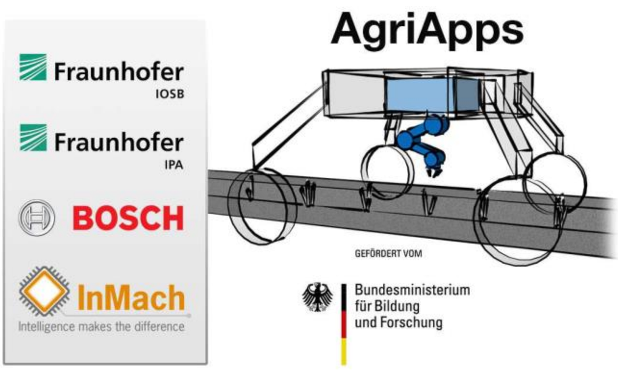
AgriApps –
An App-based Solution for Field-Robot-Based Agriculture

Christian Frese, **Christian Frey**, Fraunhofer IOSB, Karlsruhe
Felix Meßmer, **Kai Pfeiffer**, Fraunhofer IPA, Stuttgart
Slawomir Sander, **Daniel Di Marco**, **Maximilian Wenger**, **Amos Albert**, Robert Bosch Start-up GmbH, Renningen
Manuel Wopfner, **Alexander Burkhardt**, **Siegfried Hochdorfer**, **Matthias Bittner**, **Johannes Bosch**, **Matthias Strobel**, InMach Intelligente Maschinen GmbH, Ulm

Abstract

Today agriculture has to deal with an increasing cost pressure. Innovative Field-Robot-Based solutions can help to handle this challenge. Until now, such systems are not profitable and technically mature for commercial use.



In the project AgriApps these issues are addressed. An application based system is developed where a carrier platform can be equipped with different apps. This allows the usage of the system throughout the year to handle different tasks. Thus such a system is profitable for end users. To evaluate the developed concepts. an app for mechanical weed control in special cultures was developed. It consists of a multi-sensor setup for plants and weed detection and a manipulator for digging the soil around the plants. This not only removes the weed but also provides better water absorption for the plants. The app is finally tested on the BoniRob platform, which is the reference platform in this project.
A novelty and major contribution of the project is the general definition and specification of the interface between robot and the apps, which covers mechanical and electrical issues as well as the information flow, static and dynamic working area descriptions and energy demand management. The general approach of the interface specification is not limited on robotics, especially it can be transferred to working implements on mobile machinery.

1. Motivation and Overview of the AgriApps Aproach

Today field robots systems are leaving the labs and prototype stage and entering the market [6], especially in vegetable gardening, nurseries, viniculture and orcharding.

Beside the technical feasibility and robustness of this robots, economic efficiency is a key factor for a successful and sustainable commercial usage and acceptance of these systems. Generally there are two ways to meet the demands of economic efficiency: Either to built specialized cost effective machines for a single task which has to be performed often or like our approach in the AgriApps project design a multi-purpose system: An application based system was developed where a carrier platform can be equipped with different apps. This allows to use the system throughout the year to handle different tasks. Thus such a system is profitable for end users.

To evaluate the developed concepts an app for mechanical weed control (see Fig. 1) in special cultures, like boxwood in nurseries, has been developed. It consists of a multi-sensor setup for plants and weed detection and a manipulator for digging the soil around the plants. This not only removes the weed but also provides better water absorption for the plants.

In order to achieve an efficient and areal regulation of weed in boxwood cultivations, the AgriApps project follows a mechanical approach poaching up the ground thus disrooting weeds while in addition aerating the soil at the same time. For this approach, an application module consisting of a manipulator and a camera setup has been designed and constructed. The manipulator uses several rotating harrows mounted on a linear unit which can be lowered to the ground along a cantilever.

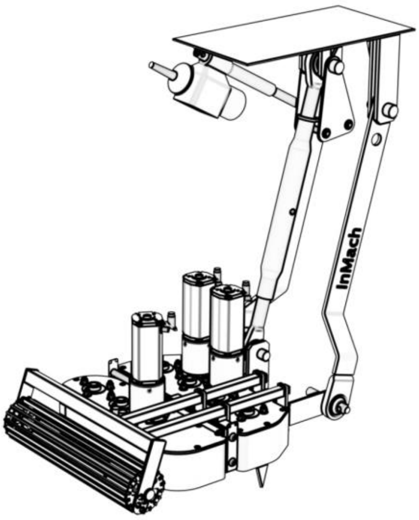


Fig. 1 Field robot Bonirob and App-Concept for mechanical weed regulation:

Other Apps like e.g. for cutting of the plants or precision plant protection / spraying purposes can be used on the same robot and in the same application domain.

This paper is organized as follows: in Chapter 2 the functionality of the mobile robot base is discussed and in Chapter 3 the functionalities of the app for mechanical weed regulation are explained. These are the functionalities of sensing and classification and the synchronized mobile manipulation which have been designed and built with a more general approach. In Chapter 4 the scope of the general specifications in the project of the interface between mobile base and app is shown. Finally in Chapter 5 first results in field tests are shown.
Further aspects in the project AgriApps which are beyond the scope of this paper are the evaluation of safety issues and the later admission for operation for such a robotic system as well as the development of an innovative energy sourcing system (see [1]) based on fuel cell technology.

2. Functionality of the Mobile Robot Base

The BoniRob autonomous driving capabilities are divided into two functional components. The low-level command execution running on the DriveECU within a real-time operating system and a high level command generation component (Navigation-Server) running on an industrial PC (Ubuntu OS). The Navigation Server communicates directly with the BoniRob low-level controller via a serial interface and sends twist-commands on demand that are executed and tracked on the low-level controller. A command from the Navigation-Server has a life-time of 400ms and the platform stops when no new twist command was send within this period reflecting a potential communication loss to the Navigation-Server. The Navigation-Server can load a Navigation-Plugin dynamically during a mission. A Navigation-Plugin hereby implements a specific driving behavior like “follow the row”. In this case, the semantic localization module fuses sensory data like odometry, 3D Laser, to estimate the current position within a row-based cultivation, determines the shape and extend of the row, and computes the next waypoint to allow the robot to follow that row. The Navigation-Server herby takes the role of a sand-box environment and ensures commands are within the mission limits (f.e. speed) and allows the application module to shift the Navigation-Reference (the point on the robot chassis that is used to approach the target, f.e. the center of the robot). It also allows the application module to slow down the robot, if in any case the application need more time to perform its action. Shifting the navigation-reference point allows the robot to drive with a laterally shift along the row which is necessary if the track-width configuration is not symmetric anymore.
A Navigation-Plugin has a defined life-cycle that allows monitoring the internal state. Similarly, the Navigation-Server also follows a defined life-cycle. Combined, both life-cycles reflect a flow-control of the driving behavior for the mission to be carried out.

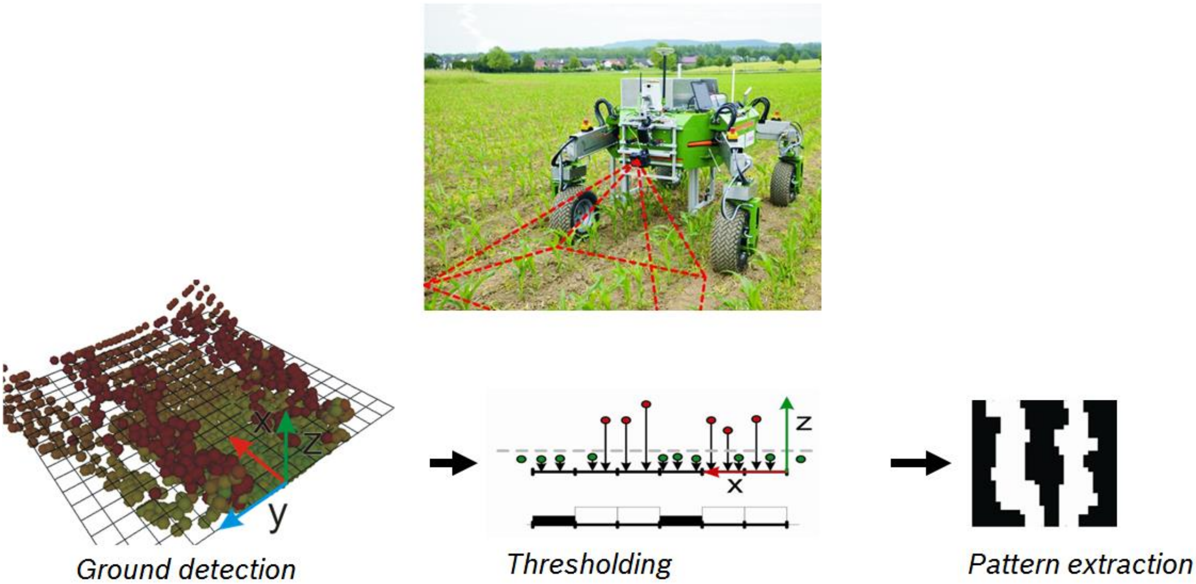


Fig. 2: Extracting a binary image from the latest measurement

Row-Navigation over Box-Tree cultivations is divided into two parts. First, the latest 3D Pointcloud from the Nippon FX8 Laser scanner is retrieved and the posture is corrected wrt the current odometry information. In the second step box-tree rows are interpreted into the current scan and the next target waypoint is estimated. This estimated target waypoint is passed to the navigation plugin that subsequently computes the needed twist-commands to approach this waypoint.

Detection of Box-Tree rows can be summarized in the following three steps: 1. Configuration of the semantic localization 2.Template Matching of the latest measurement to a database 3. Estimation of the next valid target waypoint and local-map-building.

The semantic localization module herby estimates the current position of the BoniRob in a topological map. The states the BoniRob can be are: OPEN-FIELD (the BoniRob observes free space only), ROW-BEGIN: (detect the begin of a row), ROW-CONTINUOUS (only rows of box trees are detected), ROW-GAP (box-tree rows, follow by free space, followed by box trees), ROW-END (box-tree rows are detected but in some distance (> 3m) also free space is perceived), ERROR (sensory data cannot be explained by any of the above).

The configuration step of the semantic localization module is hereby a mandatory pre-requisite since the typical height, extend, row-width, distance between rows etc are set. This data is the used in a pre-processing step to generate measurement templates. Subsequently, the latest laser measurement is transformed into a binary image reflecting the different heights in the actual laser measurement as shown in Fig. 2. The binary image is then correlated with the previously built data base of potential measurements. A score is computed for each candidate from the data base. This score can be transformed into a measurement likelihood of the current scan wrt to the template. The a-priori probabilities are encoded in a state-transition automaton. Applying Bayes-Rule allows computing the posterior for each semantic state as described above. Indeed, we compute the belief over all semantic states. Note that the state automaton is realized via a Hidden-Markov-Model. For each of the semantic states the next target waypoint can be extracted from the database templates. For more details about the semantic localization see [5]. Fusing this information with short-time odometry allows computing a target waypoint for the BoniRob that is sent to the navigation plugin which computes the appropriate twist commands for the platform.

3. Functionalities of the App for mechanical weed regulation.

The major functionalities of sensing and classification and the synchronized mobile manipulation are shown in the following two subsections.

3.1 Crop plant and weed detection

For effective weed control, it is essential to locate the weed and to discriminate it from the plants to be cultivated. A multi-sensor approach is used to detect crop plants and weed reliably. The sensor setup consists of a color camera and a linescan laser range finder (2D lidar) inclined towards the ground at an angle of about 45 degrees. The sensors are calibrated so that for each 3D point of the laser scan line, the corresponding image pixel is known. As the robot moves along the plant row, the laser scans can be aggregated to a 3D point cloud representation of the field.

Feature extraction and classification: From the sensor data, a multitude of features are extracted which help to discriminate the cultivated crop from the weed. The feature set includes range features computed from the lidar data, color features and texture features. In order to obtain robust color features with respect to varying illumination conditions, a preprocessing step is applied to the image: the color histogram of the image is fitted to a reference histogram which is selected from a set of typical training images according to the histogram intersection metric in the CIE La*b* color space [4]. Texture features capture image structures in a neighbourhood of the pixel under consideration. Different texture features have been evaluated, selected and parameterized for an optimal classification result [3]. A support vector machine [2] is used to classify the individual data points into one of the three classes crop plant, weed, and soil. The classifier model is learned from training data labelled manually.

The feature extraction and classification pipeline depicted in Fig. 3 achieves a computational performance of about 7 frames per second. However, the lidar scans come in at a higher data rate. The classes of the intermediate scan points which could not be processed in time are ‘interpolated’, i.e., each point is assigned the class of the nearest neighbour point in a scan which has been classified.

Shape estimation: For the control of the manipulator, it is desirable to obtain a representation which is more concise than the classified point cloud and free of outliers. This is achieved by fitting geometric primitives to the point clusters classified as crop plants. For the boxwood application, spherical models are employed.

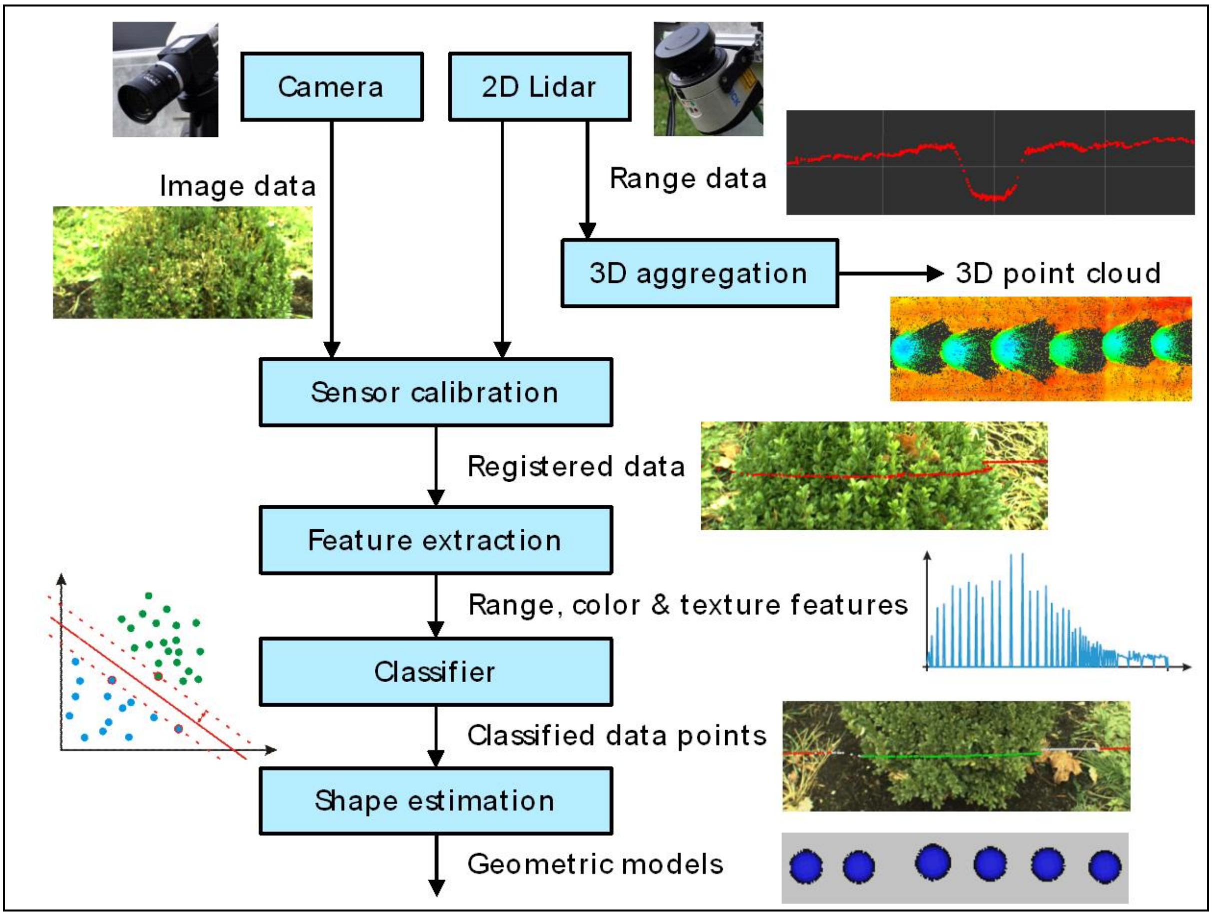


Fig. 3: Processing pipeline for crop and weed detection.

Results: The correct classification rate of the individual data points is above 95%, as reported in [3] For this contribution, the shape estimation results have been evaluated on data sets recorded at a test field consisting of 10 boxwood plants with a roughly spherical shape of about 40cm diameter. The sensor data has been acquired on 8 different days spread over a whole year. Thus, the data contains considerable variations of vegetation state, weed occurrence, and illumination conditions. For training the classifier, only data from one of the days has been used. Nevertheless, all crop plants could be successfully detected in 7 of the 8 data sets without any false positives. Only in one data set, some boxwood plants were missed because they were largely occluded by tall weed.

The reliable performance of the plant and weed detection component has also been confirmed by a first field test of the complete app with integrated manipulator control. The multi-sensor classification module may also be used for other apps such as cutting the boxwood plants into the desired spherical shape, and of course for weed control in other plant cultures.

3.2 Synchronized Mobile Manipulation for Weed Regulation

Using BoniRob as a carrier which autonomously navigates along the boxwood rows, the movements of the manipulator have to be coordinated with both the motion of the mobile platform as well as the perception module detecting the present boxwood trees. Furthermore, additional constraints are given such as consideration of hardware limits (position, velocity) of the manipulator or maximization of the area to be poached,

Control architecture: The architecture diagram shown in Fig. 4 illustrates the control system used for synchronized mobile manipulation within AgriApps. It relates the various modules involved in the AgriApps control software and depicts the respective data channels between those modules. As the mobile platform is moving along a boxwood row, the sensor system attached scans the field. Using this sensor data, the perception modules detects boxwood trees around which weeds are to be regulated. The detection results, i.e. tree location and extent, are passed to the manipulator control system. Based on this information, a Cartesian path is generated by the Cartesian Trajectory Generator, where Trajectory Models from a database can be used e.g. circular, sinoid or trapezoidal path around the boxwood tree. The Cartesian Position Controller uses this target path to generate Cartesian velocities for the end-effector based on a PID control loop. As the AgriApps manipulator actively uses just a single linear motor (the other joints are decoupled while the manipulator is dragged), this Cartesian velocity can directly be commanded to the motor actuating the linear unit. Due to hardware limits of the linear unit, cases might occur where the desired Cartesian motion cannot be achieved by the linear unit alone, e.g. for bigger boxwood trees, the position limits might be exceeded. In such cases the control system sends according adjustments to the navigation module. By predictively adjusting the navigation reference coordinate system, the mobile platform slightly changes its alignment with respect to the row so that the required Cartesian position can be reached by the manipulator. Similarly, in case velocity limits of the linear motor would be violated in order to reach the desired Cartesian position on time, the control system adjusts the speed of the mobile platform, giving the linear motor more time to follow the desired Cartesian path.

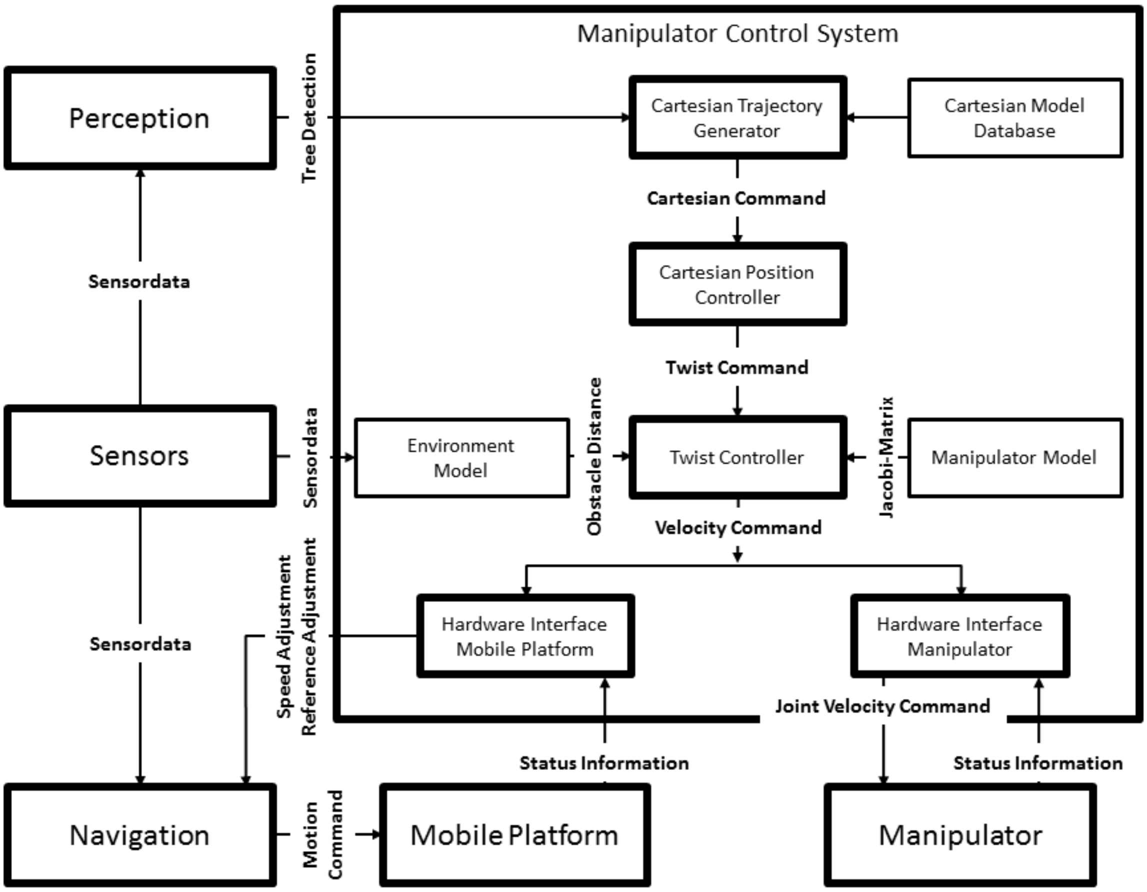


Fig. 4: Control architecture for synchronized mobile manipulation for weed regulation.

General use-cases: As AgriApps also aims for deploying the developed control system to a wider range of agricultural applications, the control system can be enhanced with an additional software module, i.e. the Twist Controller. This module allows for computing joint commands for manipulators consisting of arbitrary kinematic chains and any number of degrees of freedom. This module transforms the desired Cartesian velocity (Twist) from the Cartesian Position Controller into respective joint commands for each degree of freedom of the kinematic chain. For this, the Twist Controller module uses inverse differential kinematics calculations (based on inverse Jacobian) augmented by dynamically configurable and task-prioritized constraints to resolve possible redundancies of the manipulator, avoid singular configurations, as well as hardware limits and collisions both with itself or the environment.

4. Scope of the general specifications of the interface between mobile base and app.

A novelty and major contribution of the project is the general definition and specification of the interface between robot and the apps, which covers the classical issues

1. mechanical interfacing, and
2. electrical interfacing for control and power units

as well as the new topics

3. bidirectional information flow (safety related fieldbus and high-bandwidth data link),
4. bidirectional static and dynamic working area descriptions (symbolic and geometric),
5. kinematic and dynamic models as well as actual joint states to support coordinated synchronized motion and shared autonomy mission planning of mobile base and one or more apps, and
6. energy supply demand management.

A comprehensive specification document for a suggestion for this interface between mobile base and app(s) is planned by the AgriApps consortium to be released in spring 2016.

The general approach of the interface specification is not limited on robotics, especially it can be transferred to working implements on mobile machinery.

Therefore the AgriApps consortium (partner INMACH) is member of and watching the specification process within the AEF - the Agricultural Industry Electronics Foundation [7]: Here seven international agricultural equipment manufacturers and two associations established the Agricultural Industry Electronics Foundation in 2008. The initiative is an independent, international organization with 190 members today. As a user platform, it provides resources and know-how for the increased use of electronic and electrical systems in farming.

First, the main focus was on areas associated with ISOBUS. But that is now no longer everything. The significance of the AEF in the standardization of agricultural applications has created additional challenges for the organization. Farm management information systems (FMIS), electric drives, camera systems, high speed ISOBUS and wireless in-field communication have been included as new areas of interest.

5. Example Application Scenario

In Fig. 5 the operation scenario and first results of the mechanical weed regulation app are shown. These results look promising and ongoing tests, analysis and optimization should lead us to a robust and effective system. Another intended possibility to increase the power of impact is to use up to three apps to be able to work on the middle strip and left and right half-strips simultaneously.

Fig. 5 Operation and results of the mechanical weed regulation app

Acknowledgement

This work has been supported by the German Federal Ministry of Education and Research (BMBF) within the project "AgriApps" under grant No. 01IM12002A-D and Baumschule Braun

References

[1] Bosch, J., et al., Sustainable energy for agricultural applications: Working prototype and possibilities of fuel cell energy sourcing, Proceedings of 73rd International Conference on Agricultural Engineering AgEng 2015.

[2] Chang, C.-C., & Lin, C.-J. LIBSVM: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2 (3), 2011.

[3] Frese, C., Meyer, J., & Frey, C., Discrimination of plants and weed by multi-sensor fusion on an agricultural robot. Proc. Int. Conf. of Agricultural Eng., 2014.

[4] Lee, S.M., Xin, J.H., & Westland, S., Evaluation of image similarity by histogram intersection. COLOR research and application, 30 (4), 2005.

[5] Ruckelshausen et al. BoniRob–an autonomous field robot platform for individual plant phenotyping. Precision agriculture, 2009, 9. Jg., Nr. 841, S. 1.

[6] Traction, article: Tend Feldrobotik – Jäter und Sammler, May/June 2015

[7] AEF – Agricultural Electronics Industry Foundation, www.aef-online.org, 2015