Autonomous control of field robot

Individually study activity in field robotics

Authors:
Christian Liin Hansen

Instructors:
Kjeld Jensen

University:
The Technical Faculty, University of Southern Denmark

Course:
Extension of X-ROBO-U1 by individually study activity. 5 ECTS point

Project period:
September 1^{st} - January 1^{st} 2014

Contents

1 Introduction

- General introduction for the course and the different tasks of the project

1.1 Project description

The University of Southern Denmark (SDU) was providing a summer course in Field Robotic in summer of 2014. A test field with growing maize was established outside the university. The goal was to design an autonomous field robot capable of making its way through the field from START position to FINISH position, which is sketched in figure 1.2. The result was a mechanical working robot, which is shown in figure 1.1. The field robot had a web camera mounted in front of the robot, adjusted to make visual contact to the rows of maize in the field. An image processing algorithm with row extraction was implemented during the summer course.

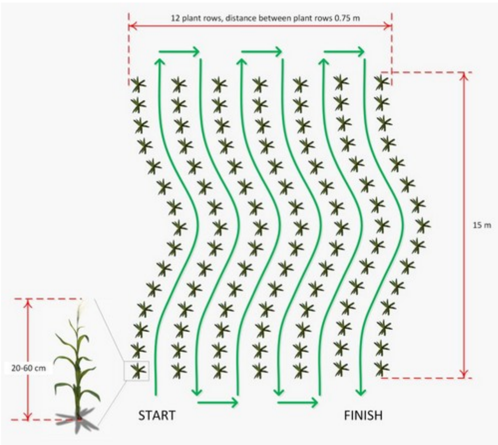

Figure 1.1: The test field at FRE 2014 [1]

Figure 1.2: The final construction of the field robot

1.1.1 Image processing - row extraction

The image processing algorithm contains three components. The first component is extracting the green nuance for each image and together with a threshold, contours finding and raw moments calculation, the central mass coordinates for each green segment in each images is found. The second component is applying the RANSAC [2] algorithm on the dataset that contains the (x,y) central mass coordinates. The RANSAC stands for Random Sample Consensus, which means a line is spanned by two random points and the number of inliers is counted for that specific line. An inlier is defined as a point, where the perpendicular distance from the point to the line spanned by the two random is below a given distance threshold. If a given points exceed that given threshold it is an outlier. The RANSAC algorithm repeats its procedure 1000 times for each image and the line that contains the most inliers is defined to be the best fitted line in the current image. The third and last component in the image processing algorithm is to calculate the perpendicular distance r from the best fitted line and to the origo, which is defined to be the upper left corner of the image. Additional the angle of the best fitted line in respect to the horizontal x-axis in the images is calculated. The image processing node with the RANSAC algorithm was tested both inside RoboLab and outside in the field. The performance is illustrated in figure 1.3 and figure 1.4, where the distance r and angle theta is displayed.

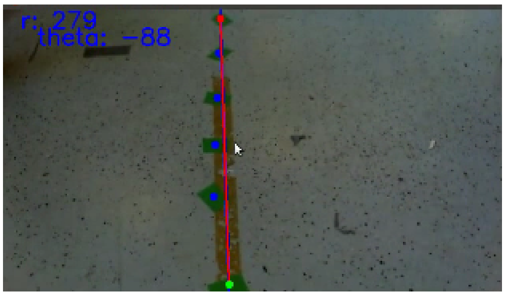

Figure 1.3: Testing algorithm inside RoboLab

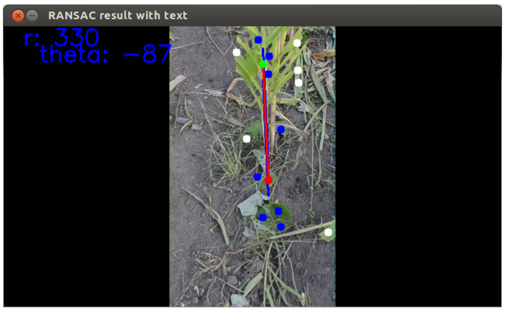

Figure 1.4: Testing algorithm outside in field

1.1.2 Problem statement

In figure 1.3 and figure 1.4, the row extraction algorithm is illustrated. When testing inside RoboLab the algorithm was performing better, since the environment was more ideal compared to the field. The result described in figure 1.3 was only produced by controlling the robot using a keyboard. The result in figure 1.4 was produced by a handheld webcam. That means the field robot never came to the point, where it could drive autonomously through the field or even inside in RoboLab.
The problem in this project is to make the field robot able to drive autonomously by using at least one webcam for navigation. To start with the indoor environment, the robot must be able to follow the line of green tape on the floor, by having the two driving wheels on each side of the line. Further extension is to have the field robot be able to follow the line of maize out in the field. This would include more optimizing work on the image processing node, due to the fact that the environment is much more complex.
An solution for the problem is to implement the FroboMind [3] architecture, which is shown in figure 1.5. The component in the figure has been color coded. Green color indicates a component is already implemented using FroboMind, blue indicates components that needs optimization and red indicates components that has not been touch and needs implementation. The Field robot frame has been color coded blue, to indicate that the physical robot might need optimization of the camera mounting location. The image processing needs to handle “End-of-Line” detection and optimization in general when it has to work in the field.

1.1.3 Related work

From previous work [4] in the summer course 2014, ideas for implement the mission control was discussed. The image processing node output the distance r and the angle θ. The image processing node has the following interface:

- Input: 512x384 RGB image from webcam at minimum 5 fps, maximum 30 fps
- Output: The distance r, which is the pendicular distance from the line extracted to the upper left corner of the images. r is between 0 and 640 pixels
- Output: The angle θ, which is the angle in degrees the horizontal x-axis and the line extraction. θ is between -90 and +90.

1.1.4 Hypothesis

In order to test the field robot the requirements for the test field is needed. The standard measurements is 750 mm between the rows [5].The dimension of the field robot is 830 mm x 815 mm. The inner distance between the wheels is 600 mm. The robots translation offset is illustrated in figure 1.6. The robots orientation is illustrated in figure 1.7. A

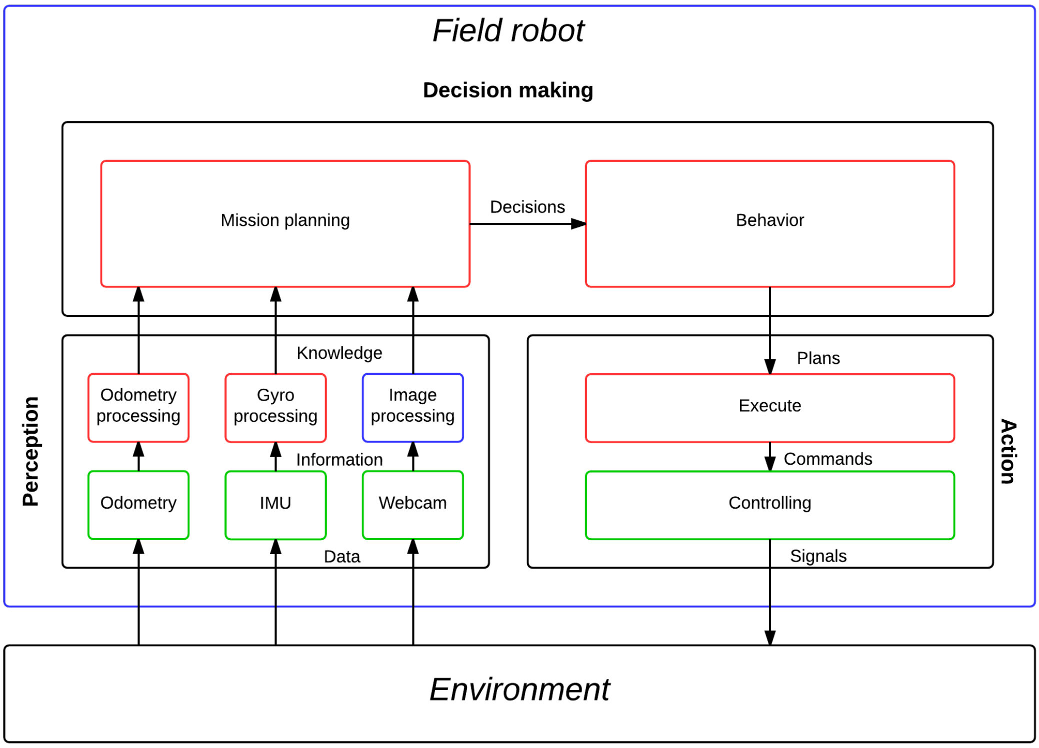

Figure 1.5: The FroboMind architecture of the field robot. Each component has been color coded. Green color indicates already implemented components, blue indicates components that needs optimization and red indicates components that is not implemented.

combination between the scenarios can occur. At the moment the project will only focus on the two scenarios separately. However it the combination will iteratively be considered when testing the hypothesis. As a starting point, the hypothesis will relate to test inside RoboLab. The green line indicate green tape on the floor. This scenario will be extended later in the project to the outside test field, where the green lines indicate rows of maize. The hypothesis is stated as followed: The robot will navigate through the line by having the driving wheels between the rows. The robot is able to drive sufficiently smooth by having a maximum offset in rotation and translation. These parameters is defined as followed:

- The field robot is able to follow the line with a maximum offset in translation of 100 mm. I.e. the distance from the inner line toward the driving wheels will be in range 200-400 mm.
- The field robot is able to follow the line with a maximum offset in orientation of 25 degrees. I.e. the angle between the direction vector and the horizontal x-axis is in range from -90 to -65 and from 90 to 65 degrees.
- The robot is able to turn 180 degrees with an maximum offset of 20 degrees. Each turn is within 1,5 meters, measured from the end of line to the center point of the robot. Atbthe end of a turn the robot is able to follow the next row of maize.

1.1.5 Aim of project

The aim of the project is decomposed into different phases. The aim of the project is to complete all the phases, without rejecting the projects hypothesis. To follow a structured approach towards the aim of the project, the phases has been prioritized. In

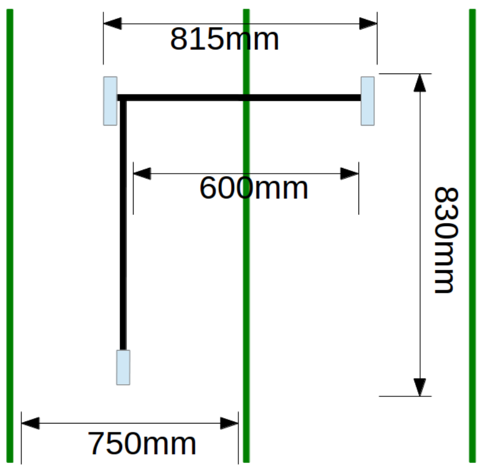

Figure 1.6: Translation offset equal to 0

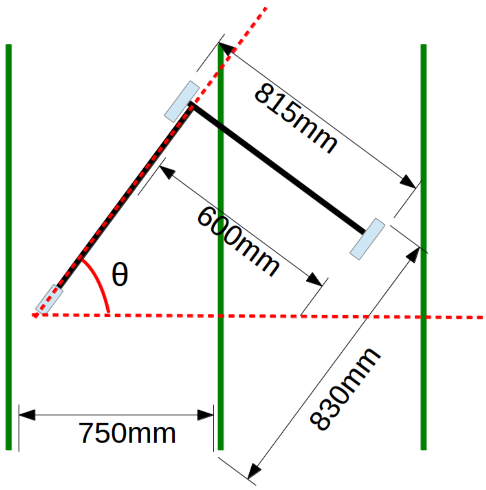

Figure 1.7: Orientation offset equal to θ

this project the focus is to have a field robot running autonomously outside in a real test field. Therefore the turning phases has been down prioritized compared to follow a line of maize. The phases is defined as followed:

- Phase 1: The robot is able to sense the green tape inside RoboLab by only using the webcam. The FroboMind architecture in figure 1.5 is implemented and hence the robot can follow the a line of green tape autonomously. A PID regulator will be implemented in software to regulate the linear and angular velocity commands.
- Phase 2: The robot is able to sense the row of maize outside in the test field with only using the webcam. The robot can follow a row of maize autonomously.
- Phase 3: The robot is able to sense the green tape inside RoboLab and together with inputs from the interfaced gyroscope an implemented Kalman filter will provide the sensor fusion and stabilizing the line-following.
- Phase 4: Same as phase 3, but now outside in the field.
- Phase 5: The robot is able to detect “End-of-Line” zones in the image processing algorithm, which can set the robot in different behaviors, i.g turning or line-following. The gyroscope and odometry is used to perform a 180 degree turning. The image processing algorithm is able to detect “Start-of-Line” zones. The robot is able to follow the next line.
- Phase 6: Same as phase 5, but now outside in the field.

An overview of the different components, where all the phases has been completed is shown in figure 1.8. The PC with ROS contains the FroboMind architecture, which is illustrated in figure 1.5.

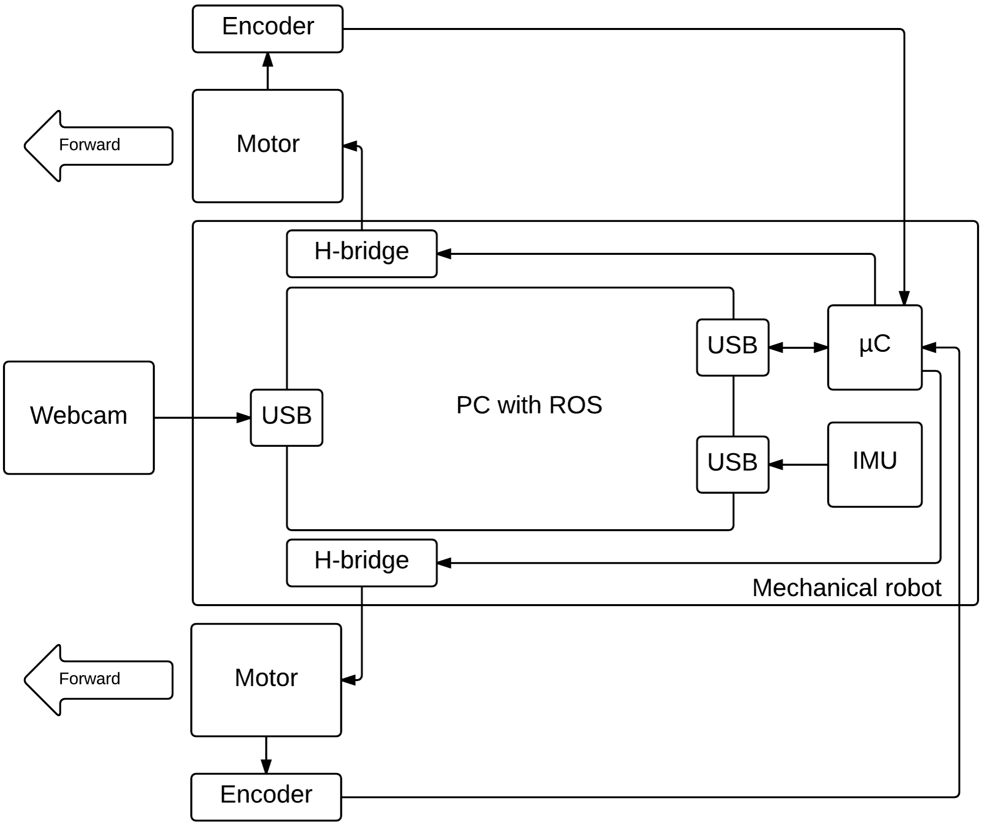

Figure 1.8: Implementation of all the physical components on the field robot.

2 Materials And Methods

In this section the different component in the project will be presented. For better overview and more efficient process, the project has been divided into several components in a *Work-Breakdown-Structure* diagram, shown in figure 2.1.

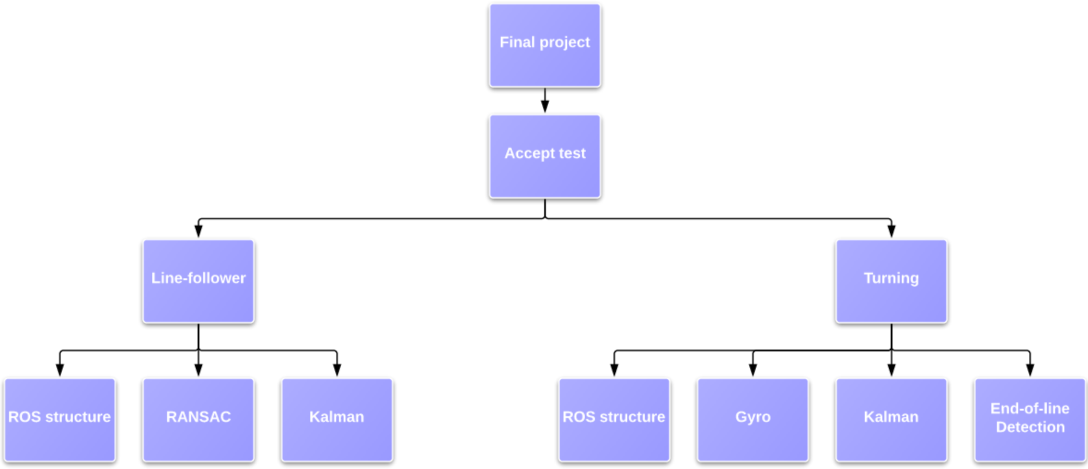

Figure 2.1: *Work-Breakdown-Structure* illustration shows how the project has been devided into several components

2.1 Simulation line-follower - week 46

A line-follower has been implemented by extending previous work [4], which was the RANSAC algorithm, with a ROS structure. The Kalman filter has not yet been implemented in this component, either has the PID regulator for velocity commands. The result is a simulation of the following nodes:

- A vision node that publish the distance r and angle theta. The parameters is described is section 1.1.1.
- A decision node that subscribe the parameters with a callback function and publish the decided angular and linear velocity commands.

Using *rqt_graph* from the rqt package, the nodes and topic is illustrated in figure 2.2

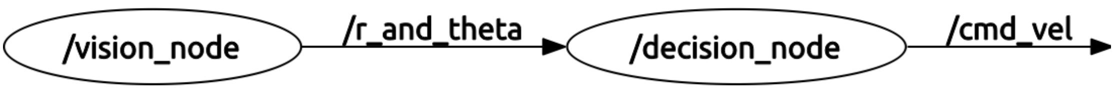

Figure 2.2: ROS structure with vision node that publish parameters on the *r_and_theta* topic. The decision node subscribes on the same topic and publish a linear and angular velocity on the *cmd_vel* topic. A dummy node listener was created to illustrate the last topic, which is not illustrated

To simulate the line-follower, the video input stream is based on a still image, figure 2.3, which can be rotated as illustrated in figure 2.4.
The result of the simulation shows that is is possible to steer the robot. The simulation result is shown in figure 2.5

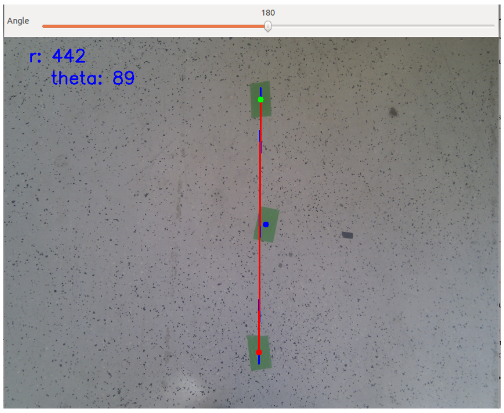

Figure 2.3: 89 degrees

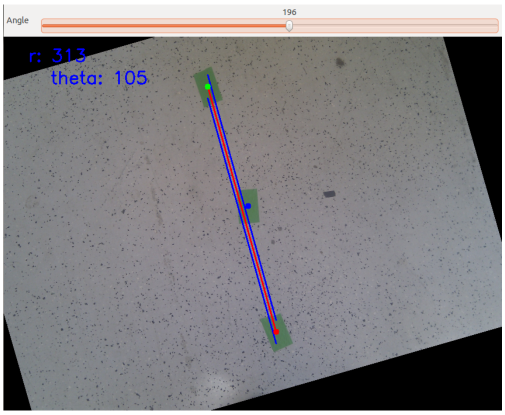

Figure 2.4: 105 degrees

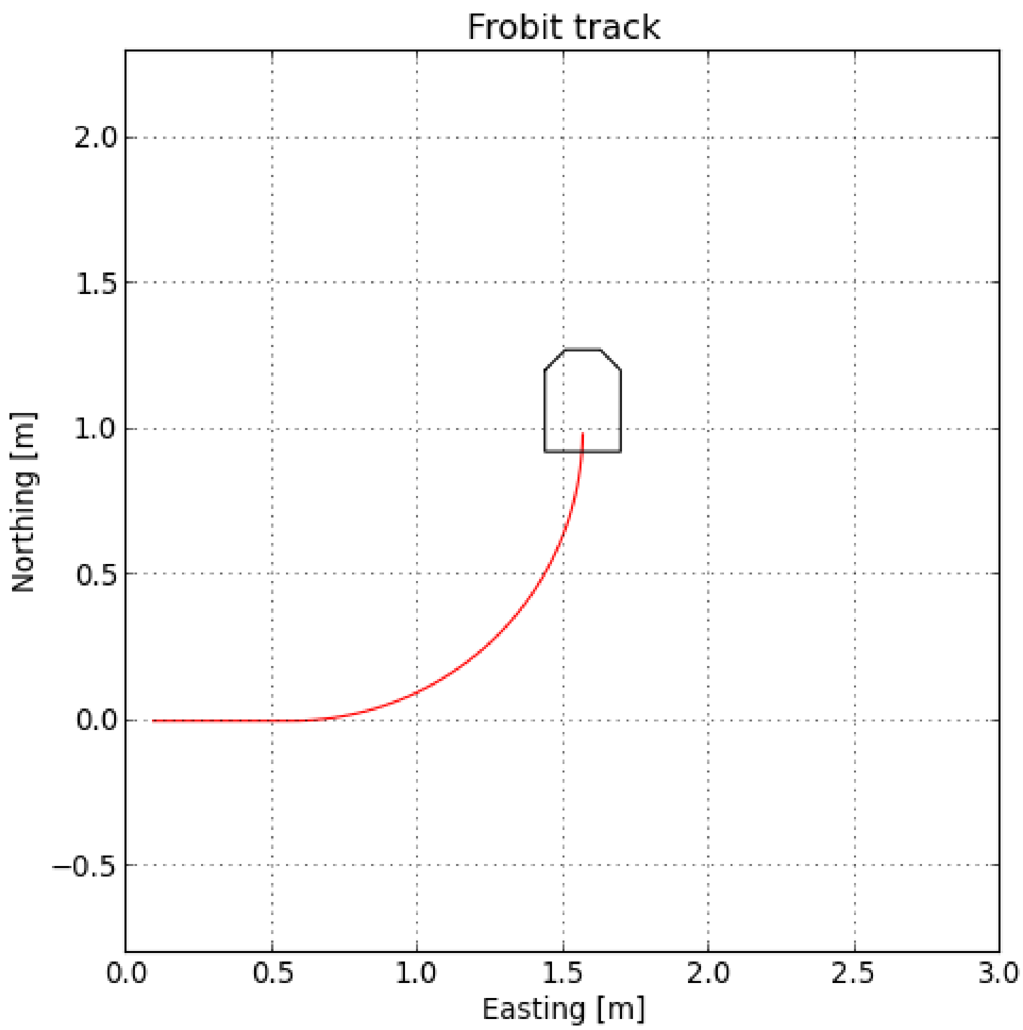

Figure 2.5: Simulating video stream with rotating still images result in change of angular velocity of robot in simulation. The linear velocity is constant.

2.2 Testing line-follower inside RoboLab - week 47

Here documentation for week 47 will be stated.

3 Deliverables

The project will be reviewed by the supervisor about halfway the trough the project period which is November 1st, 2014. By this date the following will be delivered:

- The sections “Introduction” and “Materials and methods” of the project report.

The project deadline is January 1st, 2015. By this date the following will be delivered:

- A project report documenting the fulfilment of:
  - Project aims
  - Deliverables
  - Learning goals
  - Accept test results
- FroboMind software modules and documentation
  - Mission control node implementation
  - Image processing node optimization

4 LearningGoals

At the end of this project, the student will be able to:

- Identify a specific problem domain through literature study.
- Apply engineering skills in the evaluation of existing commercial and research solutions related to the identified problem.
- Propose an optimal solution based on domain knowledge and literature review.
- Apply engineering approach to construct a prototype demonstrator based on the proposed solutions.

Furthermore the student has through the prototype demonstrated engineering skills within:

- Optimizing row detection algorithms for a web camera.
- Using FroboMind architecture to interface different sensors.
- Using a Kalman filter for sensor fusion between IMU, encoders and webcam.
- Using a PID regulator for regulate velocity commands.
- Project management.

5 Bibliography

[1] Robatic Group 2014.
Field robot event 2014 - task description.
`http://www.robatic.nl/index.php/field-robot-event`, 2014.
Available: 7-10-2014.

[2] Martin A. Fischler and Robert C. Bolles.
Random sample consensus - a paradigm for model fitting with apphcatlons to image analysis and automated cartography.
SRI International, 6:381–395, 1981.
doi:10.1145/358669.358692.

[3] Kjeld Jensen, Morten Larsen, Søren H. Nielsen, Leon B. Larsen, Kent S. Olsen, and Rasmus N. Jørgensen.
Towards an open software platform for field robots in precision agriculture.
Robotics, 3:207–234, 2014.
doi:10.3390/robotics3020207.

[4] Christian Liin Hansen.
Field robot summer course 2014.
`https://www.dropbox.com/s/ew82lvrkpvb3pjh/FRSC14.pdf?dl=0`, 2014.
Available: 7-10-2014.

[5] Germany University of Hohenheim.
Field robot event 2014.
`https://fre2014.uni-hohenheim.de/uploads/media/fre-2014-tasks-v06.pdf`, 2014.
Available: 7-10-2014.