

Article

Precise Navigation of Small Agricultural Robots in Sensitive Areas with a Smart Plant Camera

Volker Dworak ^{1,*}, Michael Huebner ^{2,†} and Joern Selbeck ^{1,†}

¹ Leibniz Institute for Agricultural Engineering Potsdam-Bornim e.V., Max-Eyth-Allee 100, D-14469 Potsdam, Germany; E-Mail: jselbeck@atb-potsdam.de

² Embedded Systems for Information Technology, Ruhr-University of Bochum, Universitätsstraße 150, D-44801 Bochum, Germany; E-Mail: michael.huebner@rub.de

† These authors contributed equally to this work.

* Author to whom correspondence should be addressed; E-Mail: vdworak@atb-potsdam.de; Tel.: +49-331-5699-420; Fax: +49-331-5699-849.

Academic Editors: Gonzalo Pajares Martinsanz and Francisco Rovira-Más

Received: 4 September 2015 / Accepted: 30 September 2015 / Published: 13 October 2015

Abstract: Most of the relevant technology related to precision agriculture is currently controlled by Global Positioning Systems (GPS) and uploaded map data; however, in sensitive areas with young or expensive plants, small robots are becoming more widely used in exclusive work. These robots must follow the plant lines with centimeter precision to protect plant growth. For cases in which GPS fails, a camera-based solution is often used for navigation because of the system cost and simplicity. The low-cost plant camera presented here generates images in which plants are contrasted against the soil, thus enabling the use of simple cross-correlation functions to establish high-resolution navigation control in the centimeter range. Based on the foresight provided by images from in front of the vehicle, robust vehicle control can be established without any dead time; as a result, off-loading the main robot control and overshooting can be avoided.

Keywords: vision-based navigation; infield navigation; steering control; plant camera

1. Introduction

In the field of agricultural research, precision farming and bio-farming agricultural robots are becoming more important because of the growing availability of robots as well as new and alternative applications that robots can provide or will provide in the near future. These alternative applications include testing and measurement applications, and research has focused on the placement of small chemical or insect bombs at precise infield positions for pest control. When wind from helicopter propellers prevents precise application, the payload is too small for deployment using helicopters, or views close to the ground [1] or mechanical manipulation are required, infield robots are the right choice. However, only robots with small wheels and low weight can be used when young or expensive plants need to be protected. Therefore, the vehicle cannot drive over the plant lines, and a method for inter-row weeder guidance is required. The typical drilling distance for wheat is 16 cm in Germany; thus, navigation precision must be in the centimeter range, which can be accomplished with a vision-based system [2–4]. Standard Global Positioning System (GPS) approaches often fail to produce high resolution over the entire field because when a satellite is hidden by an obstacle, such as a tree, hill, building, or the horizon, the resolution of the calculated position jumps to the meter range. Even a high-resolution real-time kinematic GPS that uses two GPS devices and a radio connection to transfer the correction data faces similar problems when the radio connection is lost. For a ground-based radio connection, such a loss of connection can occur without obstacles because ground-reflected radio waves experience a 180° phase shift of directly transmitted waves, which results in the attenuation of the transmitted wave [5]. Under good satellite and radio conditions, real-time kinematic GPS can reach centimeter-scale resolution for slow vehicle speeds [6] or at larger time steps of one second [7]. In addition, typical maps uploaded to agricultural machineries have a grid size in the meter range [8]. A meter-scale grid is sufficient for large agricultural machines but not for small field robots. Accordingly, alternative or complementary techniques are required for precise navigation control. The best practice is to use an actual view of the plant lines to determine the correct direction to navigate [2,9–11]. For an automobile, navigational laser scanners and camera systems are common [12]; however, laser scanners are expensive and optimized for automobile applications; thus, they have large and overlapping spots to ensure the safe detection of all potential obstacles instead of centimeter-scale resolution [13,14]. Low-cost laser scanners are produced for indoor use and are not designed to operate in conditions with high amounts of water, dust and vibration, and they fail when exposed to direct sunlight, which is the greatest disadvantage for infield applications. In research applications, laser scanners are versatile under good weather conditions for a range of applications, including measurements of tree-row crops [15] or corn crops [16]. Although the cost of camera systems is decreasing, they lack robust plant detection software, which must be implemented in the system by the user. However, a normalized difference vegetation index (NDVI) processing system may be implemented along with plant detection software. NDVI is commonly used in agriculture to detect chlorophyll activity and separate plants from the soil. Although different formulas can be used to process NDVI signals and images, they always result in a grayscale or binary image with an adequate threshold. Both types of images can be used for high-resolution navigation control, and most high-resolution navigation applications use Hough transformations to detect plant rows [2,9–11,17], the cross-correlation is a simpler approach in terms of processing power for a small embedded system. The combination of these

images with a mask representing the plant line, for cross-correlation result in a precise position signal, which can be used to (lock-in) follow along the plant line. The cross-correlation function has a high filtering effect and is thereby good for noise reduction and outlier suppression. Using the binary image and mask results in a dramatic reduction of calculation power for microcontrollers/processors and field programmable gate arrays (FPGA), because they can use the logical “AND” and a counter for the whole math. Additionally, the horizontal use of the images has the grade advantage for parallelizing the calculation process, and thereby accelerating the result determination. The foresight provided by the image in front of the vehicle provides sufficient calculation time and timely decision support. Accordingly, vehicle control can be established without any dead time, and overshooting of the control output can be avoided. The calculating power costs in a small embedded system are low and will continue to decrease in the future, thereby enabling effective image-based navigation control systems for infield robot applications.

2. Methods

The plant-based navigation described in this article is a combination of two robust concepts: the image of a known scene with a known plant camera system and the cross-correlation mathematical operation that determines the degree of similarity of two functions. In this case, the cross-correlation determines the degree of similarity of a pixel line from both an image and mask, which corresponds to the periodic plant line structure.

2.1. Plant Camera and Imaging

The plant camera should be mounted as high as possible on the field robot. The mounting angle should provide a good compromise between foresight and high-resolution views close to the robot, which also depends on the viewing angle of the objective. A typical arrangement is shown in Figure 1.

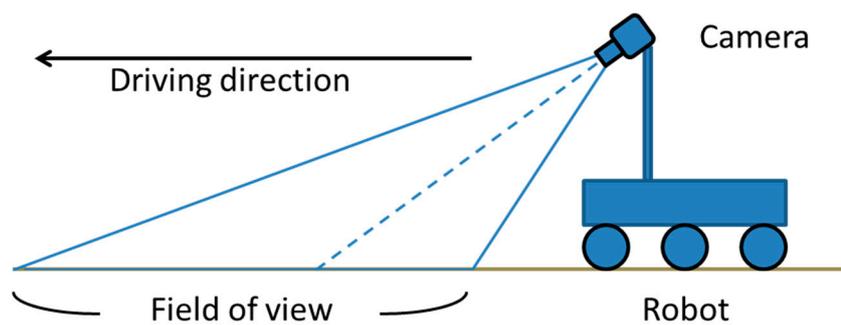


Figure 1. Mounting position of the plant camera on top of the field robot.

With respect to costs, any color complementary metal-oxide-semiconductor (CMOS) charge-coupled device (CCD) cameras with near infrared (NIR) sensitivity (~800 nm to 900 nm) can be used. The IR-cutoff filter must be removed, and a low-pass filter from approximately 645 nm to 950 nm (RG645, SCHOTT AG, 55122 Mainz, Germany) should be used. Better results can be achieved with an adapted double band-pass filter [18], although this configuration has a higher cost. An adequate CMOS chip is

the Aptina MT9V032STC (Aptina Imaging Corporation, San Jose, CA, USA), which has high NIR sensitivity at 850 nm (Figure 2) and provides an image with 752 × 480 pixels.

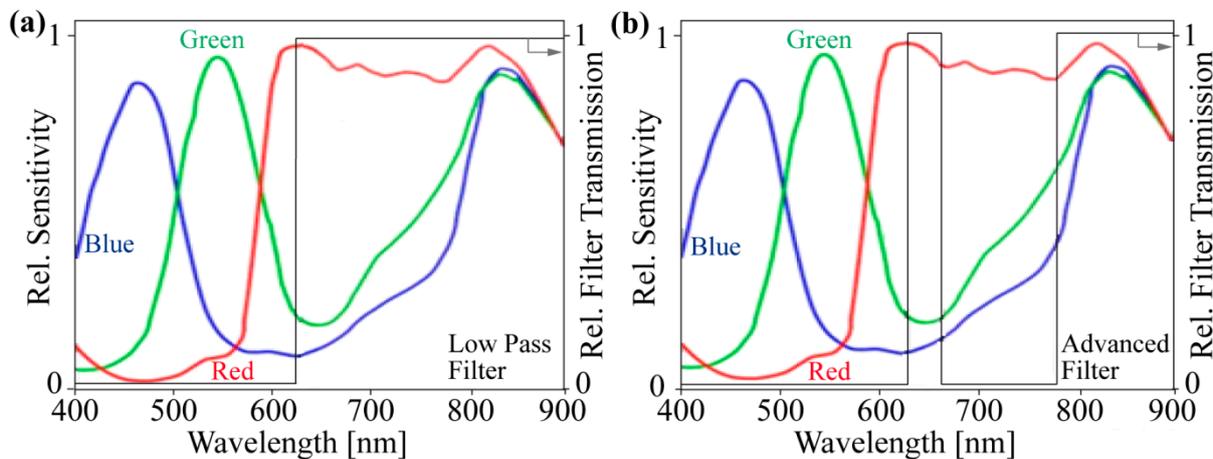


Figure 2. Spectral distribution of the RGB color chip MT9V032STC. Additional spectral range of the low pass filter (a) and optimized double band-pass filter (b) [18].

Figure 2 shows the usable spectral range of the camera chip after implementing an optical filter. The green and blue channels are only sensitive in the NIR range, and the red channel is sensitive to red and NIR light. Figure 3 shows the spectral characteristics of a plant, and the highest amplitude of spectral response is between the red and NIR range. Therefore, these spectral components are often used for the NDVI.

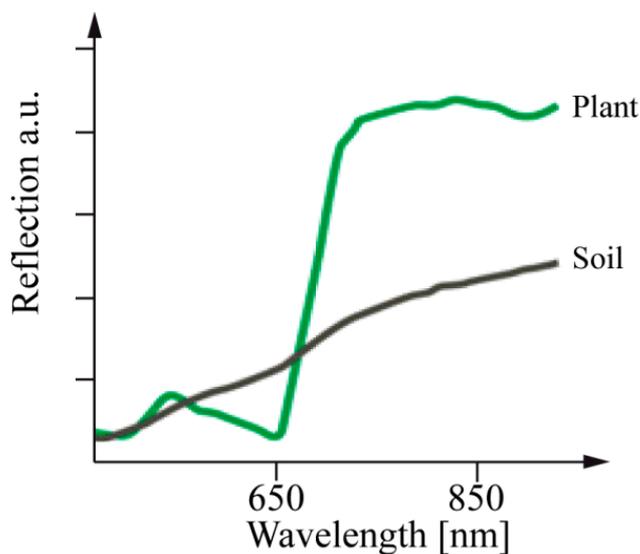


Figure 3. Characteristic spectral distribution of plants and soil. Measured 2012 at a campaign by Gebbers *et al.* with a spectrophotometer (400 to 1000 nm, build of MMS1 NIR enhanced optical modules (Zeiss, Jena Germany) and LOE-USB controller (tec5, Oberursel, Germany)) [19].

The formula for the NDVI must be adapted to the spectral composition of the new “RGB” channels of the color chip.

$$(NIR - R) / (NIR + R) \rightarrow ((B_{channel} + G_{channel}) - R_{channel}) / R_{channel} \tag{1}$$

Several optimizations can be used improve the contrast between plants and soil in the images [18], but the most important are debayering control and white balancing termination in custom cameras. These algorithms are usually optimized for RGB images and do not operate properly for new channel configurations. With optimized NDVI images, enhanced binary images can be produced [18], and they simplify the subsequent image processing applications [20].

Images of winter wheat were taken at the early growth state, and they were not optimal in terms of quality and viewing direction compared with images from fixed-mounted camera on a field robot platform. Although the images were not optimal, they demonstrate the robustness of the cross-correlation algorithm.

2.2. Cross-Correlation

The discrete form of the cross-correlation is shown in Equation (2). The calculation width from $-N$ to N corresponds to the search window width of the cross-correlation, which will be subsequently used for tracking the position of the resulting maximum.

$$CC(k) = \sum_{n=-N}^N B(n) \cdot A(n+k) \tag{2}$$

The cross-correlation algorithm describes the identity of two functions, with one discrete function moved over the other and each data point (pixel) then multiplied. Finally, all of the results are summed, and the position of this sum is then stored or displayed (Figure 4).

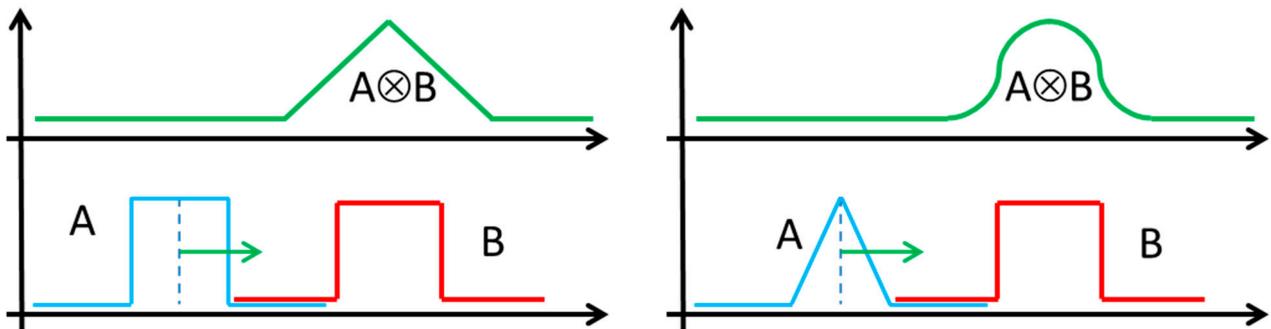


Figure 4. Correlation between simplified signals A and B.

The maximum of the correlation result provides the highest identity position of the two functions. For a periodic function, the result will have a periodic maximum. With respect to the round shape of the maxima peak, the center of the peak can be calculated by the distance between the -3 dB ($1/\sqrt{2}$) and -6 dB (0.5) points or displayed in a small search window for peak tracking of the median position width. Navigation control must follow the peak signal; therefore, the mask function does not have to be moved over the entire pixel line of the image, which reduces the amount of calculation cycles to the number of pixels from the mask function ($-N$ to N). During perfect navigation control, the peak signal of the correlation is always in the middle of the results (Figure 5), and the input signals for the control loop to

follow the specific plant line represent deviations from optimal conditions. Therefore, the search window for tracking can be even smaller than the mask window. For example, the window size for the cross-correlation is four or five periods wide, and the window size for the tracking search is one period.

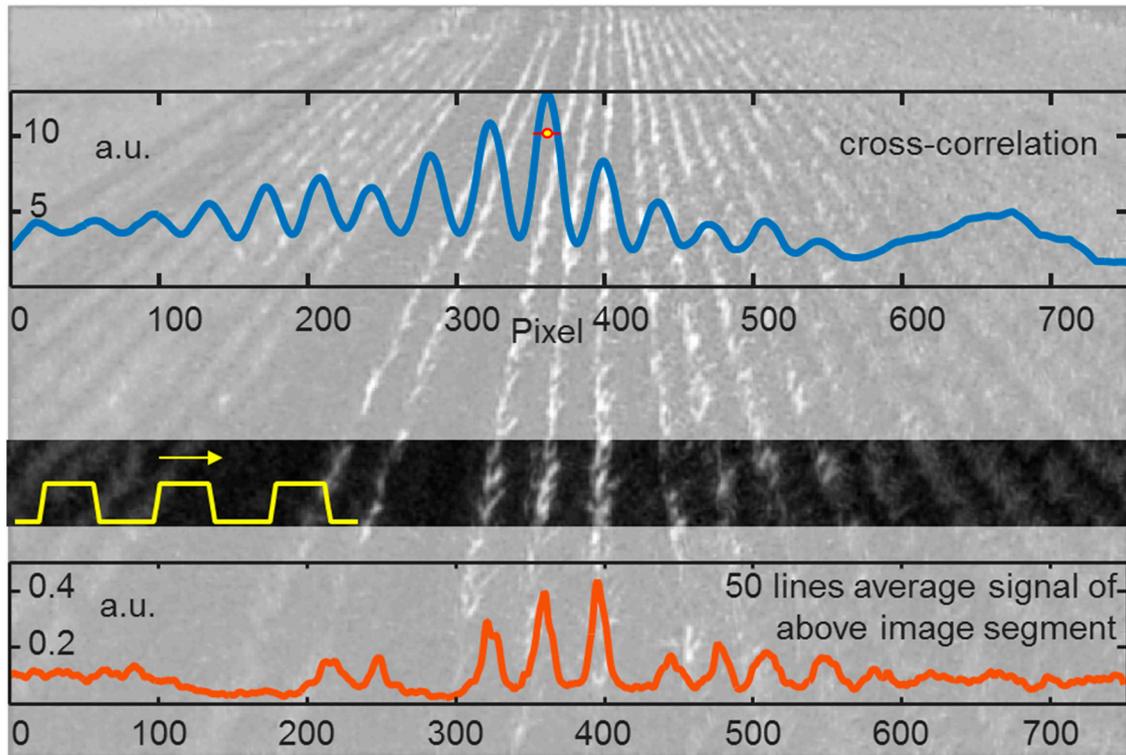


Figure 5. Application of the discrete cross-correlation between a 50-line averaged image signal and three-pulse rectangular shape mask. The center point of the correlation result is marked with a yellow and red point.

Because the middle range of the image is optimal for this application, an inexpensive objective for the camera system can be used. Optical distortions caused by the objective affect the outer part of the image, although they can be ignored for this application. The image will also be used for additional image processing applications [21], with a higher quality lens used in practice.

3. Strategies

Several strategies can be used to obtain adequate results for this navigation approach, including mask design, average number of image line determination, error correction establishment, and embedded system design.

3.1. Mask

The mask function can be estimated or calculated by different functions or methods. With a fixed camera mount and known drilling distance, an empirical mask can be used in most cases. At the starting point in front of the field, the robot system can calculate the actual mask. These calculations do not affect the control loop because the mask will be calculated only once at the initializing/starting phase after each turnaround at field ends. After stretching out the perspective in the image, a fast Fourier transform (FFT)

function can be used to determine the basic frequency of the plant line distance, which should be the frequency for the periodic mask function. Based on the stretched image, the maximum peak intensities of the cross-correlation can be used by varying the periodicity of the rectangular-shaped mask from the expected nearby lower periodicity to higher periodicity. Alternatively, a low-pass filter, such as a Gaussian-shaped filter, can be applied to the image lines in the x-direction. The threshold based binary result can be used to obtain a rectangular signal; in this case, a median duty cycle is a good choice for the mask function.

In addition, the mask function length, or the number of periods to be used for the correlation, must be set. A short mask produces a small number of calculations, whereas a large mask provides a better filtering effect and is more robust against outliers. For this application, a very large mask must be adapted at the border side because the perspective image causes changes in both the periodicity and duty cycle (Figure 5). For the same camera mounting position, a correction function or lookup table can be used to adapt the mask limb. However, with respect to calculation power, a medium-size mask is a good compromise. One rectangle is too sensitive and can lead to frequent missing plants in the line. Four rectangles are robust as long as four plants are not missed; this exception will be discussed later. Image stretching evens the linear mask correction function for the y position, with the mask decreasing in size from the bottom to the top y lines. In addition, small variations of mask scaling can be used to calculate results with the highest minimum-to-maximum distance or the best fit. Increasing differences from the best mask scale to expected scale provide additional terrain information. Larger scales indicate that a hill is coming, whereas smaller scales indicate that a hill has passed. For a field robot with an integrated hybrid power system, this information can be used to direct the robot to provide a higher amount of power, such as by increasing the generator turns per minutes.

3.2. Number of Tracking Points and Averaged Image Lines

In natural scenes, the field arrangement is never perfect. Therefore, an individual image line in the x direction can appear as noise, which is useless for a tracking result or point and may be caused by drilling errors or animal interference at the specific area. Significant effort is required to write a program that can manage all of the existing or possible exceptions, and a more effective solution is to reduce the number of tracking points and calculate them with an averaged image line. Due to the perspective distortion in the image, a limited number of lines is available for averaging in the y direction because the result is approaching an increasingly flat line. Depending on the mounting position and camera resolution, each setup will have its own optimal compromise for the number of tracking points and averaged image lines, which can be performed by averaging 10 or 50 lines or using a certain percentage of lines above and below the actual y position, such as 20 lines combined with an additional 20 lines above and below. This process results in a high degree of filtering, but the average of each 20-line package must be calculated only once. These packages can be weighted by 0.25, 0.5 and 0.25, which results in a Gaussian filter response. Thousands of combinations of filter types and lengths are possible. With respect to calculation power, simple algorithms are preferable because the cross-correlation can filter as well.

Figure 6 demonstrates that the differences at the center region are minimal as long as plants are not missing in the line.

The number of calculated tracking points is not constrained because the averaged regions can overlap. A larger overlap results in smaller potential movements of the tracking points, although it requires additional calculation power to perform averaging and has an additional disadvantage of reduced tracking at small curve radius.

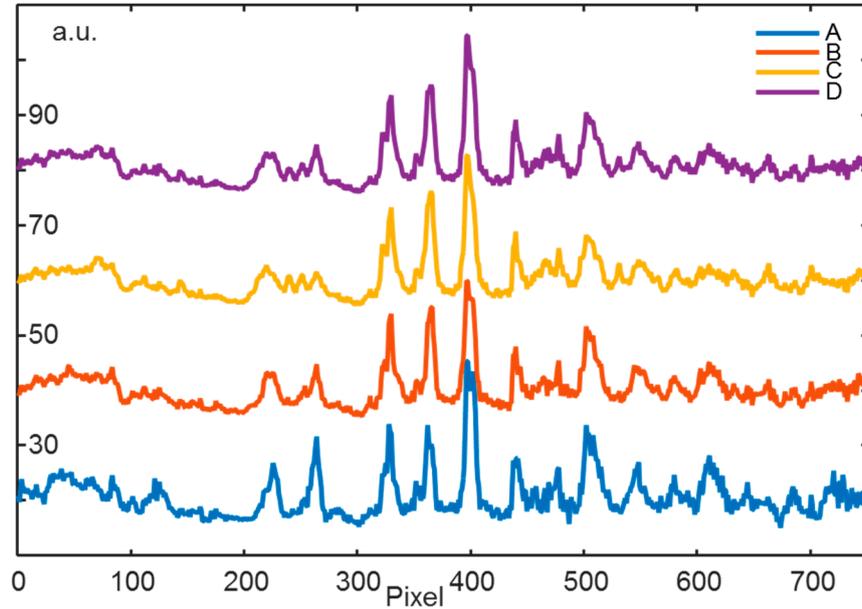


Figure 6. Four plotted lines showing the normalized average result at the same y position. Line A is an average of 20 lines, line B is an average of 40 lines, line C is an average of 60 lines, and line D is the Gaussian result ($20 \times 0.25 + 20 \times 0.5 + 20 \times 0.25$).

3.3. Error Correction

Because of the imperfect field situation, the cross-correlation can produce higher maxima to the left and right of our tracking maximum. Therefore, the search window size should be reduced to one maximum. Determining the field conditions can reduce the window size [22]. If the maximum jump is towards the window corner or the distance between minimal and maximal values is too small, then a warning or error signal is indicated. If the error signal is missing for results at higher y positions, then certain errors can be ignored and compensated by a linear regression as described below. Figure 7 shows a typical search window result with the reduced width during a normal operation cycle, and the gray areas indicate the warning region for the center point. The warning region is an example and can be adapted to the camera resolution and view field.

Larger field areas may not be in the correct order, which might be caused by animals or drilling errors. Drilling errors are not frequent in gardening, but such errors occur more frequently during grain cultivation. At the drilling machine stopping position for grain refilling, islands of excessive or missing plants can occur, and the cross-correlation cannot find a tracking point in such areas; as a result, the entire solution will be adapted with a spline or linear regression calculation for the expected driving direction. This function helps to find outliers and can bridge gaps in an image. Possible tracking points in line and close to the spline position will be used in the algorithm, and the robot can follow the spline interpolation over the gap areas. This process is the correct procedure for use in an area with excessive

numbers of plants, which is determined by the NDIV information. For missing plants, this strategy is justified if the gap is smaller than the robot. For larger gaps, the robot should ask the supervisor to drive around the gap or use additional image analysis techniques to ensure that holes in the ground are not present. Due to the reduced curve-driving capability of drilling machines, the interpolation function requires only several terms for fitting the curve shape. The main driving direction is the y direction in the image; therefore, the function depends on the y coordinate:

$$f(y) = a + by + cy^2 + dy^3 \tag{3}$$

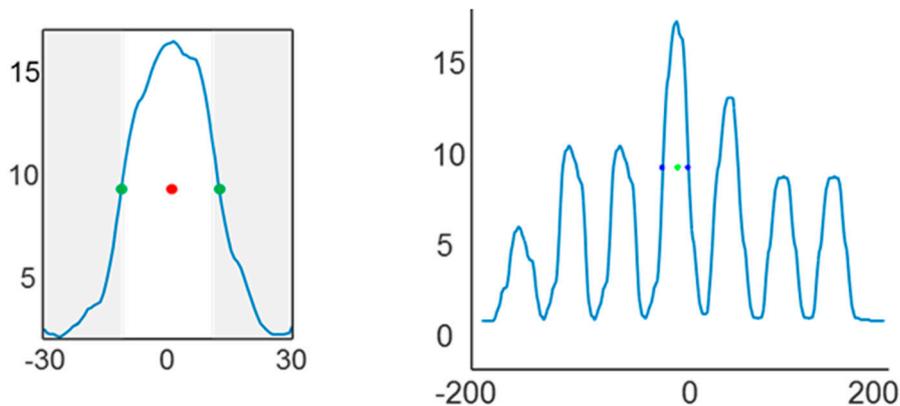


Figure 7. Search window at the center position of the cross-correlation result. The median is used to detect the nearest thickness points of the peak function, and the middle position is marked. Both plots were obtained from the results of a binary image. The gray areas indicate the warning regions for tracking point quality.

If the regression coefficient is inadequate, then historical tracking points from previous images can be used to strengthen the interpolation function. Therefore, the movement should be stored in memory, with at least the last image saved.

Additional error corrections can be implemented by using alternative information sources [17], such as gyroscopes, accelerators, magnetic compasses, barometers and GPS. This process is called “sensor fusion”, and in combination with calculated information from the sensor signals, it helps to resolve critical issues.

All needed parameters and functions for the cross correlation are summarized in the workflow diagram in Figure 8.

3.4. Embedded System

To offload the main computer of the robot system, the direction correction data should be calculated by an independent system. Many semiconductor chips can be used to perform such calculations at low cost and low power consumption. For simple solutions, it is impractical for the robot to transport a large computer workstation because of the increased power consumption and payload, although the advantage of this solution is the high degree of potential parallelization. XMOS has developed an XS1-L16A-128-QF124 microprocessor with 16 processor cores for \$20, and FPGAs are available from multiple companies at prices ranging from \$2 to \$10,000. The NVIDIA Tegra K1 chip has quad ARM

processors and 192 graphics processing units (GPUs), and a computer-like evaluation kit (Jetson TK1) costs €170. However, a more optimized solution is to use modern software to design and evaluate application specific optimized processors.

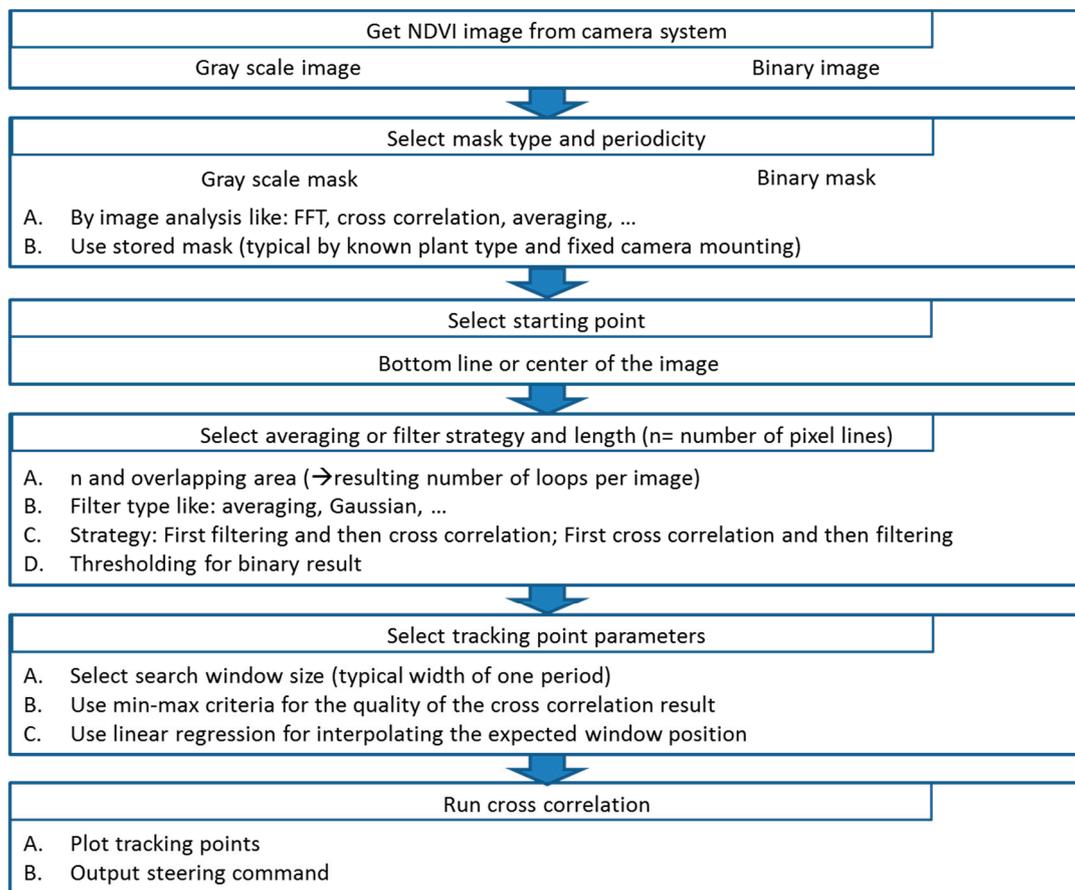


Figure 8. Simplified workflow diagram for the main path.

The required processing architecture for robot applications must be heterogeneous because both robotics control applications as well as dataflow-oriented applications are used. Robot control is clearly part of the application, which is more control-flow oriented. Therefore, the most suitable target architecture is a standard central processing unit (CPU) and varies according to the algorithm used for image processing. Here, a clear data-flow orientation is most suitable; therefore, hardware with maximal parallelization would be most beneficial to host this part of the application and specific processor architectures, such as very long instruction word (VLIW) processors, GPUs and FPGAs, are most suitable. Because the FPGA architecture can combine control- and data-flow oriented processors, this hardware is most suitable for the target application described in this paper. Promising FPGA architecture is found in the Zynq platform by Xilinx, which combines a dual core ARM 9 processor with reconfigurable hardware and a number of standard interfaces, such as the controller area network (CAN), Peripheral Component Interconnect Express (PCI Express), Serial Peripheral Interface (SPI), analogue, and a high number of digital I/Os [23]. More suitable architecture, Zynq Ultrascale, will be available in future and will include an ARM Mali GPU, quad core ARM A53, dual core R5 and many more features.

The aforementioned architectures exhibits the trends of current and future embedded system platforms, which clearly follow the trend of heterogeneity in terms of their processing units because of

applications that have different requirements, particularly for embedded applications. These requirements can be either functional or non-functional. Functional requirements include specific algorithms that deliver results according to a quality of service request (e.g., image resolution, frame rate), whereas nonfunctional requirements include real-time operation, high throughput, high reliability, and availability; these functions are crucial because real-time requirements are essential for the proper and safe operation of a system. Most of the recent achievements have enabled embedded platforms to measure and control themselves to adapt to these requirements, even during run-time operations. [24]. Here, a specific adaptation of the processing element is used to self-tune the complete architecture according to the current status of the processor and environment [25] to resolve the issue of a static computing architecture that cannot be optimized for a specific and dynamic application. These architectures are able to self-tune the processor, accelerator and specific interface cores using the reconfigurable portion of the chip by exploiting dynamic and partial reconfigurations. Here, a chip component is updated during run-time operations, whereas the rest of the chip remains in operation. This feature enables chip configurations according to the changing requirements of an application, thus increasing the flexibility of an embedded system tremendously.

4. Results

Figures 9–11 illustrate the cross-correlation algorithms implemented on plant images with real in-field conditions. Figures 9 and 10 use the average of 20 lines and were calculated every 10 lines. Figure 9 shows two scenes with small angles and middle position errors, and the mounting height and viewing direction provide an adequate foresight that is typically 20 m.

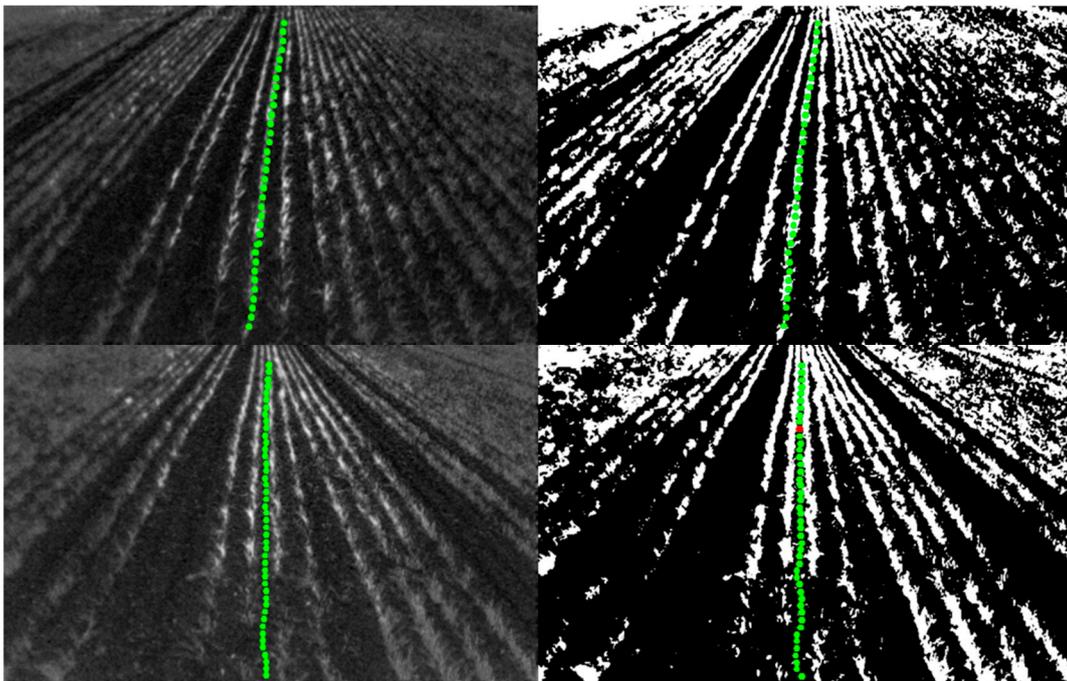


Figure 9. Two winter wheat scenes photographed using the low-cost plant camera. The left side shows a gray-scale NDVI image, and the right side shows a binary image. Both sides are overlain with the individually calculated tracking points.

Figure 9 shows that the tracking points exhibit small variations from a straight line. The differences from a linear regression line and between grayscale and binary images are discussed later. For this camera perspective, the tracking point presents excellent following of the plant lines, and variations from the linear regression are small (see Table 1). Figure 10 shows a similar field scene but with different mounting angles. The resulting images include the horizon and have a maximum field of view, although with higher restrictions for pixel resolutions in the upper 30% of the images.

Figure 10 demonstrates the enormous potential foresight of this plant-based navigation solution, with the algorithm losing tracking at several pixel lines before the horizon. Nevertheless, tracking is possible for over 50 m using a low-resolution camera. Figure 11 shows two field scenes with curves in the plant line.

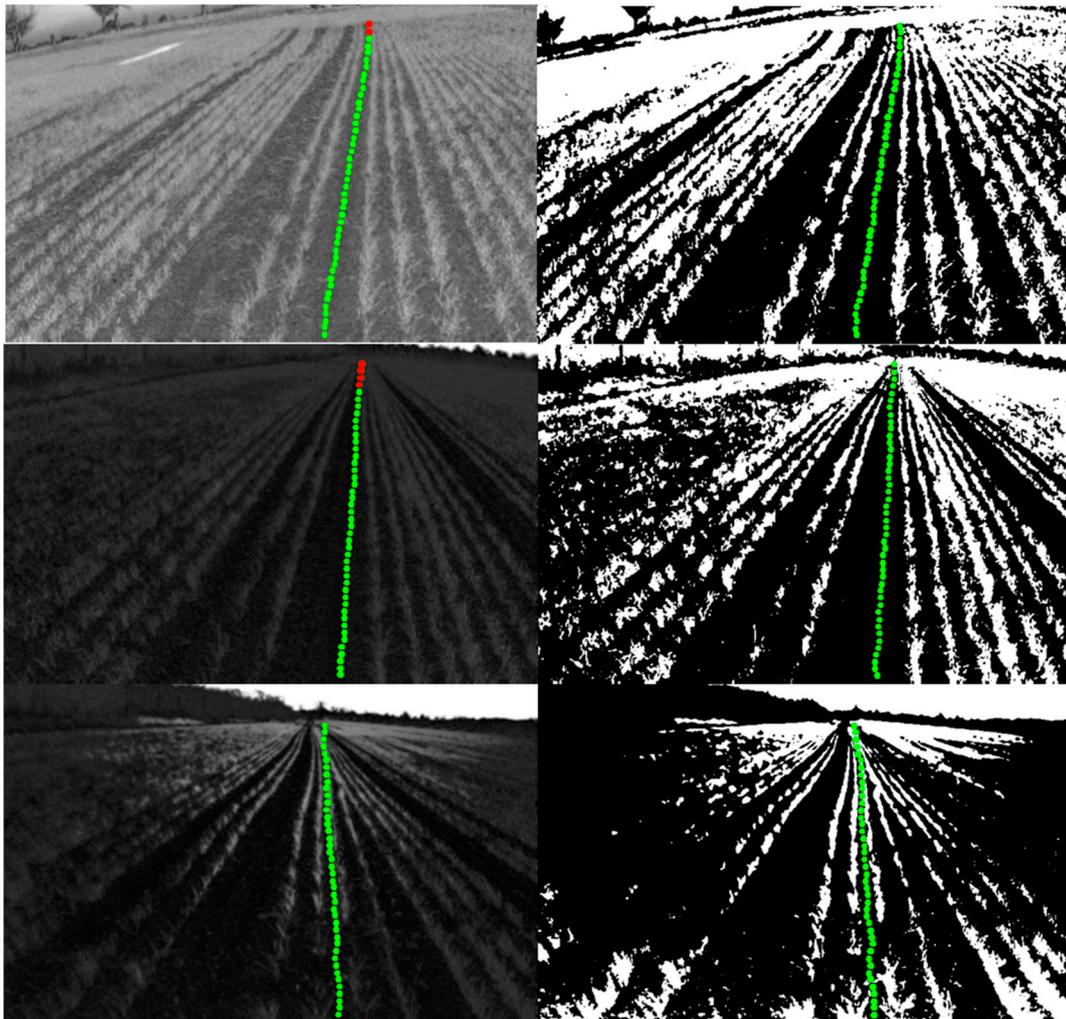


Figure 10. Three winter wheat scenes photographed using the low-cost plant camera. The left side shows a gray-scaled NDVI image, and the right side shows a binary image. Both sides are overlain with the individually calculated tracking points. Scenes one and two in the grayscale images show two and four red tracking points, respectively, which indicate exceedance of the warning level and an excessively small difference between the maximum and minimum.

The results in Figure 11 demonstrate that the algorithm can find tracking points without requiring straight lines in the image. The fitting curves demonstrate the good predictive potential of future

directions, even with poor quality images in which the bright-sky pixel intensities reduce the dynamic range for infield pixels. Figure 12 shows two extreme situations for which the field of view is inadequate for this application.

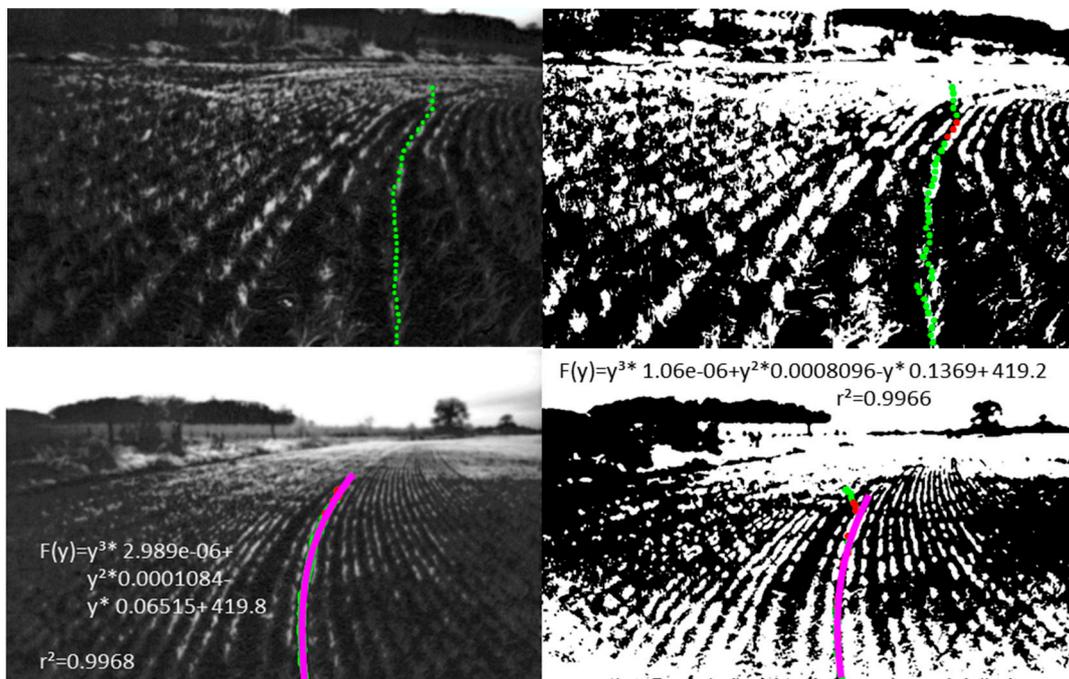


Figure 11. Two winter wheat scenes with a curved path. The left side shows the gray-scaled NDVI image, and the right side shows the binary image. Both sides are overlain with the individual calculated tracking points. Scene 1 uses an average of over 11 lines for the gray-scale image, over 15 lines for the binary image and a three period mask. Scene 2 uses an average of over seven lines for both the gray-scale and binary image and a four period mask. Scene 2 is overlain with a third-order fit.

Figure 12 demonstrates the robustness of the algorithm, even under the worst image quality conditions. Here, the viewing angle is too flat and the image contrast is reduced by the bright-sky pixels. In addition, the size of the plants is highly variable, and larger gaps with missing plants are observed. Under these extreme conditions, it is important to determine if the mask is appropriate for the actual scene or image. The width of the mask can be analyzed with additional loops that vary the width. The median level in Figure 7 is caused by the maximum and minimum values in the search window, and this difference also indicates the quality of the tracking point, is used as a warning signal and can be used to manipulate the mask size for the cross-correlation calculation. Figure 13 shows the position of the maximum difference for the calculations using masks varying from -2 to $+2$ pixel widths per period. Multiple points at the $+2$ level indicate that the mask should be wider, and multiple points at the -2 level indicate that the mask should be narrower.

As shown in Figure 11, a third-order fit is a simple but adequate function for interpolating or evaluating individual tracking points. Table 1 presents the R^2 and root mean square error (RMSE) statistics for the third-order regression for the demonstrated scenes.

A stability index with values greater than 0.9 demonstrates accuracy in the tracking points determined with the cross-correlation application. The binary results exhibit nearly the same stability as long as the image quality is adequate. For low-quality images from Scenes 6 to 9, the binary results differ from the gray-scale results. Therefore, the faster binary approach requires an adequate camera mounting and good plant camera system with efficient binary results independent of lighting [18]. The first scene shows an orthogonal tracking line, for which the R^2 value is useless for typical straight driving directions; as a result, the RMSE value should be used as the quality indicator. With respect to the different viewing angles and view fields, the RMSE values at the starting point can be compared after normalization to the given plant line distance of 160 mm. This line distance is equivalent to the mask periodicity.

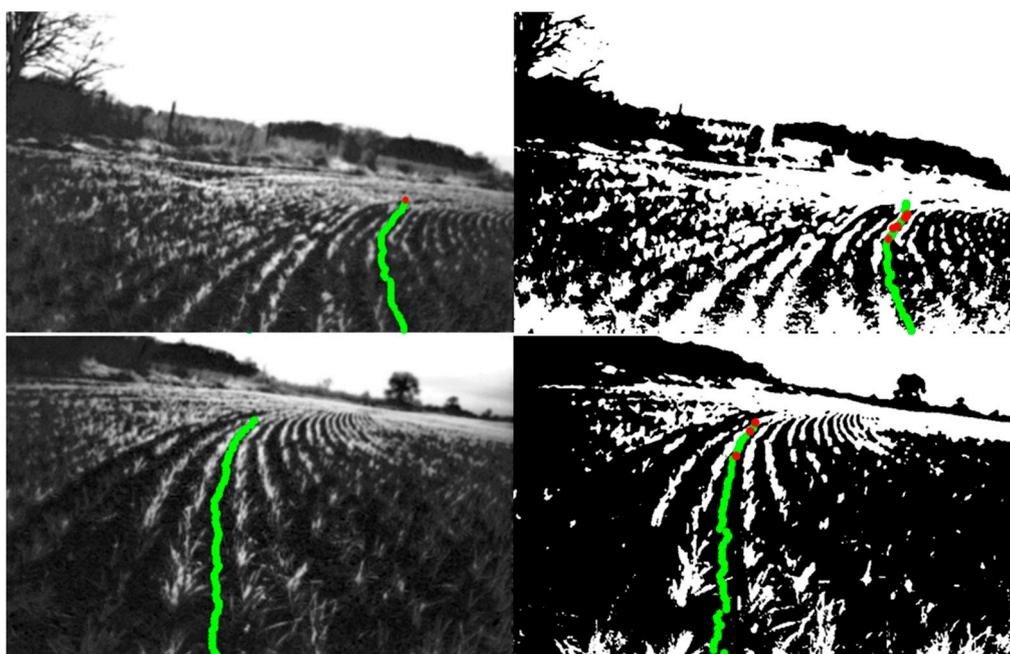


Figure 12. Two winter wheat scenes with both curved and flat viewing directions. The left side shows the gray-scaled NDVI images, and the right side shows the binary images. Both sides are overlain with the individual calculated tracking points.

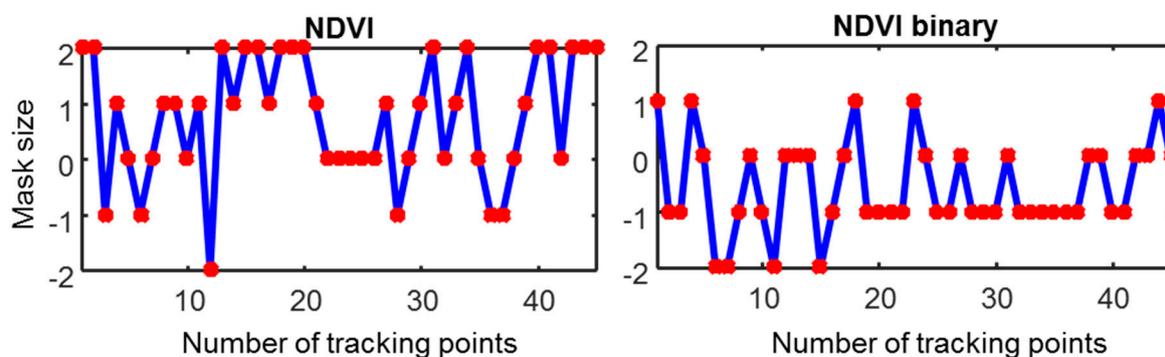


Figure 13. The quality of the mask size is indicated by the maximum difference between the maximum and minimum values in the search window for each tracking point. The mask size varies from +2 to -2 additional pixels for the periodic structure of the mask. This example plot was produced from the first image in Figure 10.

Table 1. Results for the third-order regressions.

Scene	R ²		RMSE		Pixel/Row	Normalized RMSE in mm	
	Gray	Binary	Gray	Binary		Gray	Binary
1	0.3362	0.1477	1.508	2.299	64	3.77	5.75
2	0.9962	0.9909	0.9015	0.9328	47	3.07	3.18
3	0.9984	0.9935	0.8452	1.345	64	2.11	3.36
4	0.9904	0.9753	0.870	1.071	59	2.36	2.90
5	0.9728	0.9205	1.758	1.210	77	3.65	2.51
6	0.9303	0.5164	2.844	4.820	80	5.69	9.64
7	0.9968	0.9866	0.8468	1.029	38	3.57	4.33
8	0.9463	0.9744	2.726	1.957	58	7.52	5.40
9	0.9644	0.6476	2.365	4.021	114	3.32	5.64

Binary Results

The binary images should be processed by appropriate algorithms in the plant camera system; however, this processing is beyond the scope of this paper. Regardless of how the binary images are obtained, proper binary image processing improves the performance of the cross-correlation because the resulting binary images exhibit sharper peaks as shown in Figure 14.

Figure 15 shows a small terrain effect in the image. Variations from the perspective function used to shrink the masks width indicate changes in the terrain.

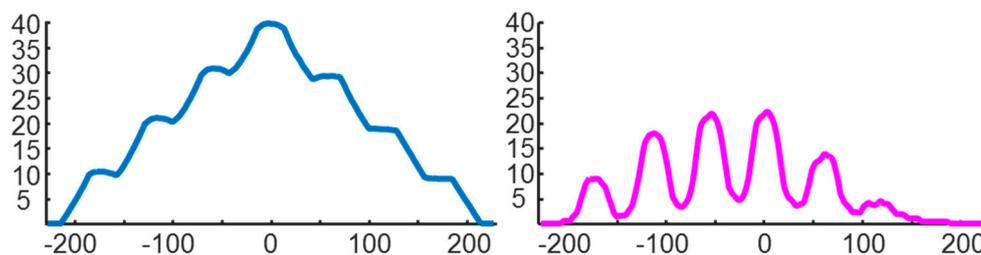


Figure 14. Cross-correlation results from one averaged pixel line over a window size with three periods. The left side shows the gray-scale results, and the right side shows the binary results.

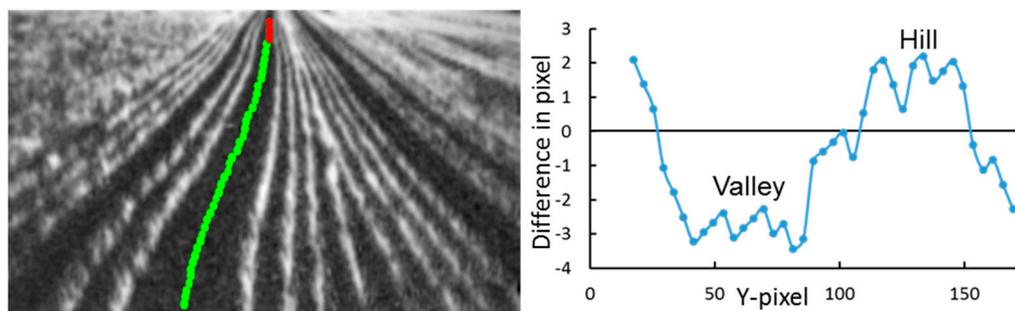


Figure 15. Terrain effect is indicated by a difference between the linear mask shrinking by the perspective in the image, and the mask size with highest maximum. Below zero indicates a valley and above zero indicates a hill.

The diagram in Figure 15 demonstrates the possibility for detecting terrain features. For a more difficult situation with a curve, the mask shrinking factor caused by the curve must be additionally considered.

5. Conclusions

The combination of a plant camera and cross-correlation algorithm results in a robust in-field navigation solution for robots working with sensitive plants. This solution uses the plant lines themselves to follow precise tramlines. The results of this study demonstrate that the proposed approach avoids driving over plants and provides accurate navigation control in the centimeter range. In addition, the proposed approach overcomes the issue of jumps that hinder GPS-driven solutions. Because plant lines were drilled with large agricultural machinery, the minimum curve radius is restricted, which provides a number of possibilities for reducing the power required to calculate the algorithms.

1. Image lines in the x direction can be concentrated by averaging in the y direction.
2. The cross-correlation function does not have to move over the entire pixel line.
3. The moving mask of the cross-correlation can be reduced to a few periodic replications with a rectangular shape, thereby reducing the length of the used pixels and number of multiplications.
4. The reduced mask must only move over a length smaller than one period to have only one maximum peak of the cross-correlation in the inspection window.
5. Missing tracking points can be interpolated using a simple linear regression function.

In addition, the required calculation power can be reduced using the binary result from the plant camera. The multiplied binary image line and a binary mask can be replaced by the logical conjunction “AND”, which can be performed in parallel during one clock cycle in a FPGA.

All of the algorithms required for this application are reduced to multiplications and summations, which is an important point for implementing algorithms in a small embedded system with restricted resources. All line averaging can be performed in parallel, and for reduced packages, even the cross-correlation can be performed in parallel. The inspection window for the cross-correlation must follow shifts in the tracking points and is therefore a cascaded operation. After two or three parallel correlations, the shift must be added and then the new run starts with the shifted window position. Using this window provides a substantial advantage in potential error detection. If the calculated tracking points fall in the edge region of the window or if the difference between the maximum and minimum value in the window is too small, then these tracking points can be ignored. A simple linear regression can be used to calculate a tracking function, and outlier tracking points can be overbridged, which also illustrates the robustness of this navigation solution. The detailed description of this application and its resulting simplicity are significant advantages in the establishment of steering control in small embedded systems that offload the main system of the field robot. The steering command is not the only output of the solution, and the foresight provides additional information on field conditions, such as direction, hills and valleys, assumed obstacles, and the field ends. Such additional information is important for providing correct overall field driving plans or management. When this information is combined with sensor signals, such as gyros and accelerators, the main system can determine if the approaching hill has too great of an ascending slope for the robot.

The next step in this line of study will be changing the correlation direction from the x axis to the orthogonal axes of the fitted line. This modification could enable the application to follow sharper curves

in the image because the averaging of lines in the y direction can filter out the plant lines, while the averaging in the expected direction cannot perform this filtering. Figure 16 shows the difference between the averaging directions. In addition, the cross-correlation will output higher peaks in the expected direction, thus improving the quality of the tracking points.

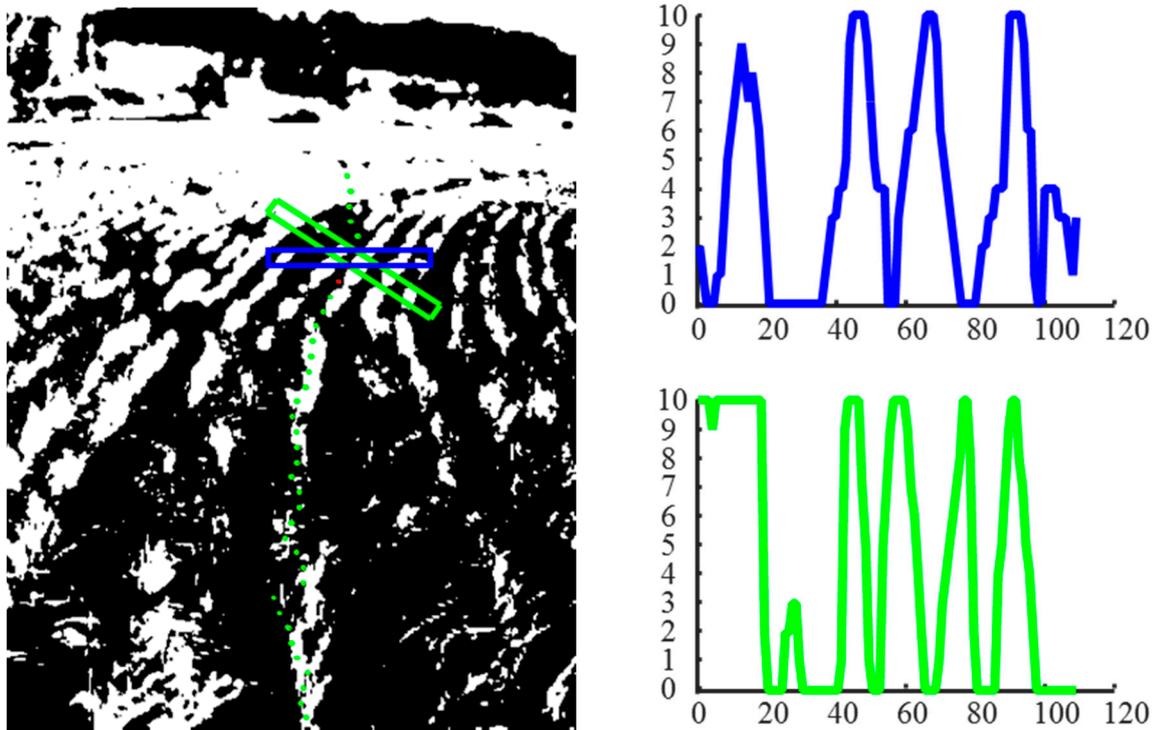


Figure 16. Different averaging directions determining different resulting input functions for the cross correlation. Upper plot is the result of the averaging in y-direction. Lower plot is the result of the 33° direction.

Acknowledgments

We thank Mathias Hoffmann for his work during the measurement campaign and for his support for the first MATLAB implementation.

Author Contributions

Volker Dworak conceived, designed, and performed the experiments. Volker Dworak, Michael Hübner, and Jörn Selbeck analyzed the data and programmed the solution in a MATLAB environment. Volker Dworak and Michael Hübner wrote the paper.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Shi, Y.Y.; Wang, N.; Taylor, R.K.; Raun, W.R.; Hardin, J.A. Automatic corn plant location and spacing measurement using laser line-scan technique. *Precis. Agric.* **2013**, *14*, 478–494.
2. Astrand, B.; Baerveldt, A.J. A vision based row-following system for agricultural field machinery. *Mechatronics* **2005**, *15*, 251–269.
3. Olsen, H.J. Determination of row position in small-grain crops by analysis of video images. *Comput. Electron. Agric.* **1995**, *12*, 147–162.
4. Máthé, K.; Buşoni, L. Vision and control for UAVs: A survey of general methods and of inexpensive platforms for infrastructure inspection. *Sensors* **2015**, *15*, 14887–14916.
5. Wallace, R. *Achieving Optimum Radio Range*; Application Report from Texas Instruments Incorporated SWRA479: Dallas, TX, USA, 2015.
6. Norremark, M.; Griepentrog, H.W.; Nielsen, J.; Sogaard, H.T. The development and assessment of the accuracy of an autonomous GPS-based system for intra-row mechanical weed control in row crops. *Biosyst. Eng.* **2008**, *101*, 396–410.
7. Mathanker, S.K.; Maughan, J.D.; Hansen, A.C.; Grift, T.E.; Ting, K.C. Sensing miscanthus swath volume for maximizing baler throughput rate. *Trans. ASABE* **2014**, *57*, 355–362.
8. Molin, J.P.; Colacco, A.F.; Carlos, E.F.; de Mattos, D. Yield mapping, soil fertility and tree gaps in an orange orchard. *Revista Brasileira de Fruticultura* **2012**, *34*, 1256–1265.
9. Jiang, G.Q.; Wang, Z.H.; Liu, H.M. Automatic detection of crop rows based on multi-ROIs. *Expert Syst. Appl.* **2015**, *42*, 2429–2441.
10. Bakker, T.; Wouters, H.; van Asselt, K.; Bontsema, J.; Tang, L.; Muller, J.; van Straten, G. A vision based row detection system for sugar beet. *Comput. Electron. Agric.* **2008**, *60*, 87–95.
11. Torres-Sospedra, J.; Nebot, P. A new approach to visual-based sensory system for navigation into orange groves. *Sensors* **2011**, *11*, 4086–4103.
12. Fernandez, J.; Calavia, L.; Baladron, C.; Aguiar, J.M.; Carro, B.; Sanchez-Esguevillas, A.; Alonso-Lopez, J.A.; Smilansky, Z. An intelligent surveillance platform for large metropolitan areas with dense sensor deployment. *Sensors* **2013**, *13*, 7414–7442.
13. Dworak, V.; Selbeck, J.; Ehlert, D. Ranging sensors for vehicle-based measurement of crop stand and orchard parameters: A review. *Trans. ASABE* **2011**, *54*, 1497–1510.
14. Martínez, M.A.; Martínez, J.L.; Morales, J. Motion detection from mobile robots with fuzzy threshold selection in consecutive 2D Laser scans. *Electronics* **2015**, *4*, 82–93.
15. Sanz, R.; Rosell, J.R.; Llorens, J.; Gil, E.; Planas, S. Relationship between tree row LIDAR-volume and leaf area density for fruit orchards and vineyards obtained with a LIDAR 3D Dynamic Measurement System. *Agric. For. Meteorol.* **2013**, *171*, 153–162.
16. Hoefle, B. Radiometric correction of terrestrial LiDAR point cloud data for individual maize plant detection. *IEEE Geosci. Remote Sens. Lett.* **2014**, *11*, 94–98.
17. Hague, T.; Marchant, J.A.; Tillett, N.D. Ground based sensing systems for autonomous agricultural vehicles. *Comput. Electron. Agric.* **2000**, *25*, 11–28.
18. Dworak, V.; Selbeck, J.; Dammer, K.H.; Hoffmann, M.; Zarezadeh, A.A.; Bobda, C. Strategy for the development of a smart NDVI camera system for outdoor plant detection and agricultural embedded systems. *Sensors* **2013**, *13*, 1523–1538.

19. Gebbers, R.; Tavakoli, H.; Herbst, R. Crop sensor readings in winter wheat as affected by nitrogen and water supply. In *Precision Agriculture'13*; Stafford, J.V., Ed.; Wageningen Academic Publishers: Gelderland, The Netherlands, 2013; pp. 79–86.
20. Tillett, R.D. Image analysis for agricultural processes: A review of potential opportunities. *J. Agric. Eng. Res.* **1991**, *50*, 247–258.
21. Peteinatos, G.G.; Weis, M.; Andujar, D.; Ayala, V.R.; Gerhards, R. Potential use of ground-based sensor technologies for weed detection. *Pest Manag. Sci.* **2014**, *70*, 190–199.
22. Billingsley, J.; Schoenfisch, M. Vision-guidance of agricultural vehicles. *Auton. Robot.* **1995**, *2*, 65–76.
23. Zynq-7000 AP SoC Technical Reference Manual, UG585 (V1.10). Available online: www.xilinx.com (accessed on 23 February 2015).
24. Janßen, B.; Schwiegelshohn, F.; Hübner, M. Adaptive computing in real-time applications. In Proceedings of the 13th IEEE International NEW Circuits And Systems (NEWCAS) Conference, Grenoble, France, 7–10 June 2015; pp. 166–173.
25. Janssen, B.; Mori, J.Y.; Navarro, O.; Gohringer, D.; Hubner, M. Future trends on adaptive processing systems. In Proceedings of the 12th IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA 2014), Milan, Italy, 26–28 August 2014.

© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).