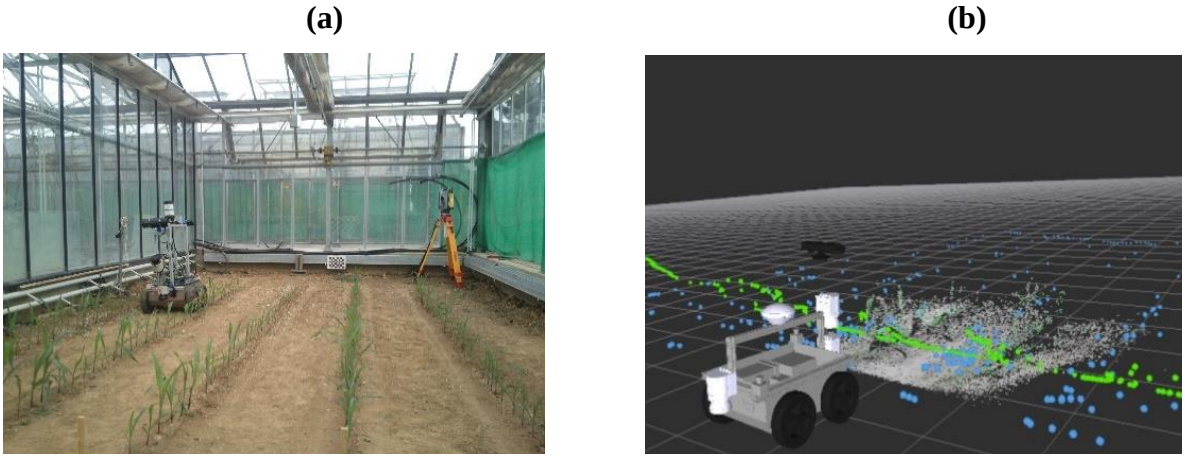

Figure 3: (a) The greenhouse environment and (b) a visualisation of the transformed sensor data in the ROS visualisation environment.

4.2.4 RANSAC algorithm

The RANSAC is a commonly used algorithm for evaluating plane parameters in noisy point cloud data (Choi et al. 2014), (Weiss et al. 2010). But also for row or line estimation in image analysis (English et al. 2014), and 2D-LIDAR data (J. Zhang et al. 2014), (Marden and Whitty 2014). To evaluate the general algorithm behaviour, the RANSAC was chosen due to its previously mentioned robustness against outliers. The implemented RANSAC algorithm is part of the Point Cloud Library (Rusu and Cousins 2011) and was integrated in the ROS environment, for direct analysis of the published scans. To get always precise reference of the extracted lines, the LIDAR data was first transformed to the robot body frame and then to the overall greenhouse frame.

As the distance between maize crop rows is 0.75 m, this parameter was used to filter roughly the row area with the known robot position with a rectangle. The resulting point cloud was separated to have for each crop row an individual point cloud. This was done by using the known robot position and robot direction. The RANSAC was then applied to each distinct point cloud. The maximum iteration limit was set to the input point number and the maximal distance range for the line to 0.5 m. These parameters were fixed for all performed line fittings.

4.3 Experiments

Five rows of maize were planted with a length of 5.2 m each. The row spacing was defined according to common agricultural practice to 0.75 m, with 41 plants per row. The maize was planted in a greenhouse to be independent of external weather conditions. The measured positions of the plants, total station and reference points can be found in Figure 4. After every data acquisition, the height, stem width and leaf numbers of each single plant had been measured. This was done manually with a measuring tape and a sliding calliper.

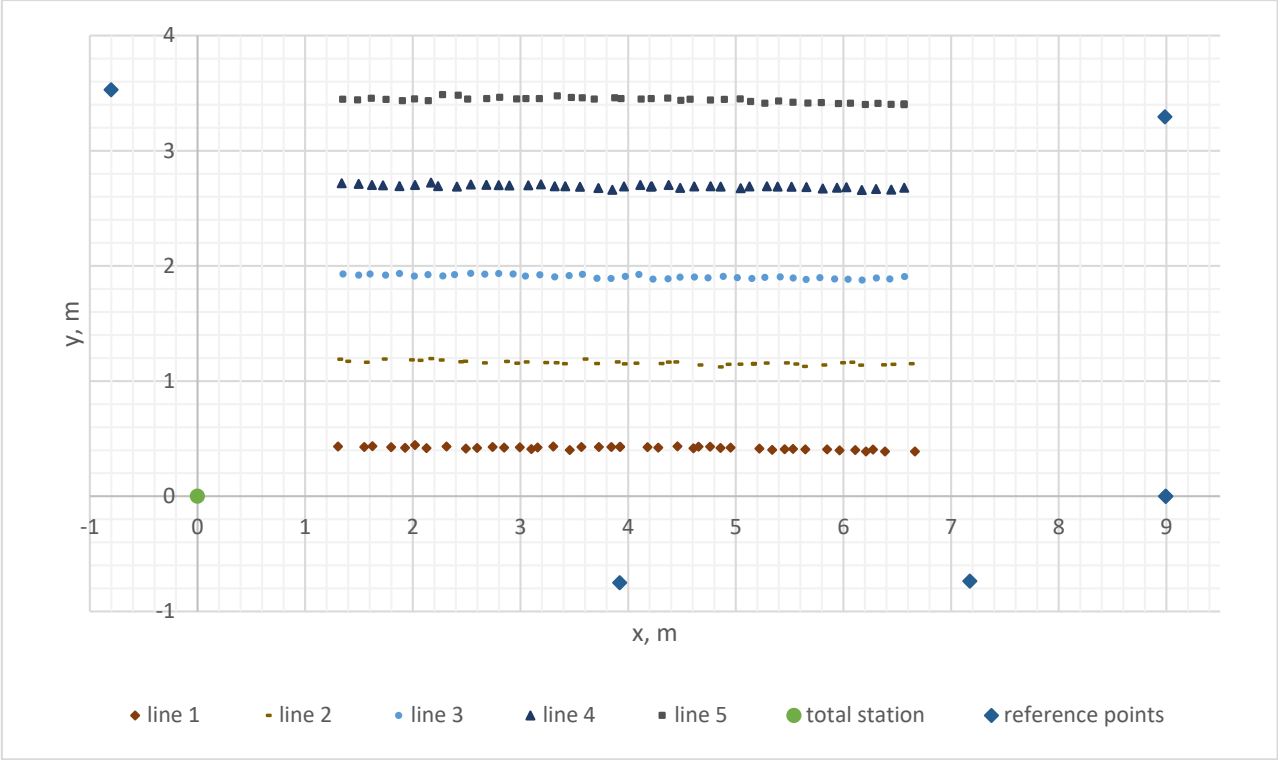

Figure 4: Manually measured plant positions, reference points and total station position.

The ideal line parameters were evaluated by the plant germination positions, measured with the total station. Because this line has no outliers, the least square algorithm can result in the most accurate line fitting. As reference, a 2D line in the XY-plane was estimated using the equation $f(x) = ax + b$. The residual r of every data point $P_i(x_i, y_i)$ can be described with:

$$r_i = f(x_i) - a * x_i - b \tag{1}$$

Using the least square estimation, the best line fit can be estimated as:

$$\min_{a\,b} \sum_{i=1}^{n} {r_i}^2 \tag{2}$$

This was adapted to the emerging points of the plants, resulting in the line parameters presented in Table 1.

As it can be seen in the intersection points, shown in Table 1, the row with the smallest angle difference between the lines is the path between crop row 2 and 3; here, the intersection point had the longest distance to the row centre. The best performance was expected from the most parallel lines for evaluating the row detection algorithm. So crop row line 2 and 3 were selected. The sensor data recording took place from 23.04.15 until 1.06.15 in Stuttgart Hohenheim. In total 7 tests were performed. In every test, the robot was driven by a remote joystick with a constant speed through the crop rows. The average speed was around 0.05 m/s in order to acquire a high data density. Both rows

were recorded twice, once for each travel direction. For all 7 test days, each travel direction was evaluated, resulting in a total of 14 recorded and analysed datasets. The laser scanner data was used, when the robot reference prism was in an area between 2 to 5 m in the x direction. For filtering purposes, the scans were first transformed to Cartesian coordinates and then to the greenhouse coordinate frame. The reflected points of the robot vehicle were removed and the point cloud was separated as described above. By doing this, the RANSAC could be performed for each crop line separately. Each of the line fittings was addressed directly to one single point cloud set without using any prior knowledge about the last dataset or the robot position. Scans with less than three points in the line area were ignored. In total 10277 different laser scans were evaluated. In Table 2 the number of analysed scans per line are presented.

Table 1: Listing of the line parameters.

Line number	Line equation in [m]	Intersection point with last row in [m]
1	$f(x) = -0.0062x + 0.4441$	-
2	$f(x) = -0.0079x + 1.1913$	$P\,(439.53, -2.28)$
3	$f(x) = -0.0085x + 1.9414$	$P\,(1250.17, 8.69)$
4	$f(x) = -0.0068x + 2.7186$	$P\,(-457.18, 5.83)$
5	$f(x) = -0.0101x + 3.4811$	$P\,(231.06, 1.15)$

Table 2: Numbers of analysed scans per line.

Test number:	Date	Days after seeding	Analysed scans line 3	Analysed scan line 2
1	23.04.2015	28	67	688
2	27.04.2015	32	601	990
3	30.04.2015	35	910	1041
4	05.05.2015	40	897	941
5	13.05.2015	48	754	763
6	18.05.2015	53	653	652
7	01.06.2015	67	660	660

The difference between ideal line and the algorithm solved line, was evaluated with the help of the Root Mean Square Error (RMSE), defined by the following equation:

$$\boldsymbol{RMSE} = \sqrt{\frac{\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} (\delta - \beta_{\boldsymbol{i}})^{\,2}}{\boldsymbol{n}}} \tag{3}$$

With δ as ideal line parameters and β_i as the resolved algorithm parameters at the scan i.

4.4 Results and discussion

As the greenhouse soil was not homogeneous, there was a huge diversity in growth status. For tracking the crop development, the highest point of each plant was measured and the mean value was evaluated for each crop row. The variability is expressed by the standard deviation of all 41 plants heights per crop row. The results are shown in Figure 5. The average height of the plants at line 3 had been lower than at line 2. 48 days after seeding, most of the plants reached the level of 0.2 m height. At all tests afterwards the number of analysed scans had been almost the same for both sides (see Table 2). As in the first two tests the average plant height of line 3 was below the height of the laser scanner, the RANSAC algorithm for line 3 detected points, just when the vehicle was turned downwards, because of uneven ground. The absolute mean value for the height of the line 3 was 0.47 m while for line 2 the mean value was 0.65 m. The tallest plant reached 0.82 m at line 3 and 0.86 m at line 2. For later growth stages the standard deviation was increased. Along with the height, the numbers of leaves, covering the row, were also increased. This caused limited sight of view for the LIDAR.

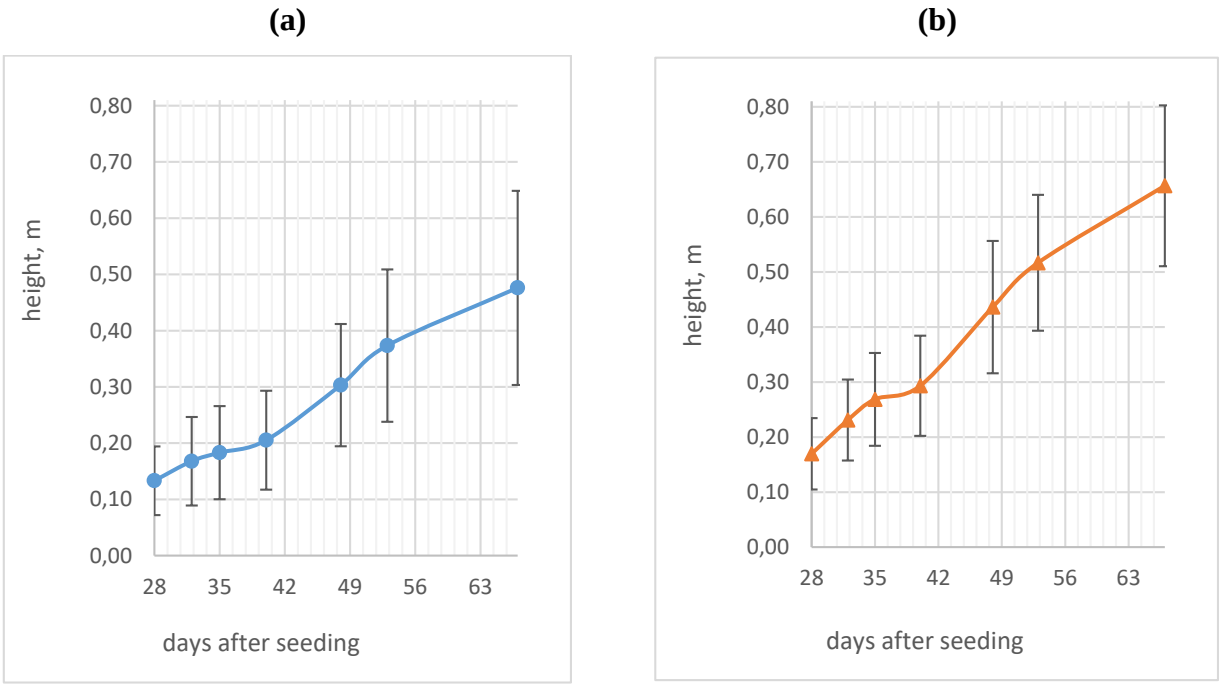

Figure 5: Mean growth status of the plants for (a) line 3 and (b) line 2.

For evaluating the change between crop row and RANSAC output, the RMSE for every data set was estimated. The results for the row position (Figure 6a) showed a higher error in the first tests for all mean values below 0.2 m. For all other heights, the RMSE value fluctuated at a value around 0.1 m.

The best matches were after 35 and 67 days at line 2 with an RMSE value of 0.07 m and 0.06 m, respectively.

Also the heading error (Figure 6b) had a higher value at the first tests with low mean plant height. For both lines a local minima could be detected after a 35 days (see Figure 6b). As it could be seen in Table 2, this was the first test with almost equal number of detected lines out of the scans. With the growth of the plants, the precision decreased back again. Only the last measurement of line 3 did performed better than the first minima of the same line. A reason for this could be the inhomogeneous growth of the crop plants. In total the RANSAC performed better at line 2 than in line 3. Reasons for that could be the more homogenous growth of the plants, which is expressed by the standard deviation of the two lines (see Figure 5). Especially line 2 had almost constant RMSE values between 35 and 53 days after seeding.

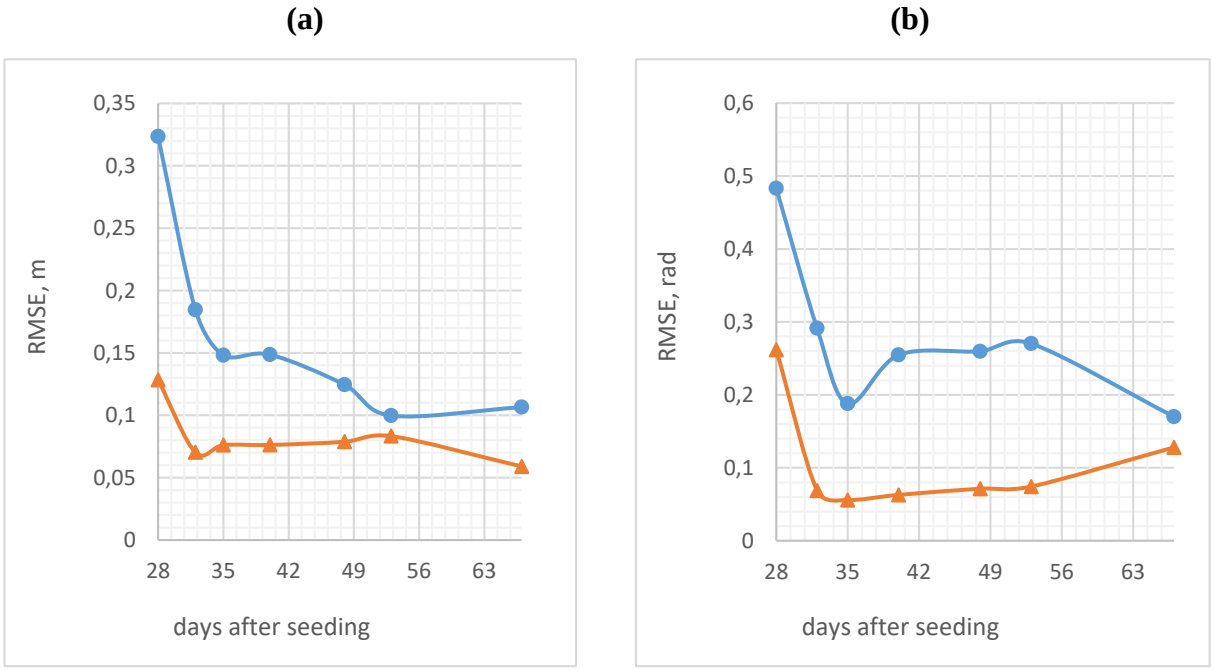

Figure 6: RMSE of (a) position and (b) heading of row line 2 (▲) and line 3 (●).

To better understand the evaluated error of the real line parameters, the direct RANSAC output is shown in Figure 7. For evaluating the values, two tests after 35 days and two tests after 67 days are visualized. For both test days, the robot moved through the row in each direction once. 200 measurements were evaluated and compared in the following graph (see Figure 7). The RANSAC heading output after 35 days is shown in Figure 7a and the output after 67 days can be seen in Figure 7b. The inclination parameter of line 2 is -0.0079, which under ideal conditions should be the same like the computed RANSAC parameter. The nearest heading to this theoretical value can be seen at 35 days after seeding (see Fig 6b). After 67 days, the computed values increased and produced out of

these a bigger shift to the reference value. At the end of the row a higher shift can be noticed. A reason for this could be, the lower number of points, detectable at the end of a row, to balance the outliers. After 67 days this performance was worse compared to the values after 35 days as illustrated in Figure 7b. First this could be reasoned, by the high amount of leaves hanging into the laser scan area. These leaves blocked the detection of the stem positions that were necessary to evaluate the line orientation. For every direction the robot moved, a static shift of the heading was observed (see Figure 7). This can be explained by some reflections of leaves, which caused a shift of the detected line to the middle of the row. This effect was stronger after 67 days and caused a narrow detection of the plant stems. The minimal reached RMSE was 0.05 rad for the line detection with the RANSAC after 37 days. After 67 days this value increased.

In worst cases the noise could be much higher than 5 degrees (0.1 rad) compared to the real value. This can cause problems on line following, especially when there is not enough space between the vehicle and the rows. The failure rate could be seen in many cases of the evaluated data. A part of the analysed error could also be resulted by the inaccuracy of the LIDAR measurements.

For getting a RANSAC algorithm robust for navigation, this heading uncertainty must be compensated. Higher algorithm robustness could be accomplished using a Kalman filter. When the growth status is known, the heading error could also be decreased by a static offset, which must be evaluated before starting the line following. Filtering for outliers or mean filter methods could also bring better results.

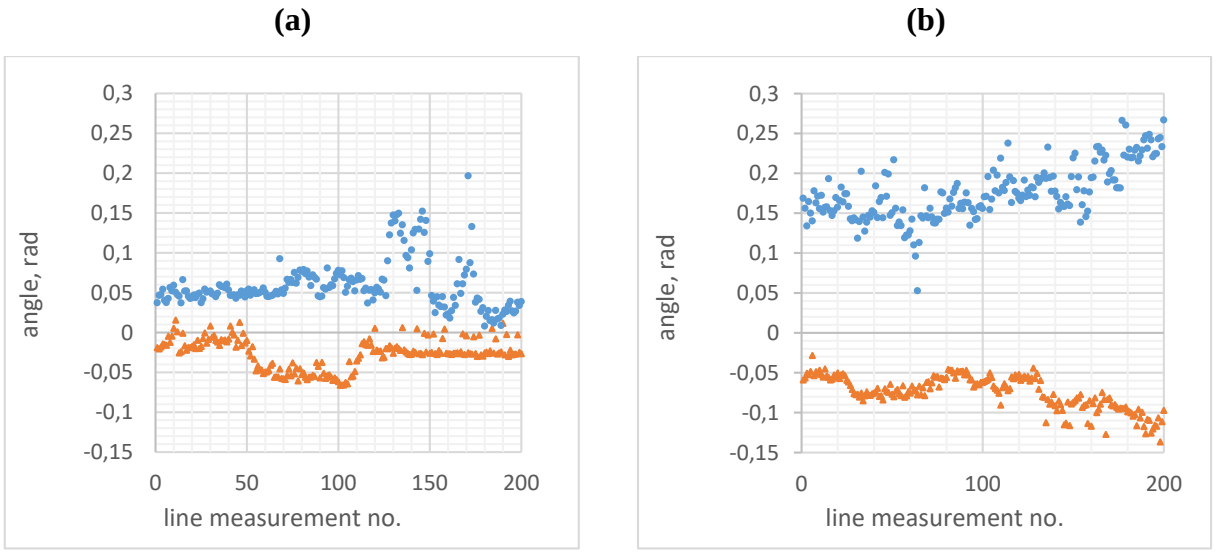

Figure 7: Heading values of the RANSAC for (a) line 2, 35 days after seeding and (b) 67 days after seeding. Robot movement in positive x direction (orange triangles) and in negative x direction (blue dots).

The results of the RANSAC showed a high variability in the dataset, with different outcomes of the algorithm. So it could be assumed, that the variability in the dataset brings additional options for

testing the robustness of line following algorithms for different growth states. The experimental setup allowed to detect a heading offset, which is dependent on the growth status of the crop plants. It could be shown that this offset is dependent on the sensor position and the movement of the robot. Also the best height for the line detection with the given laser position could be evaluated. This was at a mean height of 0.265 m of the plants.

4.5 Conclusions

The results of the collected data set showed high precision and good referenced sensor data for all measured growth stages. The application of a RANSAC algorithm for line detection to the horizontal laser data showed high diversity in heading and positioning. The smallest heading error was detected 35 days after seeding and at an average plant height of 0.268 m. After that, the error increased and brought a higher RMSE value to the detection. Also a drift dependent on the travel direction of the robot was observed, which was caused by leaves inside the row. This effect increased with the growth of the plants. In many cases of the given data set, the deviation of the line heading was higher than 5 degrees. This would cause problems for precise row navigation. The position error was for most cases acceptable. For line following applications in maize with a RANSAC algorithm, robust filtering of the laser data and algorithm results should be considered. In total the approach was helpful in order to evaluate some basic problems of outdoor line detection with LIDARs and a RANSAC algorithm. Aside of that, the accurate reference of the heading difference could be evaluated.

Acknowledgements

The project is conducted at the Max-Eyth Endowed Chair (Instrumentation & Test Engineering) at Hohenheim University (Stuttgart, Germany), which is partly grant funded by the DLG e.V.

References

Barawid, O. C., Mizushima, A., Ishii, K., & Noguchi, N. (2007). Development of an Autonomous Navigation System using a Two-dimensional Laser Scanner in an Orchard Application. Biosystems Engineering, 96(2), 139–149. doi:10.1016/j.biosystemseng.2006.10.012

Bochtis, D., Griepentrog, H. W., Vougioukas, S., Busato, P., Berruto, R., & Zhou, K. (2015). Route planning for orchard operations. Computers and Electronics in Agriculture, 113(APRIL), 51–60. doi:10.1016/j.compag.2014.12.024

Choi, S., Park, J., Byun, J., & Yu, W. (2014). Robust Ground Plane Detection from 3D Point Clouds. In: 14th International Conference on Control, Automation and Systems (ICCAS 2014) (pp. 1076–1081).

English, A., Ross, P., Ball, D., & Corke, P. (2014). Vision Based Guidance for Robot Navigation in Agriculture. In: IEEE International Conference on Robotics and Automation (ICRA) (pp. 1693–1698). doi:10.1109/ICRA.2014.6907079

Fischler, M. A., & Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM, 24, 381–395. doi:10.1145/358669.358692

Hansen, S., Bayramoglu, E., Andersen, J. C., Ravn, O., Andersen, N. A., & Poulsen, N. K. (2010). Derivative free Kalman filtering used for orchard navigation. In: 13th international Conference on Information Fusion. doi:10.1109/ICIF.2010.5712041

Hiremath, S. A., van der Heijden, G. W. A. M., van Evert, F. K., Stein, A., & ter Braak, C. J. F. (2014). Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter. Computers and Electronics in Agriculture, 100, 41–50. doi:10.1016/j.compag.2013.10.005

Jiang, G., Zhao, C., & Si, Y. (2010). A machine vision based crop rows detection for agricultural robots. In: Proceedings of the 2010 International Conference on Wavelet Analysis and Pattern Recognition (pp. 11–14).

Marchant, J., & Brivot, R. (1995). Real-Time Tracking of Plant Rows Using a Hough Transform. Real-Time Imaging, 1(5), 363–371. doi:10.1006/rtim.1995.1036

Marden, S., & Whitty, M. (2014). GPS-free Localisation and Navigation of an Unmanned Ground Vehicle for Yield Forecasting in a Vineyard. In: Proceedings of the 13th International Conference IAS-13.

Papari, G., & Petkov, N. (2011). Edge and line oriented contour detection: State of the art. Image and Vision Computing, 29(2–3), 79–103. doi:10.1016/j.imavis.2010.08.009

Russell, S. J., & Norvig, P. (1995). Artificial Intelligence: A modern approach (1st ed.). Englewood Cliffs, NJ: Prentice-Hall.

Rusu, R. B., & Cousins, S. (2011). 3D is here: point cloud library. In: IEEE International Conference on Robotics and Automation, 1–4. doi:10.1109/ICRA.2011.5980567

Weiss, U., & Biber, P. (2011). Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. Robotics and Autonomous Systems, 59(5), 265–273. doi:10.1016/j.robot.2011.02.011

Weiss, U., Biber, P., Laible, S., Bohlmann, K., & Zell, A. (2010). Plant species classification using a 3D LIDAR sensor and machine learning. In: Proceedings - 9th International Conference on Machine Learning and Applications, ICMLA 2010, 339–345. doi:10.1109/ICMLA.2010.57

Zhang, J., Maeta, S., Bergerman, M., & Singh, S. (2014). Mapping Orchards for Autonomous Navigation. In: ASABE Annual International Meeting (pp. 1–9).

CHAPTER 5

Paper D

Iterative Individual Plant Clustering in Maize with Assembled 2D LiDAR Data[4]

David Reiser, Manuel Vázquez-Arellano, Dimitris S. Paraforos, Miguel Garrido-Izard and Hans W. Griepentrog

Abstract

A two dimensional (2D) laser scanner was mounted at the front part of a small 4-wheel autonomous robot with differential steering, at an angle of 30 degrees pointing downwards. The machine was able to drive between maize rows and collect concurrent time-stamped data. A robotic total station tracked the position of a prism mounted on the vehicle. The total station and laser scanner data were fused to generate a three dimensional (3D) point cloud. This 3D representation was used to detect individual plant positions, which are of particular interest for applications such as phenotyping, individual plant treatment and precision weeding. Two different methodologies were applied to the 3D point cloud to estimate the position of the individual plants. The first methodology used the Euclidian Clustering on the entire point cloud. The second methodology utilised the position of an initial plant and the fixed plant spacing to search iteratively for the best clusters. The two algorithms were applied at three different plant growth stages. For the first method, results indicated a detection rate up to 73.7% with a root mean square error of 3.6 cm. The second method was able to detect all plants (100% detection rate) with an accuracy of 2.7 – 3.0 cm, taking the plant spacing of 13 cm into account.

[4] The publication of Chapter 5 is done with the consent of the Elsevier Publishing. The original publication was in *Computers in Industry*. It can be found under the following link: http://doi.org/10.1016/j.compind.2018.03.023

Keywords: LiDAR, individual plant detection, context, iterative plant clustering, total station, stem detection

5.1 Introduction

Detecting individual plants could bring high benefits to future farming. When the precise position of the plants is known, variable rate application could be scaled down to individual plant treatment. By obtaining this information, autonomous mechanical or thermal precision weeding could be implemented (Griepentrog et al. 2005; Pérez-Ruiz et al. 2012). This could also help to decrease the use of chemicals such as fertilizers, pesticides and herbicides (Chéné et al. 2012; Gonzalez-de-Soto et al. 2016). The integration of autonomous machines into every-day agricultural practice could result in increasing yield, saving labour cost and time but also valuable resources. This could make the use of robots in agriculture economically feasible as it is still difficult to justify such an investment (Pedersen et al. 2006).

Autonomous machines need robust object classification of the environment and an optimal behaviour to the classified objects (Fountas et al. 2007; Reina et al. 2015). The machines need sensors that will enable them to react appropriately to any unknown circumstances that may occur in the unstructured agricultural environment. It is necessary to determine everything in this environment, which could be potentially damaged by the robot or damage the robot itself. This would be highly important in order to find the right strategy for robot navigation and the performed agricultural applications. Individual plant observation could even help to improve decisions when the question arises if the acquired sensor data belong to a rigid or a flexible obstacle, but also if the data are measurement noise. The availability of this information could enable an optimised path navigation system for future applications involving agricultural robotics.

Mapping the individual seed positions and using this information for later applications has been already successfully performed by using a Real Time Kinematic Global Navigation Satellite System (RTK-GNSS) (Griepentrog et al. 2005; Pérez-Ruiz et al. 2012). The main advantage of this method was that it was not dependent on the crop type, as the shape and the morphology of the plants were not considered (Pérez-Ruiz et al. 2012). However, this method was not able to define the individual plant parameters, which are necessary to determine the demands of each individual plant.

For plant classification in 2D images, many different algorithms and methods have been used. The results varied depending on the environment, the object diversity, the used sensors and the implemented algorithms. Shrestha et al. (2004) used an Otsu thresholding algorithm to discriminate between maize plants and weeds in video data. Sugar beet has been also analysed for crop/weed

discrimination using 2D images (Åstrand and Baerveldt 2004). Haug et al. (2014) reached a detection rate of 80.4%, using a random forest classifier for plant discrimination in images. Two methods based on support vector machine (SVM) and ant colony optimisation were used to discriminate different plant species out of 2050 images with an accuracy of 95.5% (Ali Jan Ghasab et al. 2015). Neural networks were used to classify sunflower crops in images reaching a classification rate of 85 – 90% (Arribas et al. 2011). Dyrmann et al. (2016) used a deep convolutional neural network to classify 22 different weed species out of a training dataset of 10413 upper-view pictures. The reached success rate fluctuated between 33 – 97%. Bayesian classification and unsupervised learning for isolating weeds in crop rows were investigated with a success rate of 85 – 94% by De Rainville et al. (2014).

3D point clouds for agricultural analysis were created by scanning the area of interest with terrestrial light detection and ranging (LiDAR) scanners (Escolà et al. 2016; Rosell et al. 2009). The point cloud clustering for separating single plants was already performed with fixed distances (Arnó et al. 2013), convex-hull approach (Auat Cheein et al. 2015), hidden semi-Markov model (Underwood et al. 2015), by clustering the highest points inside a defined grid (Reitberger et al. 2007) or using machine-learning approaches (Gleason and Im 2012). Weiss and Biber (2011) reached a detection rate of 60% while examining early stage maize plants with a 3D LiDAR scanner. They analysed the data with a k-d-tree based Euclidean Clustering algorithm. Garrido et al. (2014) used a LiDAR and a light curtain to detect stem positions of almond trees, with a detection rate of 99.5%. The position of the detected stems was obtained with the help of an optical wheel odometer at clearly separated plants with no overlapping parts. Sonar sensors were also used to detect plants, but the results were not that precise due to the detection cone of the sensor (Harper and McKerrow 2001; Reiser, Martín-López, et al. 2017). Weiss et al. (2010) tested different machine-learning algorithms for 3D point clouds generated by a 3D laser scanner under laboratory conditions, with a success rate of 34.5 – 98%. Conditional random fields were used in point clouds, created by structure from motion, to discriminate between grapes, leaves and branches with a success rate of 96% (Dey et al. 2012). Point clouds of an RGB-D camera were analysed to detect broccoli heads with the help of histograms and SVM classifier with a detection rate of 95.2% (Kusumam et al. 2016). A Bayesian classifier used the combination of spectral data and 3D features for crop/weed discrimination with a success rate of 85 – 95% (Strothmann et al. 2017).

The basic limitation of machine-learning principles is the high amount of labelled training data, which are necessary to create robust and reliable classifiers. Inappropriate data sets could cause serious issues in terms of safety for autonomous machines (Steen et al. 2016). As the plant diversity is immense, robust and accurate classifiers for plant discrimination under outdoor conditions are

indispensable. Machine-learning models are vulnerable to adversarial examples, which can cause unexpected classification errors (Rozsa et al. 2016). By correctly contextualising sensor data, more robust results could be obtained compared to machine-learning as proven by Garrido et al. (2014), where context awareness helped to reach a stem detection rate of 99.5%, better than all machine-learning principles in literature.

One of the biggest challenges for robotics is the sensing and the correct environment classification. Bechar and Vigneault (2016) considered it as the weakest link of autonomous robots. The best examples are harvesting robots, which are still not performing sufficiently under high uncertainties for detecting and picking fruits (Back et al. 2014). For reliable autonomous agricultural robots, a robust and easy object and plant detection system is necessary. However, this should be performed with affordable and robust sensors, so that the investment costs do not overcome the saving of resources and labour (Pedersen et al. 2006). Vision and machine-learning algorithms are facing difficulties when the objects of interest change their morphology over time. A unique characteristic in agriculture is that the plant detection algorithms are highly impacted by the plant growth stage.

In general, it is possible to extract information such as plant position, plant height, leaf area, yield or even plant health status out of 3D representations. As Vázquez-Arellano et al. (2016) point out, it is necessary to have 3D sensor data to obtain this information. Many of the autonomous outdoor robots today are using horizontally mounted 2D LiDAR scanners for simultaneous localisation and mapping (SLAM) and navigation, which are still more robust and efficient compared to camera-based systems (Hiremath et al. 2014; Reiser et al. 2016; Steen et al. 2016). Vertically mounted 2D LiDAR scanners have been mainly used for remote sensing (Andújar et al. 2013; Jiang et al. 2016). Most projects for individual plant detection used vertical laser scanners or already 3D sensors (Chéné et al. 2012; Garrido et al. 2014, 2015; Lin 2015). Horizontally mounted LiDAR scanners are not suitable for creating 3D representations because their detecting is limited due to the parallel scanning to the terrain. When the sensor is mounted vertically, every robot movement provides a new position or orientation of the sensor, allowing a better observation of the 3D environment. However, a vertically positioned sensor makes it difficult to detect obstacles in front of the robot. A compromise is to mount the LiDAR scanner in an inclined position. This makes it possible to detect obstacles before the robot reaches them, and at the same time to assemble the data into a 3D point cloud representation.

The aim of this publication is to use a terrestrial 2D LiDAR for obtaining georeferenced 3D point clouds of maize plants at different growth stages and use this information to cluster individual plants. Until now, the use of context information for plant clustering in 3D point clouds has not been investigated. Therefore, the proposed methodology should show that it is possible to significantly

increase the accuracy of the plant detection when an algorithm is used together with the right context information (plant spacing, row position). For the plant detection and calculation of the plant position, two different approaches are applied and compared:

- Search for plant objects in the complete 3D representation by removing the ground points and by searching for objects in the remaining data points with a Euclidian distance clustering.
- Estimate plant positions using the fixed plant spacing parameter and search for the nearest fitting cluster. Use the obtained cluster position to estimate the next plant position.

5.2 Materials and methods

5.2.1 Hardware and sensors

A small 4-wheel autonomous robot with differential steering was the carrier vehicle to move the sensors through the crop rows (Figure 1a) (Reiser et al. 2016). The Robot Operating System (ROS) visualisation software "rviz" (Figure 1b) was used to present the output of the data processing and assembling of the inclined LiDAR in real-time.

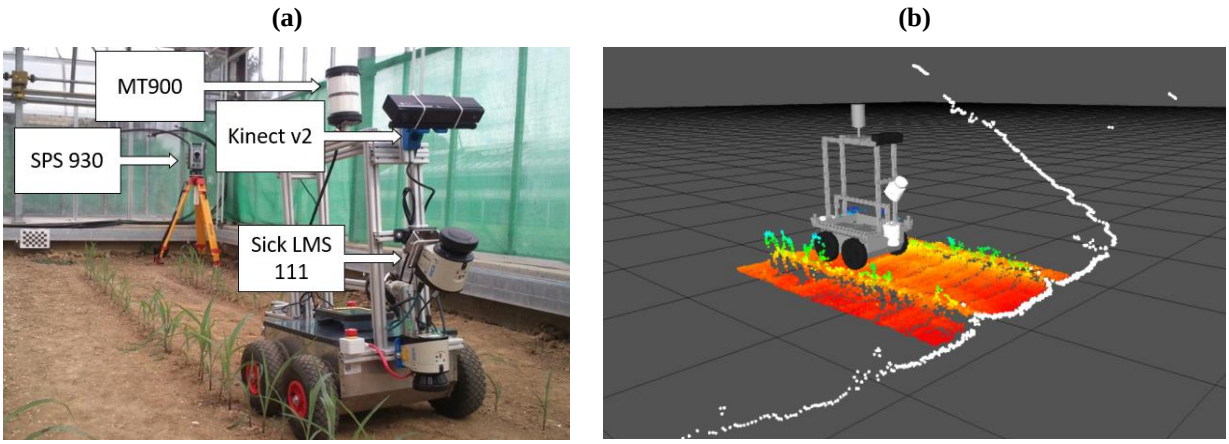

Figure 1: The robot platform for the data acquisition (a) and the representation of the robot in the ROS visualization tool "rviz" while assembling the LiDAR data (b).

The used laser scanner in this research was an LMS111 2D-LiDAR laser scanner (SICK, Waldkirch, Germany), mounted at a height of 0.58 m, pointing downwards at an angle of 30 degrees. The LMS111 sensor is robust against direct sunlight and has a separate heating system to work properly even under rainy and foggy conditions. To measure the robot orientation in all three directions, the VN-100 Inertial Measurement Unit (IMU) (VectorNav, Dallas, USA), was included in the sensor setup. The SPS930 Universal Total Station (Trimble, Sunnyvale, USA) tracked a Trimble MT900 Machine Target Prism (Trimble, Sunnyvale, USA), which was mounted on top of the robot at a height of 1.07 m (see Figure 1a) to evaluate the robot position. To qualitatively compare the assembled point

clouds with RGB images, a Kinect v2 (Microsoft, Washington, DC, USA) sensor was mounted at the front part of the robot at a height of 0.95 m. The specifications of the used sensors for the 3D point cloud generation can be found in Table 1.

Table 1: Sensor specifications.

Sensor	Specification	Value
LMS 111 (Sick AG Waldkirch 2017):	Operating range:	0.5 m to 20 m
	Field of view/scanning angle:	270º
	Data rate:	25 Hz
	Angular resolution:	0.5º
	Systematic error:	± 30 mm
	Statistical error (1σ):	12 mm (0.5-10 m)
SPS930 (Trimble 2017):	Operating range:	0.2 to 700 m
	Field of view/scanning angle:	360°
	Data rate:	20 Hz
	Distance measurement accuracy:	± (4 mm + 2 ppm)
VN100 (Vectornav 2017) :	Data rate:	40 Hz
	Angular resolution:	0.05°
	Accuracy heading:	2.0°
	Accuracy pitch/roll:	1.0°

The robot used an embedded computer system with 3.3 GHz, 4 GB RAM and SSD Hard drive. In addition, the Total Station data were sent with a radio antenna at the 2.4 GHz range (IEEE 802.11) to a Yuma 2 Rugged Tablet Computer (Trimble, Sunnyvale, USA) with an Intel Atom CPU N2600 dual-core processor with 1.6 GHz, 4 GB RAM and SSD Hard drive. The Yuma 2 Tablet was placed on the robot. The data exchange between the computers was done via serial RS232 interface. The whole data flow inside the robot, observation system, computers and the sensors are depicted in the following data flow chart (Figure 2).

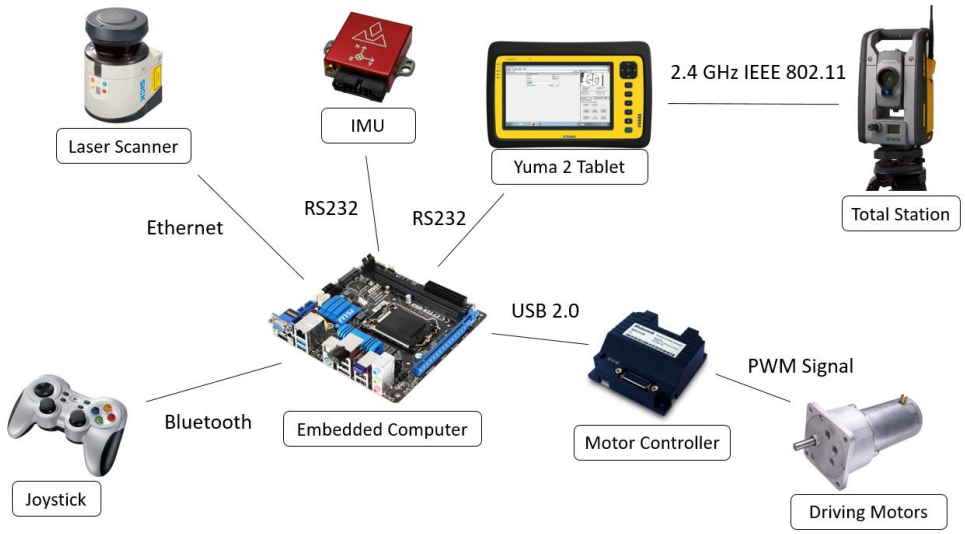

Figure 2: Data flow diagram of the robot and sensor architecture used in the experiment (Reiser et al. 2016).

5.2.2 Software

The robot computer was running on Linux Ubuntu 14.04 and used the ROS-Indigo middleware for sensor control, actuator control and data recording (Quigley et al., 2009). This middleware provided the architecture for sensor time-synchronisation, data processing, data filtering and linear data interpolation. As ROS is providing monitoring (rqt) and 3D visualisation tools (rviz), it was possible to monitor the sensor data and the robot behaviour in real-time and to visualize the outputs. The software components of the robot were programmed in a combination of C++ and Python programming languages. The Trimble SCS900 Site Controller Software (Software Version 3.4.0) was used to recalibrate the system for every test and to export the total station data to ROS. The plant detection accuracy of the programmed algorithms was analysed in Matlab R2015b (MathWorks, Natick, MA, USA).

5.2.3 Calibration and experiments

To reference all sensor data to the same Cartesian coordinate frame, five fixed locations were defined at the test area inside a greenhouse (Garrido et al. 2015; Reiser et al. 2016). The positions of these five points were measured by the total station at the beginning of every test by fixing the prism on the defined locations on the greenhouse's surrounding concrete wall. By doing this, the positioning system of the total station was every time calibrated to the same fixed coordinate frame. Based on these static measurements, the shift between the first reference points and the actual measurements was below 4 mm for all three dimensions in all tests. First, the roll, pitch and yaw angle of the IMU were fused with the total station data and were used to define the prism position and orientation. From the prism, a static transformation to the robot geometric centre and to the sensor position was performed. This procedure allowed to track the precise sensor position and orientation in the same reference frame (Garrido et al. 2015; Reiser et al. 2016).

In total five rows of maize were seeded in the greenhouse, with a length of 5.2 m and a width of 0.75 m. Each row had a fixed number of 41 plants. In this study, just two rows were considered, one row to the left and one to the right of the robot driving direction. The spacing between the plants was defined by different Gaussian distributions for every crop row, to emulate diverse real scenarios. The rows used in this paper were planted with a standard deviation for the plant spacing of 1.7 cm (left row) and 0.6 cm (right row). The ground truth positions of the plants were measured using the Total Station after plant emergence. With the help of a tripod, a pendulum was pointed over the stem position to obtain the correct position. The resulting distance between plants had a mean spacing of 13.0 cm (left row) and 12.8 cm (right row). The corresponding standard deviation was 5.9 cm left and 2.2 cm at the right crop row. This agrees with the results reported by Sun et al. (2010), were a standard

deviation between planting and emergence of 5.1 cm was measured with an RTK-GNSS system. Due to the limited space inside the greenhouse, it was possible to visualize 38 of the plants in the 3D representation. The tests were performed at three different growth stages with plants between V1 and V3 leaf stage (Ritchie et al. 1993). As the soil in the greenhouse was not previously cultivated, and no fertiliser was added to the soil in the experiment, the development of the maize plants was slower than it would be under real field conditions. Different lighting conditions occurred during the experiments because the greenhouse was not shadowed. The average height and standard deviation of the plants in the collected datasets are described in Table 2.

Table 2: Description of the maize plant heights in the used datasets.

	Left row		**Right row**	
Dataset no.	**Average plant height [m]**	**Height standard deviation [m]**	**Average plant height [m]**	**Height standard deviation [m]**
1	0.14	0.0567	0.12	0.0556
2	0.17	0.0624	0.13	0.0609
3	0.23	0.0831	0.16	0.0786

After every test, the height and the stem width of every single plant were manually measured and used as a reference. All three tests were performed in the same driving direction and in the same path. The number of used sensor readings, recording time and other details for all datasets are described in Table 3.

Table 3: Description of used datasets including date of the data acquisition, duration and number of sensor readings.

Dataset [no.]	**Days after seeding [d]**	**Date**	**Duration [s]**	**IMU data [no.]**	**Tilted LiDAR scans [no.]**	**Total station data [no.]**
1	26	21.4.2015	78	3146	1965	1573
2	28	23.4.2015	54.8	2189	1366	1086
3	32	27.4.2015	60	2426	1499	1200

The trajectories of the robot for all three performed tests are depicted in the following Figure 3, together with the ground truth of the maize plants.

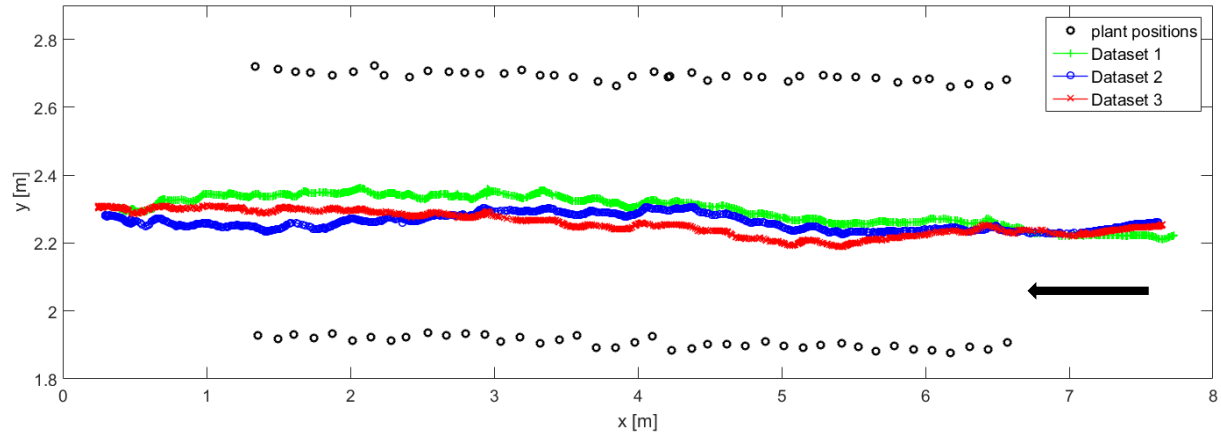

Figure 3: Path moved by the robot at the three test days. For reference, also the plant positions are depicted. The driving direction was from the right to the left (see arrow).

The speed of the robot was kept as constant as possible, to ensure similar densities for the assembled point clouds. The mean speed was 0.0965 m s^{-1}, 0.1086 m s^{-1} and 0.1064 m s^{-1} for the first, second and third test, respectively. The exact speed distribution over the x-axis for the three tests can be seen in Figure 4.

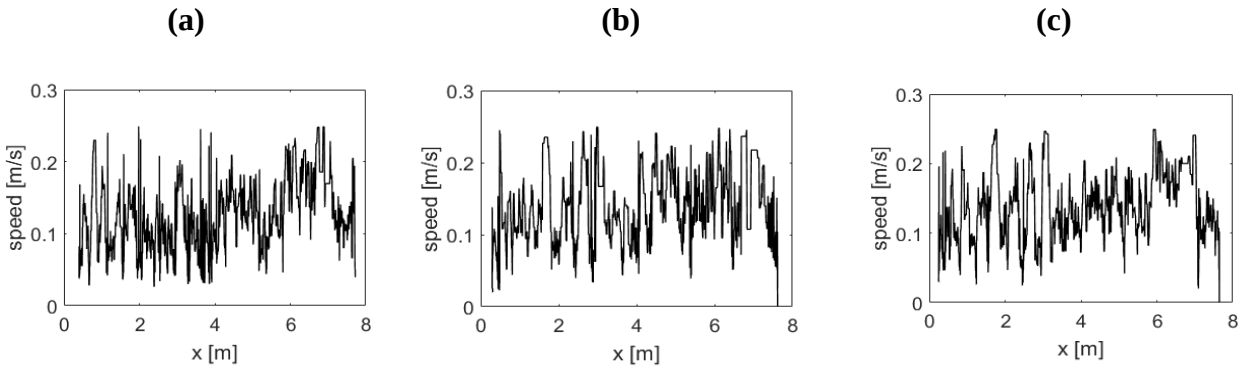

Figure 4: Description of the three test runs (a) speed of the first, (b) second, and (c) third dataset.

5.2.4 Data processing and assembling

To obtain a representative and accurate plant map out of the raw sensor data, the first task was to interpolate the LiDAR position by using the total station and IMU data. This position was then fused with the LiDAR data in order to create a 3D point cloud representation. Subsequently, the two different approaches to filter plant structures were applied to the dataset and the results were compared (Figure 5). The first approach used Plant Detection with Euclidian Clustering (PDEC) and the second an Iterative Plant Clustering Method (IPCM).

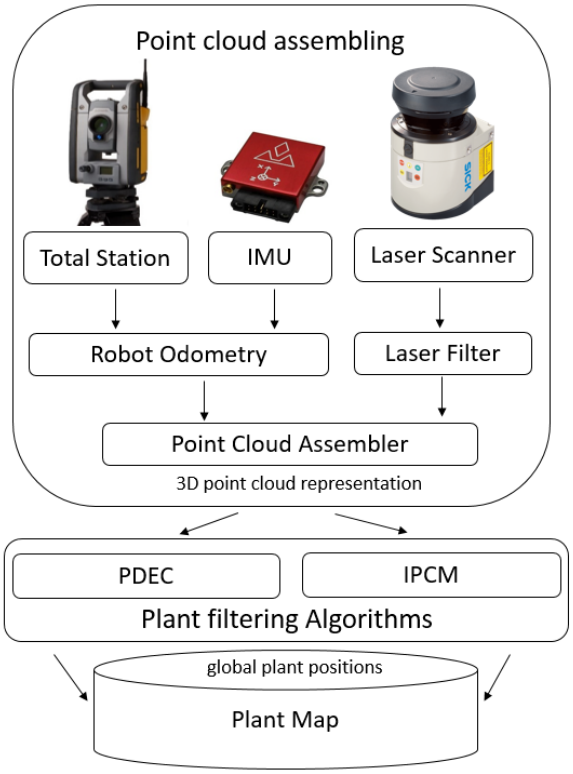

Figure 5: Data flowchart, describing how the sensor data were converted to a resulting plant map.

5.2.5 Point cloud assembling

To obtain usable 3D point clouds from the referenced 2D sensor data, the raw data of the laser scanner were filtered to remove noise and points outside the area of interest. Initially, the reflections of the vehicle and the greenhouse wall were removed from the sensor data, using a distance filter. In addition, the scanned distances by the LiDAR were limited to 1 m so that only one row to the left and one to the right of the robot could be observed. Sharp edges and reflective surfaces can cause the effect of misaligned LiDAR beams and create the so-called ghost points (Balduzzi et al. 2011). This is mainly the case when the laser beam partially hits a plant, or the angle of incidence is higher than a specific threshold. Balduzzi et al. (2011) found that an angle of incidence higher than 60 degrees can cause variable intensities and make ghost points more probable. The plant edges were responsible for a high percentage of this measurement noise. To remove these ghost points from the raw data, a shadow noise filter was applied (see Algorithm 1).

Algorithm 1 Shadow Noise Filter

```
Require: scan, sensorOrigin, minAngle, maxAngle
for all points p in scan do
        angle_to_last_point = getAngleBetweenPointsComparedToSensorOrigin (p_i, p_i -1)
        angle_to_next_point = getAngleBetweenPointsComparedToSensorOrigin (p_i, p_i+1)

        if angle_to_last_point >minAngle or angle_to_last_point < maxAngle then
            removeFromOutput (p_i)
        end if
        if angle_to_next_point >minAngle or angle_to_next_point < maxAngle then
            removeFromOutput (p_i)
        end if
end for
```

For all three datasets, the same filter settings were used. The *minAngle* was set to -60 degrees and the *maxAngle* 60 degrees. Then the data were transformed together with the fusion of the robot total station position and the IMU orientation to a fixed world coordinate system (Garrido et al. 2015).

5.3 Plant detection algorithms

The resulting 3D representations were later processed to find the plant clusters and then predict the individual plant positions. For every plant point cloud cluster, the 3D centroid was evaluated and assumed as the resulting plant position. The centroid c was correlated to the n number of P points in a point cloud cluster in Equation 1.

$$c = \frac{1}{n}\sum_{i=1}^{n} P_i \tag{1}$$

Only the x and y coordinates of the plant positions were considered in order to compare the calculated centroids to the ground truth measurements acquired by the total station. To assess the accuracy of the plant pose, the mean value, the standard deviation and the Root Mean Square Error (RMSE) of the position error were calculated and compared.

5.3.1 Plant Detection with Euclidian Clustering (PDEC)

The PDEC method follows a basic approach for detecting and separating objects by removing the ground plane, which is widely used for object recognition in point clouds (Owens et al. 2015; Vosselman et al. 2004; Weiss and Biber 2011). Consequently, the ground points were detected and removed from the point cloud. The remaining points above the detected ground plane were further processed. This method does not take the order of points into account and can be applied to all point clouds, with clearly separated objects such as early-growth-stage crop plants. Initially, the point cloud was separated into two parts to obtain two point clouds, one for each row. To these point clouds, a basic Random Sample Consensus (RANSAC) plane algorithm was applied as implemented in the

Point Cloud Library (PCL) (Fischler and Bolles 1981). A heuristic algorithm calculated the distance of each point *P(x,y,z)* to the plane model, to get the point-to-plane distance. This *ransac_dist* was solved using the Hessian Normal form, describing a plane with the plane equation parameters constants a, b, c and d.

$$ransac_{dist} = ax + by + cz + d \quad (2)$$

When the distance of the point to the plane equation was below the defined threshold, this point was removed from the point cloud. To eliminate noise and outliers, the remaining plant points were filtered using a radius outlier filter (PCL 1.7.0, RadiusOutlierRemoval class). To separate the resulting points, a k-d-tree clustering was utilised (PCL 1.7.0, EuclideanClusterExtraction class) (Bentley 1975; Rusu et al. 2009), assuming that the plants were spatially separated in the point cloud. The min and max cluster size of the point clouds were fixed in the range between 5 and 1000 points. To ensure the best possible results, the parameter values were individually adjusted for each row. The used parameters for the algorithm are listed in Table 4.

Table 4: Algorithm settings first method (PDEC).

Dataset [no.]	Row side	RANSAC distance [m]	Noise radius [m]	Min points inside radius [no.]	Cluster distance [m]
1	Left	0.045	0.05	20	0.02
1	Right	0.035	0.03	10	0.05
2	Left	0.06	0.05	15	0.03
2	Right	0.035	0.05	15	0.05
3	Left	0.06	0.05	15	0.05
3	Right	0.03	0.05	15	0.03

5.3.2 Iterative Plant Clustering Method (IPCM)

The fixed plant spacing was used to facilitate the clustering of single plants from the entire point cloud. A rough estimate of the starting position inside the plant row was used to limit the search area. All points around the starting point with a distance less than the fixed plant spacing multiplied by a factor of 1.5 were selected (Algorithm 2). All ground points were removed from the point cloud. As the terrain was not level, an additional threshold parameter, the RANSAC distance, was necessary in order to remove all ground points (Equation 2). All remaining points after the ground removal were considered as plant points. The resulting points were clustered and the 3D centroid of every cluster was calculated. These centroids were compared to the initially estimated plant position. The nearest cluster centroid was estimated as the real plant position and was stored in a plant position array. On

the resulted plant position, the IPCM added the fixed plant spacing along the row direction and then restarted an iterative process until the end of the row was reached.

In the experiment, the row structure was parallel to the x-axis of the world coordinate frame. The plant spacing of 13 cm was added to the x-axis value of the last plant position. All resulting plant clusters were saved in an array (*plant_positions_array*) to be used later for plant description and mapping.

The height of three plant clusters was evaluated and was then compared to the ground truth data as an example to assess the algorithm capability. As the terrain plane was almost horizontal, the distance from the highest point to the terrain plane was considered as the height of the plant (Equation 2). The RANSAC plane reduction removed all points with a smaller distance to the plane than the RANSAC distance. This factor was added to the final estimated plant height.

Algorithm 2: *IPCM*

Input: *point_cloud, first_plant_position, plant_spacing, RANSAC_distance, cluster_distance, spacing_factor*
Output: *plant_position_array*

```
actual_plant_pose = first_plant_position
result_points=GetAllPointsAroundPlant(actual_plant, point_cloud, plant_spacing * spacing_factor)
result_points=RemoveGroundPoints(result_points, RANSAC_distance)

while result_points include points do
        clusters = DoEucledianClustering(result_points,cluster_distance)
        for all clusters do

                2D_centroid = Get2DCentroid(cluster)
                distance_to_actual_plant = GetDistance(2D_centroid, actual_plant_pose)

                if distance_to_actual_plant < than the other cluster centroids then

                        addToPlantArray(2D_centroid, cluster)
                        best_centroid = 2D_centroid

                end if
        end for

        actual_plant_pose = best_centroid + plant_spacing
        result_points = GetAllPointsAroundPlant(actual_plant, point_cloud, plant_spacing* spacing_factor)
        result_points=RemoveGroundPoints(result_points, RANSAC_distance)

end while
```

The whole process was programmed in a ROS node so that it could be used and visualised in real-time. The following Figure is illustrating the estimated plant positions (blue dots), the obtained point cloud and the ground truth (green sticks) in the ROS visualization tool “rviz” (Figure 6).

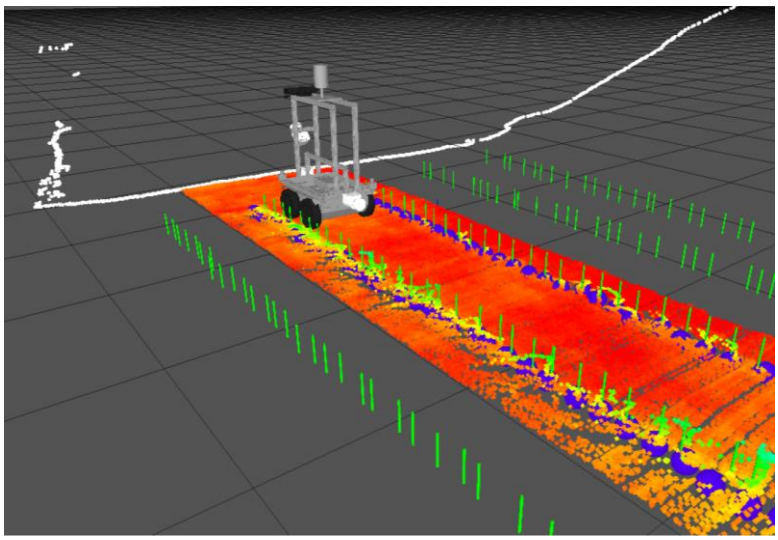

Figure 6: Processed point cloud is shown in a 3D representation of the ROS visualisation tool "rviz". The blue spheres present to the automatically detected plant positions and the green sticks indicate the ground truth positions.

The following Table 5 is presenting the used parameters for the three datasets. For each row side, one separate setting was used. The cluster distance defined the max spacing that was necessary between the points to separate the points into two different clusters. The "max cluster deviation" defined the max distance between the calculated cluster centres and the estimated plant position.

Table 5: Algorithm settings second method (IPCM).

Dataset [no.]	Row side	RANSAC distance [m]	Noise radius [m]	Minimal points inside radius [no.]	Cluster distance [m]	Max cluster deviation [cm]
1	Left	0.04	0.020	4	0.020	7.2
1	Right	0.045	0.015	5	0.024	5.4
2	Left	0.04	0.020	4	0.015	7.2
2	Right	0.045	0.015	5	0.024	6.5
3	Left	0.04	0.020	4	0.015	6.5
3	Right	0.045	0.015	5	0.024	6.5

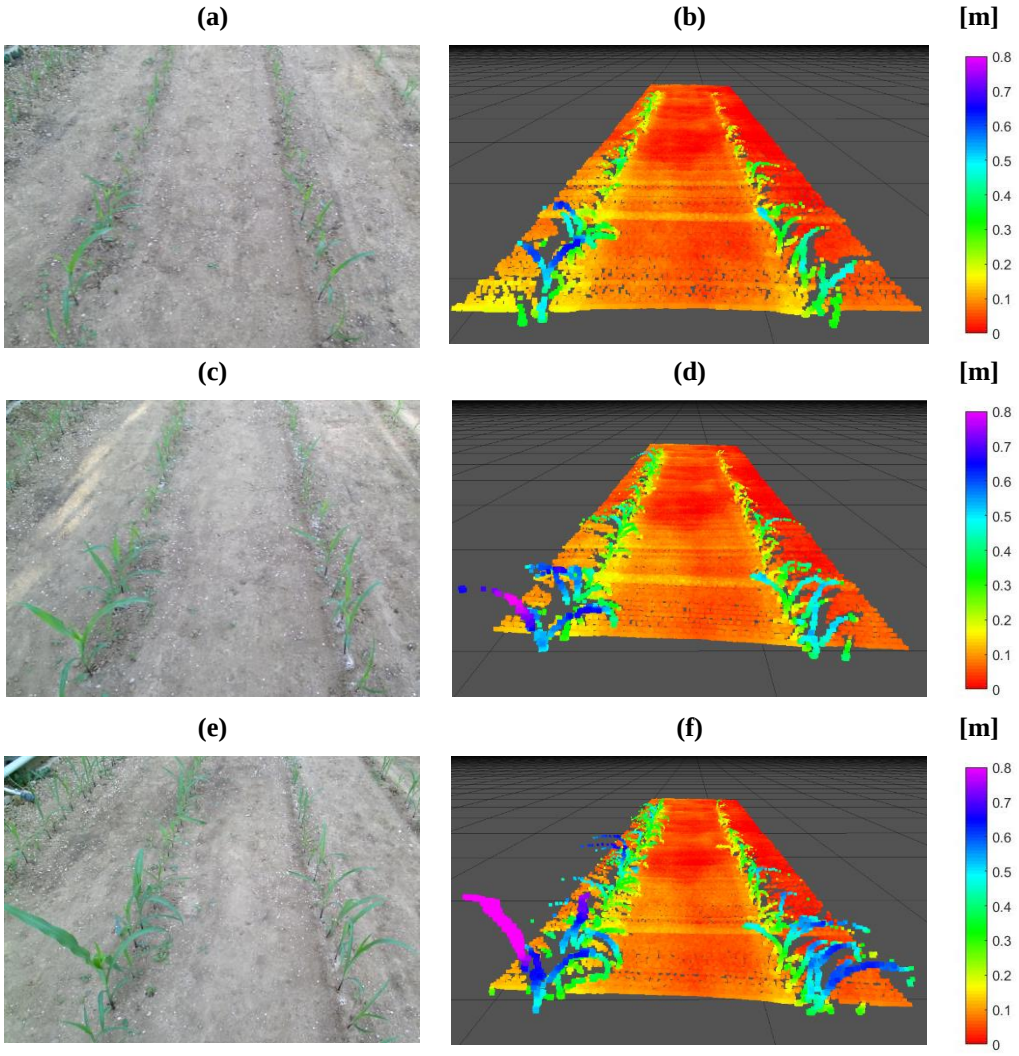

Figure 7: Assembled point clouds used by the algorithm for the three different growth stages: (a, b) 26, (c, d) 28, (e, f) and 32 days after seeding. The colorbar is defining the value of the points in the *z*-axis.

5.4 Results and discussion

The resulting 3D representations of the point cloud assemblies are depicted in Figure 7b, 7d and 7f. To indicate the difference in the growth stage, the corresponding RGB image by the attached Kinect v2 Sensor is also presented (Figure 7a, 7c, 7e). The Figure illustrates the plant development over time, starting 26 days (Figure 7a, 7b), 28 days (Figure 7c, 7d) and 32 days after seeding (Figure 7e, 7f). It could be seen that the plant development was not uniform, resulting in plant height differences, which introduced some difficulties to the plant detection algorithms. Another factor that caused problems to the algorithms was that many plants had overlapping parts. Sometimes points belonging to plants were removed and leaves were separated because of the low-density data of the point clouds.

The detection rate of the first algorithm (PDEC) resulted in varying rates with a max value up to 73.7%. The lower detection rate was 52.6% (Table 6). This was achieved with a best standard deviation of 2 cm and an RMSE of 3.6 cm at the right row of dataset 1. The percentages for the

detected plants were calculated considering 38 plants in every single row. From the results of Table 6, it can be seen that the distance accuracy decreases as the plants were growing. The RMSE increased from 3.6 cm to 6.5 cm and from 4.8 cm to 6.6 cm for the right and the left row, respectively. The mean distances also increased as the plants were growing. This was not the case for the plant detection rate that appeared not to have any direct correlation with the growth stage. The total evaluated plant number was 228 plants, with 148 correct detected plants, corresponding to a detection rate of 64.9%. The confusion matrix of the PDEC algorithm is shown in the following Table 7.

Table 6: PDEC algorithm results.

Dataset no.	Row side	Correct detected plants no.	Plants detected [%]	Mean distance [m]	Standard deviation [m]	RMSE [m]
1	Left	23	60.5	0.044	0.023	0.048
1	Right	27	71.1	0.030	0.020	0.036
2	Left	28	73.7	0.046	0.026	0.053
2	Right	20	52.6	0.037	0.019	0.042
3	Left	26	68.4	0.050	0.043	0.066
3	Right	24	63.2	0.052	0.040	0.065

Table 7: PDEC algorithm confusion matrix.

		True value		
		Plant	**no Plant**	**%**
predicted	**Plant**	148	25	85.5
	no Plant	80	0	0
	%	64.9	0	

In total 80 plants of the three different datasets were not detected by the PDEC algorithm and were therefore regarded as false negatives. A detected plant by the software was regarded as false positive when the estimated plant centre was more than 6.5 cm away (half of the plant spacing) from the real plant centre. The number of false positives was equal to 25 while the results of the true positives were quite sufficient with an accuracy of 85.5%. The spatial distribution of the detected plants by the PDEC algorithm can be seen in Figure 8.

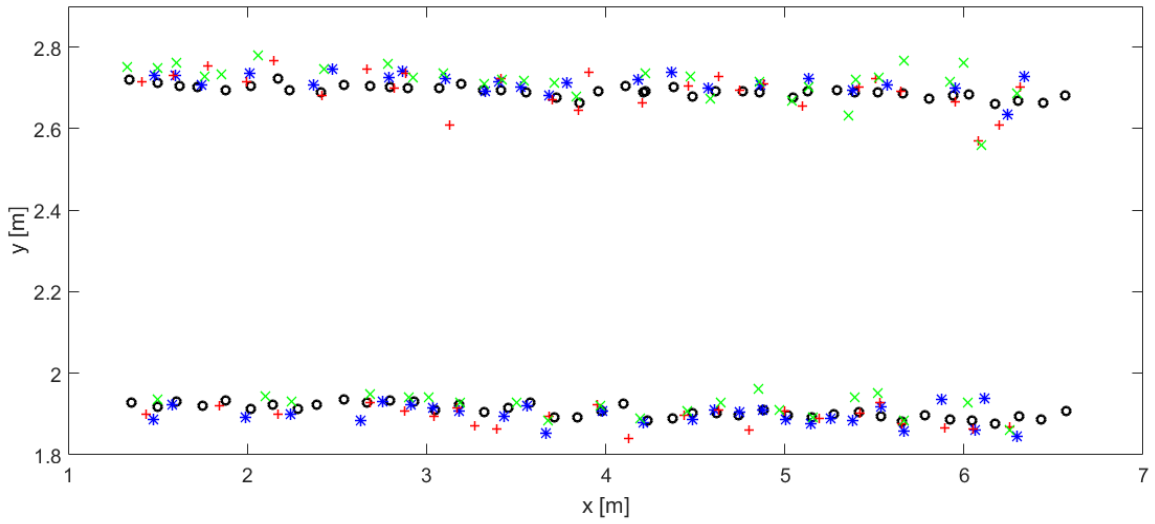

Figure 8: Plant positions obtained by the PDEC algorithm, with (o) ground truth of the plant positions, (*) result of dataset 1, (x) result of dataset 2 and (+) result of dataset 3.

The false positives were mainly triggered by leaves which were not connected to the rest of the plant points. This was caused by low point cloud density and the fixed scanning direction. Sometimes two plants could not be separated because of the close distance between them. This led to the detection of one plant directly in the middle of two real ones.

The overall detection rate of the algorithm was low but was mainly affected by the height differences of the plants. This caused that the algorithm settings could not be calibrated to work consistently at the whole row. Small plants were shadowed by the bigger ones beside them. At areas of low height variability, the detection rate was high and accurate. The precision on small plants was considerably good, where some plants had a detection rate of 100% at all three datasets. The precision of the position (less than 3 cm deviation), made the detection rate good enough to allow single plant treatment like mechanical weeding.

The IPCM algorithm used the same clustering method as the PDEC, but in combination with the known spacing parameters to search for the best fitting cluster around the estimated plant position. This allowed to define the algorithm settings more specifically and more precisely. The results of the iterative IPCM method are shown in the following Table 8.

Table 8: IPCM algorithm results.

Dataset no.	Row side	Correct detected plants [no.]	Plants detected [%]	Mean distance [m]	Standard deviation [m]	All-over RMSE [m]	RMSE without false positives [m]
1	Left	35	92.1	0.031	0.015	0.0345	0.0297
1	Right	38	100.0	0.029	0.009	0.0303	0.0303
2	Left	35	92.1	0.029	0.016	0.0331	0.0286
2	Right	38	100.0	0.027	0.010	0.0290	0.0290
3	Left	36	94.7	0.026	0.016	0.0304	0.0266
3	Right	36	94.7	0.028	0.017	0.0324	0.0293

As the IPCM method was searching for the best fit around the estimated plant position, a cluster was chosen for every plant, as long as some plant points were in the area. This guided to a plant detection up to 100% for the datasets 1 and 2 at the right row. The worst detection rate was 92.1% at the left row. The overall results were quite accurate, even when considering all false positives. The IPCM algorithm resulted in a mean deviation of less than 3.1 cm and a standard deviation less than 1.7 cm at all datasets. The worse RMSE of the method was 3.4 cm and was better than the best result of the PDEC method. This result even gets better when excluding the false positives, guiding to an RMSE between 3.0 and 2.7 cm. All false positives were caused by overlapping leaves between the plants so that the clusters could not be separated correctly with the Euclidian Clustering. The detection rates can be seen in Table 9.

Table 9: IPCM algorithm confusion matrix.

		True value		
		Plant	**no Plant**	**%**
predicted	**Plant**	218	10	95.6
	no Plant	10	0	0
	%	95.6	0	

The overall detection rate was 95.6% in a total sum of 228 analysed plants. As the algorithm always tried to detect the best spot for the estimated plant position, each false positive resulted in one true negative. The threshold for defining false positives was set, when the distance between the true plant position and the detected spot was more than 6.5 cm apart.

The spatial distribution of the plant positions is shown in Figure 9. It is possible to see that the main distributions and deviations from the real plant positions were in the y-axis direction. Especially the small and even separated plants on the left row side could be detected perfectly (Figure 9).

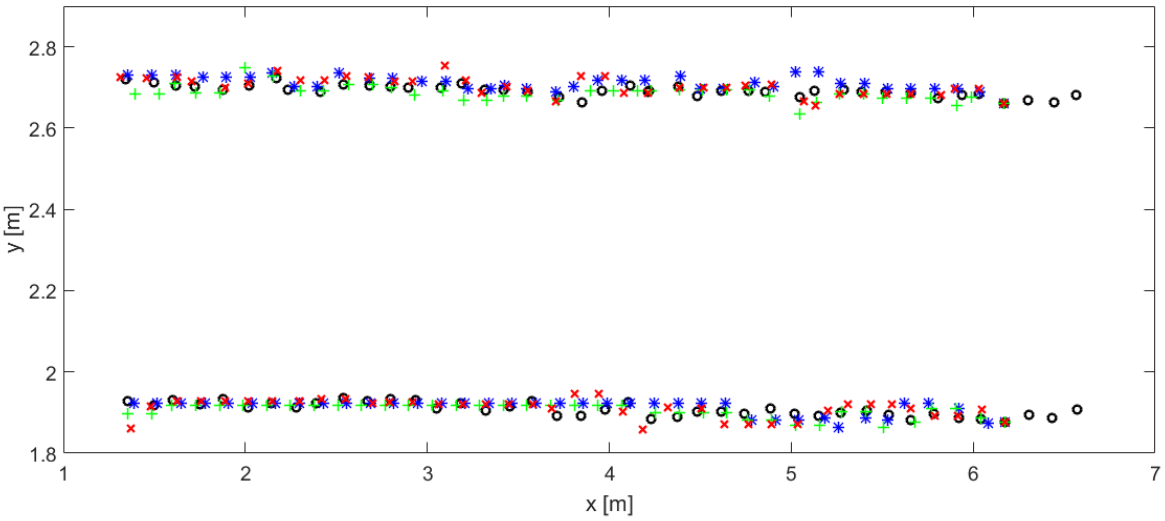

Figure 9: IPCM algorithm results of (*) dataset 1, (x) dataset 2, and (+) dataset 3 and (o) ground truth.

Some of the occurred errors were produced by the sensor and data acquisition setup. The positioning error of the Total Station could have added errors to the plant position estimation. Some deviations could be caused by the inaccuracy of the laser scanner, as the system can have a statistical error of +/- 30 mm (Table 1).

Figure 10 shows the automatically created point cloud clusters of the IPCM algorithm for one plant example and the change of the plant over the datasets. The considered plant is framed in the reference pictures (Figure 10a-c).

The shape of the reference plant varied for each dataset (Figure 10a-c). The height of the plant increased from 23 to 30 cm and the stem size from 0.25 to 0.48 cm. The IPCM algorithm managed to detect the plant in all three datasets, even when the plant leaves were touching the adjacent plant (32 days after seeding dataset). The resulting point cloud clusters are shown in Figure 10d-f (black dots). In all three cases, the algorithm did not manage to cluster the entire plant, as parts of the leaf and stem points were filtered or removed by the RANSAC ground removal. Nevertheless, this did not cause a high deviation of the obtained plant position, as the mean centre of the resulted point cloud cluster was close to the real plant position (Figure 10d and 10e). Cutting off parts of the leaves or the stem did not affect the estimated stem position considerably. A shift in the plant position of Figure 10f was noticed, which was caused by the asymmetrical clustering due to some wrongly clustered points of the adjacent plant. The accuracy was mainly affected by this kind of clustering errors. The highest point-to-plane distance correlates to the manual measured plant height. The results of the manual plant measurements to the height estimated by the programmed software of the selected plant in Figure 10, are shown in the following Table 10.

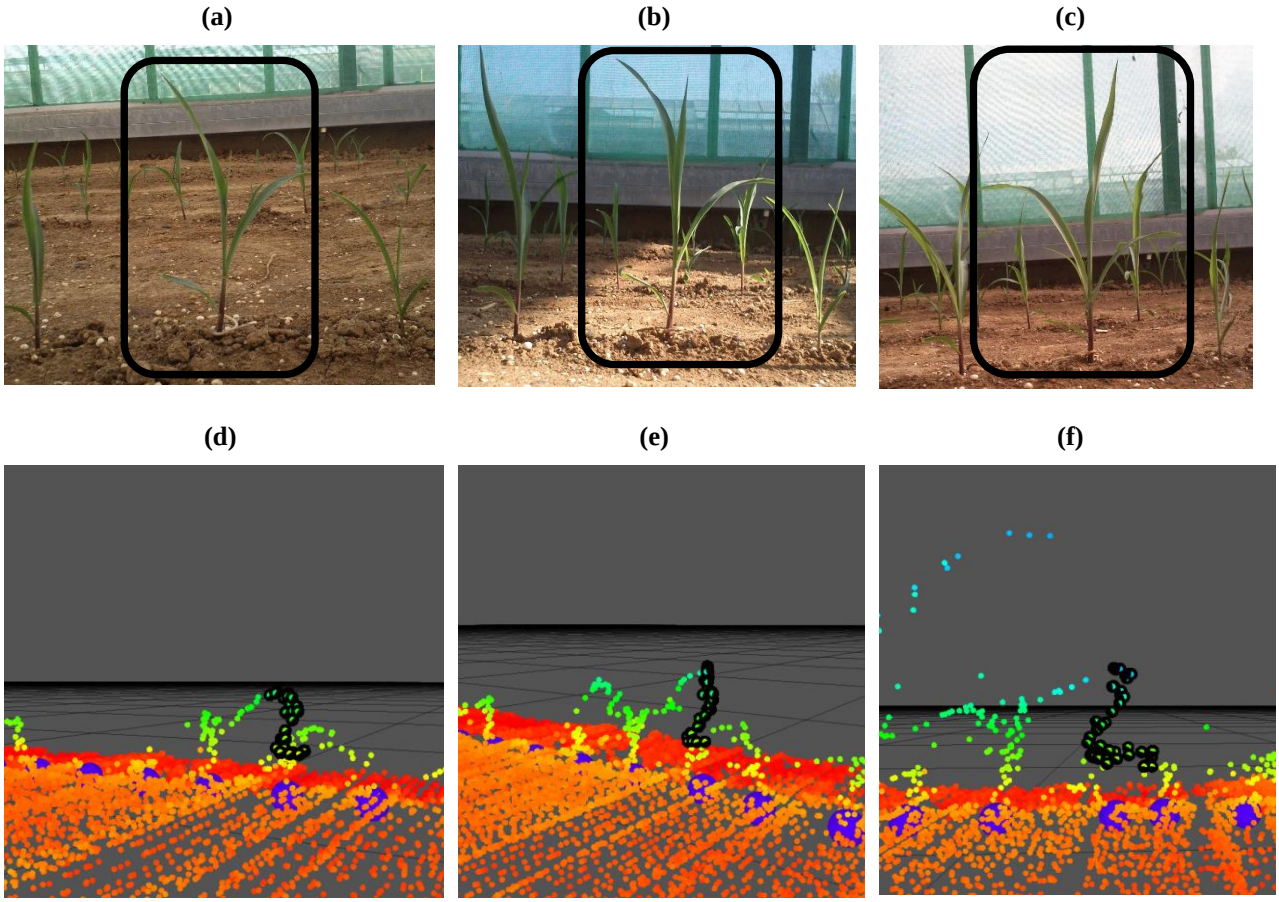

Figure 10: Pictures of the same plants at the 3 tests (a) 26, (b) 28 and (c) 32 days after seeding and the corresponding algorithm clusters (d), (e) and (f) (black dots).

Table 10: Results individual plant clustering of Figure 10.

Dataset no.	Stem width [cm]	Plant height [cm]	Algorithm plant height [cm]	Height deviation [cm]	Height deviation [%]
1	0.25	23.0	20.2	2.8	12.2
2	0.28	24.6	23.4	1.2	4.9
3	0.48	30.0	26.7	3.3	11.0

The maximum deviation of the height was 12.2% with a minimum of 4.9%. All values were below the manual measurement. One reason could be that the leaves are becoming quite small at the top of the maize plants, making it easy to get missed by the laser beams. Even when a beam hit the plant tip, the point was probably removed by the adjusted noise filters. This problem indicates the limits of the evaluated clustering method and the used sensor system.

Both described algorithms performed the plant clustering fast enough to be realised in real-time. The PDEC was faster and allowed to cluster the entire point cloud with a rate of 5 Hz. The iterative IPCM

needed more time, with a rate of 0.5 Hz for point cloud analysis. For real-time performance, the methods should keep the considered point cloud as small as possible. In combination with a global plant map, the search area could be minimised, allowing a fast and robust plant detection. With an adequate CPU or graphics card, it would be possible to increase the speed of the algorithms, in order to obtain information for navigation and application.

The main advantage of using the LiDAR LMS111 for sensing is the high reliability of the sensor under outdoor conditions. Especially the ambient light immunity until 40000 lx, the operating temperature between -30 and +50 °C and the IP 67 enclosure rating promise robustness against changing outdoor conditions (Sick AG Waldkirch 2017). Therefore, the sensor can work under bright sunlight as well as in a moonless night. This is a high advantage compared to the image based methods described in the introduction. In addition, the combination of safety and observation can be fulfilled with the same sensor system. The accuracy of the performed test was determined by the measuring accuracy of the LiDAR and the point cloud density. Using different sensor types like the Kinect v2, stereo vision or even a 3D laser could increase the number of points and the accuracy of the sensor output, which could result in a more precise plant clustering.

The biggest challenge of the algorithm settings was the discrimination between ground and plant points. Especially when the plants are small, the height difference was not enough to use a simple RANSAC plane filter. It could be helpful to discriminate ground and plant points not only by using a plane model but by taking the laser intensities or multiple echoes into account (Reymann et al. 2015). Another challenge was the density of the point clouds because the laser beams did not cover all parts of the leaves. This caused that some leaves were not clustered correctly to the right plant. More complex plant models should be used for the point cloud clustering. In addition, repeated driving in the same row and matching the data could help to increase the density of the point clouds.

In general, the algorithms could perform in other growth stages, as long as the plants could be separated in single point cloud clusters. In the described methods this is dependent on point cloud density and plant spacing. In the best case, the plants are completely isolated and can be easily distinguished from each other. As long as the objects of interest are clearly separated, but big enough to be hit by the laser beams, like in early stage maize, the described methods promise sufficient results. In other crops like cereals, this could be more challenging and have to be evaluated in future research. It is expected that when the plants are higher than the sensor position, no satisfying results could be obtained. The speed of the robot is a crucial parameter in order to gain adequate information from the laser scanner data. However, it is expected that driving faster would also increase the error. This could

be solved by using a laser sensor with a higher frequency or using a 3D sensor system for point cloud generation.

As the described methods do not take the shape of the objects into account, a discrimination between crops, weeds or other objects is not possible. For a reliable object detection of maize plants, additional algorithms such as machine-learning algorithms, or fusion with 2D image analysis could be used. Sensor redundancy could help to keep the detection rate close to 100%, even without manual parameter changes. By using other algorithms such as region growing, k-mean, graph-cut or by utilising normal orientations for the clustering, better results could be achieved with less dependency on perfectly aligned and dense point clouds (Reiser, Vázquez-Arellano, et al. 2017). To increase the reached accuracy, the stem position should be analysed with more complex models. A line fitting or a histogram-based approach could be used to estimate the stem position. To use the obtained information in a complex autonomous robotic system, the information could be converted into a suitable obstacle or cost map. This could be later used to guide an attached implement and to re-localise the robot in the field.

5.5 Conclusions

A tilted 2D LiDAR laser scanner was utilised to assemble 3D point clouds of maize plants in combination with a total station and an IMU. This point cloud generation setup allowed to process information and objects in front of the vehicle, which could be used for performing different applications and navigation purposes. Detecting individual plants with context information was investigated. The behaviour of the algorithms was considered under different growth stages. The results showed high improvement in the accuracy and robustness using context awareness. In a semi-structured and consistent environment, this could lead up to a 100% detection rate, better than the state-of-the-art technology, which does not include contextual information in the analysis. The contextualised Iterative Plant Clustering Method was accurate and reliable with an RMSE between 3.0 and 2.7 cm. The ground truth height measurements and the algorithm results had a maximum deviation of 3.3 cm for all examined datasets. Future work should deal with automatically determining the necessary parameters. A classification based on region growing or min-cut principles could increase the level of the algorithm robustness.

Acknowledgements

The project was conducted at the Max-Eyth Endowed Chair (Instrumentation & Test Engineering) at Hohenheim University (Stuttgart, Germany), which is partly grant funded by the Deutsche Landwirtschafts-Gesellschaft e.V. (DLG).

References

Ali Jan Ghasab, M., Khamis, S., Mohammad, F., & Jahani Fariman, H. (2015). Feature decision-making ant colony optimization system for an automated recognition of plant species. Expert Systems with Applications, 42(5), 2361–2370. doi:10.1016/j.eswa.2014.11.011

Andújar, D., Escolà, A., Rosell-Polo, J. R., Fernández-Quintanilla, C., & Dorado, J. (2013). Potential of a terrestrial LiDAR-based system to characterise weed vegetation in maize crops. Computers and Electronics in Agriculture, 92, 11–15. doi:10.1016/j.compag.2012.12.012

Arnó, J., Escolà, A., Vallès, J. M., Llorens, J., Sanz, R., Masip, J., et al. (2013). Leaf area index estimation in vineyards using a ground-based LiDAR scanner. Precision Agriculture, 14(3), 290–306. doi:10.1007/s11119-012-9295-0

Arribas, J. I., Sánchez-Ferrero, G. V., Ruiz-Ruiz, G., & Gómez-Gil, J. (2011). Leaf classification in sunflower crops by computer vision and neural networks. Computers and Electronics in Agriculture, 78(1), 9–18. doi:10.1016/j.compag.2011.05.007

Åstrand, B., & Baerveldt, A. (2004). Plant recognition and localization using context information. In IEEE Conference Mechatronics and Robotics 2004 special sessions Autonomous Machines in Agriculture (pp. 1191–1196). Aachen, Germany.

Auat Cheein, F. A., Guivant, J., Sanz, R., Escolà, A., Yandún, F., Torres-Torriti, M., & Rosell-Polo, J. R. (2015). Real-time approaches for characterization of fully and partially scanned canopies in groves. Computers and Electronics in Agriculture, 118, 361–371. doi:10.1016/j.compag.2015.09.017

Back, W. C., van Henten, E. J., & Edan, Y. (2014). Harvesting Robots for High-value Crops: State-of-the-art Review and Challenges Ahead. Journal of Field Robotics, 31(6), 888–911. doi:10.1002/rob

Balduzzi, M. A. F., van der Zande, D., Stuckens, J., Verstraeten, W. W., & Coppin, P. (2011). The properties of terrestrial laser system intensity for measuring leaf geometries: A case study with conference pear trees (Pyrus Communis). Sensors, 11(2), 1657–1681. doi:10.3390/s110201657

Bechar, A., & Vigneault, C. (2016). Agricultural robots for field operations : Concepts and components. Biosystems Engineering, 149, 94–111. doi:10.1016/j.biosystemseng.2016.06.014

Bentley, J. L. (1975). Multidimensional Binary Search Trees Used for Associative Searching. Communications of the ACM, 18(9), 509–517. doi:10.1145/361002.361007

Chéné, Y., Rousseau, D., Lucidarme, P., Bertheloot, J., Caffier, V., Morel, P., et al. (2012). On the use of depth camera for 3D phenotyping of entire plants. Computers and Electronics in Agriculture, 82, 122–127. doi:10.1016/j.compag.2011.12.007

De Rainville, F.-M., Durand, A., Fortin, F.-A., Tanguy, K., Maldague, X., Panneton, B., & Simard, M.-J. (2014). Bayesian classification and unsupervised learning for isolating weeds in row crops. Pattern Analysis and Applications, 17(2), 401–414. doi:10.1007/s10044-012-0307-5

Dey, D., Mummert, L., & Sukthankar, R. (2012). Classification of plant structures from uncalibrated image sequences. Proceedings of IEEE Workshop on Applications of Computer Vision, 329–336. doi:10.1109/WACV.2012.6163017

Dyrmann, M., Karstoft, H., & Skov, H. (2016). Plant species classification using deep convolutional neural network. Biosystems Engineering, 151(2005), 72–80. doi:10.1016/j.biosystemseng.2016.08.024

Escolà, A., Martinez-Casanovas, J. A., Rufat, J., Arnó, J., Arbonés, A., Sebé, F., et al. (2016). Mobile terrestrial laser scanner applications in precision fruticulture/horticulture and tools to extract information from canopy point clouds. Precision Agriculture. doi:10.1007/s11119-016-9474-5

Fischler, M. A., & Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM, 24, 381–395. doi:10.1145/358669.358692

Fountas, S., Blackmore, B. S., Vougioukas, S., Tang, L., Sørensen, C. G., & Jørgensen, R. (2007). Decomposition of agricultural tasks into robotic behaviours. Agricultural Engineering International: the CIGR Ejournal. Manuscript PM 07 006, IX.

Garrido, M., Paraforos, D., Reiser, D., Vázquez Arellano, M., Griepentrog, H., & Valero, C. (2015). 3D Maize Plant Reconstruction Based on Georeferenced Overlapping LiDAR Point Clouds. Remote Sensing, 7(12), 17077–17096. doi:10.3390/rs71215870

Garrido, M., Perez-Ruiz, M., Valero, C., Gliever, C. J., Hanson, B. D., & Slaughter, D. C. (2014). Active optical sensors for tree stem detection and classification in nurseries. Sensors (Basel, Switzerland), 14(6), 10783–10803. doi:10.3390/s140610783

Gleason, C. J., & Im, J. (2012). Forest biomass estimation from airborne LiDAR data using machine learning approaches. Remote Sensing of Environment, 125, 80–91. doi:10.1016/j.rse.2012.07.006

Gonzalez-de-Soto, M., Emmi, L., Perez-Ruiz, M., Aguera, J., & Gonzalez-de-Santos, P. (2016). Autonomous systems for precise spraying - Evaluation of a robotised patch sprayer. Biosystems Engineering, 6. doi:10.1016/j.biosystemseng.2015.12.018

Griepentrog, H. W., Norremark, M., Nielsen, H., & Blackmore, B. S. (2005). Seed mapping of sugar beet. Precision Agriculture, 6(2), 157–165. doi:10.1007/s11119-005-1032-5

Harper, N., & McKerrow, P. (2001). Recognising plants with ultrasonic sensing for mobile robot navigation. Robotics and Autonomous Systems, 34(2–3), 71–82. doi:10.1016/S0921-8890(00)00112-3

Haug, S., Biber, P., & Michaels, A. (2014). Plant Stem Detection and Position Estimation using Machine Vision. In Workshop Proceedings of IAS-13, ser. 13th Intl. Conf. on Intelligent Autonomous Systems (pp. 483–490).

Hiremath, S. A., van der Heijden, G. W. A. M., van Evert, F. K., Stein, A., & ter Braak, C. J. F. (2014). Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter. Computers and Electronics in Agriculture, 100, 41–50. doi:10.1016/j.compag.2013.10.005

Jiang, Y., Li, C., & Paterson, A. H. (2016). High throughput phenotyping of cotton plant height using depth images under field conditions. Computers and Electronics in Agriculture, 130, 57–68. doi:10.1016/j.compag.2016.09.017

Kusumam, K., Kranjík, T., Pearson, S., Cielniak, G., & Duckett, T. (2016). Can You Pick a Broccoli? 3D-Vision Based Detection and Localisation of Broccoli Heads in the Field. In IEEE International Conference on Intelligent Robots and Systems (pp. 1–6). doi:10.1109/IROS.2016.7759121

Lin, Y. (2015). LiDAR : An important tool for next-generation phenotyping technology of high potential for plant phenomics? Computers and Electronics in Agriculture, 119, 61–73. doi:10.1016/j.compag.2015.10.011

Owens, J. L., Osteen, P. R., & Daniilidis, K. (2015). MSG-Cal: Multi-sensor Graph-based Calibration. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 3660–3667). Hamburg.

Pedersen, S. M., Fountas, S., Have, H., & Blackmore, B. S. (2006). Agricultural robots—system analysis and economic feasibility. Precision Agriculture, 7(4), 295–308. doi:10.1007/s11119-006-9014-9

Pérez-Ruiz, M., Slaughter, D. C., Gliever, C. J., & Upadhyaya, S. K. (2012). Automatic GPS-based intra-row weed knife control system for transplanted row crops. Computers and Electronics in Agriculture, 80, 41–49. doi:10.1016/j.compag.2011.10.006

Reina, G., Milella, A., Nielsen, M., Worst, R., & Blas, M. R. (2015). Ambient awareness for agricultural robotic vehicles. Biosystems Engineering, (Robotic Agriculture), 1–19. doi:10.1016/j.biosystemseng.2015.12.010

Reiser, D., Garrido, M., Arellano, M. V., Griepentrog, H. W., & Paraforos, D. S. (2016). Crop Row Detection in Maize for Developing Navigation Algorithms under Changing Plant Growth Stages. In Advances in Intelligent Systems and Computing (Vol. 417, pp. 371–382). Lisbon: Springer. doi:10.1007/978-3-319-27146-0_29

Reiser, D., Martín-López, J., Memic, E., Vázquez-Arellano, M., Brandner, S., & Griepentrog, H. (2017). 3D Imaging with a Sonar Sensor and an Automated 3-Axes Frame for Selective Spraying in Controlled Conditions. Journal of Imaging, 3(1), 9. doi:10.3390/jimaging3010009

Reiser, D., Vázquez-Arellano, M., Izard, M. G., Paraforos, D. S., Sharipov, G., & Griepentrog, H. W. (2017). Clustering of Laser Scanner Perception Points of Maize Plants. Advances in Animal Biosciences, 8(2), 204–209. doi:10.1017/S204047001700111X

Reitberger, J., Krzystek, P., & Stilla, U. (2007). Combined tree segmentation and stem detection using full waveform lidar data. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, (iii), 332–337.

Reymann, C., Lacroix, S., Reymann, C., Lacroix, S., Lidar, I., Cloud, P., et al. (2015). Improving LiDAR Point Cloud Classification using Intensities and Multiple Echoes. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 5122–5128). Hamburg.

Ritchie, S., Hanway, J., & Benson, G. (1993). How A Corn Plant Develops. Special Report No. 48 (revised); Iowa State University of Science and Technology Cooperative Extension Service: Ames, IA, USA.

Rosell, J. R., Llorens, J., Sanz, R., Arno, J., Ribes-Dasi, M., Masipa, J., et al. (2009). Obtaining the three-dimensional structure of tree orchards from remote 2D terrestrial LIDAR scanning. Agricultural and Forest Meteorology, 149, 1505–1515. doi:10.1016/j.agrformet.2009.04.008

Rozsa, A., Günther, M., & Boult, T. E. (2016). Are Accuracy and Robustness Correlated? In Machine Learning and Applications (ICMLA), 2016 15th IEEE International Conference on. doi:10.1109/ICMLA.2016.0045

Rusu, R. B., Blodow, N., & Beetz, M. (2009). Fast Point Feature Histograms (FPFH) for 3D registration. 2009 IEEE International Conference on Robotics and Automation, 3212–3217. doi:10.1109/ROBOT.2009.5152473

Shrestha, D. S., Steward, B. L., & Birrell, S. J. (2004). Video processing for early stage maize plant detection. Biosystems Engineering, 89(2), 119–129. doi:10.1016/j.biosystemseng.2004.06.007

Sick AG Waldkirch. (2017). Operating Instructions LMS1xx. https://mysick.com/saqqara/im0031331.pdf. Accessed 11 January 2017

Steen, K. A., Christiansen, P., Karstoft, H., & Jørgensen, R. N. (2016). Using Deep Learning to Challenge Safety Standard for Highly Autonomous Machines in Agriculture. Journal of Imaging, 2–9. doi:10.3390/jimaging2010006

Strothmann, W., Ruckelshausen, A., Hertzberg, J., Scholz, C., & Langsenkamp, F. (2017). Plant classification with In-Field-Labeling for crop/weed discrimination using spectral features and 3D surface features from a multi-wavelength laser line profile system. Computers and Electronics in Agriculture, 134, 79–93. doi:10.1016/j.compag.2017.01.003

Sun, H., Slaughter, D. C., Ruiz, M. P., Gliever, C., Upadhyaya, S. K., & Smith, R. F. (2010). RTK GPS mapping of transplanted row crops. Computers and Electronics in Agriculture, 71(1), 32–37. doi:10.1016/j.compag.2009.11.006

Trimble. (2017). Trimble Universal Total Station. https://construction.trimble.com/sites/default/files/literature-files/2016-07/SPSx30-Data-Sheet-EN.pdf. Accessed 1 December 2017

Underwood, J. P., Jagbrant, G., Nieto, J. I., & Sukkarieh, S. (2015). Lidar-Based Tree Recognition and Platform Localization in Orchards. Journal of Field Robotics, 32(8), 1056–1074. doi:10.1002/rob

Vázquez-Arellano, M., Griepentrog, H. W., Reiser, D., & Paraforos, D. S. (2016). 3-D Imaging Systems for Agricultural Applications - A Review. Sensors, 16(618), 24. doi:10.3390/s16050618

Vectornav. (2017). VectorNav VN-100 IMU / AHRS High-Performance Embedded Navigation. https://www.vectornav.com/docs/default-source/documentation/vn-100-documentation/PB-12-0002.pdf?sfvrsn=9f9fe6b9_16. Accessed 1 December 2017

Vosselman, G., Gorte, B. G. H., Sithole, G., Rabbani, T., & Rabbanit. (2004). Recognising Structure in Laser Scanner Point Clouds. Information Sciences, 46, 1–6. doi:10.1002/bip.360320508

Weiss, U., & Biber, P. (2011). Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. Robotics and Autonomous Systems, 59(5), 265–273. doi:10.1016/j.robot.2011.02.011

Weiss, U., Biber, P., Laible, S., Bohlmann, K., & Zell, A. (2010). Plant species classification using a 3D LIDAR sensor and machine learning. Proceedings - 9th International Conference on Machine Learning and Applications, ICMLA 2010, 339–345. doi:10.1109/ICMLA.2010.57

CHAPTER 6

Paper E

Clustering of Laser Scanner Perception Points of Maize Plants [5]

David Reiser, Manuel Vázquez-Arellano, Miguel Garrido-Izard, Dimitris S. Paraforos, Galibjon Sharipov and Hans W. Griepentrog

Abstract

The goal of this work was to cluster maize plants perception points under six different growth stages in noisy 3D point clouds with known positions. The 3D point clouds were assembled with a 2D laser scanner mounted at the front of a mobile robot, fusing the data with the precise robot position, gained by a total station and an Inertial Measurement Unit. For clustering the single plants in the resulting point cloud, a graph-cut based algorithm was used. The algorithm results were compared with the corresponding measured values of plant height and stem position. An accuracy for the estimated height of 1.55 cm and the stem position of 2.05 cm was achieved.

Keywords: LiDAR, single plant detection, graph-cut, stem detection, phenotyping

6.1 Introduction

Precision farming is developing from big scale to small scale. Instead of considering the entire field, single plants and their status are getting into focus. With the continuing automation of processes, it could in future be possible to treat every plant individually, by measuring their behaviour and needs. This requires an accurate sensing system for plant shape and position and the possibility to localize

[5] The publication of Chapter 6 is done with the consent of the Cambridge University Press Publishing. The original publication was published in *Advances in Animal Bioscience, Precision Agriculture (ECPA) 2017,8:2.* It can be found under the following link: *https://doi.org/10.1017/S204047001700111X*

for every crossing the plants. Tasks like navigation, weeding, spraying, or estimating plant health status would benefit from this gained information. During sowing it is possible to map the position of every seed with real time kinematic global navigation satellite system receivers. This information would be precise enough for guidance of autonomous vehicles, but not sufficient for individual plant care (H. W. Griepentrog et al. 2005). However, when using this information in combination with sensor data, the precise position of the plant and the vehicle could be recalibrated.

Plants are elastic and shape changing objects, which are located in alternating environments. This makes perception with common sensors and algorithms a challenging task (Y. Zhang et al. 2016). Sunlight or shade strongly affects the sensor outputs (Bechar and Vigneault 2016b). Light detection and ranging (LiDAR) laser scanners are in general robust against sunlight and are not dependent on external light sources like passive sensor types (i.e. cameras, stereo cameras), what making it reasonable to use them for outdoor robotics (Vázquez-Arellano et al. 2016). LiDAR sensors measure the time of flight of a laser beam, reflected by an object. Every sensor output could be described as a perception point, defining the distance between sensor and object.

Using 3D instead of 2D data is recommended if the whole plant should be described with sensor data (Vázquez-Arellano et al. 2016). 3D-LiDARS are expensive, making them until now, unrealistic to provide affordable autonomous system solutions. Another method to gain 3D Data is to use a 2D laser and assemble 3D point clouds with the exact knowledge of the sensor position. This method could help to keep autonomous systems affordable (Escolà et al. 2016; Garrido et al. 2015). The most economical way would be to use the same sensor for navigation and plant phenotyping. This means that the sensor must look ahead of the machine, to navigate the robotic system. Using 3D point clouds of a 3D LiDAR for single plant detection, was already applied by Weiss and Biber (2011) using machine-learning and nearest neighbour classification methods. Also stem detection in point clouds was performed with different sensor types like stereo cameras, light curtains and LiDAR data (Bac et al. 2014; Garrido et al. 2014; Reitberger et al. 2007). Analysing the plant height with 3D sensor data is a well-known research topic for plant phenotyping, performed with a large variety of sensors (Y. Zhang et al. 2016). Today's variable-rate applications are performed using map-based or sensor based approaches. Nevertheless, if both methods are combined, enormous benefits could be brought together by reaching high accuracy (sensor-based approach) and high consistency (map-based approach) over long periods of time.

In the following work a graph-cut based method for the clustering of the remaining perception points of the plants is presented (Golovinskiy and Funkhouser 2009). This algorithm uses the knowledge of the object position as a reference to cluster a 3D point cloud with the use of weighted graphs and a

min-cut method. As reference parameters for the achieved precision and accuracy, the ground truth of the stem position and the maximal plant height were evaluated.

6.2 Materials and methods

6.2.1 Hardware and sensors

A small 4-wheel autonomous robot with differential steering was used as the carrier vehicle to move the sensors through the crop rows (see Figure 1a) (Reiser, Garrido, et al. 2016). A LMS111 2D-LiDAR laser scanner (SICK, Waldkirch, Germany) was used, mounted at a height of 0.58 m, above the ground, pointing downwards at an angle of 30 degrees. The sensor data was assembled with 25 Hz and an angle resolution of 0.5 degrees. This position was selected to allow 3D point cloud generation and at the same time to be able to navigate the robot system with the sensors through the rows. To measure the robot orientation, a VN-100 Inertial Measurement Unit (IMU) (VectorNav, Dallas, USA) was included in the sensor setup. The robot position was obtained through the use of a SPS930 Universal Total Station (Trimble, Sunnyvale, USA). The total station tracked a Trimble MT900 Machine Target Prism, which was mounted on top of the robot at a height of 1.07 m (see Figure 1a).

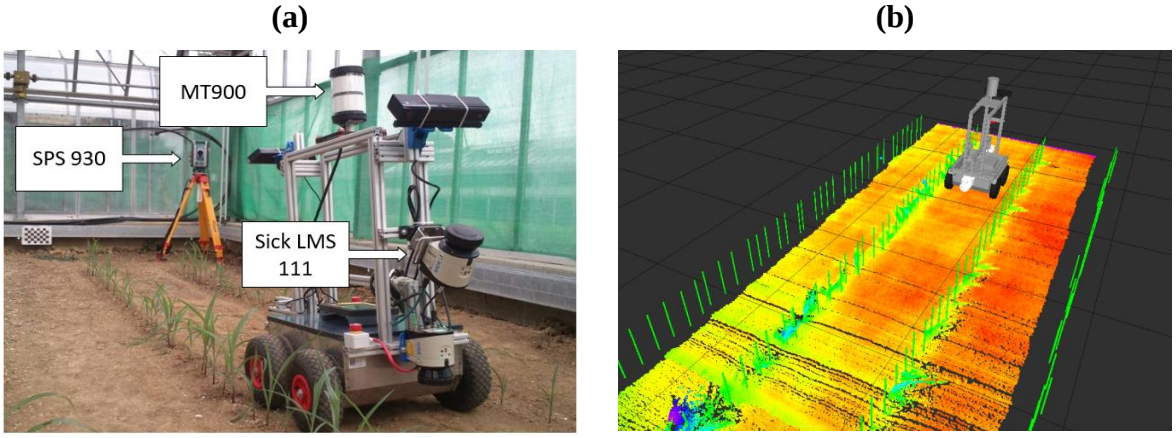

Figure 1: The robot platform for the data acquisition (a) and the visualization of the robot in the Robot Operating System (ROS) with one assembled point cloud and the ground truth of the plants as green sticks (b).

6.2.2 Software

The robot computer runs on Ubuntu 14.04 and uses the Robot Operating System (ROS-Indigo) middleware for sensor control and data recording. The system could be used for live monitoring of the sensor data and provided the necessary visualization tools (see Figure 1b). All the software components were programmed in a combination of C++ and Python programming languages. For fast calibration, point measurement and importing the total station data into ROS, the Trimble

SCS900 Site Controller (Software Version 3.4.0) graphical interface was used. The prism position data was time stamped and helped to refer the transforms to the global frame and to interpolate the data.

For the plant detection, a ROS node (executable-program) was developed, reading in the assembled point cloud and giving out the results of the algorithm. This implementation could be used directly on a real-time operating robot system. For the specific point cloud algorithms the PCL implementation (Rusu and Cousins 2011) was used and optimized for ROS.

6.2.3 Data acquisition and point cloud assembling

For referencing to the same Cartesian coordinate frame in every test, 5 fixed points were defined nearby the test area (Garrido et al. 2015; Reiser, Garrido, et al. 2016). To relocate these points for every test, a greenhouse with a solid concrete wall was selected for the data acquisition. The precise position of these 5 points could be located by just screwing a prism of the total station on fixed positions on the concrete wall. With these positions, the positioning system of the total station could be calibrated to one fixed coordinate frame. The inaccuracy in the static measurement could be estimated by reassessing each of these fixed points with the first measurement. The shift between the first reference points and the actual measurements was in all tests below 4 mm for all three dimensions. For the robot rigid body frame, carrying the sensors, a static transformation between the prism and the sensor position was assumed. First the roll, pitch and yaw angle of the IMU was fused together with the prism position and was used to create a coordinate frame for the prism position. After that, a static transformation to the robot geometric centre and to the sensor position was performed. This procedure allowed to track down the precise sensor position and orientation in the same reference frame in every test (Garrido et al. 2015; Reiser, Garrido, et al. 2016).

The spacing between the plants was defined by different Gaussian distributions for every crop row, to emulate diverse real scenarios. The rows used in this paper had a Gaussian distribution and a standard deviation of 0.02 m and 0.03 m for the spacing. In total 41 plants were planted per row. The ground truth positions of the plants were measured using the total station just after emergence with the help of a tripod.

In total six different growth stages were assessed in this paper, the first test 28 days and the last test 47 days after seeding. The average height of the plants changed in this time between 12.03 cm and 41.76 cm. The absolute plant height was varying between 5.7 and 45 cm. The real height of the plants was measured manually after every test day right after the data acquisition. The plants were between V1 and V6 stage, varying at every test day (Ritchie et al. 1993).

For the data acquisition the robot drove through the row always from the same side with an average speed of 0.02-0.04 m/s. All six tests were performed in the same row, in the same driving direction, towards the total station. The laser scanner was always at the front of the driving direction.

Before assembling the data of the LiDAR into a 3D point cloud, the single scans were filtered with a range filter, so that reflections of the vehicle and the greenhouse wall were removed from the sensor data. The limits of the points were set to a defined distance, so that just one row to the left and one to the right of the robot could be observed. Only the points in the range of 0.75 m to the left and 0.75 m to the right of the sensor position were considered. This filtered scans were transformed together with the fusion of the robot position, gained by the total station and the IMU orientation, into one coordinate system. With this new reference, all points could be transferred to one 3D point cloud, in one global world coordinate system. In Figure 2 the 3D assembled point cloud representations in six different grow stages are depicted. The colours represent the height value of the points.

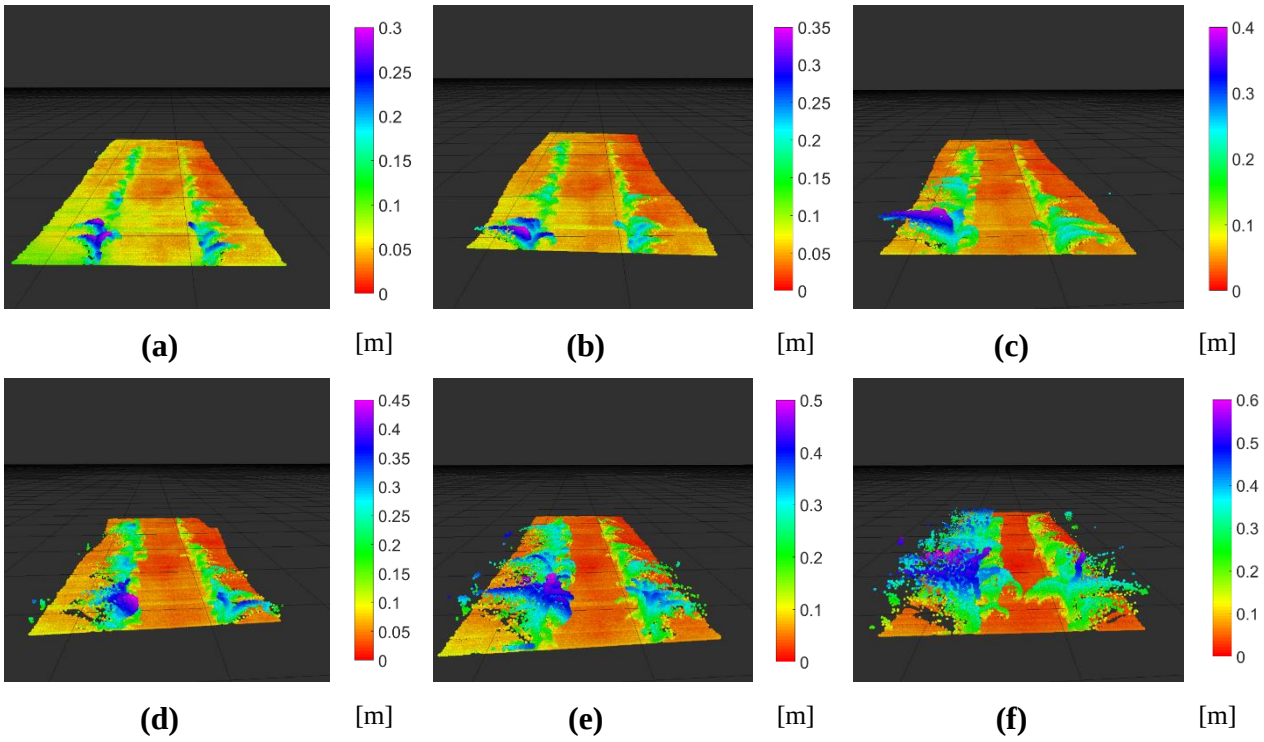

Figure 2: Assembled point clouds used for the algorithm with the six different assessed growth stages: (a) 26, (b) 28, (c) 32, (d) 35, (e) 40 and (f) 47 days after seeding.

6.2.4 Plant detection algorithm

To speed up the detection algorithm and get more precise results, the area of interest (AOI) was defined around the single particular plant, just taking one square meter around the plant into account. To test the limits of the algorithm, no noise reduction was applied. Limiting the AOI around the single particular plant, allowed to create a more precise estimation of the data points representing the soil.

As the shape of the ground was roughly planar in the data set, a random sample consensus (RANSAC) based plane fitting algorithm was used to remove the points of the ground (Fischler and Bolles 1981) and separate them from the plant points.

The plant points in the AOI were afterwards clustered to define point cloud groups of every single plant with the use of the ground truth of the plant position. A graph-cut based algorithm was used to cluster the plant points (Boykov and Funka-Lea 2006). This method fits perfect to the described problem, as it allows to separate foreground and background objects by using the known position of the object of interest. Compared to the use in 2D image analysis, the graph-cut based algorithm in point clouds cannot use colour information to define the nodes and edges of the graph. The only possible organizing methods for the graph are distances and densities between the points. Therefore, the point cloud was organized in a k-nearest neighbour graph, using the row direction as separator (Bentley 1975), correlating with the X-Axis in the point cloud reference frame (Reiser, Garrido, et al. 2016). Every point of the point cloud was defined as a node of the graph. This defined graph was then clustered with a min-cut in foreground and background points (Golovinskiy and Funkhouser 2009). For that, it is necessary to define sink and source points to set the edges, linking the nodes (points) of the graph. The source points define the assumed center of the object of interest. The sink points define background points. The edge value defines the weight that is used for the min-cut clustering. This weight decreases with the distance to the source point. The weight w_i of edge i is defined with the distance d_i to the source point and the additional fixed parameter σ.

$$w_i = e^{-\left(\frac{d_i}{\sigma}\right)^2} \quad (1)$$

In this case, the real plant position was used as source point with a predefined source weight as starting weight value. As sink points, all points with a distance radius area around 0.5 m around the real plant position were defined. The final segmentation was done by minimizing the cut cost of the nearest neighbour graph and the background penalty (Golovinskiy and Funkhouser 2009).

To assess the clustering, first the minima and maxima points of the gained point cloud were assessed. When the ground plane is shown with the point-normal form, the minimal distance h_i between a point $P_i(x_i, y_i, z_i)$ to a plane can be defined as:

$$h_i = ax_i + by_i + cz_i + d \quad (2)$$

With a, b, c, d defining the plane equation parameters. As the parameters a and b converged to zero and parameter c converged to one, the estimated height was approximated with $h = z + d$. The stem position was approximately estimated with the 3D centroid of all resulting points. The centroid c is correlated to the number of n points P in one point cloud cluster in Equation 3.

$$c = \frac{1}{n}\sum_{i=1}^{n} P_i \tag{3}$$

Just the x and y coordinates of the plants were considered for ground truth. The results of the algorithm for plant positions were compared to the measured positions by the total station to define the achieved precision. The height was compared with the manual measurements taken after every data acquisition. For assessing the accuracy of the plant pose and the height, the mean value $\bar{d}$, the standard deviation std_{dev} and the Root Mean Square Error (RMSE) were used:

$$\bar{d} = \frac{1}{N}\sqrt{(x_r - x_a)^2 + (y_r - ya)^2} \tag{4}$$

$$std_{dev} = \sqrt{\frac{1}{N} * \sum_{i=1}^{N} (d_a - \bar{d})^2} \tag{5}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\sqrt{(x_r - x_a)^2 + (y_r - ya)^2}\right)^2} \tag{6}$$

with $d_r(x_r, y_r)$ as ideal plant position measured by the total station and $d_a(x_a, y_a)$ as the resolved algorithm plant position in 2D and N as the number of assessed datasets.

6.3 Results and discussion

The algorithm was applied to six selected plants. Each was assessed in six different growth stages. For each row three plants were assessed. All plants were grouped in the row with highly overlapping leaves in the later growth stages (see Figure 2). For applying the algorithm, the settings of the adjustable parameters were fixed in all data sets. The only change between each assessment was the definition of the ideal plant position as source point. The used settings were σ = 0.5, radius = 0.5, source weight = 0.25 and the minimal neighbours acceptable with 5 points. The results of the plant 17 in row 2 are shown in the following Figure 3. The results of the stem detection and height estimation of the evaluated six plants with standard deviation and RMSE could be found in Table 1.

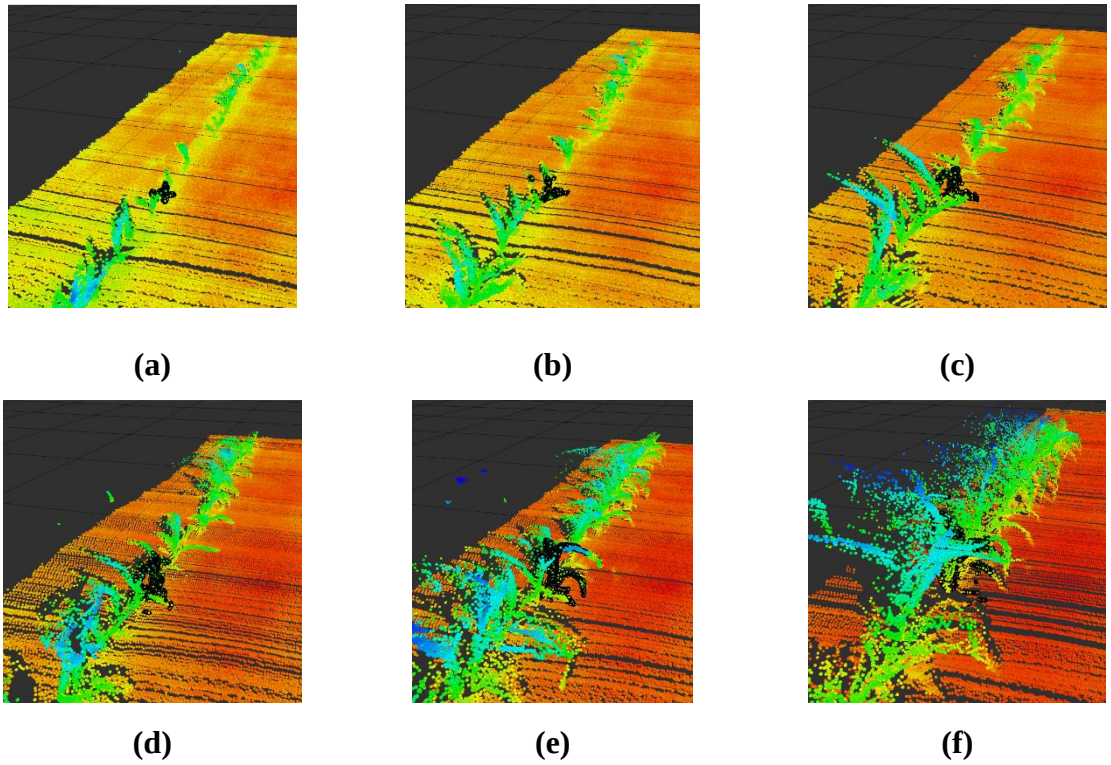

Figure 3: clustered points for plant 17 of row 2 in all six growth stages, black points are the clustered plant points: (a) 26, (b) 28, (c) 32, (d) 35, (e) 40 and (f) 47 days after seeding (V1-V6 (Ritchie et al. 1993)).

The achieved precision was accurate, so showed the mean value the best deviation of the stem position of 1.88 cm for all six growth stages and reached in some of the measurements even the position accuracy of 0.21 cm. The worst positioning error was 5.19 cm at plant 19 in row 3 what was caused by a partial covering of the plant of one big maize plant just 10 cm away from the examined plant. This caused the much higher RMSE for the stem distance. The poorly results for the plant 27 of row 3 are caused by a general offset to the ground truth, what could be caused by an error while measuring the plant position, or a special shape of the plant, what caused that the center of the plant did not fit with the stem position. Using a more complex model for the stem pose estimation in the point clouds and denser point clouds would bring improvements to the results. The RMSE height estimation ranged for the examined plants between 1.55 and 3.85 cm. The best estimations for single growth stages reached an accuracy of 1.1 mm for plant 33 in row 2 and plant 27 in row 3. The worst detection had an accuracy of 6 cm for one test in row 2 for plant 17. This was caused by not perfectly clustered point clouds, cutting off leaves at the top.

Table 1: Results of the plant-clustering algorithm combined over all six growth stages.

Plant position		Stem distance error [m]			Height error [m]		
row	no.	$\bar{d}$	std_{dev}	$RMSE$	$\bar{d}$	std_{dev}	$RMSE$
2	17	0.0188	0.0075	0.0202	0.0195	0.0332	0.0385
2	18	0.0216	0.0117	0.0245	-0.0062	0.0262	0.0269
2	33	0.0194	0.0088	0.0213	0.0015	0.0190	0.0190
3	18	0.0188	0.0116	0.0221	-0.0045	0.0288	0.0292
3	19	0.0264	0.0137	0.0297	0.0138	0.0249	0.0285
3	27	0.0318	0.0083	0.0328	0.0028	0.0152	0.0155

From visual feedback, the ground truth does not match completely with the 3D point cloud plant poses. This could be caused by the inaccuracy of the used LMS111 laser scanner depth information. In the manual the systematic error was described with +/- 30 mm with a maximum of +/- 50 mm (Sick AG Waldkirch 2017). When visually comparing the algorithm stem positions with the laser scanning data, the results seemed quite accurate. This inaccuracy in the sensor data showed, that the algorithm would be quite robust against not precise plant pose estimations. The algorithm results matched with the sensor data and not with the ground truth points. The algorithm worked well for detecting accurately the stem position of small plants, but it had the tendency to cut off small leaves and not clustering them, when they were too far apart from the assumed centre of the object.

Future work should evaluate how precise the plant estimation must be in order to detect precisely the plants and how a known plant spacing could help to perform better results. In addition, the evaluation of all plants in the row should be done to evaluate the robustness of the suggested algorithm with more data. To evaluate the quality of the point clouds and the clustering, the spatial correlation between the clustered point clouds and the leaf area and biomass change could be investigated.

6.4 Conclusions

In this work, a mobile robot was used for assembling 2D laser scanner data in six different growth stages of maize plants. The data was assembled to a 3D point cloud and clustered with a graph-cut based algorithm. In total, 6 different plants at 6 different growth stages with a varying height between 5.7 and 45 cm were assessed. It was shown that all plants could be clustered correctly, with the same algorithm settings, when using the previous known plant position. The results were better than the LiDAR sensor accuracy specifications, with accuracies for stem position of 2.02 cm and plant height estimation of 1.55 cm.

Acknowledgments

The project was conducted at the Max-Eyth Endowed Chair (Instrumentation & Test Engineering) at Hohenheim University (Stuttgart, Germany), which is partly grant funded by the Deutsche Landwirtschafts-Gesellschaft e.V. (DLG).

References

Bac, C. W., Hemming, J., & Henten, E. J. Van. (2014). Stem localization of sweet-pepper plants using the support wire as a visual cue. Computers and Electronics in Agriculture, 105, 111–120. doi:10.1016/j.compag.2014.04.011

Bechar, A., & Vigneault, C. (2016). Agricultural robots for field operations : Concepts and components. Biosystems Engineering, 9. doi:10.1016/j.biosystemseng.2016.06.014

Bentley, J. L. (1975). Multidimensional Binary Search Trees Used for Associative Searching. Communications of the ACM, 18(9), 509–517. doi:10.1145/361002.361007

Boykov, Y., & Funka-Lea, G. (2006). Graph Cuts and Efficient N-D Image Segmentation. International Journal of Computer Vision, 70(2), 109–131. doi:10.1007/s11263-006-7934-5

Escolà, A., Martinez-Casanovas, J. A., Rufat, J., Arnó, J., Arbonés, A., Sebé, F., et al. (2016). Mobile terrestrial laser scanner applications in precision fruticulture/horticulture and tools to extract information from canopy point clouds. Precision Agriculture. doi:10.1007/s11119-016-9474-5

Fischler, M. A., & Bolles, R. C. (1981). Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM, 24, 381–395. doi:10.1145/358669.358692

Garrido, M., Paraforos, D., Reiser, D., Vázquez Arellano, M., Griepentrog, H., & Valero, C. (2015). 3D Maize Plant Reconstruction Based on Georeferenced Overlapping LiDAR Point Clouds. Remote Sensing, 7(12), 17077–17096. doi:10.3390/rs71215870

Garrido, M., Perez-Ruiz, M., Valero, C., Gliever, C. J., Hanson, B. D., & Slaughter, D. C. (2014). Active optical sensors for tree stem detection and classification in nurseries. Sensors (Basel, Switzerland), 14(6), 10783–10803. doi:10.3390/s140610783

Golovinskiy, A., & Funkhouser, T. (2009). Min-Cut Based Segmentation of Point Clouds. In: IEEE Workshop on Search in 3D and Video (S3DV) at ICCV. doi:10.1109/ICCVW.2009.5457721

Griepentrog, H. W., Norremark, M., Nielsen, H., & Blackmore, B. S. (2005). Seed mapping of sugar beet. Precision Agriculture, 6(2), 157–165. doi:10.1007/s11119-005-1032-5

Reiser, D., Izard, M. G., Arellano, M. V., Griepentrog, H. W., & Paraforos, D. S. (2016). Crop Row Detection in Maize for Developing Navigation Algorithms under Changing Plant Growth Stages. In: Advances in Intelligent Systems and Computing (Vol. 417, pp. 371–382). Lisbon: Springer. doi:10.1007/978-3-319-27146-0_29

Reitberger, J., Krzystek, P., & Stilla, U. (2007). Combined tree segmentation and stem detection using full waveform lidar data. International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, (iii), 332–337.

Ritchie, S., Hanway, J., & Benson, G. (1993). How A Corn Plant Develops. Special Report No. 48 (revised); Iowa State University of Science and Technology Cooperative Extension Service: Ames, IA, USA.

Rusu, R. B., & Cousins, S. (2011). 3D is here: point cloud library. In: IEEE International Conference on Robotics and Automation, 1–4. doi:10.1109/ICRA.2011.5980567

Sick AG Waldkirch. (2016). Operating Instructions LMS1xx. https://mysick.com/saqqara/im0031331.pdf. Accessed 11 January 2017

Vázquez-Arellano, M., Griepentrog, H. W., Reiser, D., & Paraforos, D. S. (2016). 3-D Imaging Systems for Agricultural Applications - A Review. Sensors, 16(618), 24. doi:10.3390/s16050618

Weiss, U., & Biber, P. (2011). Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. Robotics and Autonomous Systems, 59(5), 265–273. doi:10.1016/j.robot.2011.02.011

Zhang, Y., Teng, P., Shimizu, Y., Hosoi, F., & Omasa, K. (2016). Estimating 3D Leaf and Stem Shape of Nursery Paprika Plants by a Novel Multi-Camera. Sensors (Basel, Switzerland), 16(874), 1–18. doi:10.3390/s16060874

CHAPTER 7

Discussion

This thesis conducts to the perception of agricultural robots to perform better context awareness within fields. In total five aspects were addressed: (I) static local sensor communication with a mobile vehicle, (II) detect unstructured objects in a controlled environment, (III) describe the influence of growth stage to algorithm outcomes, (IV) use the gained sensor information to detect single plants and (V) improve the robustness of algorithms under noisy conditions.
The following discussion addresses the findings and highlights the key points of the thesis.

7.1 Static local sensor communication with a mobile vehicle

For the first aspect of perception for context awareness, four wireless sensors were placed in a vineyard. Besides communicating with a receiver node, they were able to send data from an attached sensor to gain information about the environment. To use this information (e.g. temperature, moisture, pH) they had to be localized and the information must be set in the right context. The use of static sensors brings advantages for context awareness and could be used for precision farming purposes. With these “Internet of Things” (IoT) sensors, the environment could communicate with other technical devices like computers, mobile phones, tractors or autonomous robots. The information is dependent on the position where the sensors were placed (Bloem et al. 2014). Therefore, the user needs to define the right context, by placing the static IoT devices at specific points. The intelligence of the system is limited to static points of interests and the specific knowledge about the received sensor data. In the showed use case, a farmer would be responsible for placing the sensors at relevant locations. For understanding the sensor data, context awareness and location of the IoT system have to be provided. The exact location of the data could be assessed by the system automatically, as it was shown in the experiment. However, the user must provide the necessary context awareness. With these IoT sensors, it could be possible to realize more detailed path planning and reasoning, taking sensor values like moisture into account. For example, moisture in steep vineyards could be an interesting parameter to define if the soil and grass have enough friction to enable the vehicle to pass the steep areas, or if a different route has to be used. Even the soil compaction caused by a machine could be estimated with actual moisture measurements. However, the generated information is fixed to one spot, what makes it hard for covering big areas with a small grid size. As for autonomous

vehicles, the direct environment and the behaviour of fast changes are interesting, this method of static Wireless Sensor Networks is not sufficient. It is necessary to observe the environment with additional optical sensors simultaneously. This helps to react to direct environment changes and obstacles in front of the robot. In the described experiment, the navigation was based on a deterministic RTK-GNSS signal, combined with a direct reactive obstacle control for avoiding the trunks in the vineyard.

7.2 Detect unstructured objects in a controlled environment

The basic behaviour for robot navigation is to order the environment in two classes: free space and obstacles (Hertzberg et al. 2012). Out of the perception information of objects in the environment, a path or task could be planned and optimized (Seraji and Howard 2002). This works fine for wide and clear structures as they could be found indoors, or on a fine prepared vineyard, as it was addressed in the first paper. To increase the level of context awareness, sensor perceptions should be classified to objects, so that the system really “knows”, what is around. This can help to make more advanced decisions, leading to the implementation of self-awareness in autonomous machines (Gorbenko et al. 2012).

The task in the agriculture domain is to deal with unstructured objects in unstructured environments (Bechar and Vigneault 2016b). The objects could be for example crop plants, weed plants or soil. To distinguish first between plants and soil, this experiment addressed unstructured objects in a structured (indoor) environment with a well-known sensor position, to focus on the main aspect of object detection.

For dealing with objects in our world, it is necessary to gain three-dimensional knowledge about them, what is one big aspect of setting the sensor information into the right context. 1D or 2D sensor data could just be used for dealing with 3D objects, if the precise context of the sensor position and the environment is known. Therefore a 3D information is providing more options for environment perception and context awareness for later applications (Vázquez-Arellano et al. 2016). For gaining relative or absolute sensor positions, it is possible to use a precise positioning system to set the information in the right context. In this case, a 1D sonar sensor was used together with a highly precise 3D frame to create a 3D point cloud out of an artificial crop row. The object classification was done by detecting the planar shape (soil) and the space between the remaining points (plants). This classification uses known parameters of the semi-structured agricultural environment, instead of integrating machine-learning algorithms to detect structures in an unknown context. When you see just a small part of a picture without any context, it is even hard for a human to detect, what the image is showing. However, with the right context, the detection of objects and interpretation of the data is

a solvable task. As long as the context is known, the object classification could be performed easier. Therefore, more robust and more reliable algorithms could be developed, when the context of the sensor perception is known. However, how does it work, when the objects are not static, but changing their shape over time? This changing of unstructured objects in agriculture and their effect on algorithms were addressed in the next paper.

7.3 Influence of growth stage to algorithm outcomes

Describing the change of unstructured objects and their effect on algorithms is a challenging task for autonomous machines (Chen and Cournede 2012). Therefore, it was important to find an adequate way to describe the change of plants over different growth stages. There are two typical ways to describe unstructured objects like plants: create a model and simulate the data, or to measure real plants with sensor values in a realistic environment (Weiss and Biber 2011). In our case, the measuring of real plants was considered.

For recording real data sets, maize was planted in a greenhouse and the positions of every single crop was measured with a highly accurate total station. This total station was also used to track down a vehicle with mounted sensors. Afterwards the data could be tested with different algorithms for the performance under different growth stages. The performance of the tested line-fitting algorithm was highly affected by the different growth stages. In later growth stages, leaves covered the row, causing algorithm failure.

Another issue is, that in later growth stages the plants touch the robot from time to time. For robust performance the robot has to decide, if touching the leaves is acceptable, or if the sensor readings define an obstacle. This leaves would make a standard obstacle detection with a LiDAR challenging, as the system could not separate leaves in the row or obstacles placed in front of the robot. Here context awareness is necessary to keep the system capable of acting. An intelligent robot system would decide, if a collision would harm the robot or if a touch of the robot will harm the plant. It is challenging to achieve this goal, but it can be avoided by using a different robot operating structure. An easy way for getting the system work under the described conditions, is to use a mode changer (Griepentrog et al. 2006; Vougioukas et al. 2004). As the system could change the actual status of the program parameters and algorithms based on outer states (e.g. in row navigation, headland turning), the sensor readings could be set in the right context. When performing in row navigation, obstacles could be ignored and the navigation could be performed by detecting line structures. When the actual mode is headland turning, rigid obstacles could be expected.

A more advanced way for environment perception would be to detect single plants and to use this information for navigation. For robust navigation, the plant positions needed to be known, so that it

could be possible to distinguish between leaves and other obstacles and where the robot is allowed to drive without destroying the plants. For e.g. precision farming, mechanical weeding and precision spraying this information could be used to improve the performance (Tillett et al. 2002).

7.4 Use the gained sensor information to detect single plants

Single plant detection is still a challenging task. There are many different approaches for detecting plants, starting from camera detection to light curtains and tactile systems (Åstrand and Baerveldt 2002; Garrido et al. 2014; Ge et al. 2013; Pérez-Ruiz et al. 2012; Vázquez-Arellano et al. 2016). Anyhow, most systems were only used for single plant detection and not for simultaneous navigation. The proposed approach used a tilted laser scanner in a position suitable for navigation, to create an assembled 3D point cloud. The resulting point cloud could be used simultaneously for detecting objects and to classify single plants.

Since the environment in fields is changing rapidly, the approach tried to focus on the known parameters and used them in combination with the LiDAR. The single plant detection was performed with the knowledge of the theoretical plant position, which could be a known parameter in a semi-structured environment on an agricultural field. As the line width and the spacing between the plants were fixed, it was possible to limit the search for the plant. Without using context information like row width and theoretical plant position, a detection rate of 70% was reached. By using the context information, it was possible to detect all plants in the test area.

The semi-structured environment of crop rows helped to gain better results for the detection of unstructured plant objects, which could be the first step for increasing the context awareness of vehicles on fields. Especially the information about plant health, exact plant position and plant shape could help to improve many different applications for agricultural robots (Chéné et al. 2012; Ruckelshausen et al. 2009). This information could be used for any kind of single plant treatment, advanced navigation and advanced obstacle detection. The next step would be to show how the approach could be made more robust to work even under noisy sensor conditions.

7.5 Improve the robustness of algorithms under noisy conditions

Robustness of algorithms is one of the most challenging tasks for autonomous applications in agriculture. Especially the changing conditions of shape and illumination could make it hard to detect the position of single plants (Vázquez-Arellano et al. 2016). The goal was to find a parameter adjustment, which would work over the measured growth stages without any changes.

The research for a robust and usable algorithm resulted in a graph cut based method. It reached 100% detection rate, without any adjustment, overall measured growth stages. The general function of this

algorithm was using the weight of points to define the point cloud clusters. In combination with the assumed position of a plant, it was possible to assess the points of the plant quite accurate in a crop row, even when the point clouds were overlapping and plants were not clearly visible.

The knowledge of plant positions could help to improve perception and context awareness of machines. To reach a higher state of autonomy and algorithm robustness, the focus on the known aspects seem to be more promising besides trying to deal with all uncertainties in agriculture.

7.6 Outlook

The thesis demonstrated different approaches for context awareness of agricultural robots. The implementation of the approaches could lead to the next step of advanced farming with autonomous or semi-autonomous machines. The realisation of future machines must combine different strategies for the perception of context awareness. Similar to sensor fusion, multiple sensors and approaches for context awareness could make the object detection and environment perception more robust. Future work has to address the changes in objects in more detail and should try to adapt the algorithms in combination with an application to show the use of these methodologies.

Companies in agriculture could in future focus more seriously on small autonomous robots than on big machines guided by one operator. Small autonomous machines combine the positive aspects of less risk with a high autonomy and saving labour. This could help to keep farming in developed countries competitive on the world market, could fulfil the need for organic food and could make farming ready for future law regulations because of health and environment issues.

Even for developing countries, context-aware robots could bring advantages, as the machines can provide knowledge directly to the farmer. In addition, the trend of producing your own food could be realized with small autonomous robots, what would give individual persons the option to produce their food by themselves.

References

Åstrand, B., & Baerveldt, A. J. (2002). An agricultural mobile robot with vision-based perception for mechanical weed control. Autonomous Robots, 13(1), 21–35. doi:10.1023/A:1015674004201

Bechar, A., & Vigneault, C. (2016). Agricultural robots for field operations: Concepts and components. Biosystems Engineering, 149, 94–111. doi:10.1016/j.biosystemseng.2016.06.014

Bloem, J., Doorn, M. van, Duivestein, S., Excoffier, D., Maas, R., & Ommeren, E. van. (2014). The Fourth Industrial Revolution Things to Tighten the Link Between it and ot. VINT research report, 1–39.

Chen, Y., & Cournede, P. (2012). Assessment of Parameter Uncertainty in Plant Growth Model Identification. In: IEEE 4h International Symposium on Plant Growth Modeling, Simulation, Visualization and Applications (pp. 85–92).

Chéné, Y., Rousseau, D., Lucidarme, P., Bertheloot, J., Caffier, V., Morel, P., et al. (2012). On the use of depth camera for 3D phenotyping of entire plants. Computers and Electronics in Agriculture, 82, 122–127. doi:10.1016/j.compag.2011.12.007

Garrido, M., Perez-Ruiz, M., Valero, C., Gliever, C. J., Hanson, B. D., & Slaughter, D. C. (2014). Active optical sensors for tree stem detection and classification in nurseries. Sensors (Basel, Switzerland), 14(6), 10783–10803. doi:10.3390/s140610783

Ge, Z., Wu, W., Yu, Y., & Zhang, R. (2013). Design of Mechanical Arm for Laser Weeding Robot. In: Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013) (pp. 2340–2343). doi:10.2991/iccsee.2013.586

Gorbenko, A., Popov, V., & Sheka, A. (2012). Robot Self-Awareness: Exploration of Internal States. Applied Mathematical Sciences, 6(14), 675–688.

Griepentrog, H. W., Blackmore, S., & Vougioukas, S. G. (2006). Positioning and Navigation. In A. Munack (Ed.), CIGR Handbook of Agricultural Engineering (pp. 195–204). American Society of Agricultural and Biological Engineers.

Hertzberg, J., Lingemann, K., & Nüchter, A. (2012). Lokalisierung in Karten. In Mobile Roboter (pp. 155–220). Berlin: Springer Berlin Heidelberg.

Pérez-Ruiz, M., Slaughter, D. C., Gliever, C. J., & Upadhyaya, S. K. (2012). Automatic GPS-based intra-row weed knife control system for transplanted row crops. Computers and Electronics in Agriculture, 80, 41–49. doi:10.1016/j.compag.2011.10.006

Ruckelshausen, A., Biber, P., Dorna, M., Gremmes, H., Klose, R., Linz, A., et al. (2009). BoniRob: an autonomous field robot platform for individual plant phenotyping. In: Proceedings of Joint International Agricultural Conference (2009) (Vol. 9, pp. 841–847). doi:10.3920/978-90-8686-664-9

Seraji, H., & Howard, A. (2002). Behavior-based robot navigation on challenging terrain: A fuzzy logic approach. IEEE Transactions on Robotics and Automation, 18(3), 308–321. doi:10.1109/TRA.2002.1019461

Tillett, N. D., Hague, T., & Miles, S. J. (2002). Inter-row vision guidance for mechanical weed control in sugar beet. Computers and Electronics in Agriculture, 33(3), 163–177. doi:10.1016/S0168-1699(02)00005-4

Vázquez-Arellano, M., Griepentrog, H. W., Reiser, D., & Paraforos, D. S. (2016). 3-D Imaging Systems for Agricultural Applications - A Review. Sensors, 16(618), 24. doi:10.3390/s16050618

Vougioukas, S., Fountas, S., Blackmore, B. S., & Tang, L. (2004). Navigation Task in Agricultural Robotics. In International conference on information systems and innovative technologies in agriculture, food and environment. Thessaloniki, Greece (pp. 55–64).

Weiss, U., & Biber, P. (2011). Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. Robotics and Autonomous Systems, 59(5), 265–273. doi:10.1016/j.robot.2011.02.011

Summary

Context awareness is one key point for the realisation of robust autonomous systems in unstructured environments like agriculture. Robots need a precise description of their environment so that tasks could be planned and executed correctly. When using a robot system in a controlled, not changing environment, the programmer maybe could model all possible circumstances to get the system reliable. However, the situation gets more complex when the environment and the objects are changing their shape, position or behaviour. Perception for context awareness in agriculture means to detect and classify objects of interest in the environment correctly and react to them.
The aim of this cumulative dissertation was to apply different strategies to increase context awareness with perception in mobile robots in agriculture. The objectives of this thesis were to address five aspects of environment perception: (I) test static local sensor communication with a mobile vehicle, (II) detect unstructured objects in a controlled environment, (III) describe the influence of growth stage to algorithm outcomes, (IV) use the gained sensor information to detect single plants and (V) improve the robustness of algorithms under noisy conditions.

First, the communication between a static Wireless Sensor Network and a mobile robot was investigated. The wireless sensor nodes were able to send local data from sensors attached to the systems. The sensors were placed in a vineyard and the robot followed automatically the row structure to receive the data. It was possible to localize the single nodes just with the exact robot position and the attenuation model of the received signal strength with triangulation. The precision was 0.6 m and more precise than a provided differential global navigation satellite system signal.
The second research area focused on the detection of unstructured objects in point clouds. Therefore, a low-cost sonar sensor was attached to a 3D-frame with millimetre level accuracy to exactly localize the sensor position. With the sensor position and the sensor reading, a 3D point cloud was created. In the workspace, 10 individual plant species were placed. They could be detected automatically with an accuracy of 2.7 cm. An attached valve was able to spray these specific plant positions, which resulted in a liquid saving of 72%, compared to a conventional spraying method, covering the whole crop row area.
As plants are dynamic objects, the third objective of describing the plant growth with adequate sensor data, was important to characterise the unstructured agriculture domain. For revering and testing algorithms to the same data, maize rows were planted in a greenhouse. The exact positions of all plants were measured with a total station. Then a robot vehicle was guided through the crop rows and the data of attached sensors were recorded. With the help of the total station, it was possible to track down the vehicle position and to refer all data to the same coordinate frame. The data recording was

performed over 7 times over a period of 6 weeks. This created datasets could afterwards be used to assess different algorithms and to test them against different growth changes of the plants. It could be shown that a basic RANSAC line following algorithm could not perform correctly under all growth stages without additional filtering.

The fourth paper used this created datasets to search for single plants with a sensor normally used for obstacle avoidance. One tilted laser scanner was used with the exact robot position to create 3D point clouds, where two different methods for single plant detection were applied. Both methods used the spacing to detect single plants. The second method used the fixed plant spacing and row beginning, to resolve the plant positions iteratively. The first method reached detection rates of 73.7% and a root mean square error of 3.6 cm. The iterative second method reached a detection rate of 100% with an accuracy of 2.6 - 3.0 cm.

For assessing the robustness of the plant detection, an algorithm was used to detect the plant positions in six different growth stages of the given datasets. A graph-cut based algorithm was used, what improved the results for single plant detection. As the algorithm was not sensitive against overlaying and noisy point clouds, a detection rate of 100% was realised, with an accuracy for the estimated height of the plants with 1.55 cm. The stem position was resolved with an accuracy of 2.05 cm.

This thesis showed up different methods of perception for context awareness, which could help to improve the robustness of robots in agriculture. When the objects in the environment are known, it could be possible to react and interact smarter with the environment as it is the case in agricultural robotics. Especially the detection of single plants before the robot reaches them could help to improve the navigation and interaction of agricultural robots.

Zusammenfassung

Kontextwahrnehmung ist eine Schlüsselfunktion für die Realisierung von robusten autonomen Systemen in einer unstrukturierten Umgebung wie der Landwirtschaft. Roboter benötigen eine präzise Beschreibung ihrer Umgebung, so dass Aufgaben korrekt geplant und durchgeführt werden können. Wenn ein Roboter System in einer kontrollierten und sich nicht ändernden Umgebung eingesetzt wird, kann der Programmierer möglicherweise ein Modell erstellen, welches alle möglichen Umstände einbindet, um ein zuverlässiges System zu erhalten. Jedoch wird dies komplexer, wenn die Objekte und die Umwelt ihr Erscheinungsbild, Position und Verhalten ändern. Umgebungserkennung für Kontextwahrnehmung in der Landwirtschaft bedeutet relevante Objekte in der Umgebung zu erkennen, zu klassifizieren und auf diese zu reagieren.

Ziel dieser kumulativen Dissertation war, verschiedene Strategien anzuwenden, um das Kontextbewusstsein mit Wahrnehmung bei mobilen Robotern in der Landwirtschaft zu erhöhen. Die Ziele dieser Arbeit waren fünf Aspekte von Umgebungserkennung zu adressieren: (I) Statische lokale Sensorkommunikation mit einem mobilen Fahrzeug zu testen, (II) unstrukturierte Objekte in einer kontrollierten Umgebung erkennen, (III) die Einflüsse von Wachstum der Pflanzen auf Algorithmen und ihre Ergebnisse zu beschreiben, (IV) gewonnene Sensorinformation zu benutzen, um Einzelpflanzen zu erkennen und (V) die Robustheit von Algorithmen unter verschiedenen Fehlereinflüssen zu verbessern.

Als erstes wurde die Kommunikation zwischen einem statischen drahtlosen Sensor-Netzwerk und einem mobilen Roboter untersucht. Die drahtlosen Sensorknoten konnten Daten von lokal angeschlossenen Sensoren übermitteln. Die Sensoren wurden in einem Weingut verteilt und der Roboter folgte automatisch der Reihenstruktur, um die gesendeten Daten zu empfangen. Es war möglich, die Sendeknoten mithilfe von Triangulation aus der exakten Roboterposition und eines Sendesignal-Dämpfung-Modells zu lokalisieren. Die Genauigkeit war 0.6 m und somit genauer als das verfügbare Positionssignal eines „differential global navigation satellite system“.

Der zweite Forschungsbereich fokussierte sich auf die Entdeckung von unstrukturierten Objekten in Punktewolken. Dafür wurde ein kostengünstiger Ultraschallsensor auf einen 3D Bewegungsrahmen mit einer Millimeter Genauigkeit befestigt, um die genaue Sensorposition bestimmen zu können. Mit der Sensorposition und den Sensordaten wurde eine 3D Punktewolke erstellt. Innerhalb des Arbeitsbereichs des 3D Bewegungsrahmens wurden 10 einzelne Pflanzen platziert. Diese konnten automatisch mit einer Genauigkeit von 2.7 cm erkannt werden. Eine angebaute Pumpe ermöglichte das punktuelle Besprühen der spezifischen Pflanzenpositionen, was zu einer Flüssigkeitsersparnis von 72%, verglichen mit einer konventionellen Methode welche die gesamte Pflanzenfläche benetzt, führte.

Da Pflanzen sich ändernde Objekte sind, war das dritte Ziel das Pflanzenwachstum mit geeigneten Sensordaten zu beschreiben, was wichtig ist, um unstrukturierte Umgebung der Landwirtschaft zu charakterisieren. Um Algorithmen mit denselben Daten zu referenzieren und zu testen, wurden Maisreihen in einem Gewächshaus gepflanzt. Die exakte Position jeder einzelnen Pflanze wurde mit einer Totalstation gemessen. Anschließend wurde ein Roboterfahrzeug durch die Reihen gelenkt und die Daten der angebauten Sensoren wurden aufgezeichnet. Mithilfe der Totalstation war es möglich, die Fahrzeugposition zu ermitteln und alle Daten in dasselbe Koordinatensystem zu transformieren. Die Datenaufzeichnungen erfolgten 7-mal über einen Zeitraum von 6 Wochen. Diese generierten Datensätze konnten anschließend benutzt werden, um verschiedene Algorithmen unter verschiedenen Wachstumsstufen der Pflanzen zu testen. Es konnte gezeigt werden, dass ein Standard RANSAC Linien Erkennungsalgorithmus nicht fehlerfrei arbeiten kann, wenn keine zusätzliche Filterung eingesetzt wird.

Die vierte Publikation nutzte diese generierten Datensätze, um nach Einzelpflanzen mithilfe eines Sensors zu suchen, der normalerweise für die Hinderniserkennung benutzt wird. Ein gekippter Laserscanner wurde zusammen mit der exakten Roboterposition benutzt, um eine 3D Punktewolke zu generieren. Zwei verschiedene Methoden für Einzelpflanzenerkennung wurden angewendet. Beide Methoden nutzten Abstände, um die Einzelpflanzen zu erkennen. Die zweite Methode nutzte den bekannten Pflanzenabstand und den Reihenanfang, um die Pflanzenpositionen iterativ zu erkennen. Die erste Methode erreichte eine Erkennungsrate von 73.7% und damit einen quadratischen Mittelwertfehler von 3.6 cm. Die iterative zweite Methode erreichte eine Erkennungsrate von bis zu 100% mit einer Genauigkeit von 2.6-3.0 cm.

Um die Robustheit der Pflanzenerkennung zu bewerten, wurde ein Algorithmus zur Erkennung von Einzelpflanzen in sechs verschiedenen Wachstumsstufen der Datasets eingesetzt. Hier wurde ein „graph-cut“ basierter Algorithmus benutzt, welcher die Robustheit der Ergebnisse für die Einzelpflanzenerkennung erhöhte. Da der Algorithmus nicht empfindlich gegen ungenaue und fehlerhafte Punktewolken ist, wurde eine Erkennungsrate von 100% mit einer Genauigkeit von 1.55 cm für die Höhe der Pflanzen erreicht. Der Stiel der Pflanzen wurde mit einer Genauigkeit von 2.05 cm erkannt.

Diese Arbeit zeigte verschiedene Methoden für die Erkennung von Kontextwahrnehmung, was helfen kann, um die Robustheit von Robotern in der Landwirtschaft zu erhöhen. Wenn die Objekte in der Umwelt bekannt sind, könnte es möglich sein, intelligenter auf die Umwelt zu reagieren und zu interagieren, wie es aktuell der Fall in der Landwirtschaftsrobotik ist. Besonders die Erkennung von Einzelpflanzen bevor der Roboter sie erreicht, könnte helfen die Navigation und Interaktion von Robotern in der Landwirtschaft verbessern.

Author’s declaration

I hereby declare that this doctoral dissertation is a result of my own work, and that no other than the indicated aids have been used for its completion. All sources of information I exploited have been cited appropriately.

Furthermore, I assure that the work has not been used, neither completely nor in parts, for achieving any other academic degree.

David Reiser

Stuttgart-Hohenheim, March 2018