CORN PLANT LOCATION, SPACING AND STALK DIAMETER MEASUREMENTS USING OPTICAL SENSING TECHNOLOGIES

By

YEYIN SHI

Bachelor of Science in Mechanical Engineering
Nanjing Forestry University
Nanjing, Jiangsu, China
2007

Master of Science in Agricultural Engineering
Oklahoma State University
Stillwater, OK
2009

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
May, 2014

CORN PLANT LOCATION, SPACING AND STALK

DIAMETER MEASUREMENTS USING

OPTICAL SENSING TECHNOLOGIES

Dissertation Approved:

Dr. Ning Wang

Dissertation Adviser

Dr. Randal K. Taylor

Dr. Paul Weckler

Dr. William R. Raun

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my graduate advisor, Dr. Ning Wang, for her absolute long steady support and great advice throughout my graduate level education; and other committee members, Dr. William Raun in Plant and Soil Science Dept. for the funding support of this study; Dr. Randy Taylor and Dr. Paul Weckler, for their continuous support and the achievements I have been fortunate to obtain under their advisement.

I would also like to express appreciation to Aaron Franzen, Mike Welderman, Wayne Kiner and other staff members working in the lab for their help with the system building whenever I turned to them. Recognition goes to the co-workers who assisted in the field experiment: Natasha Macnack, Jeremiah Mullock in Plant and Soil Sciences Dept. for their work in the plot preparation and ground truth data collection in field; Jingcheng Zhang, Wesley Porter, Jorge Rascon, Bin Li, Michael Chavez, and Yongbo Wan for their great help on the field tests.

Last, but not least, I would like to thank my beloved parents and Jian for their understanding, support and encouragement along with my difficult but always determined career path far away from them.

Name: YEYIN SHI

Date of Degree: MAY, 2014

Title of Study: CORN PLANT LOCATION, SPACING AND STALK DIAMETER MEASUREMENTS USING OPTICAL SENSING TECHNOLOGIES

Major Field: BIOSYSTEMS AND AGRICULTURAL ENGINEERING

Abstract: Plant within-row spacing and stalk diameters at mid-growth stages have been demonstrated to be important variables in the by-plant yield prediction model for corn. This information would help advising the in-season variable-rate fertilizer application to increase the fertilizer use efficiency. Little study could be found in developing an automatic, non-invasive and high spatial resolution system to measure these variables at the desired growth stages. A three-year study was conducted on this topic. The overall goal of this study was to investigate the feasibility of various machine vision technologies and to develop prototype systems for corn plant location, within-row spacing and stalk diameter measurements at their mid-growth stages.

In Phase I and II of this study, a system for plant location and within-row spacing measurements based on the LiDAR technology was developed and improved. In Phase III, a system for plant stalk diameter measurement using two different approaches was developed – the LiDAR-and-RGB Approach was a combination of a LiDAR sensor and a webcam; the 3D Range Imaging Approach used a 3D range camera. In each system, sensors were mounted on a cart and viewing horizontally at the lower sections of plant stalks. These systems featured with their abilities of viewing each plant from multiple angles when the sensors were passed by, which largely increased the possibility of correct identification. At each phase, the system was tested in the field condition. Data processing algorithms were developed to identify potential stalks in a laser scan, a RGB image or a distance image; and to register information between scans or different sensors.

A total error of 5.5% in plant counting and a 1.9 cm of root-mean-squared error (RMSE) in the spacing measurement were achieved between the sensor measurements and the manually measured ground truth for data collected in year 2012. The RMSE of diameter measurement were 4.1 mm and 3.9 mm for the LiDAR-and-RGB Approach and 3D Range Imaging Approach, respectively. This study was a good basis of developing a high spatial resolution corn plant within-row spacing and stalk diameter sensing system for real-time, variable-rate fertilizer application.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Background

Nitrogen fertilizer is one of the most important inputs in agricultural production, accounting for almost two thirds of the total fertilizer consumption. As the world's largest corn producer, nearly half of the total nitrogen fertilizer is used for corn production in United States (ERS, USDA, 2012). An issue related to nitrogen fertilizer application is low nitrogen use efficiency (NUE) – around 33% in world cereal grain production (Raun and Johnson, 1999) and 37% in US corn production (Cassman et al., 2002). Excessive nitrogen loss is not only economically costly, but also the cause of environmental problems such as eutrophication in water body and greenhouse gas emission. Research and technology to minimize nitrogen loss and increase NUE is significant for the sustainability of the living environment that human beings rely on.

One of the major reasons of low NUE is the poor synchrony between nitrogen supply and crop nitrogen demand in terms of application timing and rate. If the nitrogen is not applied at the time when a plant is fast taking it up, the chance of leaching, surface runoff or other types of nitrogen loss would increase. Also, if nitrogen is applied more than the rate that is needed by a plant to achieve an optimal yield, the excessive nitrogen applied would be wasted. These findings motivated the research and adoption of in-season variable-rate nitrogen application at precise

timing and rate to replace the traditional pre-planting nitrogen application at a fixed rate through a whole field. Problems needed to be answered in variable-rate nitrogen application are how to decide the rate and what the application resolution should be.

Models have been developed by finding the correlation between crop attributes (independent variables), and yield (dependent variable) using the historic data in previous years. These models are then used to predict the yield in-season by knowing the current crop attributes. In-season nitrogen application rates vary according to the predicted yield, assuming the better the yield the more nitrogen the crop would uptake. Crop attributes vary spatially. Research demonstrated that the optimal spatial scale of nitrogen application for corn management was less than 0.5 m or by-plant (Martin et al., 2005), and the combination of corn plant chlorophyll content, plant height, interplant spacing and plant stalk diameter were highly correlated with its grain yield and biomass (Martin et al., 2012; Kelly, 2011). Technologies to automatically sense these plant attributes are demanded to replace the time-consuming manual data collection and incorporate with other field operations. This study focuses on automatic measuring within-row plant spacing and plant stalk diameters.

Though little research can be found on in-field corn plant stalk diameter measurement, various technologies have been studied or already applied for plant counting and spacing/population measurements. Remote sensing is a way to obtain general large-scale information rapidly, while ground-based method is used to collect detailed crop and soil information and usually can be incorporated with production operations. Corn plant counting systems on combine harvesters usually use mechanical method, infrared proximity sensing or capacitance sensing. For early and mid-growth stage plants, non-invasive sensing method is commonly used in order to have minimum impact on plants. Optical sensing is a widely used non-invasive sensing technique usually has less limitation in terms of the sensing distance, object orientation and sensing speed comparing with other non-invasive techniques.

As one of the most common optical sensing methods, machine vision technique includes 2D imaging and range sensing. Several studies have been conducted successfully based on 2D color images taken from the top view for early growth stage corn plant counting and spacing measurement (Shrestha and Steward, 2003 and 2005; Tang and Tian, 2008a and b). Individual plant was recognized by analyzing the collected color images. However, using color images collected from top view is not a best way to measure corn interplant spacing at mid growth stage when the in-season nitrogen application is usually operated due to canopy overlap. To overcome this, color images collected from side view is a better option. As for the stalk diameter measurement, RGB imaging is not enough because depth information is needed to estimate the actual size of an object. Another optical sensing technique – range sensing – can compensate this shortcoming.

Range sensing technique, especially laser line scanning, has been widely used in orchard studies to estimate canopy volume and density of trees. It has not been studied much for corn plant sensing yet. Previous studies conducted for corn population estimate (Luck et al., 2008) and plant counting and interplant spacing measurement (Rascon, 2012) using laser pointer sensors showed the prospect of using range sensing technology on this application. However, they also indicated the interference of leaves on the measurement accuracy and a lack of information to eliminate such interference. 3D range imaging is a newly developed range sensing technology. A 3D range camera outputs a depth reading as well as a gray-scale reading of each pixel in an image. Study has been conducted successfully using a 3D range camera to measure corn interplant spacing at early growth stage from the side view (Nakarmi and Tang, 2010). So far, the 3D range cameras for outdoor application have to use laser light source and are in fairly high cost. Most of the 3D range cameras used in industry and agriculture use LED light sources which have lower light intensity. Both sensor and plants need to be shaded during the field tests.

A LiDAR (Light Detection and Ranging) sensor is a compromise of a low-cost laser pointer sensor and a high-cost 3D range camera. It completes a line scan in very short time and provides multiple distance measurements along the scan. The LiDAR sensors are the most widely investigated range sensors in agriculture applications nowadays. They are involved in autonomous guidance on tractors, yield estimate and variable-rate spraying in orchard and cereal production (Lee et al., 2007; Saeys, et al., 2009; Chen, et al., 2012). The LiDAR sensor may have the potential for corn plant spacing and stalk diameter measurements.

In summary, automatic corn plant spacing and stalk diameter estimates at mid-growth stages is important for in-season variable-rate nitrogen applications. Development of real-time sensing systems which are accurate and feasible for field operations are in a great need.

Research Objectives

The overall objective of this study is to develop systems based on optical sensing technologies for automatic corn plant location, spacing and stalk diameter measurements to facilitate in-season variable rate nitrogen applications. The specific objectives include:

- Phase I: to develop a system based on LiDAR technique with corresponding data processing algorithms to estimate corn plant location and spacing;
- Phase II: to improve the overall system for plant location and spacing measurement and evaluate the system with more field experiment;
- Phase III: to develop two approaches to estimate corn plant stalk diameter: one system is a combination of RGB imaging and LiDAR technique; the other system is based on 3D range imaging technique; develop corresponding data and image processing algorithms for each approach; and compare and evaluate two approaches in terms of their performances on plant counting, spacing and stalk diameter measurements.

References

Cassman, K. G., Dobermann, A. R., Walters, D. T. (2002). Agroecosystems, nitrogen-use efficiency, and nitrogen management. Royal Swedish Academy of Sciences. Ambio, 31(2).

Chen, Y., Zhu, H., Ozkan, H. E., 2012. Development of a variable-rate sprayer with laser scanning sensor to synchronize spray outputs to tree structures. *Transactions of ASABE* 55 (3), 773-781.

ERS (Economic Research Service), USDA. 2012. Fertilizer use and price. http://www.ers.usda.gov/data-products/fertilizer-use-and-price.aspx. Verified 01/04/2013.

Kelly, J. (2011). By plant prediction of corn grain yield using height and stalk diameter. Master thesis. Oklahoma state university.

Lee, K. H., Ehsani, R., Schueller, J. K., 2007. Forward movement synchronization of two vehicles in parallel using a laser scanner. *Applied Engineering in Agriculture* 23 (6), 827-834.

Luck, J. D., Pitla, S. K., & Shearer, S. A. (2008). Sensor ranging technique for determining corn plant population. ASABE paper No. 084573. Providence, Rhode Island: ASABE.

Martin, K. L., Hodgen, P. J., Freeman, K. W., Ricardo Melchiori, Arnall, D. B., Teal, R. K., Mullen, R. W., Desta, K., Phillips, S. B., Solie, J. B., Stone, M. L., Octavio Caviglia, Fernando Solari, Agustin Bianchini, Francis, D. D., Schepers, J. S., Hatfield, J. L., & Raun, W. R. (2005). Plant-to-Plant Variability in Corn Production. Agronomy Journal, 97:1603–1611.

Martin, K., Raun, W., & Solie, J. (2012). By-plant prediction of corn grain yield using optical sensor readings and measured plant height. Journal of Plant Nutrition, 35:9, 1429-1439.

Nakarmi, A. D. and Tang, L. (2010). Inter-plant spacing sensing at early growth stages using a time-of-flight of light based 3D vision sensor. 2010 ASABE Annual International Meeting, Pittsburgh, Pennsylvania. Paper number: 1009216.

Rascon, J. (2012). Corn sensor development for by-plant management. Unpublished Master's thesis. Oklahoma State University.

Raun, W.R., & Johnson, G.V. (1999). Improving nitrogen use efficiency for cereal production. Agronomy Journal, 91:357–363.

Saeys, W., Lenaerts, B. Craessaerts, G. & De Baerdemaeker, J. (2009). Estimation of the crop density of small grains using LiDAR sensor. Biosystems Engineering, I02 (2009) 22-30.

Shrestha, D. S., Steward, B., and Bartlett, E. (2001). Segmentation of plant from background using neural network approach. In Intelligent Engineering Systems through Artificial Neural Networks: Proc. Artificial Neural Network Networks in Engineering (ANNIE) International Conference, 11: 903-908. New York, N.Y.: ASME Press.

Shrestha, D. S., & Steward, B. L. (2003). Automatic corn plant population measurement using machine vision. Transactions of the ASABE, 46(2): 559–565.

Tang, L., & Tian, L. F. (2008a). Real-time crop row image reconstruction for automatic emerged corn plant spacing measurement. Transactions of the ASABE, 51(3): 1079-1087.

Tang, L., & Tian, L. F. (2008b). Plant identification in mosaicked crop row images for automatic emerged corn plant spacing measurement. Transactions of the ASABE, 51(6): 2181-2191.

CHAPTER II

LITERATURE REVIEW

Importance of Corn

Corn (or maize) is one of the oldest human-domesticated plants. Its origins were grown in the form of a wild grass in Central Mexico dated back to seven thousand years ago. Corn is known as the third largest planted crop in the world after wheat and rice. It is mostly used as a primary feed crop – for instance it accounts for 95% of the total feed grain production and use in United States – but is also important as a food crop in many parts of the world, and in food processing for making starch, sweeteners, oil and beverage. Besides food and feed, nowadays corn has been playing an important role in industrial ethanol production.

World corn production was around 700 million tonnes in 2010, which shared about one-third of the total cereal production quantity. It has increased by nearly 50 percent in the past two decades. The five largest corn producers in the world in 2010 were: United States, 316.2 million tonnes; China, 177.5 million tonnes; Brazil, 55.4 million tonnes; Mexico, 23.3 million tonnes; and Argentina, 22.7 million tonnes (FAOSTAT, 2012a). United States is the world's largest producer, consumer and exporter of corn. Corn planted area in US is estimated at 35.6 million hectares which is the largest among all crop production (NASS, USDA, 2010). The US domestically used corn was around 352 million m^3, in which about 41% was used to produce animal feed; about 45% was used for industrial ethanol production; the rest was for human food, seed and other

usage (USDA, 2012).

Issues in Nitrogen Fertilizer Applications

Nitrogen fertilizer is a critical input in agricultural production. World nitrogen fertilizer consumption was 105 million tonnes in 2009 which accounted for almost two thirds of the total fertilizer consumption (FAOSTAT, 2012b). This nitrogen fertilizer consumption quantity was increased from 86 million tonnes in 2002 (FAOSTAT, 2012b). In United States, around 12 million tons of nitrogen was applied in agricultural production, in which 5.6 million tons was used for corn production (ERS, USDA, 2012).

An issue related to nitrogen fertilizer application is the low nitrogen use efficiency (NUE). NUE is defined as the total nitrogen removed by cereal plants, excluding the nitrogen coming from the soil and deposited from the rainfall, divided by the total fertilizer nitrogen applied (Raun and Johnson, 1999). The world cereal grain NUE is estimated at 33% (Raun and Johnson, 1999). Nitrogen uptake efficiency which is very similar to NUE was reported at 37% in corn production in north-central US, 31% in rice production in Asia (Cassman et al., 2002). Except for those nitrogen incorporated in soil organic and inorganic nitrogen pools, the unaccounted nitrogen is lost in ways of surface runoff by rain before the nitrogen fertilizer enters the soil, ammonia volatilization to the atmosphere, NO_3 denitrification to N_2 and N_2O gases, and leaching when dissolvable nitrate in soil moved by sufficient rain or irrigation to surface water or groundwater (Raun and Johnson, 1999; Ribaudo et al., 2011).

Excessive nitrogen in the environment has impacts on water resources, atmosphere and terrestrial resources. Concentration of total nitrogen is the highest in agricultural streams comparing with other landscapes (Dubrovsky et al., 2010). Excessive nitrate in drinking water can lead to low oxygen level in human body; excessive nitrogen in surface water can cause algal bloom (known as eutrophication) and decrease the dissolved oxygen in water that aquatic life rely on

(Dubrovsky et al., 2010; Rabalais et al., 2002). Agriculture fertilizer application is the largest source of greenhouse gas emission in US, accounting for 67.9% in 2010 (USEPA, 2012). The alteration of the nitrogen cycle also stimulated the uptake and storage of carbon stored within terrestrial ecosystem and further influenced the whole ecosystem (Vitousek et al., 1997).

Numerous researches have been conducted to investigate methods of reducing nitrogen loss to increase the NUE in agriculture production. These methods include adopting crop rotations, forage-only production systems, hybrid or cultivar with higher NUE, conservation tillage, soil injection rather than broadcasting, other forms of slow-release nitrogen fertilizer, proper irrigation, and precision farming strategy including applying the nitrogen at optimal timing and rate using proper decision-making algorithm (Raun and Johnson, 1999; Ribaudo et al., 2011; Shanahan et al., 2008).

In-season Variable Rate Nitrogen Applications

One of the major causes of low NUE is due to the poor synchrony between the nitrogen supply and the crop nitrogen demand in terms of application timing and rate (Raun and Johnson, 1999; Casman et al., 2002; Campbell et al., 1995). The traditional approach of nitrogen fertilizer application is to apply a uniform rate prior to crop planting; however, using corn as an example, the plant does not rapidly take up nitrogen until four weeks after its emergence (Baker, 2001), which increases the chance of nitrogen loss by leaching and denitrification. Nitrogen applications can be taken up by wheat plants efficiently late in the season during grain fill period without affecting the crop grain protein levels and decreasing soil nitrogen uptake (Wuest and Cassman, 1992). NUE for dryland winter wheat can be improved by in-season nitrogen application with point injection or topdressing rather than replant applications (Sowers et al., 1994). Maximum yields of corn can be achieved with delayed but sufficient in-season nitrogen application when nitrogen uptake by plant is greatest; or the yield is still highly responsive to nitrogen application

level even if the full yield could not be achieved (Varvel at al., 1997; Binder et al., 2000; Scharf et al., 2002). The nitrogen application rate also has a major effect on NUE in crop production. NUE decreases with increasing nitrogen level especially under drier soil conditions (Gauer et al, 1992). Nitrate leaching can be significant when nitrogen is applied at rates in excess of that needed for maximum yield (Raun and Johnson, 1995). These reasons support the adoption of in-season variable rate nitrogen application in crop production.

It is critical to determine an optimal spatial scale for variable rate nitrogen application for in-field variability management. The optimal spatial scale in wheat production is at 1 m^2 (Solie et al., 1999). For corn production, Martin et al. (2005) evaluated by-plant grain yield variability in the USA, Argentina and Mexico from 2002 to 2004. Their study found that average plant to plant grain yield differed by 2,765 kg/ha. They also found that variability was not significant if the yield was averaged along the row over a scale greater than 0.5 m. These results indicate that high-resolution plant management protocols may have significant impact in corn production.

Parameters commonly considered when investigating variability management are soil nutrient level, soil moisture, plant nitrogen content, plant population or spacing, plant height, canopy coverage or volume, and canopy density. In corn production, variability of plant nitrogen content, plant population or spacing, and plant height are often examined. Krall et al. (1977) found that every 2.5 cm increase in the standard deviation of plant spacing would decrease the yield by 210 kg/ha. Lauer and Rankin (2004) found a yield loss at 1.06% with every centimeter increase in the standard deviation of corn plant spacing when standard deviation was larger than 12.0 cm. A corn by-plant yield prediction model proposed by Martin et al. (2012) included plant height, plant spacing and normalized difference vegetative index (NDVI, indicating plant nitrogen content) and achieved an R^2 of 0.48. The previous model without using plant height and spacing had an R^2 of 0.22. Another parameter recently being investigated is corn stalk diameter. Kelly (2011) found

the index of 'stalk diameter × plant height' correlated well with corn grain yield with an R^2 of 0.34 at V8, 0.55 at V10 and 0.67 at V12.

Sensing Technologies for Plant Spacing/Population and Diameter Measurement

The approaches of plant population or interplant spacing measurement can be categorized as two types: airborne and ground-based (Dworak et al. 2011). Most of the airborne remote sensing approaches use hyperspectral or multispectral analysis to rapidly obtain large scale data (GopalaPillai and Tian, 1999; Huang et al. 2010; Thorp et al. 2008). Ground-based sensing methods have been used to obtain detailed crop and soil information and conducted concurrently with other in-field operations such as planting, spraying or harvesting. Ground-based approaches to plant population or spacing measurements can be further categorized as intrusive (mechanical methods) or non-intrusive methods.

Mechanical methods to measure corn plant population usually use the resistant force of stalks on a spring loaded arm or a gravity pendulum to count the number of stalks (Birrell and Sudduth 1995; Heege and Thiessen 2004). Some of these methods have already been commercialized on combine harvesters. Non-intrusive methods are more suitable for sensing corn population at early and mid-growth stages. Some of these methods are based on capacitive sensing: Nichols (2000) invented a moisture detecting sensor installed on a combine to count harvested stalks; Li et al. (2009) developed a capacitance-based biomass proximity sensor to count corn stalks during harvesting.

Optical sensing techniques (including image-based sensing) have been playing important roles in plant population/spacing measurements. Image-based sensing is one category in it. Shrestha and Steward (2003, 2005) developed and improved a machine vision based corn plant population sensing system. Algorithms were developed for color image sequencing, segmentation and plant recognition in order to count corn plants, and to estimate plant location and intra-row spacing.

Their system resulted in a 5.4% coefficient of variation for the standard error in population estimates in 2003, and 6.2% root mean square error (RMSE) in 2005. Tang and Tian (2008a, 2008b) developed a real-time crop row image reconstruction and plant identification system for automatically measuring the spacing of emerged corn plants. They achieved an overall RMSE of 1.7cm and an R^2 of 0.96. All of these studies targeted early growth stage corn plants prior to canopy closure.

Range sensing is also an optical sensing technique that has been applied to crop parameter measurements. Wangler et al. (1994) patented a laser scanning sensor which could be attached on a sprayer to selectively spray according to the presence of the tree foliage. Laser scanning technique was also used for tree foliage density and wheat stand density estimation by calculating variation in laser penetration depth (Wei and Salyani 2004, 2005; Saeys et al. 2009; Chen et al., 2012). Luck et al. (2008) used an infra-red range sensor for in-field plant population measurements and achieved an error in population estimates between 0.7% and 4.4%. They did not report results on plant counting and location estimates. They indicated that the main error source was interference from leaves. Rascon (2012) used a red light range sensor to sense corn plant stalk location at mid growth stages. He found leaf interference on the results was significant. Nakarmi and Tang (2010) developed a system to measure corn interplant spacing at growth stages V3 to V6 using a 3D range camera. They reported a 100% plant identification accuracy and RSME of 0.15 cm for interplant spacing measurements. Little research has been conducted so far on real-time corn stalk diameter measurement and plant location measurement at their mid-growth stage using range sensing techniques.

Data Processing Algorithms for Plant Sensing

Machine vision technology is based on image processing algorithms. The following literatures provide a review of related data/image processing algorithms which may inspire the algorithm

development in automatic corn plant spacing and stalk diameter measurement. Some image processing algorithms can be applied on laser scan data processing.

Segmentation

Segmentation is the process to partition an image into heterogeneous objects. It is an important step before the objects of interest in an image can be recognized and analyzed. Image segmentation algorithms for monochrome images generally have three categories: 1, thresholding; 2, point, line and edge detection; and 3, region-based segmentation or clustering (Gonzalez et al., 2002). For color images, similar approaches can be used in RGB vector space.

Thresholding is the most commonly used algorithm in image segmentation. One or multiple threshold is selected based on the difference of pixel's intensity levels or other properties. One or multiple distance thresholds are often used to eliminate background pixels such as soil and sky in 3D depth images (Wei and Salyani, 2005; Nakarmi and Tang, 2010; Chen et al., 2012). Thresholds of gray levels are used to eliminate pixels with undesired gray scale in monochrome images (Wang et al., 1998).

Point, line or edge detection looks for discontinuities in an image. The most common way of point, line or edge detection is to run a mask through the image and to compute the sum of products encompassed by the mask at each pixel. The mask varies from different applications. The detected line or edge segments in an image are often discontinued. Hough transform is an approach to link them to a meaningful line. Wang et al. (1998) implemented the Hough transform in identifying micropropagated sugarcane shoots and achieved a 93% of identification accuracy.

The region-based segmentation or clustering generally includes two approaches – region growing starting with seed points; and region splitting followed by merging. Numerous models have been developed for clustering such as distance connectivity based models, density based models,

centroid linkage region growing models, distribution based models, and trainable models. Each of them was developed for specific application.

Color image segmentation can be usually implemented in RGB color space. Shrestha et al. (2001) developed a truncated ellipsoidal surface in RGB color space to segment corn plant from soil background from top view color imaging. The parameters of the truncated ellipsoidal surface were obtained using artificial neural network method. Steward and Tian (1998) proposed an environmentally adaptive segmentation algorithm for corn plant segmentation in RGB image taken from top view at early growth stages. This algorithm included a transformation of pixel's RGB reading, a K-mean clustering and a Bayes classifier with a look-up table.

Morphological Processing

Morphological processing is used to extract shape features of regions in an image such as boundaries and skeletons. Common morphological processing techniques are dilation, erosion, opening and closing. In agriculture engineering, morphological processing is often used to eliminate noise in binary images (Wei and Salyani, 2005; Tang and Tian, 2008b), to sharpen or thinning objects in an image during pre-processing (Wang et al., 1998), or to obtain skeleton of a shape for further analysis (Tang and Tian, 2008b; Nakarmi and Tang, 2010). It is a necessary procedure to achieve the final image processing and analysis goal.

Image Registration

Image registration or image sequencing is a process to connect images taken from various points of view with overlap to one scene. It often makes use of the features in two images to register them together such as shape and color. In agriculture applications, the purpose of image registration is often to stitch images taken during the field operation together to one big map. Some previous image registration studies were reported with satisfactory results and feasibility. Shrestha and Steward (2003) developed a patch match algorithm to register corn images taken

from top view during operation. In the range image processing in which there is only color information, auxiliary method was used for image registration such as shaft encoder reading (Nakarmi and Tang, 2010). The objects in images taken in a corn field are plants and residues on the soil. The color and shape of plants are not consistent due to the wind blowing. Because of this, information obtained from non-imaging sensors such as a shaft encoder can be used to help image registration in agriculture engineering.

References

Cassman, K. G., Dobermann, A. R., Walters, D. T. (2002). Agroecosystems, nitrogen-use efficiency, and nitrogen management. Royal Swedish Academy of Sciences. Ambio, 31(2).

Chen, Y., Zhu, H., Ozkan, H. E., 2012. Development of a variable-rate sprayer with laser scanning sensor to synchronize spray outputs to tree structures. *Transactions of ASABE* 55 (3), 773-781.

Doerge, T., Hall, T. Gardner, D., 2002. New research confirms benefits of improved plant spacing in corn. *Crop Insights* 12 (2), 1-5.

ERS (Economic Research Service), USDA. 2012. Fertilizer use and price. http://www.ers.usda.gov/data-products/fertilizer-use-and-price.aspx. Verified 01/04/2013.

FAOSTAT. Food and Agriculture Organization of the United Nations. 2012a. http://faostat.fao.org/site/339/default.aspx. Verified 01/04/2013.

FAOSTAT. Food and Agriculture Organization of the United Nations. 2012b. http://faostat3.fao.org/home/index.html#VISUALIZE. Verified 01/04/2013.

GopalaPillai, S. and Tian, L. (1999). In-field variability detection and spatial yield modeling for corn using digital aerial imaging. Transactions of the ASAE, 42(6): 1911-1920.

Gauer, L. E., Grant, C. A., Gehl, D. T., & Bailey, L. D. (1992). Effects of nitrogen fertilization on grain protein content, nitrogen uptake, and nitrogen use efficiency of six spring wheat (Triticum aestivum L.) cultivars, in relation to estimated moisture supply. Canadian Journal of Plant Science, 72: 235-241.

Gonzalez, R. C., & Woods, R. E. (2002). Digital Image Processing, 2nd Edition. Prentice Hall. ISBN: 0-201-18075-8.

Heege, H., Reusch, S., & Thiessen, E. (2004). Systems for site-specific on-the-go control of nitrogen top-dressing during spreading. Proc. 7th Intl. Conf. on Precision Agriculture and Other Precision Resources Management, 133-147. St. Paul, Minn.: University of Minnesota, Precision Agriculture Center.

Huang, Y., Lan, Y., Ge, Y., Hoffmann, W. C., & Thomson, S. J. (2010). Spatial modeling and variability analysis for modeling and prediction of soil and crop canopy coverage using multispectral imagery from an airborne remote sensing system. Transactions of the ASABE, 53(4): 1321-1329.

Kelly, J. (2011). By plant prediction of corn grain yield using height and stalk diameter. Master thesis. Oklahoma state university.

Krall, J. M., Esechie, H. A., Raney, R. J., Clark, S., TenEyck, G., Lundquist, M., Humburg, N. E., Axthelm, L. S., Dayton, A. D., & Vanderlip, R.L. (1977). Influence of within-row variability in plant spacing on corn grain yield. Agronomy Journal 69:797–799.

Lauer, J. G., & Rankin, M. (2004). Corn response to within row plant spacing variation. Agronomy Journal, 96: 1464 – 1468.

Li, H., Worley, S. K., & Wilkerson, J. B. (2009). Design and optimization of a biomass proximity sensor. Transactions of the ASABE, 52(5), 1441-1452.

Luck, J. D., Pitla, S. K., & Shearer, S. A. (2008). Sensor ranging technique for determining corn plant population. ASABE paper No. 084573. Providence, Rhode Island: ASABE.

Martin, K. L., Hodgen, P. J., Freeman, K. W., Ricardo Melchiori, Arnall, D. B., Teal, R. K., Mullen, R. W., Desta, K., Phillips, S. B., Solie, J. B., Stone, M. L., Octavio Caviglia, Fernando Solari, Agustin Bianchini, Francis, D. D., Schepers, J. S., Hatfield, J. L., & Raun, W. R. (2005). Plant-to-Plant Variability in Corn Production. Agronomy Journal, 97:1603–1611.

Martin, K., Raun, W., & Solie, J. (2012). By-plant prediction of corn grain yield using optical sensor readings and measured plant height. Journal of Plant Nutrition, 35:9, 1429-1439.

Nakarmi, A. D. and Tang, L. (2010). Inter-plant spacing sensing at early growth stages using a time-of-flight of light based 3D vision sensor. 2010 ASABE Annual International Meeting, Pittsburgh, Pennsylvania. Paper number: 1009216.

National Agricultural Statistics Service, USDA. 2010. Acreage. http://usda.mannlib.cornell.edu/usda/nass/Acre/2010s/2010/Acre-06-30-2010.pdf. Verified 01/04/2013.

Nichols, S. W. (2000). Method & apparatus for counting crops. U.S. Patent No. 6073427.

Rabalais, N.N., R.E. Turner, and D. Scavia. 2002b. “Beyond Science Into Policy: Gulf of Mexico Hypoxia and the Mississippi River,” Bioscience 52(2):129-142.

Rascon, J. (2012). Corn sensor development for by-plant management. Unpublished Master’s thesis. Oklahoma State University.

Raun, W.R., & Johnson, G.V. (1995). Soil–plant buffering of inorganic nitrogen in continuous winter wheat. Agronomy Journal, 87:827–834.

Raun, W.R., & Johnson, G.V. (1999). Improving nitrogen use efficiency for cereal production. Agronomy Journal, 91:357–363.

Ribaudo, M., Delgado, J., Hansen, L., Livingston, M., Mosheim, R., and Williamson, J. (2009) Nitrogen in agricultural systems: implications for conservation policy. ERR-127. U.S. Dept. of Agriculture, Econ. Res. Serv.

Saeys, W., Lenaerts, B. Craessaerts, G. & De Baerdemaeker, J. (2009). Estimation of the crop density of small grains using LiDAR sensor. Biosystems Engineering, I02 (2009) 22-30.

Scharf, P. C., Wiebold, W. J., and Lory, J. A. (2002). Corn yield response to nitrogen fertilizer timing and deficiency level. Agronomy Journal, 94: 435-441.

Shanahan, J. F., Kitchen, N. R., Raun, W. R., and Schepers, J. S. (2008). Responsive in-season nitrogen management for cereals. Computer and Electronics in Agriculture, 61: 51-62.

Shrestha, D. S., Steward, B., and Bartlett, E. (2001). Segmentation of plant from background using neural network approach. In Intelligent Engineering Systems through Artificial Neural Networks: Proc. Artificial Neural Network Networks in Engineering (ANNIE) International Conference, 11: 903-908. New York, N.Y.: ASME Press.

Shrestha, D. S., & Steward, B. L. (2003). Automatic corn plant population measurement using machine vision. Transactions of the ASABE, 46(2): 559–565.

Shrestha, D. S., & Steward, B. L. (2005). Shape and size analysis of corn plant canopies for plant population and spacing sensing. Transactions of the ASAE, 21(2): 295−303.

Solie, J. B., Raun, W. R., and Stone, M. L. (1999). Submeter spatial variability of selected soil and bermudagrass production variables. Soil Science Society of America Journal, 63: 1724-1733.

Sowers, K.E., Pan, W.L., Miller, B.C., & Smith, J.L. (1994). Nitrogen use efficiency of split nitrogen applications in soft white winter wheat. Agronomy Journal, 86:942–948.

Steward, B. L. and Tian, L. F. (1998). Real-time weed detection in outdoor field conditions. Proceedings of Precision Agriculture and Biological Quality, G. E. Meyer and J. A. DeShazer, eds., SPIE, Bellingham, WA, pp. 266-278.

Tang, L., & Tian, L. F. (2008a). Real-time crop row image reconstruction for automatic emerged corn plant spacing measurement. Transactions of the ASABE, 51(3): 1079-1087.

Tang, L., & Tian, L. F. (2008b). Plant identification in mosaicked crop row images for automatic emerged corn plant spacing measurement. Transactions of the ASABE, 51(6): 2181-2191.

Thorp, K. R., Steward, B. L., Kaleita, A. L., & Batchelor, W. D. (2008). Using aerial hyperspectral remote sensing imagery to estimate corn plant stand density. Transactions of the ASABE, 51(1): 311-320.

United States Department of Agriculture. (2012). US domestic corn use. http://www.ers.usda.gov/media/866543/cornusetable.html. Verified 01/04/2013.

United States Environmental Protection Agency. (2012). Inventory of U.S. greenhouse gas emissions and sinks: 1990 – 2010. EPA 430-R-12-001.

Varvel, G. E., Schepers, J. S., and Francis, D. D. (1997). Ability for in-season correction of nitrogen deficiency in corn using chlorophyll meters. Soil Science Society of America Journal, 61: 1233-1239.

Vitousek, P. M., Aber, J. D., Howarth, R. W., Likens, G. E., Matson, P. A., Schindler, D. W., Schlesinger, W. H., and Tilman, D. G. (1997). Human alteration of the global nitrogen cycle: sources and consequences. Ecological Applications, 7(3): 737-750.

Wang, Z., Heinemann, P. H., Sommer III, H. J., Walker, P. N., Morrow, C. T., & Heuser, C. (1998). Identification and separation of micropropagated sugarcane shoots based on the hough transform. Transactions of the ASAE, 41(5): 1535-1541.

Wangler, R. J., Fowler, K. L. & McConnell, R. E. (1994). Object sensor and method for use in controlling an agricultural sprayer. U. S. Patent No. 5278423.

Wei, J., & Salyani, M. (2004). Development of a laser scanner for measuring tree canopy characteristics: phase 1. prototype development. Transactions of the ASABE, 47(6): 2101-2107.

Wei, J., & Salyani, M. (2005). Development of a laser scanner for measuring tree canopy characteristics: phase 2. foliage density measurement. Transactions of the ASABE, 48(4): 1595-1601.

Wuest, S. B. and Cassman, K. G. (1992). Fertilizer-nitrogen use efficiency of irrigated wheat: I. uptake efficiency of preplant versus late-season application. Agronomy Journal, 84(4): 682-688.

CHAPTER III

AUTOMATIC CORN PLANT LOCATION AND SPACING MEASUREMENT USING LASER LINE-SCAN TECHNIQUE

The material in this chapter was published in 2013 in Journal of Precision Agriculture, 14(5): 478-494.

Abstract

Identifying corn plant location and/or spacing is important for predicting yield potential and making decisions for in-season nitrogen application rate. In this study, an automatic corn stalk identification system based on a laser line-scan technique was developed to measure stalk locations during corn mid-growth stages. A laser line-scan technique is advantageous in this application because the line-scan data sets taken from various points of view of a plant stalk results in less interference and higher probability of plant recognition. Data were collected for two 10-meter-long corn rows at the growth stages of V8 and V10 using a mobile test platform in 2011. Each potential stalk cluster was identified in a scan and registered with the same stalks in previous scans. The final location of a stalk was the average of the measured locations in all scans. The current system setup with data processing algorithms achieved 24.0 % and 10.0 % of mean total errors in plant counting at the V8 and V10 growth stages, respectively. The root-mean-

squared error (RMSE) between system measured plant locations and manually measured ones were 2.3 cm and 2.6 cm at the V8 and V10 growth stages, respectively. The interplant spacing measured by the developed system had a good correlation with the manual measurement with an R^2 of 0.962 and 0.951 for the V8 and V10 growth stages, respectively. This system can be ultimately integrated in a variable-rate-spraying system to improve real-time, high spatial resolution variable-rate nitrogen applications.

Keywords Corn population · In-field variability · Data clustering · Variable-rate technology

Introduction

Nitrogen use efficiency (NUE) in cereal production worldwide is as low as 33 % (Raun and Johnson 1999). Much of the nitrogen (N) fertilizer applied to crops is lost to surface runoff and leaching. This results in environmental damage such as the contamination of groundwater supplies or eutrophication of surface waters. Research on the approaches to reduce N losses has shown that the NUE decreases with increasing N application level (Gauer et al. 1992) and nitrate leaching can be significant when N is applied at rates in excess of that needed for maximum yield (Raun and Johnson 1995). These findings have motivated research and adoption of variable-rate N applications where N is applied during the growing season rather than being applied at a fixed rate prior to planting (Sowers et al. 1994).

It is critical to determine an optimal spatial scale for variable rate N application for in-field variability management. For corn production, Martin et al. (2005) evaluated by-plant corn yield variability based on the data collected in the USA, Argentina and Mexico from 2002 to 2004. The by-plant corn yield was calculated in the unit of kg/ha by assuming the grain yield of a plant was the average grain yield in an area of one hectare. The area occupied by that plant was calculated as half the distance to and from its two nearest neighbors multiplied by the row spacing. They

reported an averaged standard deviation of plant to plant grain yield at 2,765 kg/ha. They also found that variability was not significant if the yield was averaged along the row over a scale greater than 0.5 m. These results indicated that high-resolution plant management protocols might have a significant impact in corn production.

Parameters commonly considered when investigating variability management are soil nutrient level, soil moisture, plant N content, plant population or spacing, plant height, canopy coverage or volume and canopy density. In corn production, variability of plant N content, plant population or spacing and plant height are often examined. Krall et al. (1977) found that every 2.5 cm increase in the standard deviation of plant spacing would decrease the yield by 210 kg/ha. Lauer and Rankin (2004) found that when the standard deviation in corn plant spacing was greater than 12.0 cm, relative grain yield reduced at 1.06 % with every centimeter increase of the standard deviation in plant spacing. A corn by-plant yield prediction model proposed by Martin et al. (2012) included plant height, plant spacing and normalized difference vegetative index (NDVI) and achieved an R^2 of 0.48. The previous model without using plant height and spacing had an R^2 of 0.22.

The research on plant population or interplant spacing measurement can be categorized as two types: airborne and ground-based (Dworak et al. 2011). Most of the airborne remote sensing approaches use hyperspectral or multispectral analysis to obtain large-scale data (Huang et al. 2010; Thorp et al. 2008). Ground-based sensing methods have been used for obtaining detailed crop and soil information. These can be done concurrently with other in-field operations such as planting, spraying or harvesting. Ground-based approaches for plant population or spacing measurements can be categorized as intrusive (mechanical methods) or non-intrusive methods.

Mechanical methods to measure corn plant population usually use the resistant force of stalks on a spring loaded arm or a gravity pendulum to count the number of stalks (Birrell and Sudduth

1995; Heege et al. 2004). Some of these methods have already been commercialized on combine harvesters. Non-intrusive methods are more suitable for sensing corn population at early and mid-growth stages. Some of these methods are based on capacitive sensing: Nichols (2000) invented a moisture detecting sensor installed on a combine head to count harvested stalks; Li et al. (2009) developed a capacitance biomass proximity sensor to count corn stalks during harvesting.

Other non-intrusive methods in ground-based crop sensing are mainly based on optical sensing technique including 2D color/gray-scale imaging and range sensing. Color imaging has been explored in several studies for corn plant counting and spacing measurement in the past few decades. Shrestha and Steward (2003, 2005) developed and tested a machine vision based corn plant population sensing system. Algorithms were developed for color image sequencing, segmentation and plant recognition in order to count corn plants and to estimate plant location and spacing. The root-mean-square errors (RMSEs) in population estimates were in the range of 5 to 6 % compared with manual counts. Tang and Tian (2008a, 2008b) developed a real-time crop row image reconstruction and plant identification system for automatically measuring the spacing of emerged corn plants. They achieved an overall RMSE of 1.7 cm and an R^2 of 0.96. All of these studies targeted at early growth stage corn plants prior to canopy closure.

Range sensing techniques are another category of optical-based sensing methods that have been applied to crop parameter measurements. A photoelectric emitter and receiver pair is a 1D range sensor. Hummel et al. (2002) developed and tested photoelectric sensors installed on a combine corn head for plant diameter, spacing and population measurements. They used an air-jet system to physically remove corn leaves and other debris from the sensors' field of view. The average normalized population and spacing estimates were reported at 0.94 and 1.08, respectively. Luck et al. (2008) used an infra-red range sensor for in-field plant population measurements and achieved an error in population estimates between 0.7 % and 4.4 %. They indicated that the main error source was the interference from leaves. A laser line scanner is a 2D range sensor. Wangler

et al. (1994) patented a laser scanning sensor which could be attached on a sprayer to selectively spray according to the presence of the tree foliage. A laser scanning technique was also used for tree foliage density and wheat stand density estimation by calculating variation in laser penetration depth (Wei and Salyani 2004, 2005; Saeys et al. 2008).

Up to recent, little research has been conducted so far on corn plant location measurement in the mid-growth stages using 2D range sensing techniques. The objective of this study was to develop a system using the laser line-scan technique for automatic corn plant location measurements to facilitate in-season variable rate N applications. The specific objectives were to:

- Develop a data acquisition system based on laser line-scan techniques to obtain corn plant location and spacing information;
- Develop data processing algorithms to estimate corn plant location and spacing;
- Evaluate the system performance at the V8 and V10 growth stages.

Materials and methods

System setup and principles

The data acquisition platform was a four-wheel cart which moved easily between rows (Fig. 1a). The key component of this system was a laser line scanner (LMS291, SICK AG, Waldkirch, Germany) which measured distances between the sensor and target objects based on the time-of-flight principle. It was configured to operate in continuous line scan mode with a field of view of 100° and a resolution of 0.25°. The laser scanner was mounted on the cart's front arm and aimed about 5-cm above the roots of the corn stalks with a downward angle of 20°. The scan plane formed a 70° angle with the plane of the plant row (Fig. 1c). This setup was selected to allow the sensing of stalks near the ground while maintaining sufficient clearance between the sensor and

ground.

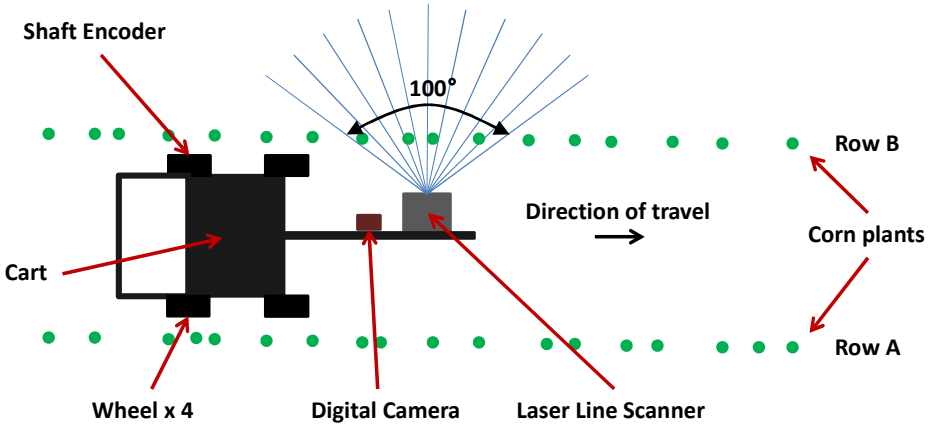
Multiple neighboring stalks within a row were sensed in a scan as illustrated in Fig. 1b. The number of stalks in a scan depended on the distance between the sensor and plant row, as well as how far apart neighboring plants were. A control program developed in LabVIEW® (National Instruments Co., Austin, Texas, USA) was used to establish the communication between a laptop computer and the laser scanner, to receive data packages, to extract distance data and convert them from polar to Cartesian coordinates, and to save the data into a file with MS Excel format. The laser scanner scanned 100° and collected 401 distance measurements in 53.28 ms. With an average 0.447 m/s moving speed of the cart, the sensor moved about 2.4 cm within the time of a scan. This offset was ignored in this study. A 500 kbps baud rate was configured with a RS-422 connection and a serial to Ethernet convertor (DeviceMaster 500, Comtrol Co., New Brighton, Minnesota, USA) between the laser scanner and the laptop to ensure a sufficient data transfer rate.

A shaft encoder was mounted on one of the rear wheels and connected to a data acquisition card (USB 6008, National Instruments, TX) to obtain the location of each scan relative to a fixed start point. Hence, the data of each laser scan was location-stamped with a corresponding encoder reading. A video camera was mounted next to the laser line scanner to record a video of each trial which could be used later to verify the measurements of the laser line scanner.

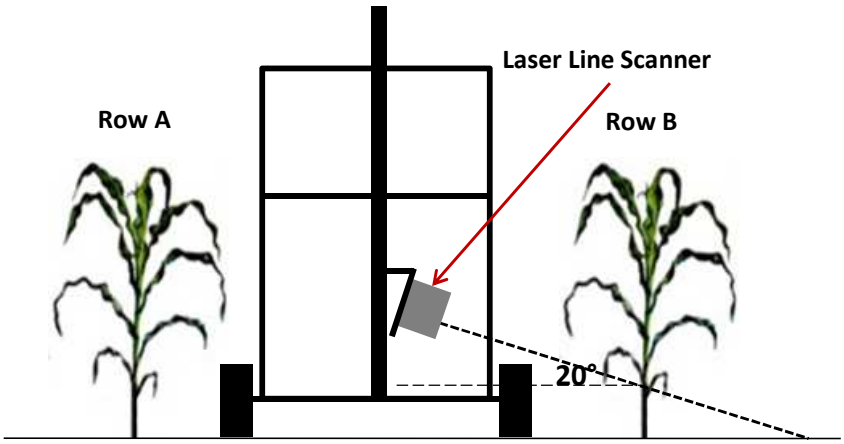
Field experiment setup

The field experiment was conducted at Lake Carl Blackwell, near Stillwater, OK, in June and July of 2011. Data were collected from two 10-m long rows, each containing 50 corn plants, at the V8 and V10 growth stages. Three trials were conducted on each row at the V8 growth stage. Due to equipment malfunctions, only the data from two trials was used on row 1 at the V10 growth stage. Fig. 2 shows the field setup for the data acquisition platform. The cart with the

(a)



(b)



(c)

Fig. 1 Laser line-scan based corn plant location and spacing measuring system: (a) the cart with a laser line-scan sensor and other data acquisition devices; (b) top view and (c) side view showing system operation

developed data acquisition platform was manually pushed between corn rows. The horizontal distance between the sensor and the corn row varied from 34 to 48 cm due to the deviation of the

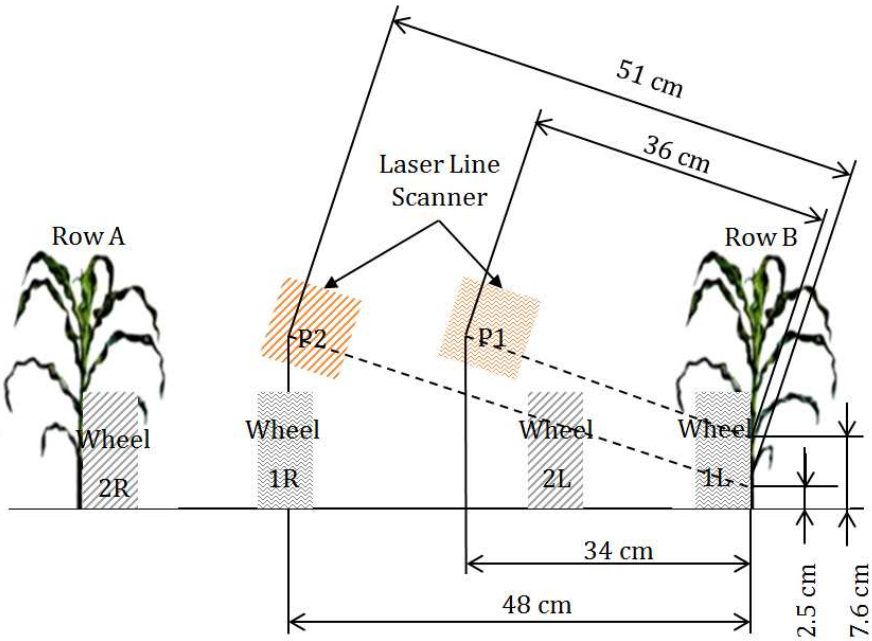


Fig. 2 Illustration of the cart's deviation between rows (not to scale): Case 1 (see P1) was when the cart was the closest to plant row B and Case 2 (see P2) was when the cart was the furthest to plant row B. Designations 1L and 2L indicated the positions of left wheel, and 1R and 2R indicated the positions of right wheel in these two cases, respectively

cart from the center line between rows. Since the laser scanner was mounted with a downward angle, when the cart travelled between the rows, the actual distance reading of the central point of a laser scan was between 36 and 51 cm. The sensor was mounted at a height so that, with this travel distance, the sensing plane on the plant stalks was between 2.5 and 7.6 cm above the plant roots. Manual location measurements were taken for the 100 plants and used as ground truth.

Data processing algorithms

Fig. 3 shows a flowchart of the main data processing algorithm for locating corn stalks. Shaft encoder data of each scan were pre-processed so that it could be synchronized with ground truth measurement. For each scan, after eliminating the soil background, potential stalk clusters were classified and each was registered with corresponding stalk clusters from previous scans if they were identified as the same plant. Except for the most recent plant entering into the sensor's field of view during movement, each cluster in the current scan not corresponding to any of the clusters from previous scans were treated as noise. The final location of each recognized plant

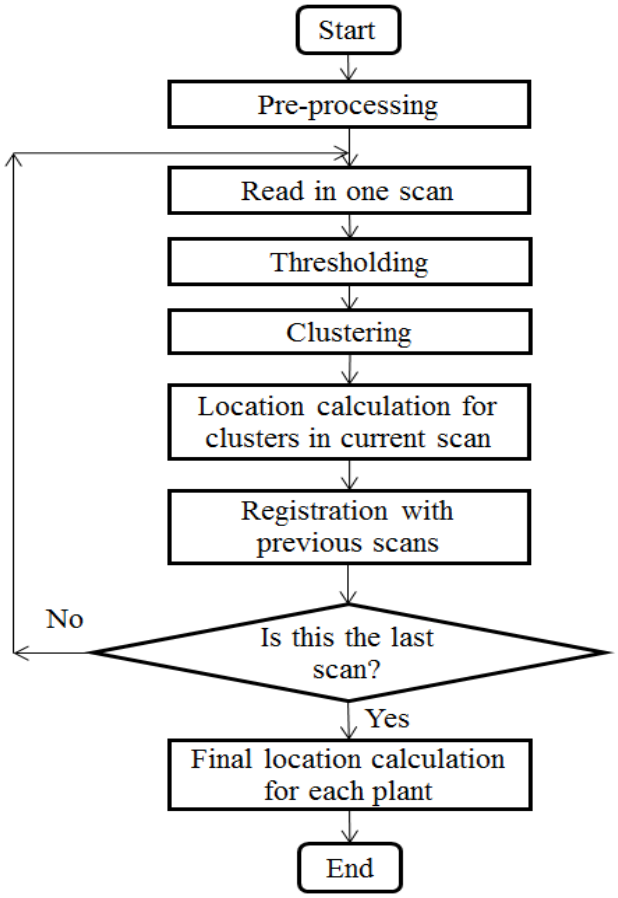


Fig. 3 Flowchart of the main data processing algorithm for locating corn stalks

was calculated as the mean of the position measurements in all related scans. All data processing algorithms were developed in MATLAB® (MathWorks Inc., Natick, Massachusetts, USA).

Synchronization of shaft encoder data and ground truth measurement

A pre-processing step was conducted on the shaft encoder data to synchronize it with ground truth measurements so that the system could be evaluated. Shaft encoder readings provided location of each laser scan. They were critical for in-row plant registration. The quality of the shaft encoder readings depended on the rotation of the wheel to which the encoder roller was attached. Intermittent rotation due to obstacles or uneven soil surface was unavoidable in a field experiment and could reduce positional accuracy. To partially correct for these deviations,

encoder readings of each trial were stretched evenly along the entire row so that they would have an equal total length in between the 1st and 50th plants.

Thresholding

Fig. 4a shows a typical scan after conversion to Cartesian coordinates. The sensor (gray square box) was at the origin of the local coordinates when this scan was taken. The clusters within 36 to 51 cm of the vertical axis of the coordinates corresponded to plant stalks. Other data points were the reflections from the soil background, leaves or other interfering objects and considered as noise. A thresholding process was conducted to eliminate noise data points that were not within the normal 36 to 51 cm range of the sensor as the cart travelled down the row.

Clustering

A clustering algorithm was implemented to identify potential stalk clusters based on the density variation of the line scan data. This algorithm was adapted from the density-based spatial clustering of applications with noise (DBSCAN) described by Ester et al. in 1996.

Three parameters must be pre-defined in this algorithm: ε was the range in which a core point searched for a neighbor point to form a cluster and was set as a specific value no less than 0.85 cm according to the distance the core point was away from the origin; MinPts was the minimum number of data points needed to identify a cluster and was set as five; MaxPts was the maximum number of data points to be included in a cluster and was set as 25. These values were determined by trial and error in this study.

The value of radius ε in which a core point searched for its neighbors was not the same for every core point in the developed clustering algorithm. It varied based on the distance a data point was away from the origin. This was necessary because the radially collected data of the laser line scanner resulted in a higher density of data points near the scanner. The section area of a laser

beam at a data point was calculated based on the divergence angle of the laser beam and the distance between that data point and the origin. The ratio of this section area to the section area of a laser beam at 44-cm away from the origin (cart centered between two rows) was obtained. It was used to multiply the minimum value of ε (which was 0.85 cm in this study) to be the specific ε value for that data point. The adjustable searching range ε made the clustering more accurate.

The values of MinPts and MaxPts were selected by assuming that the stalk diameters measured in this study would not be less than 1 cm or greater than 5 cm at both of the V8 and V10 growth stages. If the cart was moving along the center line between rows, a 5 cm or a 1 cm object would form a 25-point cluster or a five-point cluster at the central point of a laser scan when the sensing resolution was set as 0.25°. Hence, MinPts was set as five and MaxPts was set as 25.

The clustering algorithm included the following steps:

1. *All points in a dataset were initially marked as unvisited. The algorithm randomly started from one of the unvisited points p in the dataset and marked it as a visited point.*
2. *Made p a core point and searched for neighboring points within a radius ε. If no points were found within that range, marked p as noise and went back to step 1; otherwise, a new cluster was found consisting of p and its neighbors.*
3. *Each neighbor of the core point was assigned to the current cluster and was marked as visited. Each of point in the cluster was then treated as a core point and a search was conducted for its neighbors within a radius of ε. For each of the neighbor points found, step 3 was repeated until no new neighbors could be found.*
4. *Went back to step 1 until all points in the dataset had been visited.*
5. *Checked the size of each cluster. Only those clusters that had a size larger than MinPts and smaller than MaxPts were kept while others were treated as noise and eliminated.*

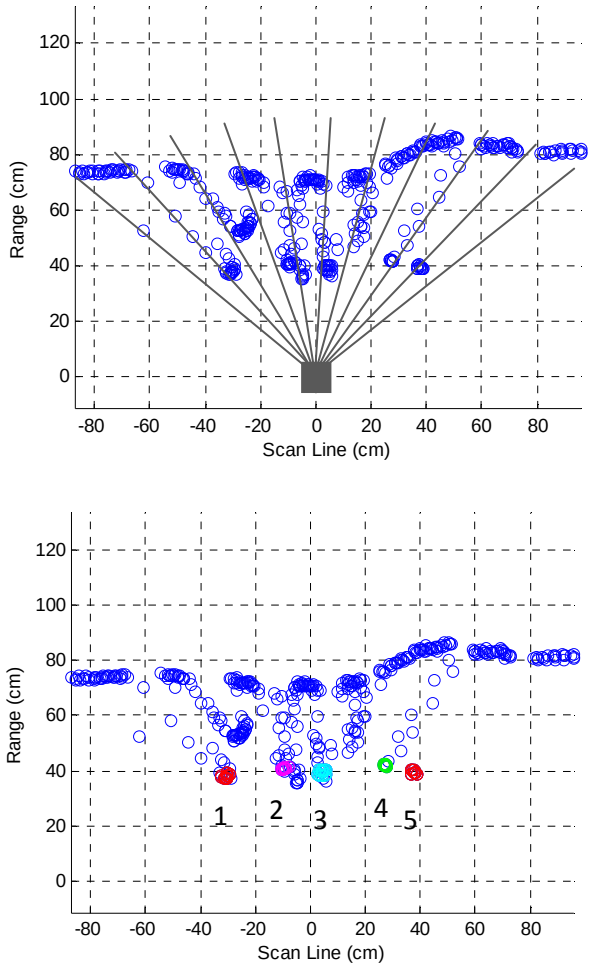


Fig. 4 Illustration of clustering algorithm for stalk identification: (a) typical line scan data after converting to Cartesian coordinates. "○" indicates reflectance data points; sensor was located at the origin (0, 0); (b) clustering result of this scan: 1-5 were the identified clusters

The clustering algorithm developed in this study changed the criterion for identifying a point as a core point in a cluster expansion. In the referred algorithm (DBSCAN) a point was usually marked as a core point when it had more than three neighbors within a certain range; otherwise, the point was marked as noise. This made it possible to initially mark border points in a cluster as noise if the cluster had a narrow shape. Though DBSCAN changed them to cluster points later, this was inefficient if most of the points in a cluster were boarder points, which was the case when using a laser line scanner in plant

stalk profiling. In the developed clustering algorithm, a cluster expansion started from a core point even if it had only one neighbor in a defined range.

Coordinate conversion

Two coordinate systems were involved in data processing – a local coordinate system and a ground coordinate system. Data from each scan had unique local coordinates due to the movement of the cart down the row. The origin of each local coordinate system was located at the laser source inside the laser line scanner. The *y-local* axis of each of the local coordinate system was along the midpoint of the sensor's field of view and increased with the distance away from the sensor; the *x-local* axis was perpendicular with *y-local* axis and increased to the right of the sensor. The origin of ground coordinates was at the start of each trial where the encoder reading was zero. The *x-ground* axis of the ground coordinates was parallel to the direction of sensor's travel; the *y-ground* axis was parallel to local coordinate axis indicating the depth measurement. The local coordinates of a plant in a scan were finally converted to the ground coordinates making use of the specific encoder reading of that scan.

Scan registration / matching between scans

Multiple scans obtained for the same stalk from various points of view gave a better chance to correctly recognize a plant. With the sensor's configured field of view while traveling at 0.447 m/s during data acquisition, a stalk generally appeared in approximately 40 continuous scans. Although it might be interfered with or blocked by leaves and other debris in some scans, a stalk still had a high possibility to be recognized in the rest of the scans. In order to match clusters from multiple scans corresponding to the same plant stalk, a scan registration procedure was implemented based on the difference of shaft encoder readings between scans.

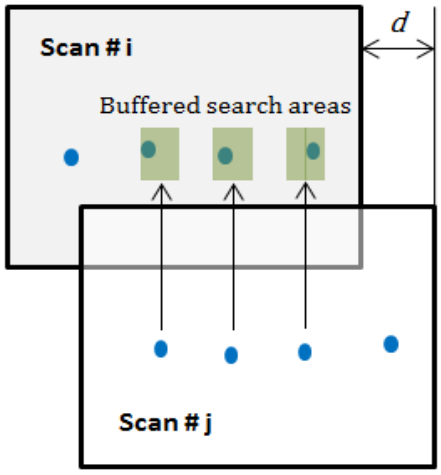


Fig. 5 Illustration of the scan registration algorithm

Fig. 5 is an illustration of this scan registration process. Assume scan # j was a current scan, scan # i was a previous scan, three clusters were shown in both scans, and the difference between encoder readings of these two scans was d. The developed algorithm used cluster locations in the current scan and d to calculate estimated cluster locations in the previous scan. It then searched for corresponding clusters within a small buffered search area around that estimated location for each cluster. The buffered search area was set as ±6 cm in this study by trial and error. If multiple corresponding clusters were found within that range for a cluster in the current scan, the one closest to the estimated location in the previous scan was selected. The first scan was exempted from registration. The clusters recognized in the first scan were assigned as plant stalks in incremental indices starting from one.

Registration started from the second scan and each scan could only register with the previous scans in order to meet the requirement by a real-time processing system. For each cluster in the current scan, the algorithm searched backward through up to thirty previous scans to find a matching cluster. Once a matching cluster was found, the one in the current scan was assigned to the same cluster index as the cluster in the previous scan; otherwise, the cluster in the current scan was marked as not being matched. Finally, all the unmatched clusters were checked to see if there was a cluster which newly entered into the sensor's field of view. If there was one, a new cluster

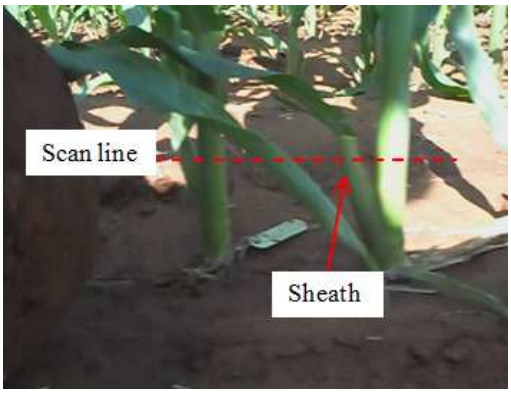


Fig. 6 Sheath interference with stalk locations

index was assigned to it. At the end of the process, all the unmatched clusters were marked as noise clusters.

The final location of a recognized plant stalk was the mean location of all clusters with the same index in different scans. Estimated locations which were within 5 cm of each other had a high likelihood of corresponding to a stalk and a sheath (Fig. 6). Sheath interference was compensated for by averaging the locations of two estimated locations less than 5-cm apart into one location.

System performance evaluation

System performance was evaluated by comparing its measurements to ground truth data which was measured manually by a ruler for each plant row. Errors in plant counting, plant location estimate and interplant spacing estimate were analyzed.

Plant counting

Three errors were defined for evaluating system's performance on plant counting: the false negative counting error (FNEr) (Eq. 1), false positive counting error (FPEr) (Eq. 2), and total counting error (TEr) (Eq. 3):

$$FNEr = \frac{Missing\ Count}{Ground\ Truth\ Count} \times 100\ \% \quad \text{Eq. 1}$$

$$FPEr = \frac{Adding\ Count}{Ground\ Truth\ Count} \times 100\ \% \quad \text{Eq. 2}$$

$$TEr = FNEr + FP \quad \text{Eq. 3}$$

If no plant was identified within ±10 cm from the ground truth location of an actual plant, there was a false negative count for that actual plant. Similarly, if no actual plant was located within ±10 cm of the location of an identified plant, this resulted in a false positive count. Only the recognized plant closest to (and within ±10 cm) the location of an actual plant was a valid count; multiple counts of an actual plant were treated as false positive counts. The total counting error was equal to the sum of the false negative error and false positive error. These errors were calculated for each of the two rows at each growth stage. The SAS (SAS Institute, Cary, North Carolina, USA) General Linear Model procedure (GLM) was used to test for significant differences in FNEr, FPEr and TEr between the V8 and V10 growth stages.

Plant location estimation

The RMSE was calculated between the manually measured ground truth locations and those measured by the developed system for all correctly recognized plants in each row at each growth stage. In this analysis, locations corresponding to false negative counts were eliminated from ground truth locations in each trial while locations corresponding to false positive counts were eliminated from system measured locations. The significant difference in plant location estimates between the V8 and V10 growth stages was also tested using GLM procedure in SAS.

Plant spacing estimation

Another parameter to investigate plant location error was the error of interplant spacing estimates. Interplant spacing was calculated as the difference between every recognized plant pair in a row excluding false positive and false negative counts for each trial. The spacing of the two plants at

each side of a false negative count was eliminated from the system measured spacing in order to correlate with the manually measured spacing. The RMSE of estimated spacing of all identified plants was calculated for each row in each trial and compared to manually measured spacing. If two successive plants were only correctly recognized in one trial, their spacing was compared to ground truth data; if two successive plants were correctly recognized in different trials, the corresponding spacing measurements were averaged before comparing to ground truth data.

Results and discussion

Plant counting error

At the V8 growth stage, an averaged 2.0 % (SD = 1.6 %) FNEr, an averaged 22.0 % FPEr and an averaged 24.0 % (SD = 6.9 %) TEr were achieved. At the V10 growth stage, a 3.0 % (SD = 1.0 %) FNEr, a 7.0 % (SD = 3.0 %) FPEr, and a 10.0 % (SD = 4.0 %) TEr were achieved (Table 1).

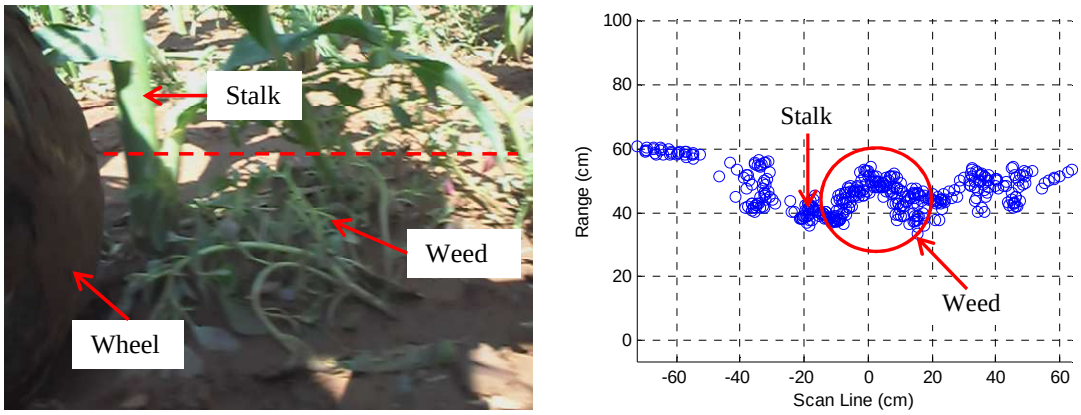
Table 1 Errors of plant counting and stalk location estimates compared to ground truth data

	FNEr (%)	FPEr (%)	TEr (%)	RMSE of location estimate (cm)	RMSE of spacing estimate (cm)
Row 1, V8	1.3 (0.94)*	16.0 (1.6)	17.3 (0.94)	1.9 (0.2)	2.1 (0.3)
Row 2, V8	2.7 (1.9)	28.0 (2.8)	30.7 (2.5)	2.8 (0.3)	2.9 (0.8)
Row 1, V10	3.0 (1.0)	7.0 (3.0)	10.0 (4.0)	2.6 (0.2)	2.0 (0.1)

* Numbers in the parenthesis represent standard deviation.

No significant difference was found in FNEr ($F_{1, 6} = 0.50$, $p = 0.51$) and TEr ($F_{1,6} = 5.51$, $p = 0.057$) between the V8 and V10 growth stages. While a significant difference was found in FPEr between V8 and V10 growth stages ($F_{1,6} = 7.61$, $p = 0.033$). At V8 growth stage, false positive errors were larger than false negative errors for both rows; while at the V10 growth stage, false positive errors decreased and false negative errors increased. The large false positive error at V8 growth stage was primarily due to weed interference which prevented the system from

differentiating weeds from stalks (Fig. 7a). Row 2 had larger errors than row 1 at V8 growth stage. The recorded video indicated more weed interference for row 2 than row 1 at that time. This suggested that the system developed in this study should be used in a weed controlled plot. A higher sensing height to avoid the weed area might help although interference from leaves on the stalk could offset improvements. At the V10 growth stage, weeds had been treated with herbicide so false positive errors decreased; however, most of the lower leaves were dehydrated and laid over the stalk (Fig. 8). A leaf cluster often had a larger size (> 25 data points) than a stalk cluster. This was used by the algorithm to differentiate the leaf clusters from the stalk clusters. Attached leaves prevented the sensor from seeing the actual shape of the stalks resulting in more false negative errors. At the V8 growth stage, most of the lower leaves were still vital and standing up.



(a) (b)

Fig. 7 Weed interference at the V8 growth stage: (a) a stalk with weed interference in row 2; (b) corresponding laser line scan data

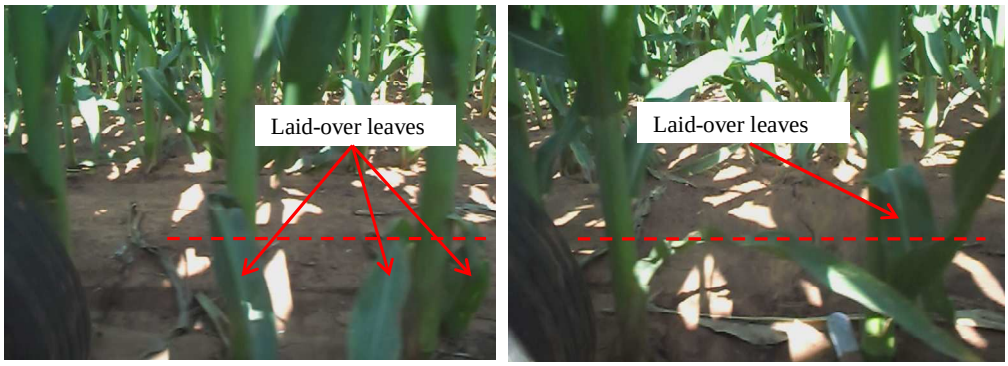


Fig. 8 Typical laid-over leaf interference at the V10 growth stage

The number of plants identified in each trial was used to assess the repeatability of system estimates. At the V8 growth stage, 49 out of 50 plants (98.0 %) were identified in all trials in row 1 though all 50 plants were identified in at least one trial; 46 plants out of 49 plants (93.9 %) were identified in all trials in row 2 excluding a plant located 3 cm apart from one of its neighbors though the other 48 plants were identified in at least one trial. At the V10 growth stage, 47 out of 50 plants (94.0 %) were identified in valid trials in row 1 though all 50 plants were identified in at least one trial.

Plant location and spacing error

The RMSEs of system measured plant locations were 2.3 cm (SD = 0.5 cm) and 2.6 cm (SD = 0.2 cm) for the V8 and V10 growth stages, respectively, for those correctly identified stalks in each row at each growth stage. No significant difference was found in RMSE of location estimate between the V8 and V10 growth stages ($F_{1,6} = 0.29$, $p = 0.61$) which demonstrated the repeatability in location estimate at two different growth stages of the system. This plant location estimate error was relatively small compared with the 20.2 cm (SD = 10.6 cm) interplant spacing which is discussed in the next section.

Factors contributing to errors in plant location estimates included errors from data acquisition, data processing and sheath and leaf interference. Small diameter wheels on the cart likely caused

some inaccuracy at some locations due to the uneven terrain in the field. The correction for encoder readings in data pre-processing distributed the encoder error evenly to every scan in a trial. This problem would be reduced if the encoder was mounted on a wheel with a larger contact area with the soil surface. Subjective error was induced in the manual measurement of plant locations. In addition, corn plants were not perfectly vertical during growth stages which caused inconsistency between manual and system measurements.

Interplant spacing estimated by the system was highly correlated to ground truth data. Comparison between them at the V8 growth stage resulted in RMSE of 2.5 cm (SD = 0.8 cm), an R^2 of 0.962 and a slope of 0.975 with an intercept of 0.643 in the regression equation (Fig. 9a). At the V10 growth stage, this comparison resulted in RMSE of 2.0 cm (SD = 0.1 cm), an R^2 of 0.951 and a slope of 0.995 with an intercept of 0.247 (Fig. 9b). The slopes were close to one and the intercepts were close to zero in both regression equations.

Distribution of manually and system measured interplant spacing was also investigated. Manually measured interplant spacing had a mean of 20.2 cm (SD = 10.6 cm); while those measured by the

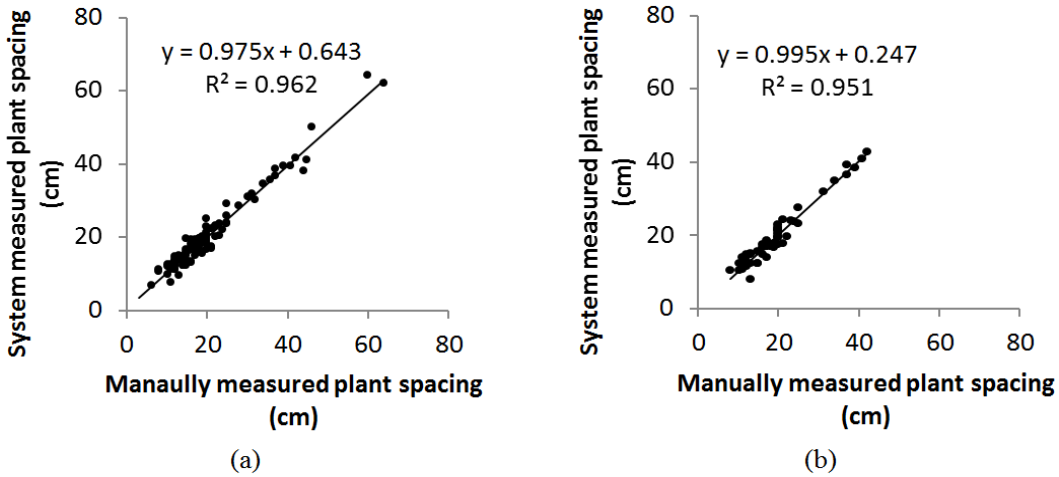


Fig. 9 Comparison between system measured interplant spacing and manually measured interplant spacing (a) at the V8 growth stage (n = 96) and (b) at the V10 growth stage

developed system had a mean of 20.6 cm (SD = 11.0 cm) at the V8 growth stage (Fig. 10a). Manually measured interplant spacing had a mean of 19.6 cm (SD = 8.5 cm); while those measured by the developed system had a mean of 19.8 cm (SD = 8.6 cm) at the V10 growth stage (Fig. 10c). The similarity in the means and standard deviations of the system measurements with the ground truth data demonstrated the repeatability of the system developed in this study.

The errors of the system measured interplant spacing had means of 0.1 cm (SD = 1.9 cm) and 0.1 cm (SD = 1.9 cm) at the V8 and V10 growth stages, respectively (Fig. 10b and Fig. 10d). Shapiro-Wilk normality tests were conducted to test the normality of the error distributions at the two growth stages. There was no significant evidence to reject the null hypothesis that they were normally distributed (SWstatistic= 0.97, $p = 0.82$ for V8; SWstatistic = 0.99, $p = 0.34$ for V10; $\alpha = 0.05$) which demonstrated the reliability of the system measurements. However, the slightly positive skewed shape in the error distribution plot of the V10 stage (Fig. 10d) indicated that there was an error source. It was very likely associated with the pre-processing procedure in data processing in which the encoder reading was stretched in order to be synchronized with the ground truth data. The stretch ratio was larger in the V10 than that in the V8 growth stage because the drought soil condition in the V10 growth stage caused more missing encoder counts. This suggested an improvement on the data acquisition platform with a better wheel encoder mechanism in the future.

Evaluation of overall system performance

The RMSEs of interplant spacing measurements were 2.5 cm (SD = 0.8 cm) at the V8 growth stage and 2.0 cm (SD = 0.1 cm) at the V10 growth stage. These results were at a centimeter or a tenth of a centimeter level, which were smaller or close to the agronomic spacing findings at a centimeter level described by Krall et al. (1977) and Lauer and Rankin (2004). This indicated that

the developed system could be used to provide useful plant location and spacing information from an agronomic point of view.

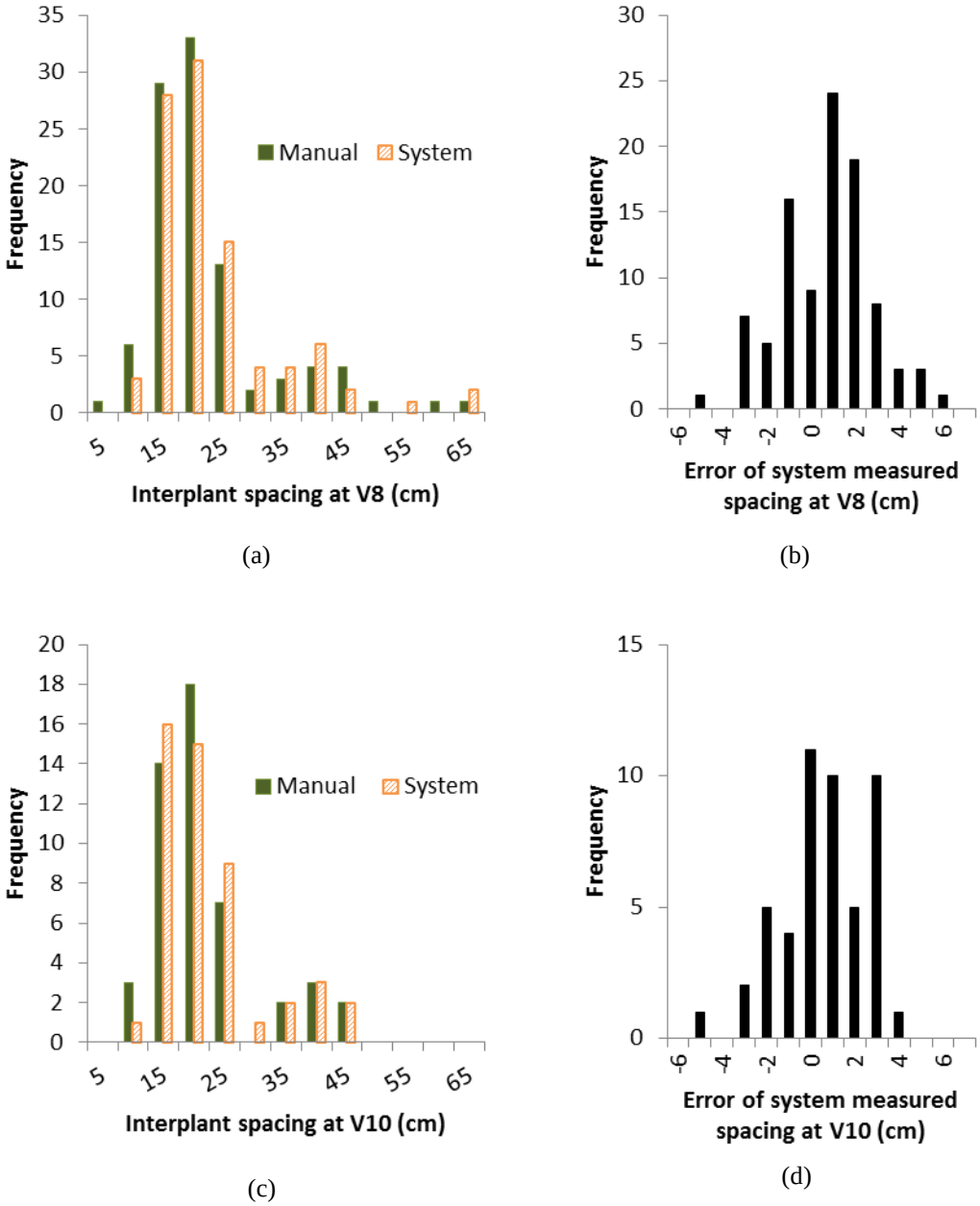


Fig. 10 Histograms of (a) manually and system measured interplant spacing, and (b) error distribution of system measured interplant spaces at the V8 growth stage; (c) manually and system measured interplant spacing and (d) error distribution of system measured interplant spaces at the V10 growth stage

In summary, the developed approach had its unique features comparing with others from previous research. Firstly, it was suitable for sensing corn plant location and spacing at mid-growth stage. Previous work based on 2D imaging and collected data from the top of the canopy could only be applied at early corn growth stages. At the mid-growth stage, the corn canopy might overlap with each other, thus made it difficult to differentiate individual plant. Secondly, the developed approach could reduce interference from leaves and other objects by scanning a plant multiple times. This greatly increased the possibility to identify each plant and improved the accuracy on corn location and spacing measurements when compared with 1D ranger sensors. Thirdly, the developed, laser-scanner-based approach acquired less data for plant measurements; hence, led to faster processing and communication speed and low demands on data processing and storage comparing with 2D imaging methods. This was very important for real-time, field implementation.

Conclusions

A system based on laser line-scan technique was developed and tested to estimate corn plant locations and interplant spacing at mid growth stages. The field experiment results demonstrated that:

- Using a laser line scanner to identify corn plant stalks from different points of view on-the-go is a feasible method for plant counting, location and spacing measurement. The current system achieved 24.0 % and 10.0 % of mean total errors in plant counting at the V8 and V10 growth stages, respectively. The mean RMSEs of the system measured locations for correctly identified plants were 2.3 cm and 2.6 cm at the V8 and V10 growth stages, respectively. Comparison between system measured and manually measured interplant spacing had R^2 of 0.962 and 0.951 for the V8 and V10 growth stages, respectively.

- The system can be enhanced for better performance in the future. Redesign of the wheel encoder mechanism will increase the reliability in data acquisition. Data processing algorithm can also be modified on reducing errors from interfering factors and fine-tuning parameter values.
- For practical deployment, the developed system can be upgraded by using a laser scanner with a faster data communication rate and an on-board fast microprocessor-based data-logging system to accommodate the travel speed of a tractor (e.g. 4-6 mph).

Acknowledgments The authors would like to thank Jeremiah Mullock and Natasha Macnack in Department of Plant and Soil Sciences for their help in plot preparation and manual measurement. Appreciation also goes to Wesley Porter, Jorge Rascon, Marshall Oldham, Bin Li, Aaron Franzen and Yongbo Wan in Department of Biosystem and Agricultural Engineering for their help in field test which made this project successful.

References

Birrell, S. J., & Sudduth, K. A. (1995). Corn population sensor for precision farming. ASAE paper No. 951334. St. Joseph, Michigan, USA.

Dworak, V., Selbeck, J., & Ehlert, D. (2011). Ranging sensor for vehicle-based measurement of crop stand and orchard parameters: a review. *Transactions of the ASABE*, 54(4), 1497-1510.

Ester, M., Kriegel, H., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (pp. 226-231). Menlo Park, CA, USA: AAAI Press.

Gauer, L. E., Grant, C. A., Gehl, D. T., & Bailey, L. D. (1992). Effects of nitrogen fertilization on grain protein content, nitrogen uptake, and nitrogen use efficiency of six spring wheat (Triticum aestivum L.) cultivars, in relation to estimated moisture supply. *Canadian Journal of Plant Science*, 72, 235-241.

Heege, H., Reusch, S., & Thiessen, E. (2004). Systems for site-specific on-the-go control of nitrogen top-dressing during spreading. Proceedings of the 7th International Conference on Precision Agriculture and Other Precision Resources Management (pp. 133-147). Minneapolis, MN, USA.

Huang, Y., Lan, Y., Ge, Y., Hoffmann, W. C., & Thomson, S. J. (2010). Spatial modeling and variability analysis for modeling and prediction of soil and crop canopy coverage using multispectral imagery from an airborne remote sensing system. *Transactions of the ASABE*, 53(4), 1321-1329.

Hummel, J. W., Drummond, S. T., Sudduth, K. A., & Krumpelman, M. J. (2002). Sensing systems for site-specific assessment of corn plants. Proceedings of the 6th International Conference on Precision Agriculture (unpaginated). Madison, WI, USA: ASA, CSSA, and SSSA.

Krall, J. M., Esechie, H. A., Raney, R. J., Clark, S., TenEyck, G., Lundquist, M., et al. (1977). Influence of within-row variability in plant spacing on corn grain yield. *Agronomy Journal*, 69, 797-799.

Kraus, K., & Pfeifer, N. (1998). Determination of terrain models in wooded areas with airborne laser scanner data. *ISPRS Journal of Photogrammetry & Remote Sensing*, 53, 193-203.

Lauer, J. G., & Rankin, M. (2004). Corn response to within row plant spacing variation. *Agronomy Journal*, 96, 1464-1468.

Li, H., Worley, S. K., & Wilkerson, J. B. (2009). Design and optimization of a biomass proximity sensor. *Transactions of the ASABE*, 52(5), 1441-1452.

Luck, J. D., Pitla, S. K., & Shearer, S. A. (2008). Sensor ranging technique for determining corn plant population. ASABE paper No. 084573. St. Joseph, Michigan, USA.

Martin, K. L., Hodgen, P. J., Freeman, K. W., Melchiori, R., Arnall, D. B., Teal, R. K., et al. (2005). Plant-to-Plant Variability in Corn Production. *Agronomy Journal*, 97, 1603-1611.

Martin, K., Raun, W., & Solie, J. (2012). By-plant prediction of corn grain yield using optical sensor readings and measured plant height. *Journal of Plant Nutrition*, 35, 1429-1439.

Nichols, S. W. (2000). Method & apparatus for counting crops. U.S. Patent No. 6073427.

Raun, W. R., & Johnson, G. V. (1995). Soil–plant buffering of inorganic nitrogen in continuous winter wheat. *Agronomy Journal*, 87, 827-834.

Raun, W. R., & Johnson, G. V. (1999). Improving nitrogen use efficiency for cereal production. *Agronomy Journal*, 91, 357-363.

Saeys, W., Lenaerts, B. Craessaerts, G., & De Baerdemaeker, J. (2009). Estimation of the crop density of small grains using LiDAR sensor. *Biosystems Engineering*, 102(1), 22-30.

Shrestha, D. S., & Steward, B. L. (2003). Automatic corn plant population measurement using machine vision. *Transactions of the ASABE*, 46(2), 559-565.

Shrestha, D. S., & Steward, B. L. (2005). Shape and size analysis of corn plant canopies for plant population and spacing sensing. *Transactions of the ASABE*, 21(2), 295-303.

Sowers, K. E., Pan, W. L., Miller, B. C., & Smith, J. L. (1994). Nitrogen use efficiency of split nitrogen applications in soft white winter wheat. *Agronomy Journal*, 86, 942-948.

Tang, L., & Tian, L. F. (2008a). Real-time crop row image reconstruction for automatic emerged corn plant spacing measurement. *Transactions of the ASABE*, 51(3), 1079-1087.

Tang, L., & Tian, L. F. (2008b). Plant identification in mosaicked crop row images for automatic emerged corn plant spacing measurement. *Transactions of the ASABE*, 51(6), 2181-2191.

Thorp, K. R., Steward, B. L., Kaleita, A. L., & Batchelor, W. D. (2008). Using aerial hyperspectral remote sensing imagery to estimate corn plant stand density. *Transactions of the ASABE*, 51(1), 311-320.

Wangler, R. J., Fowler, K. L. & McConnell, R. E. (1994). Object sensor and method for use in controlling an agricultural sprayer. U. S. Patent No. 5278423.

Wei, J., & Salyani, M. (2004). Development of a laser scanner for measuring tree canopy characteristics: phase 1. Prototype development. *Transactions of the ASABE*, 47(6), 2101-2107.

Wei, J., & Salyani, M. (2005). Development of a laser scanner for measuring tree canopy characteristics: phase 2. Foliage density measurement. *Transactions of the ASABE*, 48(4), 1595-1601.

CHAPTER IV

ALGORITHM DEVELOPMENT OF A LIDAR BASED CORN PLANT LOCATION AND SPACING MEASUREMENT SYSTEM FOR MID-SEASON FERTILIZER APPLICATION

The material in this chapter will be submitted to Journal of Computer and Electronics in Agriculture after minor revision.

Abstract

Corn plant location and within-row spacing is important information for in-season yield prediction and variable-rate fertilizer application. An improved corn plant location and spacing measurement system was developed. A LiDAR sensor with a 100° field of view horizontally scanned at the bottom section of plant stalks when the cart was moving down the row. Each stalk appeared in multiple scans from various angles of view which increased the possibility for the stalks to be correctly measured. Compared with the earlier version system, the current system enhanced the data acquisition platform to insure the quality of data collection, and the data processing algorithm especially the scan registration and stalk recognition procedure to reduce the misidentification errors. The current system was tested on 200 plants at the V8 growth stage in 2012. A total error of 5.5% in plant counting and a 1.9 cm of root-mean-squared error (RMSE) in spacing measurement were achieved between the sensor measurements and the manually measured ground truth for data collected in 2012. The improved data processing algorithm was

also tested on the data collected in 2011. The major error – false positive plant counting error – decreased to 14% from 24% for data collected at the V8 growth stage. Overall, this study was a good basis of developing a high spatial resolution sensing system for real-time, variable-rate fertilizer application.

Keywords. Plant population, in-field variability, variable rate technology, image clustering and registration.

Introduction

With the advent of genetic modified varieties and herbicides, corn plants have been planted in a much higher density than what was before the mid-19th century. Competition for resources among plants became an issue. A study conducted in more than 350 commercial corn fields in Indiana and Ohio showed a big range of standard deviation on within-row plant spacing (Nielsen, 2001). Only 16% of the fields had the standard deviation less than 10 cm; about 60% of them were between 10 and 15 cm; and the rest of the fields had this number between 15 and 33 cm. The uneven within-row plant spacing results in an unequal distribution of resources such as water, soil nutrient and sunlight which may cause plant stress and yield loss. Krall et al. (1977) found a consistent yield loss with larger standard deviation of within-row spacing in Kansas and estimated an average of 83.3 kg/ha decrease for every centimeter increase of the spacing variability. A seven-year study on farms across Indiana showed a yield loss at 62 kg/ha for every centimeter increase in the standard deviation of within-row spacing (Nielsen, 2001). Doerge et al. (2002) strengthened previous findings with data collected in Iowa, Missouri and Minnesota. For every centimeter improvement of the within-row spacing standard deviation, the data revealed an average decrease of 84 kg/ha in corn grain yield. All these studies demonstrated a negative correlation between the variation of within-row spacing and the corn grain yield. So the within-

row spacing has been included in the yield prediction model as an input variable and to further guide the in-season fertilizer management (Martin et al., 2012). Monitoring the plant population would also help the insurance companies in the loss assessment after severe stress or disease. All of these applications require an intensive within-row spacing measurement. An automatic system is in demand to replace the time-consuming and labor-intensive manual way.

Various technologies have been studied or already applied for plant counting and spacing/population measurements. Remote sensing is a way to obtain large-scale plant stand density rapidly (Thorp, et al., 2008), while ground-based methods are used to collect detailed crop and soil information and usually can be incorporated with production operations. Corn plant counting systems on combine harvesters often use mechanical, infrared proximity or capacitance sensors. For early and mid-growth stage plants, non-destructive sensing method is commonly used in order to have a minimum impact on plants. Machine vision is a widely used non-destructive sensing technique usually having less limitations on the sensing proximity, object orientation and sensing speed comparing with other non-destructive techniques such as capacitive sensing.

Machine vision technique includes 2D imaging and range sensing. When the midseason fertilizer is applied to corn, the canopy has almost closed. Color images collected from the top view are no longer a best way for the application; instead, viewing from the side of plants is a better approach. Study has been conducted successfully using a 3D range camera to measure corn interplant spacing at early growth stage from the side view (Nakarmi and Tang, 2012). The light source of the commonly available 3D range camera is modulated waves from a LED array. The imaging sensor is subjected to the saturation problem under the strong sunlight due to the relative low power light source.

Light Detection and Ranging (LiDAR) sensors complete a line scan in very short time providing multiple distance measurements along the scan. It is featured with a reliable outdoor performance, a smaller data volume and a faster sampling rate. Because of this, it becomes one of the most investigated range sensors in agriculture applications including the autonomous tractor guidance, yield estimate systems and variable-rate spraying systems in orchard and field crop production (Lee et al., 2007; Saeys, et al., 2009; Chen, et al., 2012). Little study has been conducted using the LiDAR sensors for by-plant sensing on corn yet. Rather than using the laser line scanners, previous studies used laser pointer sensors for corn population estimate (Luck et al., 2008), plant counting and within-row spacing measurement (Rascon, 2012). Both studies showed the prospects of using range sensing technology on this application; however, the interference of leaves on the measurement accuracy and a lack of effective way to eliminate such interference were concluded.

A system for corn within-row plant spacing measurement was developed based on LiDAR technology in 2011 (Shi, et al., 2013). Comparing with other work resulted with a population estimate for a whole field, this system featured by its ability of identifying the location of individual plant along a row. Its performance was tested on 100 plants at their mid-growth stages. Mean total errors in plant counting were 24.0% and 10.0% for the V8 and V10 growth stages, respectively. The root-mean-squared error (RMSE) of the plant location measurements were 2.3 cm and 2.6 cm at those two growth stages, respectively. In the results, the false positive (mistakenly added) counts were relatively high, especially at the V8 growth stage. It was largely caused by the failure of the data processing algorithms on effectively eliminating the leaf interference. Another problem was that the data acquisition platform was cumbersome and its wheel alignment was not good which made it hard to be controlled in the field and affected the quality of data collected. The overall objective of the study presented in this paper was to improve the system performance in terms of the data acquisition platform and the data processing

algorithms to achieve a better accuracy on plant counting, location measurement and within-row spacing measurement. The specific objectives were:

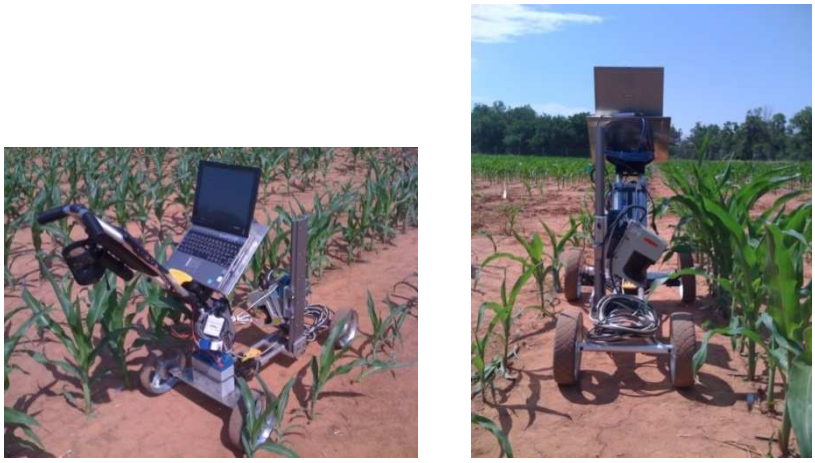
- To improve the data acquisition system to ensure the quality of data collected;
- To improve the data processing algorithm, especially the scan registration and stalk recognition procedure, to reduce the mis-identification error; and
- To evaluate the performance of the improved system by its accuracy on plant counting and within-row spacing measurement at corn mid-growth stage.

Materials and Methods

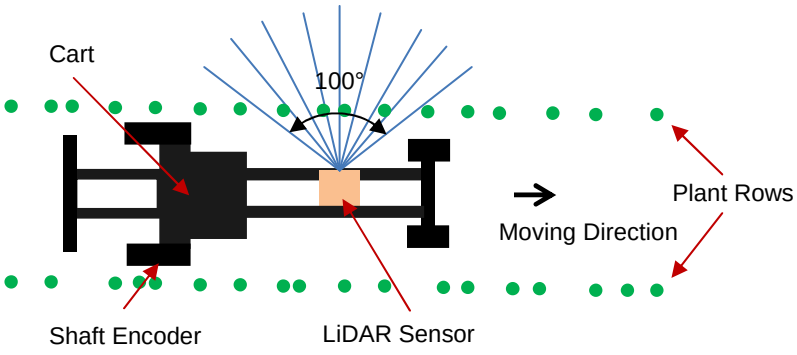
Data Acquisition System

A four-wheel golf cart was modified to serve as a data acquisition platform (Fig. 11(a)). The frame between its front and rear wheels were extended to place a LiDAR sensor, the key component of the data acquisition system. This LiDAR sensor (LMS291, SICK AG, Waldkirch, Germany) measured distances between itself and target objects along a scanning line based on the time-of-flight principle. Its light source was a pulsed laser beam at a near infrared band of 905 nm (SICK, 2006). In this study, the sensor was configured to operate with a field of view of 100° and a resolution of 0.25°. This resulted with 401 distance readings in a scan in a time interval of 53.28 ms. With an average of 0.447 m/s moving speed of the cart, the sensor moved about 2.4 cm within this interval. This offset was ignored in this study.

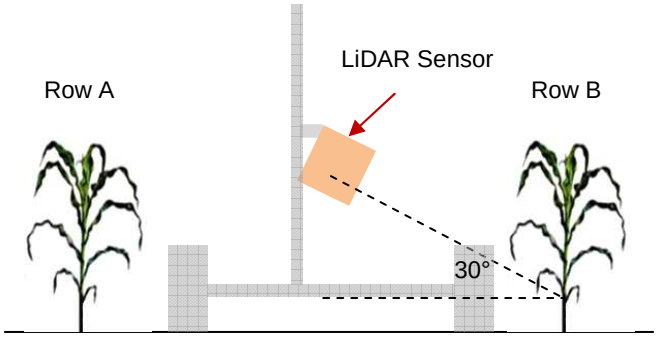
As shown in Fig. 11 (a) and (c), the LiDAR sensor was mounted on a vertical rail with a downward angle of 30° and sensing at the bottom section of the plants. The sensor's mounting height was adjusted so that the sensing height on the plant stalks was kept within 2.5 cm to 11.7 cm above the ground. The sensing height on the plant stalks was selected to be consistent with the study by Kelly (2009). The mounting height of the sensor could be easily adjusted along the rail



(a)



(b)



(c)

Fig. 11 System setup: (a) cart with sensors; (b) an illustration of the top view and (c) an illustration of the side view of the system operating in the field.

and locked on the carriage during system calibration. With a 100° field of view, multiple stalks within a row appeared in a scan as illustrated in Fig. 11(b). Other devices including a battery,

power convertors, a serial to Ethernet convertor, a data acquisition card and an on-board laptop were properly mounted on the cart so that the overall center of mass was close to the rear part of the cart. The front side of the cart could be slightly lifted for direction adjustment whenever it was necessary during the travel. Because of the location of the center of mass and the elasticity of the aluminum wheel frames, the rear wheels were kept contacting with the ground most of the time during the data acquisition. The better wheel alignment of this commercial golf cart also made the direction adjustment much easier. A shaft encoder was placed on a rear wheel and associated with the wheel rotation by the friction force created between a nylon plate and the inner side of the wheel frame (Fig. 12). The friction was strengthened by a spring. The whole design largely reduced the encoder reading loss due to the missing rotations which was an issue in the previous platform (Shi et al., 2013).

A LabVIEW® (National Instrument Co., Austin, Texas) program was developed to control the sensor and conduct data acquisition. Each scan was converted into the Cartesian coordinates and appended to a MS Excel file with the corresponding location stamp from the encoder reading. A normal digital camera was mounted under the LiDAR sensor to record video during each trial as reference.

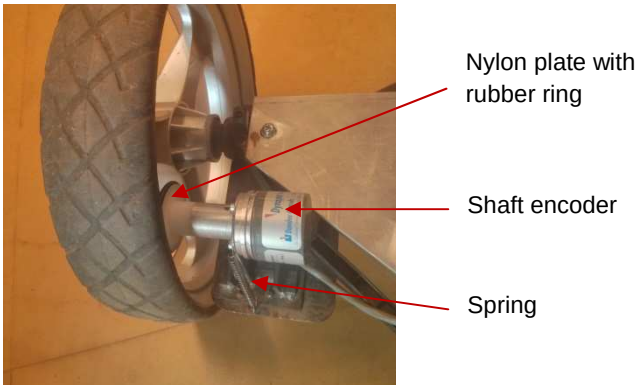


Fig. 12 Encoder associated with one rear wheel.

Field Experiment Setup

Field tests were conducted from May to June in 2012 in a corn field near Lake Carl Blackwell, Stillwater, OK. Eight corn plots were planted in two timings with an interval of three weeks (Table 2). For the four plots planted at the same time, three of them were planted in a population of 49,419 plants/ha (20,000 plants/acre) and the rest one was planted in 64,245 plants/ha (26,000 plants/acre). Different nitrogen rates were applied prior to the planting. Each plot had four rows. Part of the second row in each plot was separated and tagged. There were 25 plants in each tagged row section and 200 plants in total in the eight row sections. Data was collected at their mid-growth stage V8. Five trials were conducted for each row.

Table 2 Planting timings and treatments of the eight experiment plots in year 2012 and two experiment plots in year 2011.

		Planting Timing	Population (plants/ha)	Nitrogen Rate (kg/ha)	Growth Stage(s) when Data was Sampled	Number of Samples (plant)
2012	Plot 1	Apr 10th	49,419	0	V8	25
	Plot 2	Apr 10th	49,419	89.6	V8	25
	Plot 3	Apr 10th	49,419	179.2	V8	25
	Plot 4	Apr 10th	64,245	89.6	V8	25
	Plot 5	May 2nd	49,419	0	V8	25
	Plot 6	May 2nd	49,419	89.6	V8	25
	Plot 7	May 2nd	49,419	179.2	V8	25
	Plot 8	May 2nd	64,245	89.6	V8	25
2011	Plot 1	May 27th	74,129	0	V8, V10	50
	Plot 2	May 27th	74,129	89.6	V8	50

When the cart was pushed down the row, the horizontal distance between the sensor and the corn row was within 30 and 46cm due to the deviation of the trolley from the center line between rows (Fig. 13). The sensor was mounted at a height so that, with this horizontal distance range, the

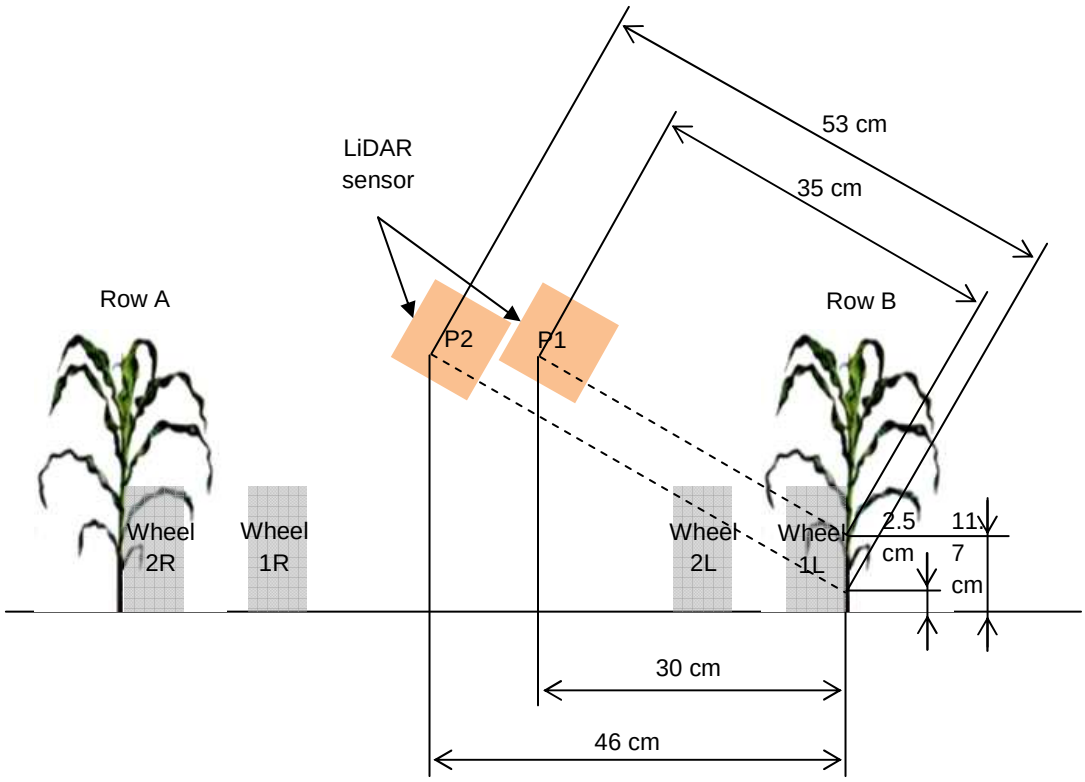


Fig. 13 Illustration of the cart's deviation between rows (not to scale). The LiDAR sensor and wheels marked as '1' show their closest position to the sensing plant row; the LiDAR sensor and wheels marked as '2' show their furthest position to the sensing plant row.

sensing height on the plant stalks was within 2.5 cm and 11.7 cm above the ground. The locations of the 200 plants were manually measured referring to the first plant in each row section used as ground truth data. Data collected in year 2011 using the previous data acquisition platform was used to validate the improved data processing algorithms.

Data Processing Algorithms

Figure 14 shows a flowchart of the data processing. A pre-processing was implemented on the encoder readings to match the start and end readings with the ground truth location data. In this way, the final location calculations could be compared directly with the ground truth data. One scan was read in at a time. Only the data within 35 to 53 cm of the vertical axis of the coordinates which corresponded to the plant row were used for the further processing. The preliminary data

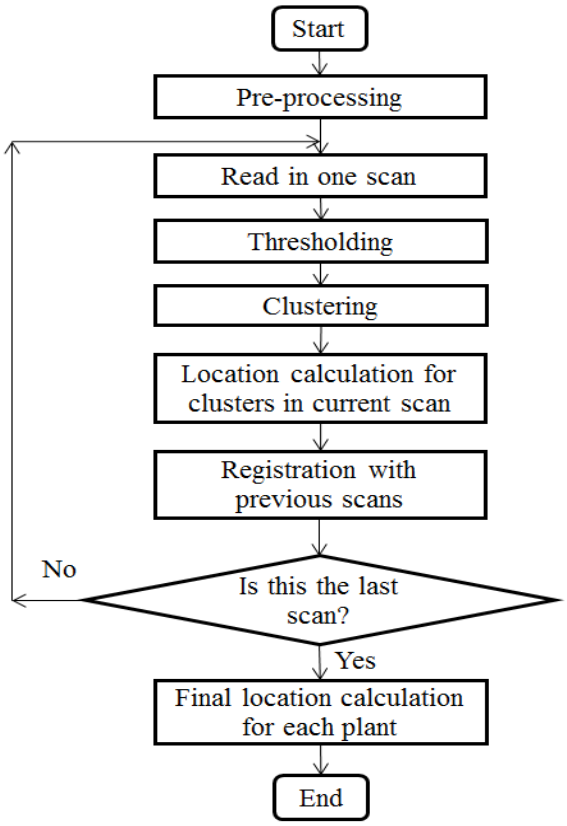


Fig. 14 Flowchart of the data processing for locating corn plants (Shi et al., 2013).

processing algorithms developed last year had an acceptable performance on achieving the functions including 1) automatically recognizing potential stalk clusters in each scan; 2) converting each stalk cluster from the coordinates of individual scan to the ground coordinates, then matching the same stalk clusters appeared in different scans, i.e. scan registration; 3) obtaining the location of each stalk. A step-by-step scrutiny revealed a major error source in the scan registration procedure during which the same stalk clusters appeared in different scans were matched with each other. In addition, the clustering procedure for individual scan sometimes failed to differentiate the noise clusters which were very close to a stalk cluster. These procedures were improved in the current algorithms to reduce the false positive counts in terms of i) a few minor improvements to eliminate noisy clusters in the clustering procedure; ii) a delayed decision-making strategy was implemented in the scan registration procedure; iii) a training process to get the optimal variable values in the algorithms.

Clustering Algorithm

The clustering algorithm for automatically recognizing the potential stalk clusters from a scan of the LiDAR point cloud had a primary clustering followed by a screening process. In the primary clustering, a density-based clustering algorithm was implemented to automatically identify stalk clusters in a scan. This algorithm was more general for processing the LiDAR point cloud. It started from a randomly selected data point p and searched for its neighboring points within a radius, ε. If at least one neighboring point was found, a new cluster was formed consisting of p and its neighbor(s); otherwise, p was marked as a noise point. Once there was a newly-formed cluster, the search then continued on each neighboring point to find their neighbors and include them in the cluster. The search stopped when no more new neighboring point could be found on any of the members in that cluster. The algorithm then tried to find another cluster using the same procedures. The whole searching process stopped when all of the data points in that scan were visited (Shi et al., 2013).

After the primary clustering procedure, a screening process was implemented in order to refine the previous result. This process included three operations designed specifically for the LiDAR data processing of corn plant stalk sensing.

Operation I: Estimating the plant row line.

The purpose of predicting the plant row line was to differentiate the plant leaves from stalks. In a laser scan, the cluster formed by a leaf hanging down often seemed very similar to the cluster formed by a stalk, but would have less possibility to be on the plant row line. The plant row line was moving-averaged by the stalk clusters detected in the previous scans to simulate a real-time operation. A width of ±5 cm was used to tolerant the non-straightness nature of the plant row. Clusters outside this area were considered as leaf cluster and eliminated.

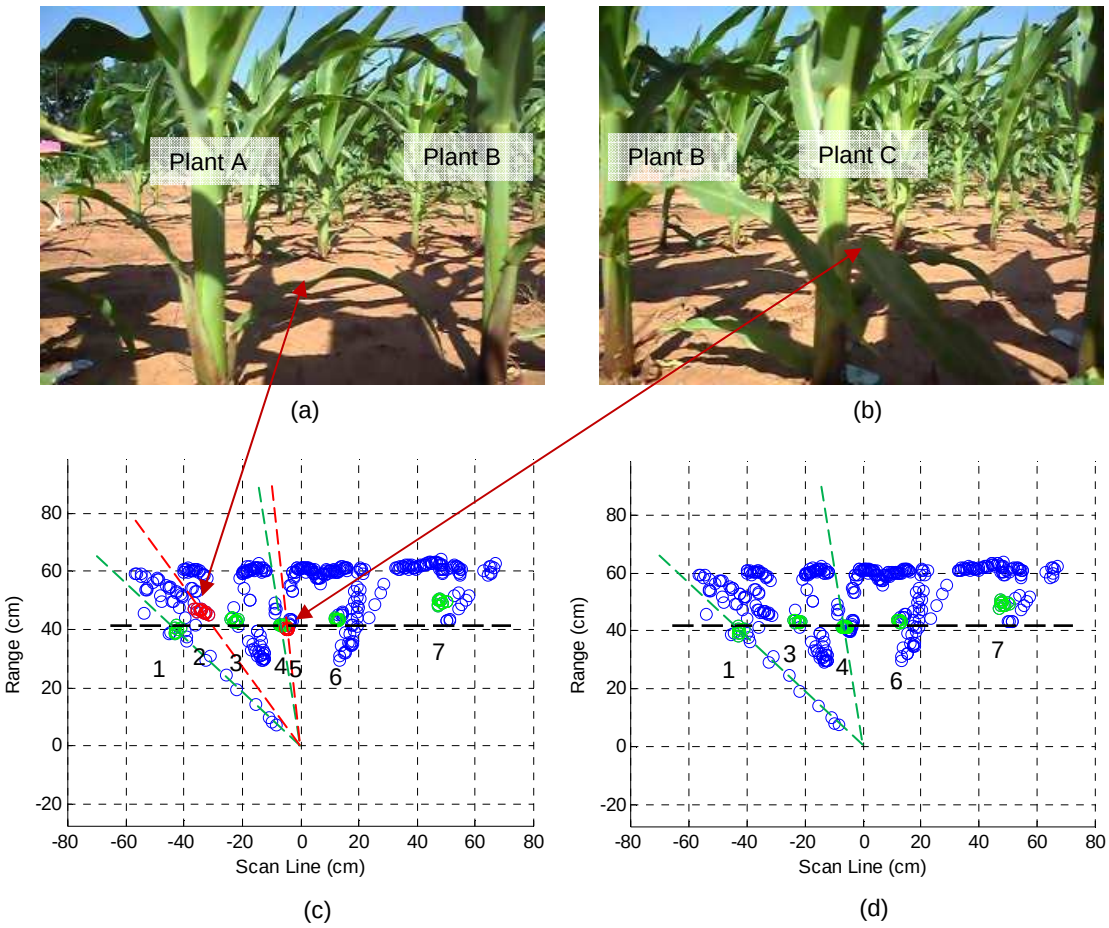


Fig. 15 Clusters close to each other on the radial axes centered the origin was screened: (a) and (b) show scenes of the leaf interference; (c) clustering result without screening out the two noise clusters marked in red; the desired clusters were marked in green; (d) clustering result of the corresponding scenes with noise clusters being screened. The black dashed lines in (c) and (d) represent the estimated plant row line.

Operation II: Screening clusters close to each other on the radial axes centered the origin of the coordinates according to their offsets from the estimated plant row line.

A typical distance between a sheath and a stalk was less than 5 cm. This resulted in a 6.5° difference between the sheath cluster and the stalk cluster at 44 cm away from the origin along the radial axes centered the origin. Giving some tolerance, a threshold of ±10° along the radial axes was selected and clusters closed to each other less than that threshold were grouped. In each group, the cluster further away from the predicted plant row plane was eliminated (Fig. 15). By this way, the sheath interference was reduced in an individual scan; however, potential errors

were induced by eliminating some clusters due to their locations further away from the row center line than their neighboring 'sheath' clusters.

Operation III: Screening all clusters again according to their size on the tangential direction centered the origin of the coordinates.

The center of mass (x_c, y_c) for each cluster was calculated (Eq. 4 and 5) and a line was drawn through it being tangential to the arc centered the origin of the coordinates (Fig. 16). (x_i, y_i) in the equations are the coordinates of individual data points in the cluster with a mass of m_i. Assume m_i equaled to one for each data point, the total mass of a cluster would equal to the number of data points it had which was n. The length of the longest line segment (D) between the projection of all the data points on the tangential line was measured and used as the size of that cluster. A cluster was considered to be a valid cluster only if its size was within 1 and 4 cm. These thresholds were decided by the size of corn stalks at V8 and V10 growth stages in this study. After these three operations, some noise clusters were effectively eliminated.

$$x_c = \frac{\sum_{i=1}^{n} m_i \cdot x_i}{m} = \frac{\sum_{i=1}^{n} 1 \cdot x_i}{1 \cdot n} = \frac{\sum_{i=1}^{n} x_i}{n} \quad \text{Eq. 4}$$

$$y_c = \frac{\sum_{i=1}^{n} m_i \cdot y_i}{m} = \frac{\sum_{i=1}^{n} 1 \cdot y_i}{1 \cdot n} = \frac{\sum_{i=1}^{n} y_i}{n} \quad \text{Eq. 5}$$

At the end of the clustering procedure, the location of each identified cluster was converted from the scan coordinates to the ground coordinates making use of the encoder reading of each scan. The data was ready for the registration process.

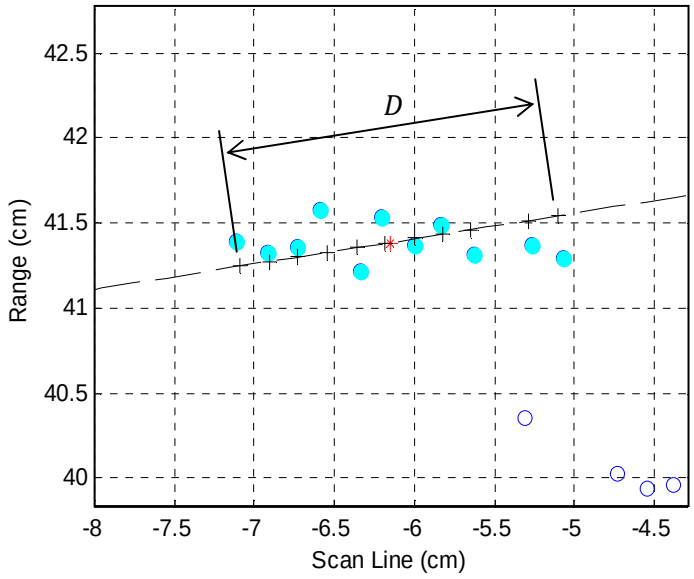


Fig. 16 Clusters were screened according to their size on the tangential direction centered the origin. '•' represents the recognized laser data cluster; '*' represents the center of mass of the recognized cluster; '— —' represents the line tangential to the arc centered the origin; '+' represents the projection of the laser data point on the tangential line; D is the measured width of this cluster.

Registration and Recognition Algorithm

In the registration and recognition process, the clusters in different scans corresponding to the same plant stalk were matched, and the stalk and the noise clusters were differentiated. When the cart passed by a plant, the plant stalk generally appeared in 30 – 60 continuous scans before it moved out of the sensor's field of view. This number of scans depended on the sampling rate of the sensor, the communication speed of the control system as well as the moving speed of the data acquisition platform. Multiple scans from various points of view gave a better chance to observe and recognize a stalk even if it was blocked by leaves in some scans. A correct registration of the same stalk in different scans as well as a correct differentiation between the stalk and leaf clusters were critical in successfully measuring the plant locations and spacing. A registration and recognition algorithm has already been developed in the previous work (Shi et

al., 2013) making use of the spatial shift obtained from the encoder readings. However, many leaf clusters were falsely treated as stalk clusters which resulted in a large number of false positive counts.

In order to solve this problem, a delayed decision-making strategy was adopted. A cluster was not assigned to a stalk or noise cluster immediately at the first time it was recognized; instead, the number of times it appeared in the consecutive scans was counted until the counts reached a certain number. A cluster first being recognized as a potential stalk cluster might turn out to be a noise cluster at the end if it did not appear enough times; on the other hand, a cluster first being recognized as a noise cluster could be considered later as a stalk cluster as long as it gained enough show-up counts. Two dynamic waiting lists – one for the potential stalk clusters and the other for the potential noise clusters – were updated after each scan was processed.

Each cluster in the current scan was registered with a cluster in its previous scans using the difference of the encoder readings between the current and previous scans. A scan was only registered with its previous scans to simulate a real-time operation. In the case that the corresponded cluster in a previous scan was blocked by leaves, the algorithm would search backward through up to 30 scans until a matched cluster being found. A buffered search area was set around the estimated cluster location in a previous scan calculated by the encoder reading to tolerate the encoder reading error (Fig. 17). The size of this buffered search area was represented using a variable '*bsa*'. A range of values was tested for '*bsa*' (±2, ±4, ±6, ±8, ±10 and ±12 cm) in order to find an optimal value resulting with a most accurate plant location measurement.

Except for the first scan, each cluster in the currently processed scan was checked for its match with a certain cluster in one of the previous scans.

if a matched cluster in one of the previous scans can be found

if the matched cluster has already been determined to be a stalk cluster or a noise cluster, then assign the cluster in the current scan to the same category

else if the matched cluster is on a waiting list of stalk clusters or noise clusters, then put the cluster in the current scan to that list also

else mark the cluster as an unmatched one

Each of the unmatched clusters in the current scan was assigned to one of the dynamic waiting lists.

if it is the latest one entering the sensor's field of view or there was no matched cluster after it, then put it on the waiting list of stalk clusters

else if it is in between two matched clusters or is the first cluster entering the current scan, then put it on the waiting list of noise clusters

At the end of the processing of each scan, the waiting lists were checked to see if there were enough counts for a particular cluster to be determined as a valid stalk or noise cluster. The minimum number of counts a cluster needed to achieve in order to be considered as a valid stalk cluster was represented using a variable '*mnc*'. A range of values was tested for '*mnc*' (10%, 20%, 30%, 40% and 50% of the averaged total number of scans in which a stalk appeared) in order to find an optimal value resulting with a most accurate plant location measurement. Finally, the interquartile of the entire location measurements corresponding to a same stalk was averaged to be its estimated location.

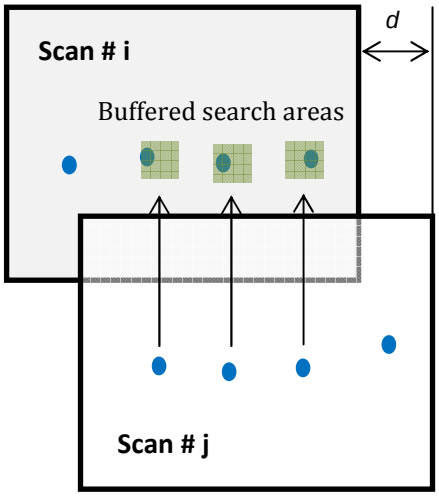


Fig. 17 Illustration of the scan registration algorithm (Shi et al., 2013). Scan #j is the current scan while scan #i is a previous scan. Blue dots represent stalk clusters. d is the spatial shift or the difference of encoder readings between the two scans. Green squares are the buffered search areas.

Selecting Optimal Values for Variables *bsa* and *mnc*

The value selection of the two variables *bsa* and *mnc* in the algorithms would influence the accuracy of plant location measurement. A range of values were tested for both variables using a training data set in order to find an optimal value combination resulting with minimum measurement error. In order to decide the number of counts (*mnc*) a cluster needed to gain in order to be treated as a valid stalk cluster, the averaged total number of scans each stalk appeared $\bar{n}$ was estimated for each trial. The sensor's field of view on the plant row when the cart was in the middle between the rows was about 105 cm. When a stalk was near the edge of a scan, its profile may be incomplete. Hence, an effective field of view of 95 cm was used. Then the average number of scans each stalk appeared could be calculated using Eq. 6. The number of counts a cluster needed to have in order to be treated as a valid stalk cluster should depend on the moving speed of the data acquisition platform, $\bar{v}$. $\bar{v}$ was estimated by the total length of the row section L and the time used to complete the data acquisition t. L varied from row to row and t varied from trial to trial. Five values related with the estimated total number of show-up scans $\bar{n}$ were tested

(Eq. 7). Except the variable '*mnc*', another variable tuned was '*bsa*' – the size of the buffered search area in the scan registration procedure. Six values were tested (Eq. 8). So there were 30 possible *bsa-mnc* combinations.

$$\bar{n} = \frac{FOV/\bar{v}}{t} = \frac{95/(L/T)}{0.053} \quad \text{Eq. 6}$$

$\bar{n}$ – average number of scans a stalk appeared in the sensor's field of view;

FOV – sensor's effective field of view when cart was in the center of two rows (in cm); set as 95 cm;

$\bar{v}$ – average speed (in cm/s);

t – time needed for a scan to be completed (in seconds); set as 0.053 seconds;

L – total length between the 1st and the 25th sampled plants in a row (in cm);

T – total time spend for the cart to move from the 1st to the 25th sampled plant in a row (in seconds).

$$mnc = [0.1\ 0.2\ 0.3\ 0.4\ 0.5] * \bar{n} \quad \text{Eq. 7}$$

$$bsa = [2\ 4\ 6\ 8\ 10\ 12]\ cm \quad \text{Eq. 8}$$

Due to the system malfunction during the data acquisition, only 35 out of the total 40 data sets collected in 2012 were valid for using. Twenty-five data sets were randomly selected out of the 35 data sets to comprise the training data set; the rest 10 data sets and the eight data sets collected in 2011 formed the test data set. An index *OF* was calculated as a weighted combination of the percentages of false negative counting error (p_{fn}), false positive counting error (p_{fp}) and the RMSE of location measurement (p_{rmse}) (Eq. 9). In this study, the weight of false negative error w_{fn} was set to 2; the weight of false positive error w_{fp} was set to 1; and the weight of RMSE w_{rmse} was set to 0.25. This was because we considered the false negative error was worse than

the false positive error. p_{rmse} was zero when the RMSE was less than or equal to 2.54 cm; and was calculated as the percentage off 2.54 cm when the RMSE was larger than 2.54 cm (Eq. 10). However, other weights combinations were also investigated and they were shown in Table 3. The OF values were calculated with each *bsa-mnc* combination for each training data set. Under each combination, the OF values of the same *bsa-mnc* combination were summed together.

$$OF = p_{fn} \cdot w_{fn} + p_{fp} \cdot w_{fp} + p_{rmse} \cdot w_{rmse} \quad \text{Eq. 9}$$

p_{fn}, p_{fp} and p_{rmse} – the percentages of false negative counts, false positive counts and RMSE, respectively;

w_{fn}, w_{fp} and w_{rmse} – the weights of p_{fn}, p_{fp} and p_{rmse} in the index function, respectively.

$$p_{rmse} = \begin{cases} 0, & RMSE \leq 2.54\ cm \\ \dfrac{RMSE - 2.54}{2.54} * 100, & RMSE > 2.54\ cm \end{cases} \quad \text{Eq. 10}$$

Table 3 Four combinations of the weights of p_{fn}, p_{fp} and p_{rmse} for calculating index *OF*.

	w_{fn}	w_{fp}	w_{rmse}
Combination 1	2	1	0.25
Combination 2	1	1	0.25
Combination 3	1	2	0.25
Combination 4	2	1	0.75

System Performance Evaluation

System performance was evaluated on the data collected in years 2011 and 2012 which consisted of 300 plants in total in terms of the plant counting accuracy and the accuracy of plant location estimate and within-row spacing estimate. Only the training data set including 10 trials collected in 2012 and 9 trials collected in 2011 were used. As for the plant counting accuracy, three errors

were defined – the false negative error (FNEr) (Eq. 11), the false positive error (FPEr) (Eq. 12) and the total error (TEr) (Eq. 13). If no plant was identified within ±10 cm from the ground truth location of an actual plant, there was a false negative count for that actual plant. Similarly, if no actual plant was located within ±10 cm of the location of an identified plant, this resulted in a false positive count. Only the recognized plant closest to (and within ±10 cm of) the location of an actual plant was a valid count; multiple counts of an actual plant were treated as false positive counts.

$$FNEr = \frac{Missing\ Count}{Ground\ Truth\ Count} \times 100\ \% \quad \text{Eq. 11}$$

$$FPEr = \frac{Adding\ Count}{Ground\ Truth\ Count} \times 100\ \% \quad \text{Eq. 12}$$

$$TEr = FNEr + FPEr \quad \text{Eq. 13}$$

As for the accuracy of plant location and within-row spacing estimates, the root mean square error (RMSE) was calculated according to the system measured and the ground truth data. In order to keep a consistency, locations corresponding to the false negative counts were eliminated from the ground truth data while the locations corresponding to the false positive counts were eliminated from the system measured data.

Results and Discussion

Optimal Values of Variables *bsa* and *mnc*

For each value combination of the buffered search area and minimum number of counts (*bsa-mnc* combination), the sums of index *OF* values for the training set were shown in Table 4-7; each table with a different weight combination. The smallest number in each table was underlined indicating the optimal *bsa-mnc* combination. The italic numbers around the smallest number

showed an area where the errors were small. When the weight combination was w_{fn}=2, w_{fp}=1 and w_{rmse}=0.25, the smallest error was at the intersection of *mnc* = 0.2$\bar{n}$ and *bsa* = 6 cm (Table 4). It moved towards the larger *bsa* and *mnc* values as the weight of the FPEr increased (Table 5). And when the FPEr was emphasized more than the FNEr, this number moved to the intersection of *mnc* = (0.2 ~ 0.3) $\bar{n}$ and *bsa* = 8 cm (Table 6). This trend was reasonable. The larger the search buffered area (*bsa*) during the scan registration procedure, the larger chance two closed clusters would be combined and so the less the FPEr would be. Also the larger the minimum number of scans (*mnc*) a cluster needed to appear in order to become a valid stalk cluster, the less chance of FPEr would happen. The weight of the RMSE of the location measurement did not affect much on the optimal *bsa -mnc* combination when the other weights were the same (Table 7); however, it had a little effect on the *bsa* value selection. The small errors shifted a little to the smaller *bsa* when the RMSE of the location measurement was weighted more. An advice according to these results was that the optimal *bsa -mnc* combination needs to be selected according to the specific application. In this study, *mnc* = 0.2$\bar{n}$ and *bsa* = 6 cm was selected with the weight combination of w_{fn}=2, w_{fp}=1 and w_{rmse}=0.25. The test set confirmed the training result (Table 8).

Table 4 Sum of the OF values at each buffered search area (*mnc*) and minimum number of counts (*bsa*) combination for the training set with weight combination of w_{fn}=2, w_{fp}=1 and w_{rmse}=0.25.

		mnc				
		0.1$\bar{n}$	0.2$\bar{n}$	0.3$\bar{n}$	0.4$\bar{n}$	0.5$\bar{n}$
bsa (cm)	2	568.9	380.8	420.1	684.3	1040.8
	4	501.8	329.1	321.1	429.3	758.3
	6	319.2	*199.3*	*231.5*	378.1	607.8
	8	271.1	*231.3*	252.1	400.1	616.6
	10	411.1	435.8	489.0	608.5	836.7
	12	835.4	878.3	949.8	1065.6	1233.7

Table 5 Sum of the OF values at each buffered search area (*mnc*) and minimum number of counts (*bsa*) combination for the training set with weight combination of w_{fn}=1, w_{fp}=1 and w_{rmse}=0.25.

		mnc				
		$0.1\bar{n}$	$0.2\bar{n}$	$0.3\bar{n}$	$0.4\bar{n}$	$0.5\bar{n}$
bsa (cm)	2	528.9	328.8	316.1	416.3	592.8
	4	473.7	301.0	249.0	269.3	430.2
	6	283.0	*171.6*	*164.1*	217.6	339.6
	8	179.0	*143.2*	*139.1*	208.1	335.8
	10	252.6	247.7	287.1	331.0	467.7
	12	509.6	525.2	576.3	635.1	731.8

Table 6 Sum of the OF values at each buffered search area (*mnc*) and minimum number of counts (*bsa*) combination for the training set with weight combination of w_{fn}=1, w_{fp}=2 and w_{rmse}=0.25.

		mnc				
		$0.1\bar{n}$	$0.2\bar{n}$	$0.3\bar{n}$	$0.4\bar{n}$	$0.5\bar{n}$
bsa (cm)	2	1020.9	632.8	512.1	564.3	696.9
	4	941.1	565.0	425.1	397.2	529.9
	6	535.1	291.1	256.2	290.1	378.8
	8	295.8	*183.2*	*160.1*	229.9	357.1
	10	290.8	289.5	304.0	352.0	467.6
	12	568.2	612.0	628.5	715.9	774.4

Table 7 Sum of the OF values at each buffered search area (*mnc*) and minimum number of counts (*bsa*) combination for the training set with weight combination of w_{fn}=2, w_{fp}=1 and w_{rmse}=0.75.

		mnc				
		$0.1\bar{n}$	$0.2\bar{n}$	$0.3\bar{n}$	$0.4\bar{n}$	$0.5\bar{n}$
bsa (cm)	2	554.5	390.5	428.2	677.0	1082.7
	4	488.3	342.9	327.1	439.7	738.6
	6	319.0	*210.2*	*234.6*	386.9	615.3
	8	269.6	263.5	288.4	409.4	660.0
	10	516.4	518.2	609.2	729.6	961.8
	12	1026.6	1109.5	1185.7	1345.0	1562.1

Table 8 Sum of the OF values at each buffered search area (*mnc*) and minimum number of counts (*bsa*) combination for the validation set with weight combination of w_{fn}=2, w_{fp}=1 and w_{rmse}=0.25.

		mnc				
		$0.1\bar{n}$	$0.2\bar{n}$	$0.3\bar{n}$	$0.4\bar{n}$	$0.5\bar{n}$
bsa (cm)	2	347.3	*248.0*	*253.3*	378.2	622.6
	4	363.3	*242.1*	*234.3*	292.2	478.9
	6	297.5	*219.5*	*226.5*	258.2	406.8
	8	324.0	308.8	323.8	375.1	465.2
	10	460.5	491.6	559.2	624.3	704.7
	12	860.0	911.7	970.1	1027.8	1102.0

Plant Counting Error

The current system achieved an averaged 1.0% (SD = 1.7%) FNEr, an averaged 4.5% (SD = 5.1%) FPEr and an averaged 5.5% (SD = 5.3%) TEr on the testing data set collected in 2012. It also made an improvement on the data collected in year 2011 with an average 3.6% (SD = 2.2%) FNEr, an 8.1% (SD = 4.2%) FPEr and an averaged 11.7% (SD = 5.5%) TEr (Table 9). Comparing with the performance of the previous version data processing algorithm (Shi, et al.,

2013), the current data processing algorithm achieved a large improvement of the plant counting accuracy on the same data collected in 2011 (Fig. 18). The averaged FPErs dropped from 22.0% to 10.7% for data collected at V8 growth stage and from 7.0% to 3.0% for data collected at V10 growth stage. The averaged FNErs increased a little by using the current data processing algorithm – from 2% to 3.3% and from 3.0% to 4.0% on V8 and V10 growth stages, respectively – which was mainly due to the adoption of the '*mnc*' variable to eliminate clusters not appearing enough times in the scans. However, due to the large reduction of the FPEr, a significant overall improvement on the averaged TErs was still achieved on the 2011 data – it decreased from 24% to 14% for V8 data and from 10% to 7% for V10 data. It was also noticed that the standard deviations of the errors resulted from the current algorithm were also smaller than those resulted from the old algorithm which indicated a more constant system performance by the current algorithm.

Table 9 Errors of plant counting and stalk location estimates compared to ground truth data.

	FNEr (%)	FPEr (%)	TEr (%)	RMSE of location estimate (cm)	RMSE of spacing estimate (cm)
2012	1.0 (1.7) *	4.5 (5.1)	5.5 (5.3)	1.9 (0.2)	1.9 (0.4)
2011	3.6 (2.2)	8.1 (4.2)	11.7 (5.5)	2.4 (0.6)	2.6 (0.8)

* Numbers in the parenthesis represent standard deviation

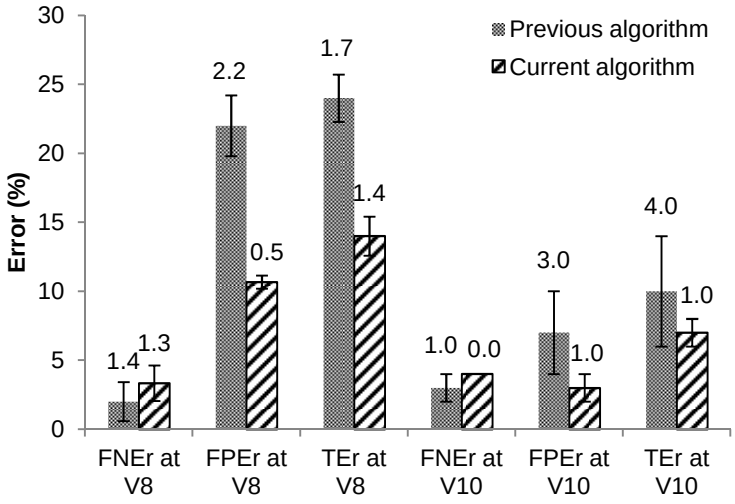
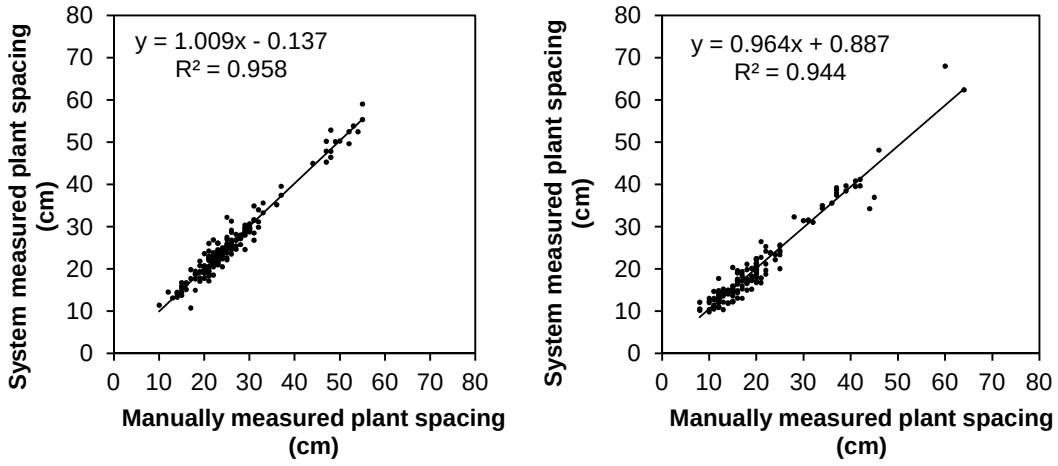


Fig. 18 Comparison of errors between old and current algorithms on data collected in year 2011. Data labels on top of the bars were the standard deviations across the trials.

Plant Location and Spacing Measurement Error

The current system achieved averaged RMSEs of 1.9 cm (SD = 0.2 cm) and 2.4 cm (SD = 0.6 cm) on the plant location measurement for the testing data set collected in 2012 and 2011, respectively (Table 4). Similarly, the system measured within-row spacing was highly correlated to the manually measured ground truth data in both years (Fig. 19). For the 2012 data, an RMSE of 1.9 cm (SD = 0.4 cm) and an R^2 of 0.958 with a slope of 1.01 and an intercept of -0.1 cm were achieved. For the 2011 data, an RMSE of 2.6 cm (SD = 0.8 cm) and an R^2 of 0.944 with a slope of 0.964 and an intercept of 0.9 cm were achieved. Considering with the average within-row plant spacing which was about 20 ~ 25 cm (Fig. 20) and the unavoidable inaccuracy in the ground truth data because the manual measurements were conducted by different operators, these errors on plant location and within-row spacing measurements were within an acceptable range.



(a) (b)

Fig. 19 Comparison between system measured and manually measured within-row spacing on (a) data collected in year 2012 (n = 188) and (b) data collected in year 2011 (n = 140).

This performance was close to the performance of the previous system on the same data collected in year 2011. The previous system had an RMSE of 2.5 cm (SD = 0.8 cm), an R^2 of 0.962 and a slope of 0.975 with an intercept of 0.643 in the regression equation on the within-row spacing measurement at V8 growth stage; and an RMSE of 2.0 cm (SD = 0.1 cm), an R^2 of 0.951 and a slope of 0.995 with an intercept of 0.247 in the regression equation on the within-row spacing measurement at V10 growth stage (Shi, et al., 2013). Since the data was collected using the previous data acquisition platform, the differences on the results were only due to the change in the data processing algorithms. The slight decrease on the R^2 values was trivial considering with the large increase on the plant counting accuracy. It was reasonable that the successful detection of those 'not easily detected' and previously missing plants by the improved algorithm induced more errors on the plant location and within-row spacing measurements.

The manually and system measured within-row spacing had similar distributions (Fig. 20). For the 2012 data, the manually measured within-row spacing had a mean of 25.6 cm (SD = 8.6 cm), while those measured by the current system had a mean of 25.7 cm (SD = 8.8 cm). For the 2011 data, the manually measured within-row spacing had a mean of 20.4 cm (SD = 9.9 cm), while those measured by the current system had a mean of 20.5 cm (SD = 9.8 cm). Each column in Fig.

20 represented the frequency of the certain within-row spaces no greater than its tick marked number. We could see that the system tended to miss the measurements on the small spaces which were less than 10 cm on both years' data. This was largely due to the filtering algorithm in the clustering procedure eliminating the sheath interference as well as the 6 cm buffered search area in the registration procedure. The normal distribution pattern of the errors of the system measured within-row spacing on both years' data partially demonstrated the reliability of the system performance (Fig. 21). Each column in Fig. 21 also represented the frequency of the certain errors no greater than its tick marked number. The measurement error had means of 0.10 cm (SD = 1.8 cm) and 0.16 cm (SD = 2.4 cm) on the data collected in 2012 and 2011, respectively. The error of the 2011 data resulted from the previous data processing algorithm had a mean of 0.1 cm with a SD of 1.9 cm (Shi, et al., 2013). The slight increase of the spacing measurement error was also very possibly due to the large decrease of the plant counting error – more error sources were included in.

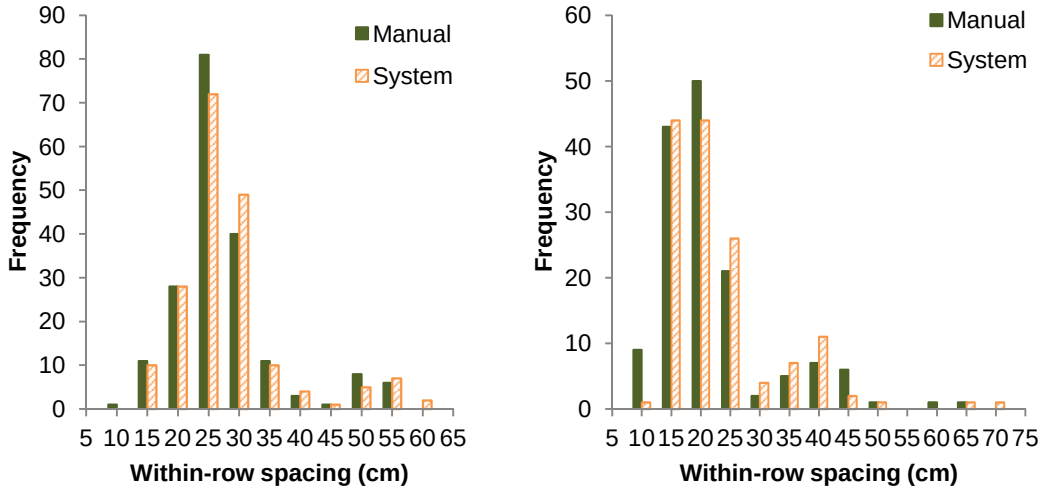


Fig. 20 Histograms of manually and system measured within-row spacing in (a) year 2012 and (b) year 2011. Each column in the plots represented the frequency of the within-row spaces no greater than its tick marked number.

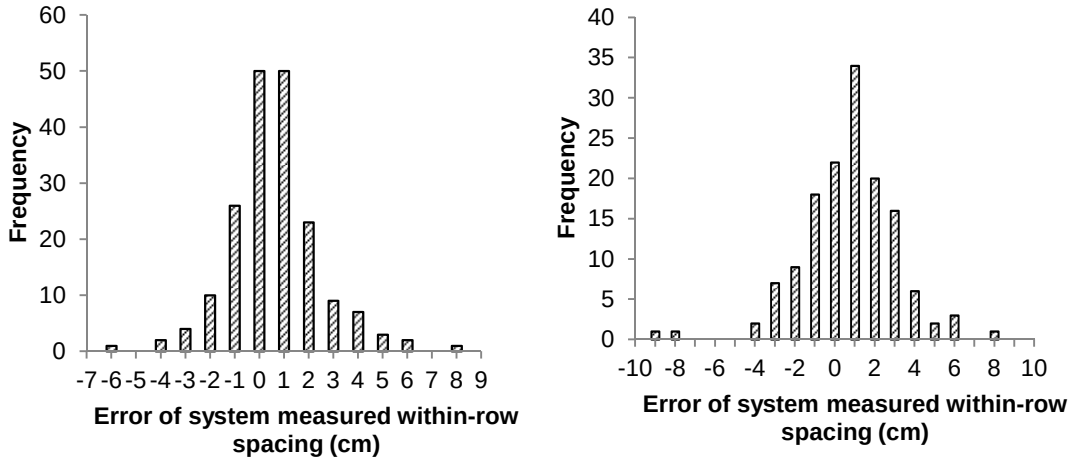


Fig. 21 Error distribution histograms of within-row plant spacing measured by the current system on data collected (a) in year 2012 and (b) in year 2011. Each column in the plots represented the frequency of the errors no greater than its tick marked number.

Conclusions

A corn plant location and spacing measurement system based on LiDAR technology was successfully improved in terms of its data processing algorithm and its data acquisition platform. The performance of the current system was evaluated and compared to the previous system on existing and newly collected field data. Major achievements were:

- By a delayed decision-making strategy in the scan registration procedure and an optimization on the variable values in the current data processing algorithm, the major error in the previous system – the false positive errors in the plant counting at V8 growth stage – was largely reduced from 24.0% to 14.0% on the same data collected using the previous data acquisition system.
- Current system resulted with an averaged 5.5% and 11.7% mean total errors in plant counting, 1.9 cm and 2.6 cm mean RMSEs in plant within-row spacing measurement on the data collected in year 2012 and 2011, respectively.

Acknowledgement

The authors would like to thank Natasha Macnack and Jeremiah Mullock in the Department of Plant and Soil Sciences for their help in the plot preparation and manual measurement; Michael Chavez in the Department of Biosystems and Agricultural Engineering for his help in the field experiment in 2012.

References

Chen, Y., Zhu, H., Ozkan, H. E., 2012. Development of a variable-rate sprayer with laser scanning sensor to synchronize spray outputs to tree structures. *Transactions of ASABE* 55 (3), 773-781.

Doerge, T., Hall, T. Gardner, D., 2002. New research confirms benefits of improved plant spacing in corn. *Crop Insights* 12 (2), 1-5.

Krall, J. M., Esechie, H. A., Raney, R. J., Clark, S., TenEyck, G., Lundquist, M., Humburg, N. E., Axthelm, L. S., Dayton, A. D., Vanderlip, R.L., 1977. Influence of within-row variability in plant spacing on corn grain yield. *Agronomy Journal* 69, 797–799.

Lee, K. H., Ehsani, R., Schueller, J. K., 2007. Forward movement synchronization of two vehicles in parallel using a laser scanner. *Applied Engineering in Agriculture* 23 (6), 827-834.

Li, H., Worley, S. K., & Wilkerson, J. B. (2009). Design and optimization of a biomass proximity sensor. Transactions of the ASABE, 52(5), 1441-1452.

Luck, J. D., Pitla, S. K., Shearer, S. A., 2008. Sensor ranging technique for determining corn plant population. ASABE paper No. 084573. Providence, Rhode Island: ASABE.

Martin, K., Raun, W., Solie, J., 2012. By-plant prediction of corn grain yield using optical sensor readings and measured plant height. *Journal of Plant Nutrition* 35 (9), 1429-1439.

Nakarmi, A. D., Tang, L., 2012. Automatic inter-plant spacing sensing at early growth stages using a 3D vision sensor. *Computers and Electronics in Agriculture* 82, 23-31.

Nielsen, R. L., 2001. Stand establishment variability in corn. AGRY-91-1 (Rev. Nov. 01). W. Lafayette, Ind.: Purdue University, Department of Agronomy.

Rascon, J. A., 2012. Corn sensor development for by-plant management. Master's thesis, Oklahoma State University.

Saeys, W., Lenaerts, B., Craessaerts, G., De Baerdemaeker, J., 2009. Estimation of the crop density of small grains using LiDAR sensor. *Biosystems Engineering* 102(1), 22-30.

Shi, Y., Wang, N., Taylor, R. K., Raun, W. R., Hardin, J. A., 2013. Automatic corn plant location and spacing measurement using laser line-scan technique. *Precision Agric.* 14, 478-494.

Shrestha, D. S., Steward, B. L., 2003. Automatic corn plant population measurement using machine vision. *Transactions of the ASABE* 46 (2), 559–565.

Shrestha, D. S., Steward, B. L., 2005. Shape and size analysis of corn plant canopies for plant population and spacing sensing. *Transactions of the ASAE* 21(2), 295−303.

Tang, L., Tian, L. F., 2008a. Real-time crop row image reconstruction for automatic emerged corn plant spacing measurement. *Transactions of the ASABE* 51 (3), 1079-1087.

Tang, L., Tian, L. F., 2008b. Plant identification in mosaicked crop row images for automatic emerged corn plant spacing measurement. *Transactions of the ASABE* 51 (6), 2181-2191.

Thorp, K. R., Steward, B. L., Kaleita, A. L., Batchelor, W. D., 2008. Using aerial hyperspectral remote sensing imagery to estimate corn plant stand density. *Transactions of the ASABE* 51(1), 311-320.

CHAPTER V

CORN STALK DIAMETER MEASUREMENT USING TWO MACHINE VISION APPROACHES

The material in this chapter will be submitted to Transactions of ASABE after medium revision.

Abstract

Corn plant stalk diameter correlates well with final grain and biomass yield and has been recognized as an important variable in predicting potential yield in season. In this study, two approaches based on machine vision technology for automatic on-the-go corn plant stalk diameter measurement were developed and compared. The LiDAR-and-RGB Approach was a combination of a ground-LiDAR sensor and a common webcam. The 3D Range Imaging Approach used a 3D range camera which offered the distance and shape information simultaneously. The sensors were sensing at the lower section of plant stalks from the side view when they were moving along the row. Data and image processing algorithms were developed for each approach to identify the existence of a plant stalk and estimate its diameter using both shape and range information. Registration was implemented to match the same stalk in different scans/images. Both approaches were tested with four row sections in total 98 plants at their V12 growth stage in 2013. The averaged root-mean-squared-errors (RMSE) of diameter measurement were 4.1 mm and 3.9 mm

for the LiDAR-and-RGB Approach and 3D Range Imaging Approach, respectively. The major error source of the LiDAR-and-RGB Approach was from the stalk segmentation in color images and the mis-synchronization between the two sensors' view. The major error source of the 3D Range Imaging Approach was noise pixels due to the failure of background light resistance. This study provided a basis of developing a real-time corn plant stalk diameter measurement system for in-season yield prediction and variable-rate application system.

Keywords

Plant stalk diameter, variable-rate technology, LiDAR, 3D camera, image processing

Introduction

Corn stalk diameter is one of those common indicators of corn plant growth status. A plant with a thicker stalk diameter is usually considered to be healthier and have a higher yield potential than other plants at the same growth stage. A study conducted at both irrigated and non-irrigated tilled locations in Oklahoma and Ciudad Obregón, Mexico, from 2009 to 2011, found that the index of 'stalk diameter × plant height' for individual corn plant correlated well with its final grain yield (Kelly, 2011). The R^2 was 0.67 at the V12 growth stage. Another study conducted during the same time period at non-tillage and non-irrigated locations in Alabama also found a high correlation between final corn grain yield and a predicting model using plant height, plant population and stalk diameter (Mourtzinis et al., 2013). The R^2 was 0.77 at R1 growth stage. The model also had a high correlation with final stover biomass (R^2 = 0.85). These studies not only confirmed the importance of stalk diameter as an indicator of corn plant health, but also proved the possibility of predicting corn potential yield in season using stalk diameters.

Not much research has been conducted so far focused on automatic, non-contact, on-the-go corn stalk diameter measurement. Most of the relevant research was in corn population estimation and plant morphological characteristics sensing. The available technologies included capacitance sensing and optical sensing. Since the capacitance of a plant stalk could change with its moisture content, the capacitance of a stalk could be indirectly correlated to its diameter or directly related to its growth status. A single-sided capacitive sensor was patented to detect the existence of a plant stalk using the moisture change in the sensor's proximity (Nichols, 2000). The sensor was mounted on the row separator of a combine harvester. This sensor had little interference from leaves and other debris in the field due to their low moisture content. Li et al. (2009) designed a single-sided biomass proximity sensor based on the capacitance measurement which was installed on the row dividers of the combine harvester. The sensor had an accuracy of more than 95% on corn plant population estimate. However, both of these capacitance-based designs required a very close proximity (less than 3 cm) between the sensors and the detected plant stalk; otherwise the accuracy could not been assured and the interference would be induced in. In practical fertilizer applications, a close proximity to the plants from the sensing mechanism is often difficult to achieve.

Okiror (2012) conducted a preliminary study on the microwave dielectric property of corn stalks which could be related with the stalk characteristics such as stalk diameter. Corn plants harvested in field were brought to the lab to take their dielectric property measurements using a bistatic (two-sided) microwave scattering system. Comparison was conducted between the system measured and manually measured stalk diameter data collected from V8 to VT growth stages during which the stalk diameters varied from about 1 cm to 3 cm. The R2 of this comparison was 0.66 when the data was taken with the leaves on the stalks; and the R2 was 0.84 when the leaves were stripped off. They did not give the statistics of these models on stalks with smaller diameter difference, such as stalks at the same growth stage.

A photoelectric sensor with an emitter and receiver pair was designed and installed on a combine harvester to measure corn plant stalk diameter and spacing (Lobdell and Hummel, 2001). The plant stalks interrupted the light beam of the sensor which was mounted on the corn header during harvesting. The interrupting and non-interrupting time were multiplied by the vehicle speed obtained from a radar transducer to calculate stalk diameters and spacing. Too small or large estimated diameters were eliminated by software filtering. A problem they mentioned was plant leaves being falsely counted as plant stalks. A 'flexible finger' was tried to reduce the false counts but resulting with no significant improvement. An air-jet system was proved as a more effective way to remove leaves out of sensor's field of view. They reported an error of 9.5% in the diameter measurement. The inaccuracy of vehicle travel speed measurement as well as the inaccuracy and delay in the interrupting time measurement were discussed as major error sources. If the stalk diameter could be measured in a way which less relies on the vehicle speed measurement, the measurement error could be reduced. Similarly, Rascon (2012) used a 1D laser pointer horizontally shooting at the bottom sections of corn plant stalks to measure the stalk diameters based on the time-of-flight principle. The sensor was mounted on a cart moving along the crop row and its locations were measured by a shaft encoder. The displacements when the laser beam was blocked by stalks were used as the estimated stalk diameters. The unevenness of the field surface caused the error between the measured displacements and actual stalk diameters. Leaves and other debris also caused false positive errors. Both studies indicated that a method independent on the measurement of sensor displacement or vehicle speed may improve the accuracy in stalk diameter measurement.

Many studies can be found on using machine vision approaches to measure plant morphological characteristics. Laser scanning technique was used for tree foliage density and wheat stand density estimation by calculating variations in laser penetration depths (Wei and Salyani 2004, 2005; Saeys et al. 2008; Chen et al., 2012). Color imaging has been explored in several studies

for corn plant counting and spacing measurement in the past few decades. Shrestha and Steward (2003, 2005) developed and tested a machine vision based corn plant population sensing system. Algorithms were developed for color image sequencing, segmentation and plant recognition in order to count corn plants and to estimate plant location and spacing. The root-mean-square errors (RMSEs) in population estimates were in the range of 5–6 % compared with manual counts. Tang and Tian (2008a, b) developed a real-time crop row image reconstruction and plant identification system for automatically measuring the spacing of emerged corn plants. They achieved an overall RMSE of 1.7 cm and an R^2 of 0.96. All of these studies targeted at early growth stage corn plants prior to the canopy closure.

Nakarmi and Tang (2010) developed a system to measure corn plant within-row spacing at growth stages V3 to V6 using side-view images from a 3D range camera. Background was eliminated by distance thresholding. The skeletons of the vegetation objects corresponding to the bottom section of the stalks were used for stalk detection. Images were stitched together to re-construct a field map based on the recognized stalk locations and encoder reading of each image. They reported a 2.2% plant mis-identification error and RSME of 1.7 cm in within-row spacing measurements. They suggested a potential improvement on the system accuracy by viewing plants from various perspectives.

Similar to the corn stalk diameter measurement, machine vision technology was applied on the stem diameter measurement of pine seedlings (Rigney and Kranzler, 1988). Grey scale images were captured for individual pine seedling with a diameter ranged from 2.3 to 6.0 mm laid on a conveyor belt using a high resolution industrial camera. The location of the root collar area of a seedling was identified by detecting rows with less black-to-white transitions in an image. The average width between the transitions along those rows was used to estimate the stem diameter of a seedling. A mean coefficient of variation of 7.6% was achieved for 100 seedlings with 20

replications. This study demonstrated the feasibility of using machine vision to accurately measure plant stem diameters.

One thing needs to be pointed out is that a corn plant is bilateral symmetric and the section of its stalk is an ellipse. The orientation of the ellipse is random when a shoot germinates but it changes during its growth to optimize the sunlight interception. This random orientation makes the corn plant stalk diameter measurement even more complicated. In this study, approaches were proposed and tested to profile the stalk widths from various angles of view to approximate the actual ellipse shape.

A corn plant location and within-row spacing measurement system was developed based on LiDAR technology (Shi, et al., 2013). Line scan profiles were collected continuously from a LiDAR sensor mounted on a moving cart scanning horizontally at the bottom sections of plants. Each stalk appeared as a cluster of data points in the laser scan and would keep showing up in several successive scans from various angles as the cart moving down the row. However, due to the relative large beam divergence (0.29°) and the edge effect of the sensor's laser beam, it was concluded that the LiDAR sensor itself was not sufficient to accurately measure the stalk diameters. In this study, two sensing approaches were proposed – the LiDAR-and-RGB Approach based on a combination of a LiDAR sensor and a common RGB camera and the 3D Range Imaging Approach based on a 3D range camera – for the corn stalk diameter measurement. The overall objective was to develop systems for the two proposed sensing approaches, and to evaluate their feasibilities and performances through field experiments. The specific objectives were:

- To develop data acquisition systems for each proposed approach and collect field data for corn plant stalk diameter measurement;

- To develop data and image processing algorithms for corn plant stalk diameter measurement; and
- To evaluate system performance at the V12 mid-growth stage of the corn plants.

Materials and Methods

System Setup

Considering with the shortcomings of using a LiDAR sensor alone in the width measurement for small objects, two approaches were proposed: the LiDAR-and-RGB Approach was based on the combination of the LiDAR sensing and the RGB imaging and the 3D Range Imaging Approach was based on a 3D range imaging technology. Both approaches targeted at the lower section of corn plant stalks where less leaves existed. The LiDAR-and-RGB Approach quickly detected the existence of potential stalk objects in a laser scan using their range information; then verified those potential stalk objects and measured their width in a corresponding RGB image; and finally measured the stalk diameters using their widths in the RGB images and their distance away from the sensors. The 3D Range Imaging Approach completed the stalk recognition, the distance measurement and the stalk diameter measurement using the data from the 3D range camera.

All the sensors were mounted on a modified golf cart. The LiDAR sensor (LMS291, SICK AG, Waldkirch, Germany) was mounted on a rail so that its height could be adjusted (Fig. 22). With a NIR light source at 905 nm, this sensor measured the distance based on the time-of-flight principle. Instead of a fixed light beam, a rotating mirror inside the sensor deflected the light source to sweep in a circular pattern so that a fan-shape scan was formed out of the sensor (SICK, 2006). The LiDAR sensor was configured with a 100° field of view and a 0.25° resolution (Fig. 23a). It was mounted with a 30° downward angle to keep the scanning level at the lower section

(a) (b)

Fig. 22 Data acquisition system: (a) modified golf cart; (b) close view of sensors mounted on cart.

of the plant stalks (about 12 cm above the plant roots) (Fig. 23b). Each scan took 53.28 ms to be completed. With an average speed of 0.447 m/s, the cart could move about 2.4 cm during this period. This offset was ignored in this study. The sensor was configured with its highest communication rate of 500 kbps to make sure a fast data acquisition. Details can be found in Shi et al., 2013.

A webcam (LifeCam, Microsoft, WA) was mounted directly under the LiDAR sensor in a way that the mid-point of the LiDAR sensor's field of view almost overlapped with the horizontal centers of the webcam's field of view (Fig. 23a). However, the spatial projection of the locations of the two sensors' light sources had an offset due to the way they were mounted (Fig. 23b). The offset needed to be precisely determined in order to match the laser scans and the RGB images. This webcam was a common low-cost RGB camera with a 73° diagonal field of view (Microsoft Corporation, 2012) and a 62° calibrated horizontal field of view. It was configured with a resolution of 360 × 640 (vertical × horizontal) with a bitmap picture format in order to have a minimum picture storage size while keeping enough color and shape information. The webcam's frame rate could be up to 30 fps depending on the operating system. Because it was viewing

horizontal to the ground, not only the plants in the adjacent row but also those in the further rows were all inside its field of view. The webcam's focal length was adjusted by software so that only the adjacent plant row was focused in the images.

A 3D range camera (SR4000, MESA Imaging AG, Zuerich, Switzerland) was mounted next to the LiDAR-webcam combination (Fig. 22, 23). This sensor collected grey-scale images and distance images simultaneously. With a NIR light source at 850 nm, the distance measurement was based on the phase shift principle (MESA Imaging, 2011). The resolution of this sensor was 144 × 176 (vertical × horizontal) with a corresponding field of view of 34° × 43°. The major problem of this sensor was its relative less ability of background light resistance. Though its integration time could be adjusted, the detection pixels were easily saturated if strong sunlight was received. To address this problem, a shade was added on the top of the cart and the sensor's integration time was set to automatically adjust to an optimal number.

A LabVIEW® (National Instruments Co., Austin, Texas, USA) program was developed to control the three sensors simultaneously. Each laser scan or image was location-stamped with its corresponding encoder reading, time-stamped with the system time of the laptop and saved on the laptop for offline processing.

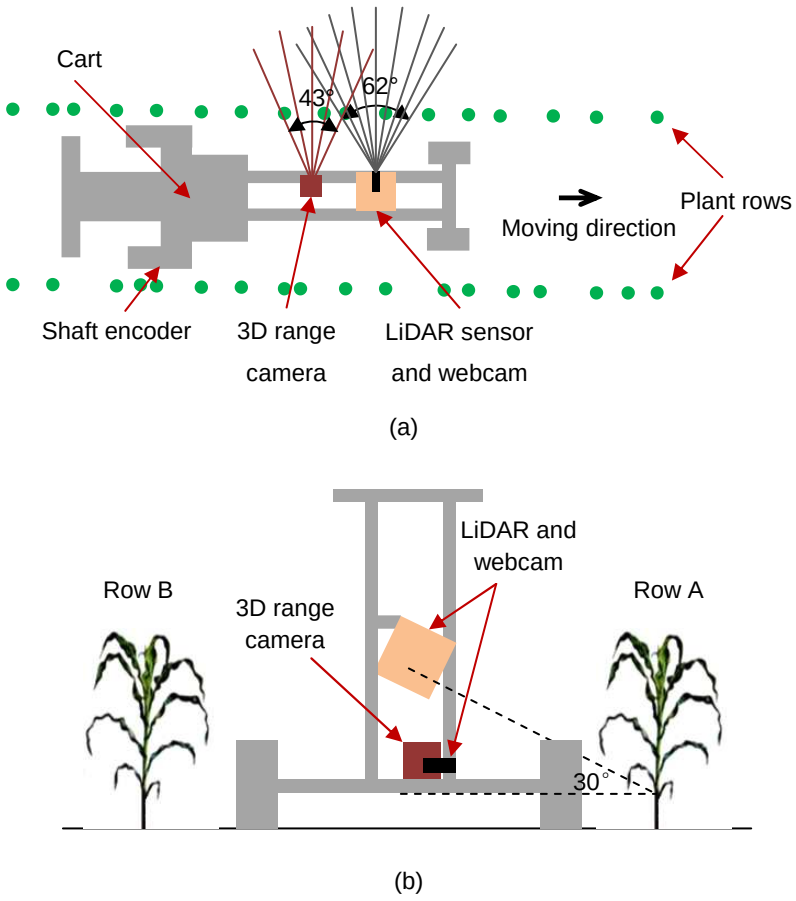


Fig. 23 Schemetic diagram of the corn plant diameter measurement system with two machine vision approaches (not to scale): (a) Top view and (b) side view.

Field Experiment Setup

Field tests were conducted in June, 2013, in a tilled corn field near Lake Carl Blackwell in Stillwater, OK. Four plots were planted with two populations and three nitrogen pre-application rates (Table 10). Part of the second row in each plot was tagged. Originally each tagged section contained 25 plants; however, due to damages, only 24 plants were survived in plot 1 and 3 which resulted with a total of 98 plants. Data was collected using the two approaches at V12 growth stage. Manual measurements were taken for plant location/spacing using a tape ruler and the stalk

diameter along the row direction using a caliper for each sampled plant. Fig. 24 is the box plot of the manually measured ground truth stalk diameters for the four plots.

The horizontal distance between the LiDAR sensor and the corn row varied from 30 to 46 cm when the cart was pushed along a row due to the deviation of the cart from the center line between rows (Fig. 25). The mounting height of the LiDAR sensor was adjusted so that the scan line on the plant stalks was kept between 7.6 and 16.8 cm above the plant roots.

Table 10 Plot treatment design.

	Planting Date	Population (plants/ha)	Nitrogen rate (kg/ha)	Number of samples	Growth stage
Plot 1	Mar 20th	49,419	0	24	V12
Plot 2	Mar 20th	49,419	89.6	25	V12
Plot 3	Mar 20th	49,419	179.2	24	V12
Plot 4	Mar 20th	64,245	89.6	25	V12

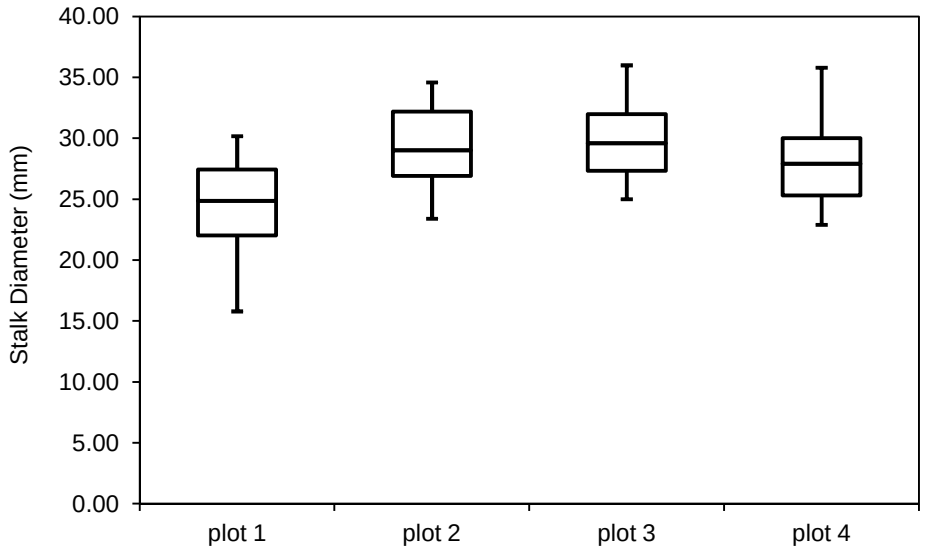


Fig. 24 Box plot of manually measured stalk diameters of four testing plots.

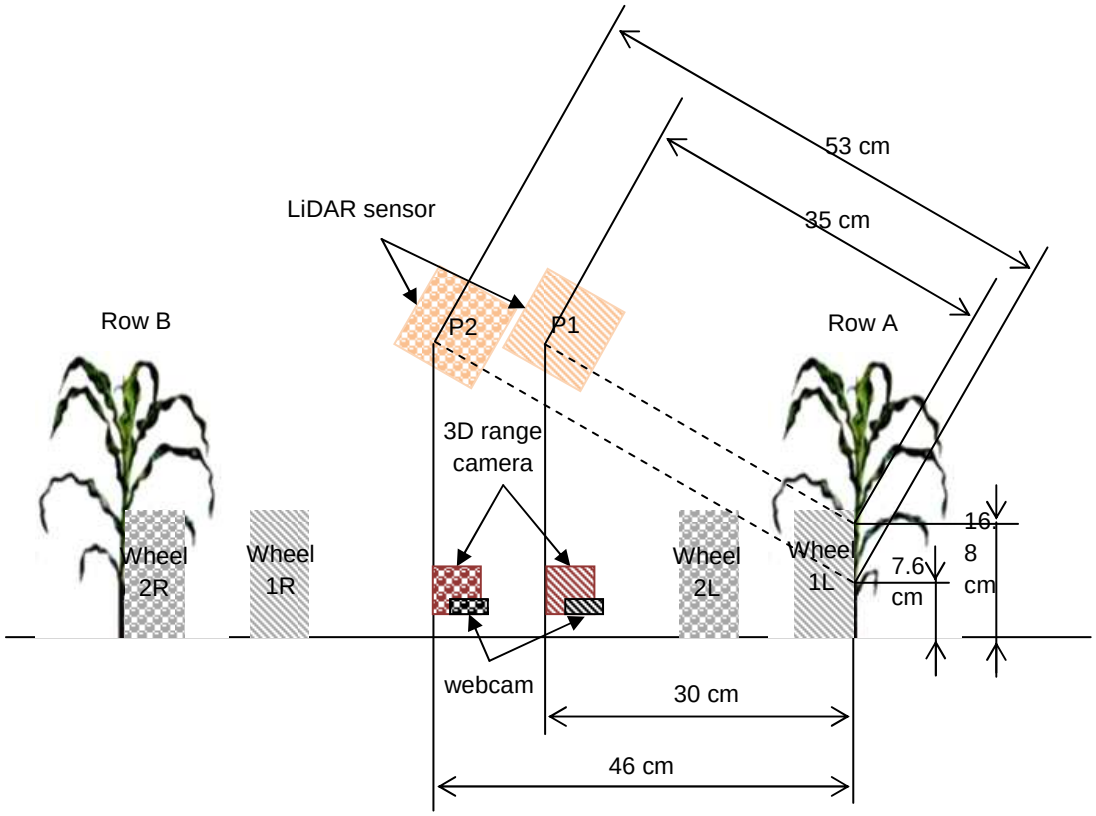


Fig. 25 Illustration of the cart's deviation between rows (not to scale): Case 1 (see P1) was when the cart was the closest to plant row A and Case 2 (see P2) was when the cart was the furthest to plant row A. Designations 1L and 2L indicated the positions of left wheel, and 1R and 2R indicated the positions of right wheel in these two cases, respectively.

Sensor Calibrations and Data Processing Algorithms

LiDAR-and-RGB Approach

Sensor Calibrations

1. Distortion correction of the webcam

A chess-board with 11 × 11 effective squared grids, each 30 × 30 mm^2, was made to calibrate the webcam. The result showed the distortion of this webcam was trivial enough to be ignored (Fig. 47 and 48 in Appendix E).

2. Matching two sensors' data

The most important procedure of the sensor calibration in the LiDAR-and-RGB Approach was to match the data points in a laser scan with the pixels in a corresponding RGB image. The webcam's datasheet only gave a diagonal field of view of 73°. To calibrate its horizontal field of view, a paper board marked with grids was placed in front of both sensors at known distances (Fig. 26). Fig. 27 shows the calibration process. O_{laser} and O_{rgb} in the figure were where the light sources of the LiDAR and the webcam located, respectively, with a distance of Δd. $A1O_{rgb}B1$ comprised the webcam's field of view, α; $RO_{laser}S$ comprised the LiDAR sensor's field of view which was 100°. The shaded area was their common field of view. $O_{laser}M'$ was the mid light beam of the LiDAR sensor while $O_{rgb}M1$ was the horizontal center of the webcam's field of view. The offset between $O_{laser}M'$ and $O_{rgb}M1$ formed an angle, $\Delta\alpha$. Parameters unknown and needed to be calibrated were: α, Δd and $\Delta\alpha$.

Fig. 26 In order to obtain the webcam's horizontal view angle, a paper board marked with grids was placed at three distances away from the sensors.

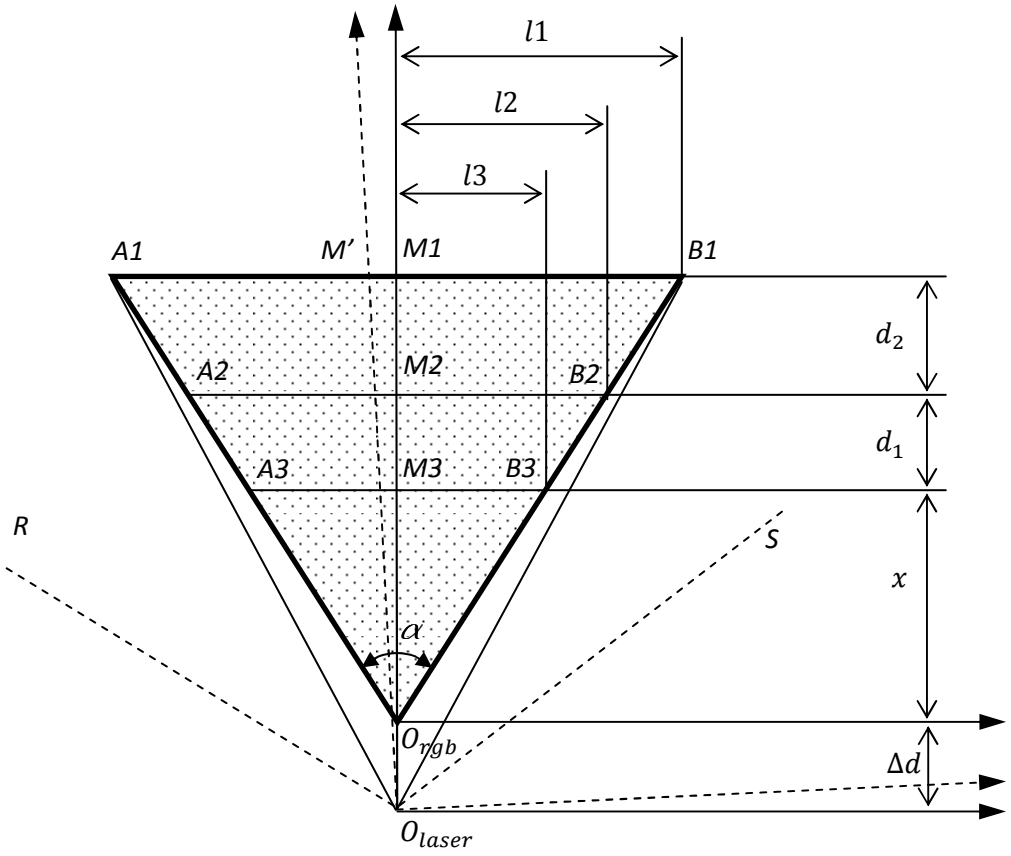


Fig. 27 Top view geometry of the LiDAR and webcam's relative locations and their coordinates. O_{laser} and O_{rgb} were the light sources of the LiDAR and the webcam, respectively. $O_{rgb}A1$ and $O_{rgb}B1$ comprised the webcam's field of view. $O_{laser}R$ *and* $O_{laser}S$ *comprised the LiDAR sensor's field of view.* $O_{laser}M'$ was the mid light beam of the LiDAR sensor while $O_{rgb}M1$ was the horizontal center of the webcam's field of view. The shaded area was their common field of view.

a. Decide the distance, Δd, between two sensors' light sources and the webcam' horizontal field of view, α

A paper board was first placed at location $A1B1$ (Fig. 27), then was moved a distance d_1 = 6 cm closer to the location, $A2B2$, and was finally moved a distance d_2 = 5 cm closer to the location $A3B3$. At each location, the halves of the horizontal field of view at those three distances, l_1, l_2 and l_3, were obtained by counting the grids on the board. The width of a minor grid was 1 cm. This resulted l_1 = 25 cm, l_2 = 21 cm and l_3 = 18 cm. Then three different x values were

calculated from Eq. 14, 15, and 16. Three x values were averaged resulting with $x \approx 28$ cm. The reason to over define x by three equations was to reduce the error in measurement. Knowing x and $l1$, $l2$, $l3$, the webcam' horizontal field of view α was calculated to be 62° (Eq. 17).

$$\frac{x + d_2}{x + d_1 + d_2} = \frac{l_2}{l_1} \quad \text{Eq. 14}$$

$$\frac{x}{x + d_2} = \frac{l_3}{l_2} \quad \text{Eq. 15}$$

$$\frac{x}{x + d_1 + d_2} = \frac{l_3}{l_1} \quad \text{Eq. 16}$$

$$\tan\left(\frac{\alpha}{2}\right) = \frac{l_1}{x + d_1 + d_2} \quad \text{Eq. 17}$$

Knowing the LiDAR sensor was mounted with a 30° downward angle, the distance data measured by the LiDAR sensor was converted to the horizontal distance between the sensor and the object. The LiDAR sensor's coordinates were then defined as its converted horizontal coordinates. At the same time, the LiDAR sensor data was taken to measure the distance of the paper board at each of those three locations. The difference between the distances measured by the LiDAR sensor and the corresponding distances calculated using the webcam's data, Δd, was 5.5 cm. The mounting of the webcam resulted in a small incline angle of the objects in the RGB images. This angle was measured in the images to be 0.85° and used to correct the stalk diameter.

b. Estimate $\Delta\alpha$

As shown in Fig. 27, the paper board was placed at a known distance away from the LiDAR sensor and the webcam. The area out of the webcam's field of view was then cut off so that only the area as wide as the webcam's field of view was kept. The LiDAR sensor's data was collected then to see which light beams corresponded to the edges of the paper board. For example, when

the paper board was placed at location in Fig. 27, the laser light beams #88 and #305 corresponded to the edges of the paper board. Instead of laser light beam #200, laser light beam #196 corresponded to the horizontal center of the webcam's field of view. the center offset $\Delta\alpha$ of the two sensors was known by this way. Several replications were conducted and an averaged final result was $\Delta\alpha = 1.25°$. So far, the geometry of locations and field of view for the LiDAR sensor and the webcam was known.

c. Develop dynamic look-up-table for matching

For a random data point *P* in the LiDAR sensor's coordinates $x_{laser}o_{laser}y_{laser}$, angle β and θ in Fig. 28 were known by knowing $\Delta\alpha$, d_{laser} and Δd. Comparing θ with the webcam's total field of view, α, the column of pixels corresponding to *P* was identified in the RGB image.

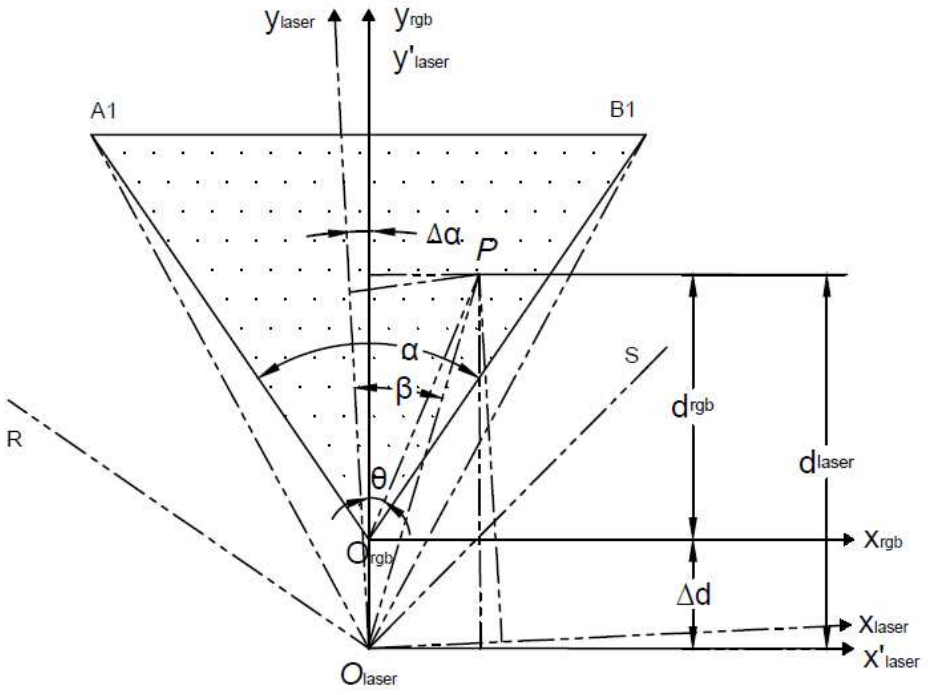


Fig. 28 Matching data point *P* from the LiDAR sensor's coordinates $x_{laser}o_{laser}y_{laser}$ to the webcam's coordinates $x_{rgb}o_{rgb}y_{rgb}$.

3. Actual object width calculation

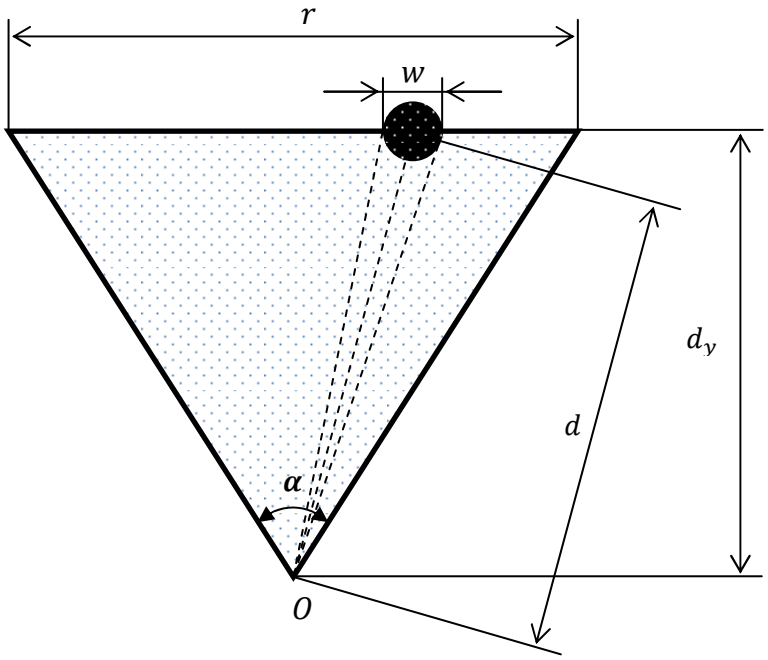


Fig. 29 Illustration of diameter calculation based on object's width in an image. The black round area represents the section of the object. O is the sensor location, α is the sensor's view angle, r is the sensor's field of view at distance d_y, w is the object diameter.

As shown in Fig. 29, the webcam's field of view r at a random distance d_y could be calculated (Eq. 18). The horizontal resolution of the webcam was 640 pixels. Assume the object had a width of w_{image} pixels in the image. The proportion of the object width to the image width in terms of pixel numbers should be the same as the proportion of the actual object width to the sensor's field view (Eq. 19). Substitute r with Eq. 18, we could get the actual object size, w (Eq. 20).

$$r = 2d_y \cdot \tan\left(\frac{\alpha}{2}\right) \quad \text{Eq. 18}$$

$$\frac{w_{image}}{640} = \frac{w}{r} \quad \text{Eq. 19}$$

$$w = \frac{w_{image} \cdot r}{640}$$

$$w = \left[\frac{2 \cdot \tan\left(\frac{\alpha}{2}\right)}{640}\right] \cdot \left(w_{image} \cdot d\right) \quad \text{Eq. 20}$$

Data/Image Processing Algorithms

Step 1. Laser scan and RGB image matching using time stamps

In this study, RGB images were collected at a lower frequency than that for the laser scans. For each RGB image, a laser scan with a time stamp closest to the time stamp of the RGB image was found to form a laser-RGB pair.

Step 2. Clustering in laser scan

A density-based clustering procedure (Shi, et al., 2013) was implemented for the paired laser scan. Laser data points close enough to each other were grouped together. Unspecified number of clusters was formed until all the data points being visited. Potential stalk clusters were further screened according to their width along the tangential direction centered the coordinate origin. Only those with widths between 1 and 5 cm were kept. As shown in Fig. 30a, four clusters were found and marked in colors. The area compassed by the red lines representing the field of view of the webcam which means the clusters in pink and cyan were in the paired RGB image (Fig. 30b).

Step 3. Stalk recognition in the paired RGB image

The vegetation area in the paired RGB image was segmented based on the red (R), green (G) and blue (B) values of each pixel. Generally four types of pixels could be found in an image: the stalk pixels, the leaf pixels, the soil pixels and the bright pixels due to the over exposure under sunlight. Stalk pixels and leaf pixels usually had larger G values than their R values; while soil pixels usually had larger R values than their G values. Most of the RGB images collected showed that the over exposed pixels usually had all the R, G and B values greater than 200 while the rest

of the pixels had values less than 100 with a uint8 data type. Leaf pixels could have similar or slightly larger B values than G values while the G components of the stalk pixels were usually the largest. Based on these observations, a pixel was defined as a stalk pixel if the rule shown in Eq. 21 was met. Fig. 30c shows the binary image after the vegetation segmentation for Fig. 30b. A threshold was set as the minimum number of pixels that a stalk object should have. Connected areas with sizes greater than this threshold were kept (Fig. 30d) and small holes in these areas were filled (Fig. 30e). Finally, the sum of the pixel values along each column of Fig. 30e was obtained (Fig. 30f). The center location of each detected plateau with readings of 360 on the vertical axis was calculated as the detected location of a potential stalk in the RGB image. Plateaus with too small or large width were eliminated. Also if a plateau was close to or touched the edges of an image which indicated the stalk profile might not be complete, it was also ignored.

$$(Max(R,G,B) < 200) \;\; \& \;\; (R < G) \;\; \& \;\; (B < G) \qquad \text{Eq. 21}$$

Step 4. Cross-verification of detected stalks

In each laser-RGB pair, a stalk had two profiles – one was from the laser; the other was from the RGB image. The locations of a stalk measured from the two sensors were usually different due to the delay during data acquisition. These two measured locations also needed to be cross-verified in order to eliminate false detection. The corresponding location in the RGB image of a detected cluster in the laser scan was estimated by the dynamic matching strategy (Fig. 28). Then it was checked to see if a detected stalk in the RGB image could be found within a certain buffered area (set as ±70 pixels by trial-and-error in this study) of this estimated location. A stalk measurement was valid only when it was verified in both the laser scan and the RGB image.

Step 5. Diameter calculation

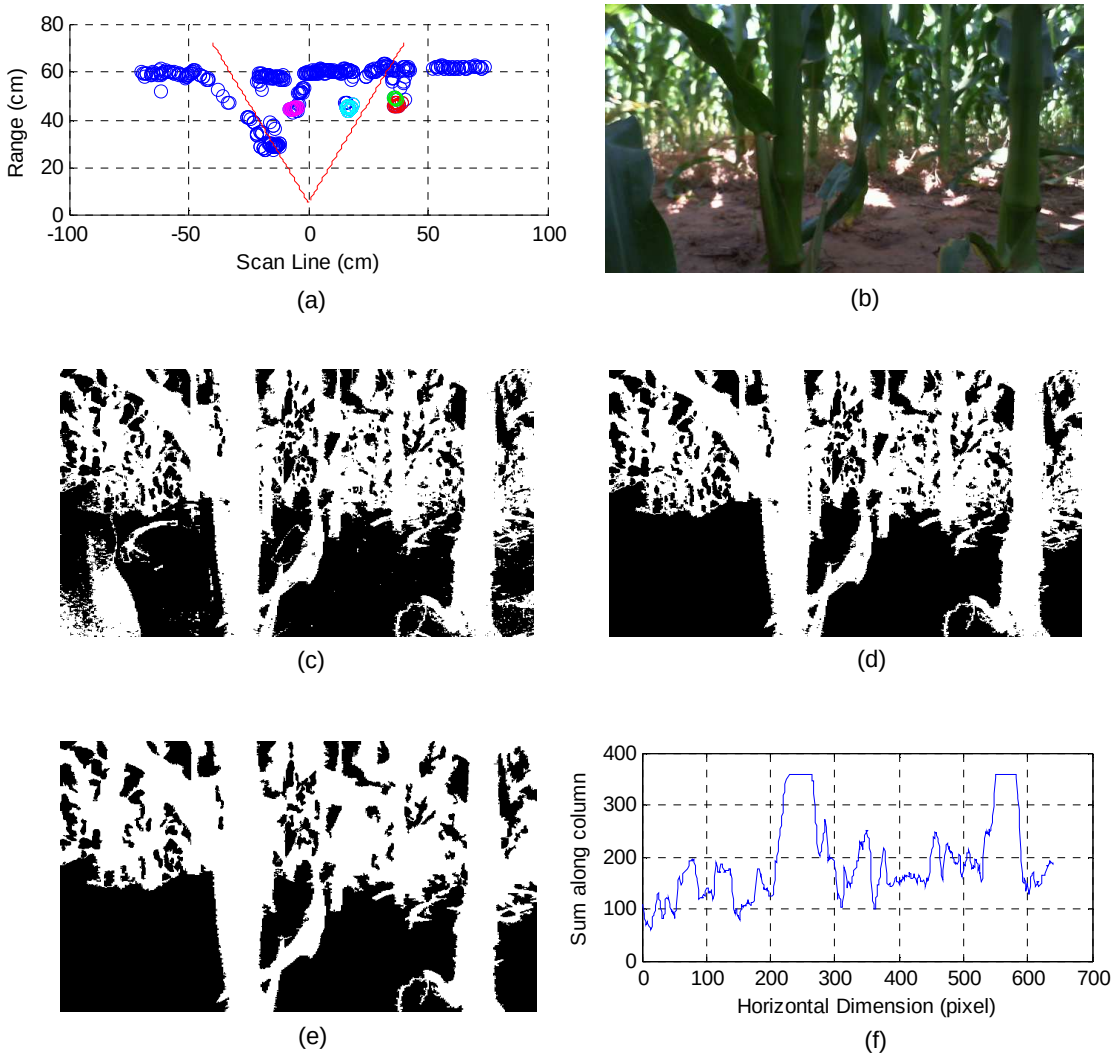


Fig. 30 The data/image processing for LiDAR-and-RGB Approach: (a) and (b) show a laser-RGB pair. In (a), blue '○'s represent laser data points; color '○'s represent recognized clusters; the area compassed by the red lines represents the webcam's field of view. (c) Result after the vegetation area segmentation. (d) Connected area(s) with a size greater than 10800 pixels. (e) Result after filling in holes smaller than 100 pixels big. (f) Sum of pixel values along the columns of (e).

The diameter of a detected stalk was calculated based on its width (number of pixels) in the RGB image and its distance measured in laser scan (Eq. 20).

Step 6. Registration between laser scans w/ counter variables

After obtaining the diameters and locations of detected stalks in each laser-RGB pair, the same stalk appeared in different laser-RGB pairs were registered based on their encoder readings (Shi, et al., 2013).

3D Range Imaging Approach

Sensor Calibration

1. 3D range camera distortion correction

In order to correct the sever barrel distortion of the 3D range camera used in this study, a calibration was conducted to obtain the rectification model. The distortion was successfully corrected for both amplitude and distance images (Fig. 31).

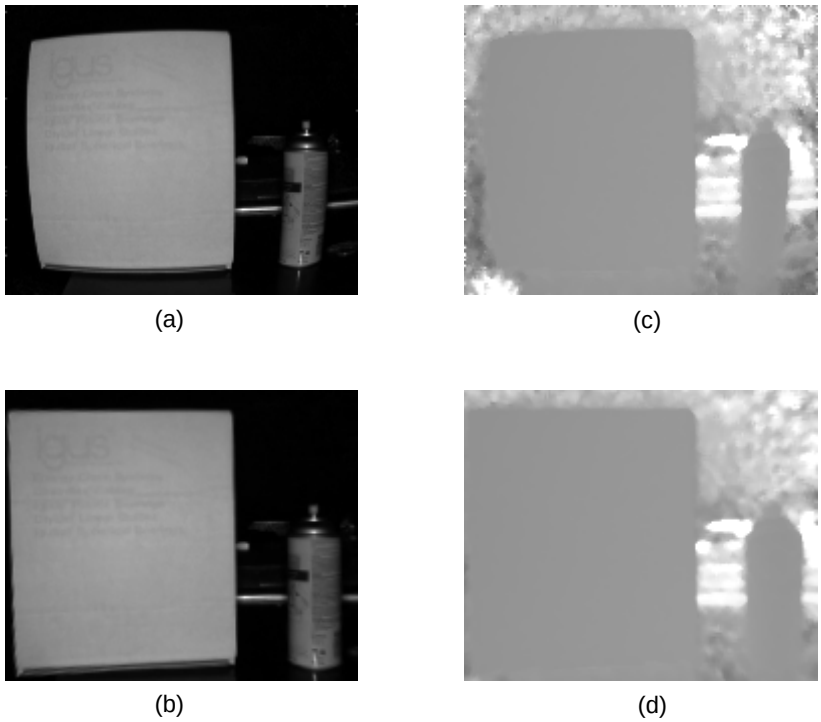


Fig. 31 Scene with a box and a bottle: (a) distorted amplitude image, (b) undistorted amplitude image, (c) distorted distance image, (d) undistorted distance image.

2. Actual object size calculation

The width calculation in a distance image was similar with that in an RGB image. In Eq. 22, w was the calculated width of an object with the same unit as d; α was the view angle of the 3D range camera which was 43°; w_{image} was the width of the object in the distance image represented by number of pixels; and d was the distance the object away from the camera. The only different was the horizontal resolution of the distance image was 176 pixels.

$$w = \left[\frac{2 \cdot \tan\left(\frac{\alpha}{2}\right)}{176}\right] \cdot \left(w_{image} \cdot d\right) \quad \text{Eq. 22}$$

Image Processing Algorithm

Step 1. Stalk feature extraction in distance image

The potential stalk pixels in a distance image was segmented based on the pixels' distance readings. A binary image was resulted from the distance image by eliminating any pixel with the distance reading out of the range of 25 to 51 cm (Fig. 32c). A threshold was set as the minimum number of pixels that a stalk object should have in a distance image (1500 pixels in this study). Connected areas with sizes greater than this threshold were kept and small holes (less than 200 pixels) in these areas were filled (Fig. 32d). Similar to the LiDAR-and-RGB Approach, the sum of the pixel values along each column of Fig. 31d was obtained (Fig. 32e).

Step 2. Diameter calculation

The plateaus with readings of 144 on the vertical axis corresponded to the detected stalk locations and widths w_{image} in the distance images. Detected plateaus with too small or large width (less than 10 pixels or larger than 30 pixels) were eliminated. Also if a detected plateau was close to or

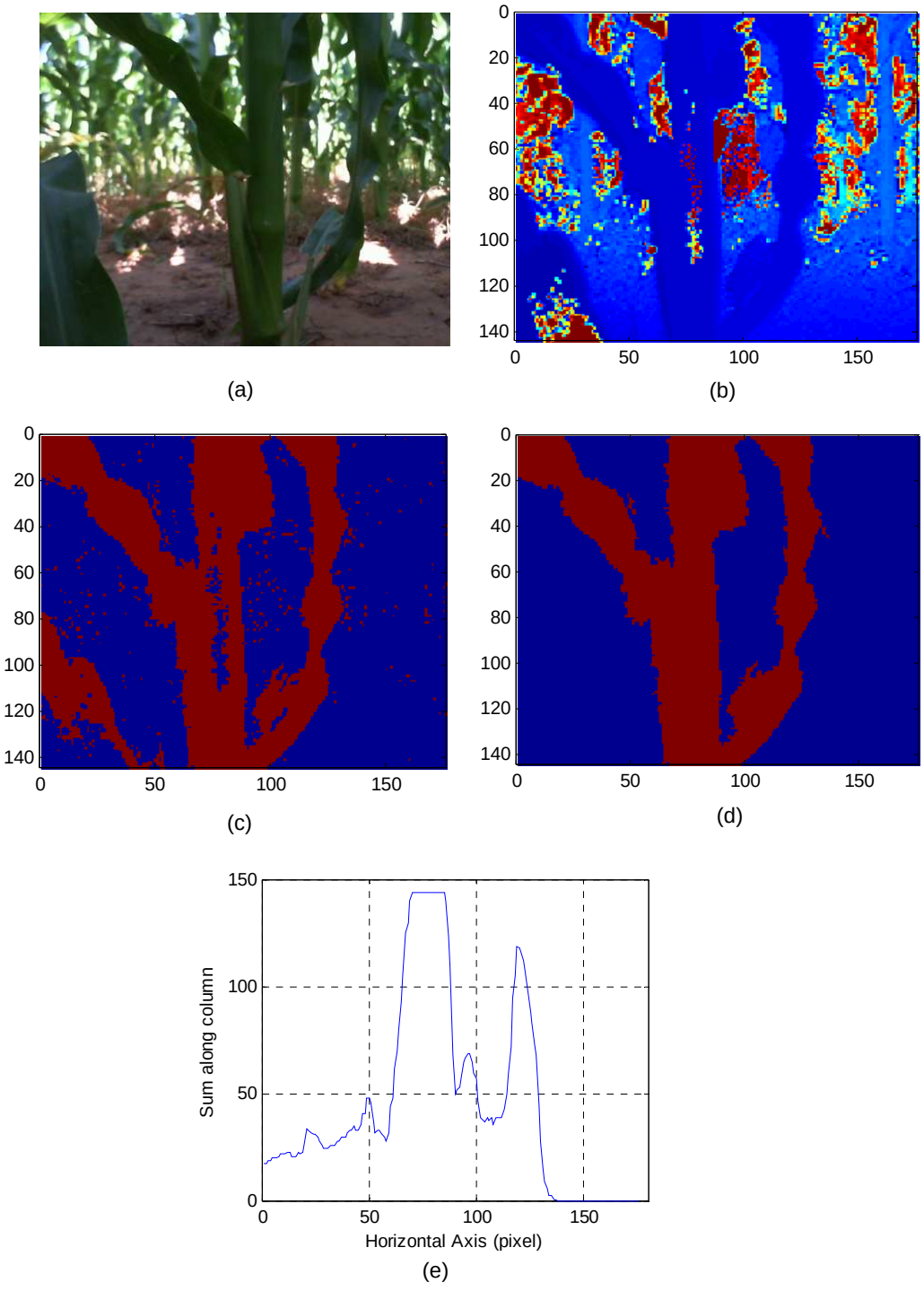


Fig. 32 The image processing for 3D Range Imaging Approach: (a) actual scene; (b) distance image after distortion correction (c) after thresholding; (d) connected area(s) with a size greater than 1500 pixels and after filling in holes smaller than 200 pixels big; (e) Sum of pixel values along the columns of (d).

touched the edges of an image which indicated the stalk profile might be incomplete, it was also

ignored. The center location of each detected plateau was calculated as the detected location of a potential stalk in the distance image. The average of the distance readings of all pixels consisted of in a recognized stalk was calculated as d in Eq. 22 for the diameter estimation.

Step 3. Registration between distance images

After obtaining the diameters and locations of detected stalks in a distance image, the location of the recognized stalk in the coordinates of individual image was converted to the ground coordinates based on the encoder reading of that image and the location of the stalk in that image. Then the same stalk appeared in different distance images were registered based on their encoder readings (Shi, et al., 2013).

Performance Evaluation

System performance was evaluated in terms of plant counting error, stalk diameter measurement error as well as the within-row spacing measurement error. As for the plant counting accuracy, three errors were defined – the false negative error (FNEr) (Eq. 23), the false positive error (FPEr) (Eq. 24) and the total error (TEr) (Eq. 25). If no plant was identified within ±10 cm of the ground truth location of an actual plant, there was a false negative count for that actual plant. Similarly, if no actual plant was located within ±10 cm of the location of an identified plant, this resulted in a false positive count. Only the recognized plant closest to (and within ±10 cm of) the location of an actual plant was a valid count; multiple counts of an actual plant were treated as false positive counts.

$$FNEr = \frac{Missing\ Count}{Ground\ Truth\ Count} \times 100\ \% \quad \text{Eq. 23}$$

$$FPEr = \frac{Adding\ Count}{Ground\ Truth\ Count} \times 100\ \% \quad \text{Eq. 24}$$

$$TEr = FNEr + FPEr \quad \text{Eq. 25}$$

As for the accuracy of stalk diameter measurement, the root mean square error (RMSE) was calculated for each trial. As for the accuracy of within-row spacing estimates, the RMSE was calculated according to the system measured and the ground truth data. In order to keep a consistency, locations corresponding to the false negative counts were eliminated from the ground truth data while the locations corresponding to the false positive counts were eliminated from the system measured data.

Results and Discussion

Plant Counting Errors

Both approaches achieved similar results on plant counting accuracy. The LiDAR-and-RGB Approach resulted with a TEr of 4.5% including a 3.2% FNEr and a 1.3% FPEr; the 3D Range Imaging Approach resulted with a TEr of 3.4% including a 1.5% FNEr and a 1.9% FPEr (Table 11). The LiDAR-and-RGB Approach had a higher FNEr indicating more missing counts. The results of Row 1 using this approach were excluded from the overall results due to the failure in vegetation area segmentation during the image processing. The image processing algorithm differentiated vegetation pixels from soil pixels based on their different RGB compositions. However, the leaf edges of plants in Row 1 were dark brown caused by short of nutrients and that color was closer to the soil than the green vegetation area which resulted with the missing identification of the plants in Row 1(Fig. 33). Besides, the major reason caused the missing identification with the LiDAR-and-RGB Approach was the out of synchronization between the two sensors' data. The data processing algorithm was developed so that a plant was identified only if it was detected at a same location in both sensors' fields of view. In some cases, the delay between a paired laser scan and RGB image was too much due to the malfunction during data

acquisition to make the two sensors' data match with each other which resulted with missing identifications (Fig. 34).

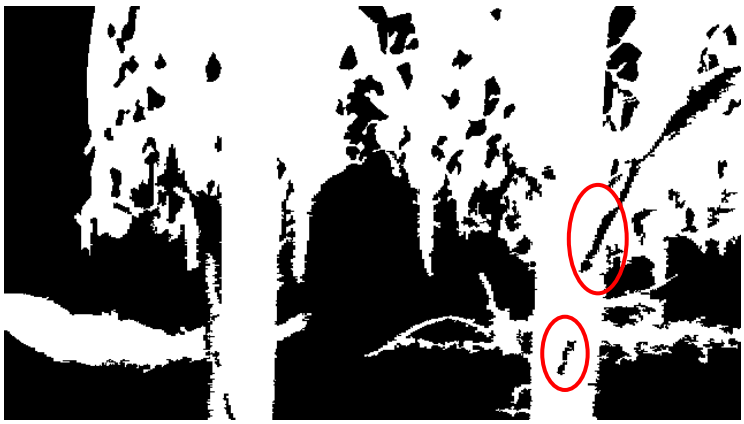
These two problems did not bother the data collected with the 3D Range Imaging Approach. The 3D range camera collected intensity and distance readings of each pixel simultaneously, and its data processing algorithm implemented the vegetation area segmentation according to distance readings rather than RGB compositions. Nevertheless, the over-exposure during the data acquisition affected the results of both approaches (Fig. 35 and 36) and it was even severe for the 3D Range Imaging Approach. The major reason caused the missing identification for the 3D Range Imaging Approach was because the sensor's imaging pixels were saturated under over-exposure or the sensor was not able to adjust its integration time fast enough to response the lighting change (Fig. 36).

Table 11 Errors of plant counting, spacing estimates and stalk diameter estimates compared to ground truth data for two approaches.

	Counting Error						RMSE of spacing estimate (cm)		RMSE of diameter estimate (mm)	
	FNEr (%)		FPEr (%)		TEr (%)					
	Approach		Approach		Approach		Approach		Approach	
	1	2	1	2	1	2	1	2	1	2
Row 1	na*	0	na	0	na	0	na	1.6	na	2.9
Row 2	1.3	0	0	4.0	1.3	4.0	1.0	0.9	3.5	4.0
Row 3	4.2	4.6	0	1.7	4.2	6.3	0.9	1.2	4.6	4.4
Row 4	4.0	0	4.0	0	8.0	0	1.3	1.0	4.2	4.2
Mean	3.2	1.5	1.3	1.9	4.5	3.4	1.1	1.2	4.1	3.9

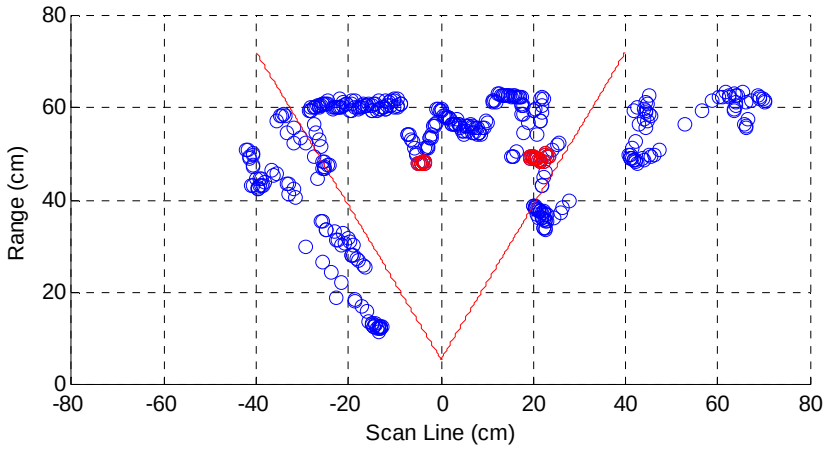
* na means data were eliminated due to bad quality

(a)



(b)

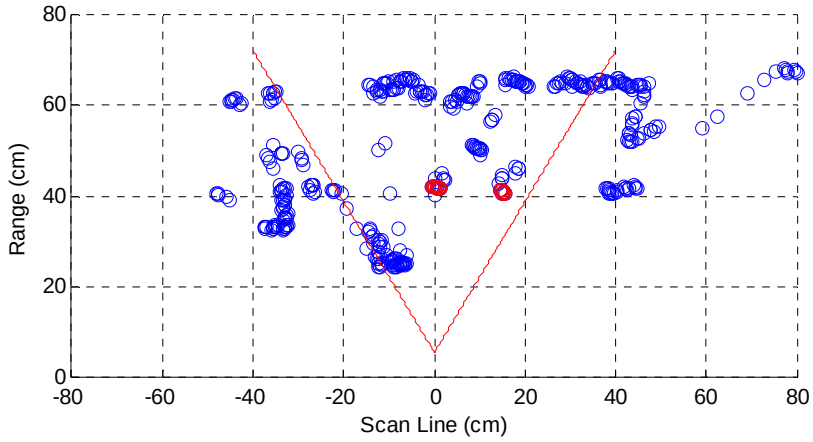
Fig. 33 Example of the yellow-edge problem in Row 1 data: (a) an RGB image and (b) after the vegetation segmentation and hole filling processing. The yellow-edged leaves and sheaths caused FNEr.



(a)

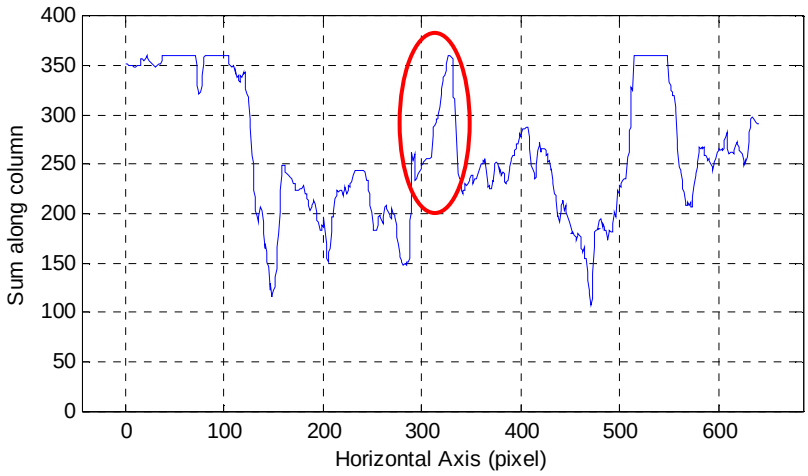
(b)

Fig. 34 A laser-RGB pair showing the out of synchronization problem: (a) laser scan with two detected stalk clusters marked in red; area compassed by two red line corresponded to the webcam's field of view; (b) RGB image with estimated stalk locations from the laser scan showing as red lines. Dashed lines represent the buffered search area. Stalk matching between the two sensors' data was failed.



(a)

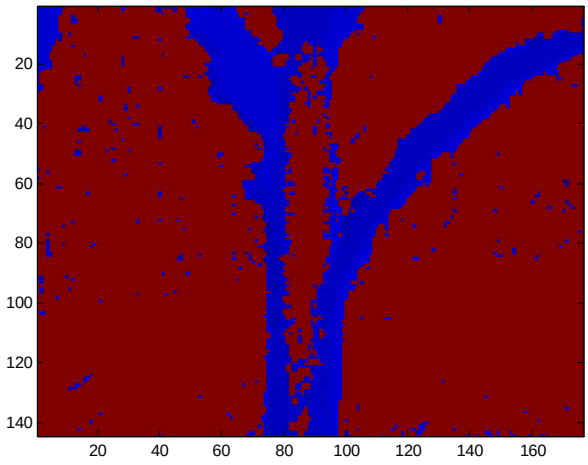
(b)



(c)

Fig. 35 A laser-RGB pair showing the problem of over exposure on the vegetation segmentation: (a) laser scan with two detected stalk clusters marked in red; area compassed by two red line corresponded to the webcam's field of view; (b) RGB image containing two stalks of which the left one suffered with the over-exposure problem (highlighted in red circle) and failed to be identified by the algorithm (c).

(a)



(b)

Fig. 36 An example of noisy pixels in a distance image obtained by the 3D Range Imaging Approach. The plant on the right side in (a) suffered with severe noise in its distance image (b).

As for the FPEr (adding counts), the major reason for the LiDAR-and-RGB Approach was the failure in segmenting the plants in adjacent row (foreground) with those in the background rows. When the foreground and background overlapped and there happened to be some weed or sheath interference, the desired plant stalk could not be segmented correctly which resulted with FPErs in plant counting and errors in stalk diameter measurement (Fig. 37). Even though the multi-angle sensing method combined with the shape-based detection had already largely reduced the leaf and sheath interference for both approaches, FPErs caused by this reason could not be completely eliminated.

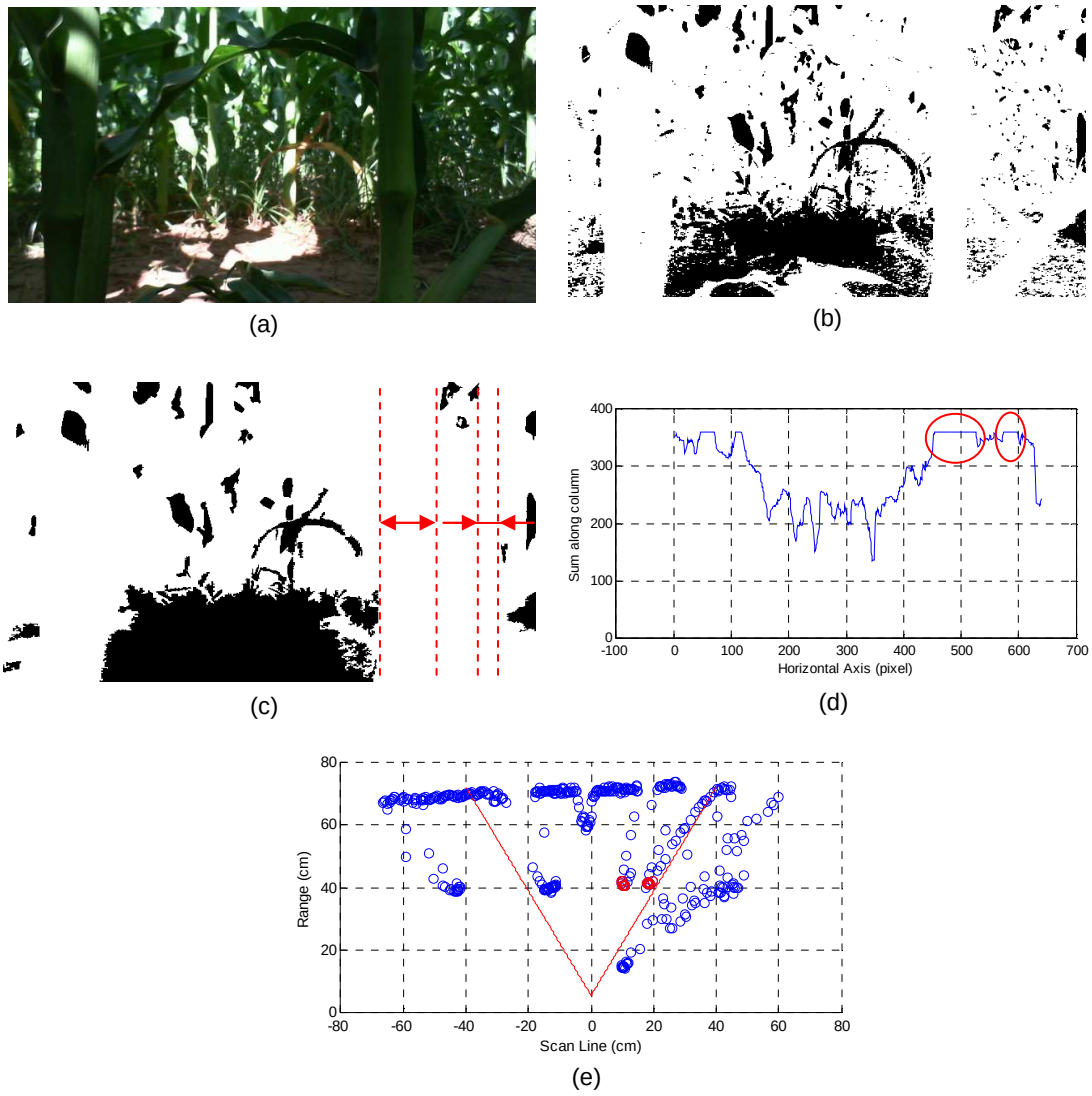


Fig. 37 An example of the failure in differentiating foreground and background vegetation pixels: (a) RGB image with an interested stalk on the right side; (b) binary image after primary vegetation segmentation of (a); the interested stalk failed to be segmented from the soil surface and background vegetation; (c) binary image after hole filling; red lines and arrows showing two identified stalk locations corresponding to the two plateaus circled in (d); the falsely identified stalk happened to correspond to the falsely identified cluster in the laser scan (e) corresponding to a sheath interference.

Within-row Spacing Measurement Errors

Both approaches achieved similar accuracies on within-row spacing measurement. The RMSE of system measured within-row spacing using the LiDAR-and-RGB Approach was 1.1 cm; the one using the 3D Range Imaging Approach was 1.2 cm (Table 11). Good correlations were achieved

between the system measured and manually measured within-row spacing. The R^2 was 0.97 for the LiDAR-and-RGB Approach (Fig. 38) and was 0.96 for the 3D Range Imaging Approach (Fig. 39). The error distribution of the LiDAR-and-RGB Approach had a mean of 0.3 cm (SD = 1.0 cm) (Fig. 40); the error distribution of the 3D Range Imaging Approach had a mean of 0 cm (SD = 1.2 cm) (Fig. 41). Both error distributions were close to normal distributions which demonstrated the reliability of both systems' performance on within-row spacing measurement.

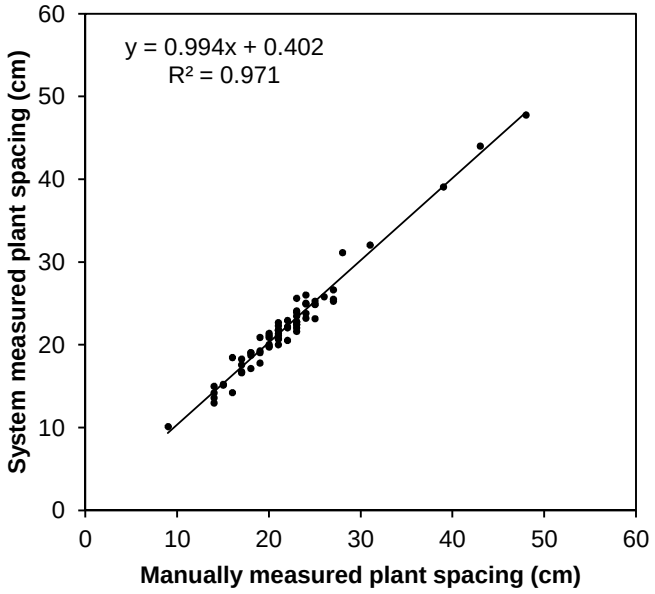


Fig. 38 Comparison between system measured within-row spacing using the LiDAR-and-RGB Approach and manually measured within-row spacing (n = 71).

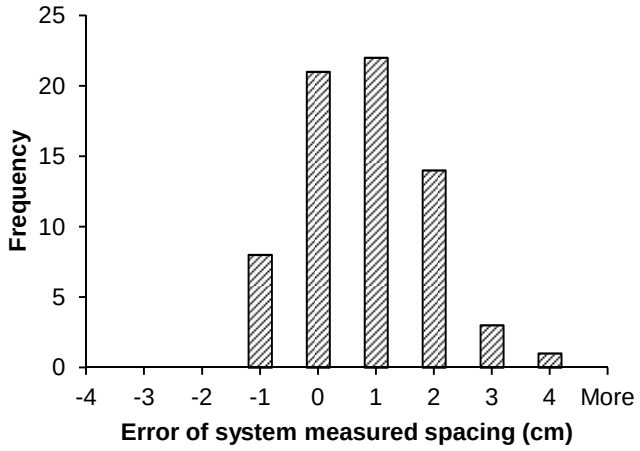


Fig. 39 Error distribution of system measured within-row plant spacing using the LiDAR-and-RGB Approach.

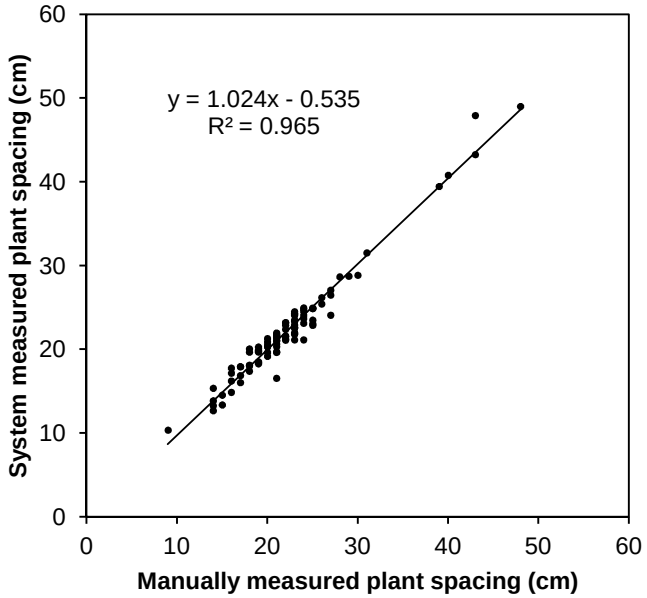


Fig. 40 Comparison between system measured within-row spacing using the 3D Range Imaging Approach and manually measured within-row spacing (n = 94).

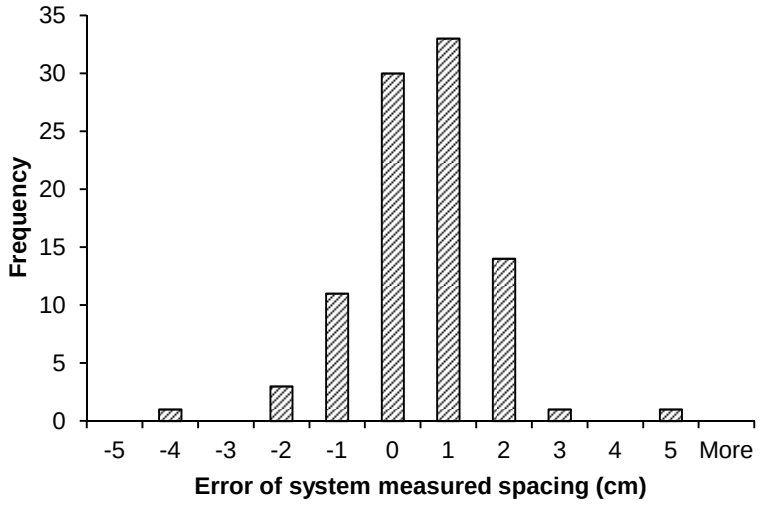


Fig. 41 Error distribution of system measured within-row plant spacing using the 3D Range Imaging Approach.

Stalk Diameter Measurement Errors

The 3D Rang Imaging Approach had a better performance on the stalk diameter measurement than the LiDAR-and-RGB Approach. The LiDAR-and-RGB Approach achieved an RMSE of 4.1 mm in the stalk diameter measurement (Table 11). With an average diameter of 28.9 mm in the corresponding ground truth data (Table 10), this RMSE resulted with a coefficient of variation (CV) of 13.9%. The correlation between the system measured and the manually measured ground truth stalk diameters was low for the LiDAR-and-RGB Approach (R^2 = 0.049 in Fig. 42). This was partly due to the small variation of the measured stalk diameters which spanned from 22.9 mm to 36 mm. If the stalk diameter data from other growth stages could be included, the correlation would be improved. The errors of the system measured stalk diameters using the LiDAR-and-RGB Approach had a mean of 0.9 mm with a standard deviation (SD) of 4.0 mm (Fig. 43). The skew towards the positive direction on the horizontal axis indicated a tendency of over-estimation in the diameter measurement using the LiDAR-and-RGB Approach. This may be because of the error happened in the two sensors' matching procedure somehow always making

the measurement larger. The sources caused the plant counting error discussed before also caused the stalk diameter measurement error if those problems were not severe enough so that a plant stalk could still be identified. Because of these, it was concluded that the LiDAR-and-RGB Approach was not a feasible method for stalk diameter measurement unless accurate sensor calibration and matching were employed.

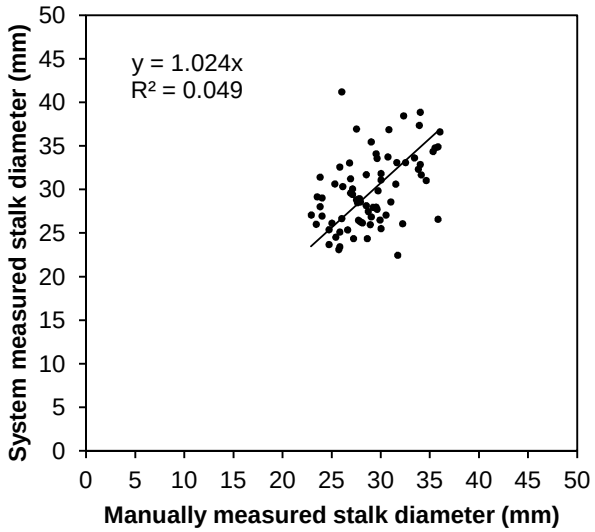


Fig. 42 Comparison between system measured stalk diameters using the LiDAR-and-RGB Approach and manually measured stalk diameters (n=73).

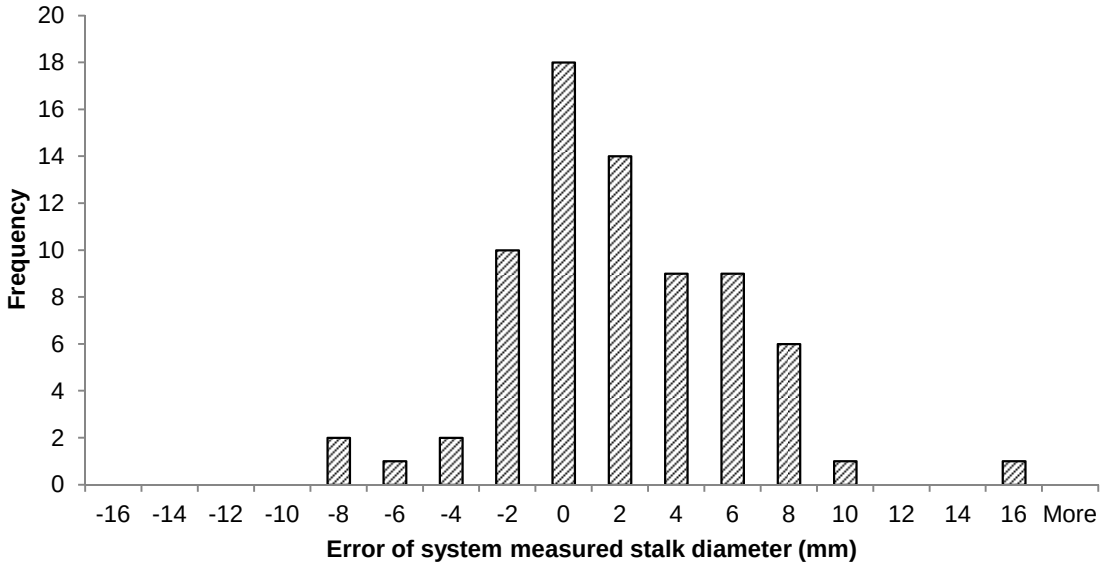


Fig. 43 Error distribution of system measured stalk diameters using the LiDAR-and-RGB Approach.

The 3D Range Imaging Approach was demonstrated to be a feasible method for stalk diameter measurement. It achieved an RMSE of 3.9 mm in the stalk diameter measurement (Table 11). With an average diameter of 27.9 mm in the corresponding ground truth data with the 3D Range Imaging Approach (Table 10), this RMSE resulted with a coefficient of variation (CV) of 13.3%. The correlation between the system measured and the manually measured ground truth stalk diameters was better than that using the LiDAR-and-RGB Approach (R^2 = 0.308 for all the data and R^2 = 0.392 for the same samples with the LiDAR-and-RGB Approach, Fig. 44). The variation of the measured stalk diameters spanned from 17.8 mm to 44.9 mm. The errors of the system measured stalk diameters using the LiDAR-and-RGB Approach had a mean of -1.1 mm with a standard deviation (SD) of 3.7 mm (Fig. 45). The little skew towards the negative direction on the horizontal axis indicated a tendency of under-estimation in the diameter measurement using the 3D Range Imaging Approach. This may be because of the edge effect of the sensor's light source or the under-measurement of the distance the stalk away from the sensor.

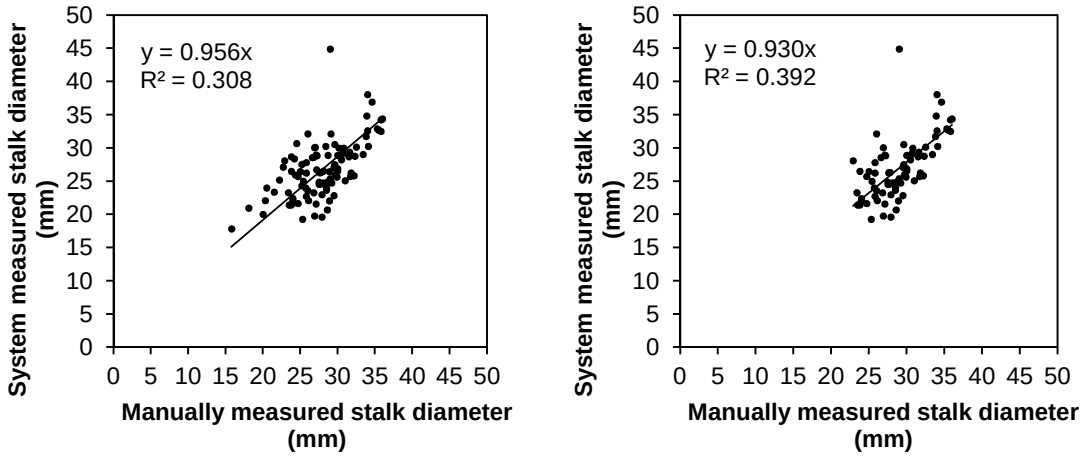


Fig. 44 Comparison between system measured stalk diameters using 3D Range Imaging Approach and manually measured stalk diameters: (a) all of the data (n=98); (b) excluded data of Row 1 to compare with the LiDAR-and-RGB Approach (n = 74).

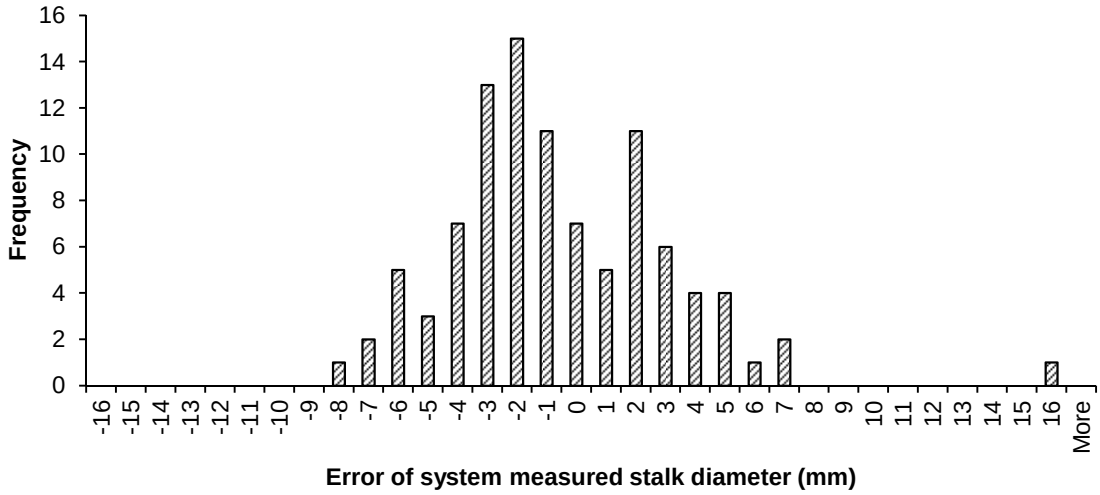


Fig. 45 Error distribution of system measured stalk diameters using the 3D Range Imaging Approach.

Current LiDAR technology usually has a wider field of view comparing with 2D or 3D imaging system due to the rotating light source intrinsically. The system developed in this study was based on the current common 2D or 3D imaging cameras. However, the concept could be adapted to similar system with wider field of view in the future. Sometimes a stalk diameter was measured inaccurate in some of the perspectives but would be more accurate in the others for both approaches. The stalk showing in Fig. 46a, b was measured larger than its actual diameter because a leaf was closely attached to it in that perspective; however, it was clearer after a few images (Fig. 46c, d). This supported the importance of measuring from various perspectives of view.

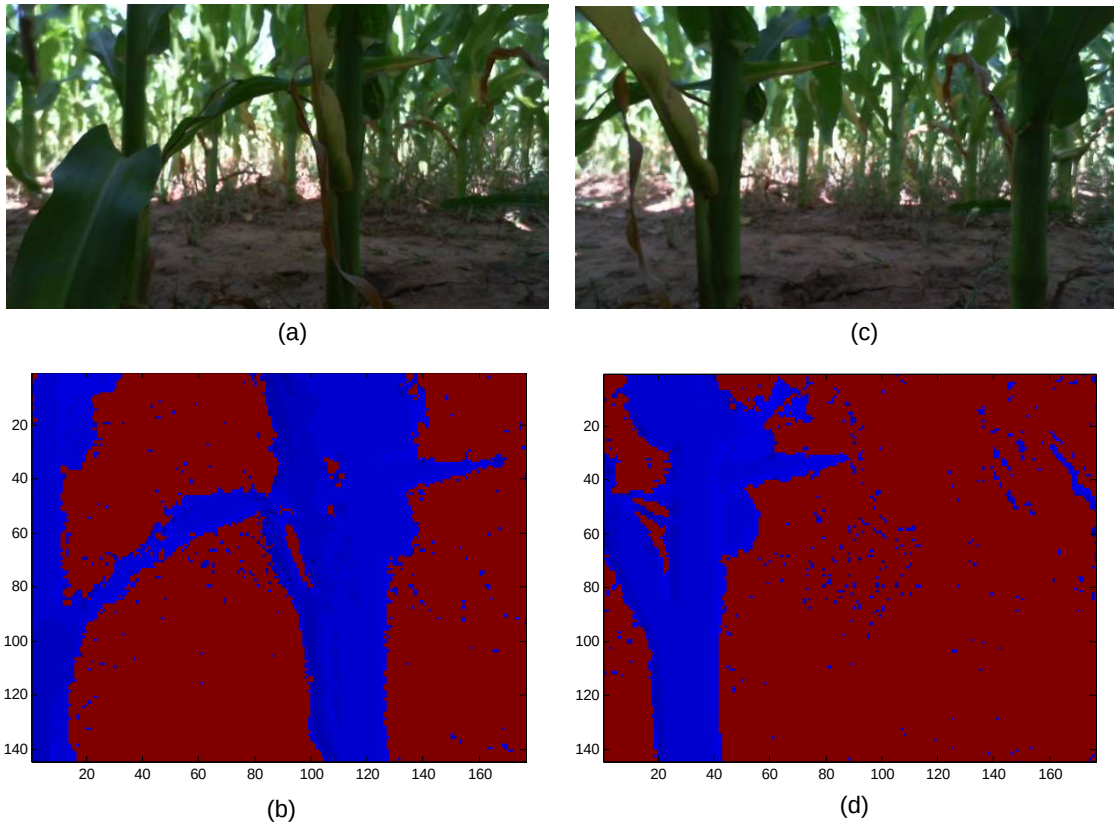


Fig. 46 Example of the advantage of measuring from various perspectives of view.

Future Work

Both approaches achieved acceptable performance on stalk diameter measurement. The problems of the LiDAR-and-RGB Approach were the error happened in the matching of the two sensors' coordinates as well as the asynchronization happened in the data acquisition. Also, to achieve a good vegetation segmentation based on RGB imaging under arbitrary illumination and uncontrolled background would always be challenging. The 3D Range Imaging Approach does not have these problems; however, its major problem was the noisy data when the 3D range camera was operated in outdoor environment. Shade mechanism needs to be well-designed and the sensor variables such as integration time, exposure time or shutter speed need to be carefully

adjusted. Other problem of the 3D Range Imaging Approach was the low pixel resolution which could be easily solved by upgrading to high pixel resolution sensors.

In this study, the concept of sensing the same object from various perspectives of view was partially realized. Due to the relative small fields of view of both approaches, the diameters measured were actually the diameters of the axis aligned with the plant row line. In order to have a better approximation to the oval-shaped corn stalks, systems with larger field of view – better greater than 90° – would be preferred.

Conclusions

Two approaches based on machine vision technology for measuring corn stalk diameters on-the-go were developed and evaluated at corn plants' mid growth stage: the LiDAR-and-RGB Approach was a combination of a LiDAR sensor and a webcam; the 3D Range Imaging Approach used a 3D range camera. The field experiment results demonstrated that:

- Using either approach to measure corn stalk diameters from various perspectives of view on-the-go is a feasible method with acceptable performance though the 3D Range Imaging Approach was little superior. The RMSEs of system measured stalk diameters using the two approaches were 4.1mm and 3.9 mm which correspond to CVs of 13.9% and 13.3%, respectively.
- Both approaches achieved good performances on plant counting and within-row spacing measurements. The total counting errors were 4.5% and 3.4%, and the RMSEs of within-row spacing measurement were 1.1 cm and 1.2 cm for each approach, respectively.
- The 3D Range Imaging Approach performed superior than the LiDAR-and-RGB Approach in terms of the simplicity of the system deployment and of the data processing which otherwise would be the error sources.

References

Chen, Y., Zhu, H., Ozkan, H. E. (2012). Development of a variable-rate sprayer with laser scanning sensor to synchronize spray outputs to tree structures. Transactions of the ASABE, 55(3): 773-781.

Heege, H. J., S. Reusch, and E. Thiessen. 2008. Prospects and results for optical systems for site-specific on-the-go control of nitrogen-top-dressing in Germany. Precision Agric, 9: 115-131.

Kelly, J. P. 2011. By plant prediction of corn (Zea mays L.) grain yield using height and stalk diameter. Master's thesis, Oklahoma State University.

Li, H., Worley, S. K., & Wilkerson, J. B. (2009). Design and optimization of a biomass proximity sensor. Transactions of the ASABE, 52(5), 1441-1452.

Lobdell, B. M, and J. W. Hummel. 2001. Optical plant spacing and stalk diameter sensor with air-jet assist. In Proc. 3rd Intl. Conf. on Geospatial Information in Agriculture and Forestry (unpaginated CD-ROM). Ann Arbor, Mich.: Veridian.

Microsoft Corporation. 2012. Technical data sheet of LifeCam Cinema.

MESA Imaging AG. 2011. SR4000 datasheet, Rev 5.1.

MESA Imaging AG. 2013. SR4000/SR4500 user manual, version 3.0.

Mourtzinis, S., F. J. Arriaga, K. S. Balkcom, and B. V. Ortiz. 2013. Corn grain and stover yield prediction at R1 growth stage. Agronomy Journal, 105(4): 1045-1050.

Nakarmi, A. D., and L. Tang. 2012. Automatic inter-plant spacing sensing at early growth stages using a 3D vision sensor. Computers and Electronics in Agriculture, 82: 23-31.

Okiror, G. P. 2012. Development of microwave dielectric algorithms for sensing corn stalks. PhD dissertation, Oklahoma State University.

Plattner, C. E., and J. W. Hummel. 1996. Corn plant population sensor for precision agriculture. p. 785-794. In P. C. Robert, R. H. Rust, and W. E. Larson (ed.), Proc. 3rd Intl. Conf. on Precision Agriculture. ASA, CSSA, and SSSA, Madison, WI.

Rascon, J. A. 2012. Corn sensor development for by-plant management. Master's thesis, Oklahoma State University.

Raun, W. R. J. B. Solie, M. L. Stone, K. L. Martin, K. W. Freeman, R. W. Mullen, H. Zhang, J. S. Schepers, and G. V. Johnson. 2005. Optical sensor-based algorithm for crop nitrogen fertilization. Communications in Soil Science and Plant Analysis, 36: 2759-2781.

Rigney, M. P. and G. A. Kranzler. 1988. Machine vision for grading southern pine seedlings. Trans. ASAE, 31(2): 642-646.

Saeys, W., Lenaerts, B. Craessaerts, G. & De Baerdemaeker, J. (2009). Estimation of the crop density of small grains using LiDAR sensor. Biosystems Engineering, I02 (2009) 22-30.

SICK AG. 2006. Technical description of LMS200/211/221/291 laser measurement system. 8008970/QI72/2006-12.

Shi, Y., N. Wang, R. K. Taylor, W. R. Raun, and J. A. Hardin. 2013. Automatic corn plant location and spacing measurement using laser line-scan technique. Precision Agric, 14: 478-494.

Sudduth, K. A., S. J. Birrell, and M. J. Krumpelman. 2000. Field evaluation of a corn population sensor. In Proc. 5th Intl. Conf. on Precision Agriculture (unpaginated CD-ROM). Madison, Wisc.: ASA, CSSA, and SSSA.

USDA-NASS. 2007. Corn production in 2007. Washington, D.C.: USDA National Agricultural Statistics Service. Available at: www.usda.gov/nass/aggraphs/crops.htm.

Wei, J., & Salyani, M. (2004). Development of a laser scanner for measuring tree canopy characteristics: phase 1. prototype development. Transactions of the ASABE, 47(6): 2101-2107.

Wei, J., & Salyani, M. (2005). Development of a laser scanner for measuring tree canopy characteristics: phase 2. foliage density measurement. Transactions of the ASABE, 48(4): 1595-1601.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

A system for automatic corn plant location, within-row spacing and stalk diameter measurements based on optical sensing technologies was developed and tested in the field. The system realized the plant location and spacing measurements by a single LiDAR sensor viewing each plant from multiple angles when a testing platform was moving between plant rows. Comparing with previous studies that mostly used laser pointer sensors, this strategy was demonstrated to be able to effectively eliminate interference factors and largely increased the possibility of correctly recognizing plants. The system also demonstrated the feasibility of measuring stalk diameters using the concept of multi-angle measurement.

The study consisted of three phases. In phase I, the function of corn plant location and spacing measurement was realized using a moving LiDAR sensor based sensing system and corresponding data processing algorithm. This technique is advantageous in this application because the line-scan data sets taken from various points of view of a plant stalk results in less interference and higher probability of plant recognition. Each potential stalk cluster was identified in a scan and registered with the same stalks in previous scans. The final location of a stalk was the average of the measured locations in all scans. The system achieved 24.0 % and 10.0 % of mean total errors in plant counting at the V8 and V10 growth stages, respectively. The RMSE between system measured plant locations and manually measured ones were 2.3 cm and 2.6 cm at

the V8 and V10 growth stages, respectively. The interplant spacing measured by the developed system had a good correlation with the manual measurement with an R^2 of 0.962 and 0.951 for the V8 and V10 growth stages, respectively.

In phase II, the system developed in phase I was improved in terms of the data processing algorithm and data acquisition platform. The enhancement was demonstrated by higher measurement accuracy on two years' data. Compared with the system developed in phase I, this system improved the data acquisition platform to insure the quality of data collection, and the data processing algorithm especially the scan registration and stalk recognition procedure to reduce the misidentification errors. More data was collected in the field to test the system performance. A total error of 5.5% in plant counting and a 1.9 cm of RMSE in spacing measurement were achieved. The improved data processing algorithm was also tested on the data collected in phase I. The total plant counting error decreased to 14% from 24% for data collected at the V8 growth stage when weed interference existed.

In phase III, the function of stalk diameter measurement was added to the existing system by developing and comparing two approaches – one based on a combination of a LiDAR sensor and a webcam; the other based on a 3D range camera. Data and image processing algorithms were developed for each approach to identify the existence of a plant stalk and estimate its diameter using both shape and range information. The 3D Range Imaging Approach was demonstrated to be a feasible method for corn stalk diameter measurement while the LiDAR-and-RGB Approach was not. The RMSEs of system measured stalk diameters using the two approaches were 4.1mm and 3.9 mm which correspond to CVs of 13.9% and 13.3%, respectively. Both approaches achieved good performances on plant counting and within-row spacing measurements. The total counting errors were 4.5% and 3.4%, and the RMSEs of within-row spacing measurement were 1.1 cm and 1.2 cm for each approach, respectively. The 3D Range Imaging Approach performed

superior than the LiDAR-and-RGB Approach in terms of the simplicity of the system deployment and of the data processing which otherwise would be the error sources.

Future Work

- Real-time sensing and measurement for the within-row spacing and stalk diameters were not implemented in this study. However, considerations on this were made at the beginning and throughout the whole process. Data processing algorithms were developed with a flow could be used in a real-time application.
- The stalk diameter measurement in this study had a small sensing angle so few measurements were taken for estimation. In the future, imaging sensors with a larger field of view, especially larger than 90°, would be preferred to have more measurements from different perspectives and a better approximation of the oval shape stalk.
- Sensors and data acquisition system with faster communication speed are developing all the time. A faster data acquisition and processing speed would benefit the real-time application. This system could be finally incorporated with the spraying operation to realize the rea-time variable-rate fertilizer application.

Original Contribution to Community of Science

This study proposed a novel method for corn plant morphological characterization by sensing from different angles on-the-go which is advantageous with less interference and higher probability of plant recognition.

APPENDICES

APPENDIX A

SICK® LMS291™ Specifications

Scanning angle (field of vision)	100° ~ 180° (type-dependent)
Motor speed	75 Hz
Angular resolution (response time)	0.25° (53.33 ms); 0.5° (26.66 ms); 1° (13.33 ms); selectable
Range	Max. 80 m (type-dependent)
Measurement resolution	10 mm
Measurement accuracy	typical ±35 mm
Systematic error	mm-mode: typical ±35 mm at range 1 to 20 m cm-mode: typical ±5 cm at range 1 to 20 m
Statistical error	mm-mode: typical 10 mm at range 1 to 20 m/ reflectivity ≥ 10 %/ light ≤ 5 klx
Laser diode (wavelength)	Infra-red (λ = 905 nm)
MTBF of LMS2xx	50,000 h
Laser class of device	Class 1 (eye-safe), to EN/IEC 60825-1 and to 21CFR 1040.10
Optical indicators	3 x LED
Data interface	RS 232 or RS 422 (selectable in the connector plug)
Data transfer rate	RS 232: 9.6 / 19.2 kbd RS 422: 9.6 / 19.2/ 38.4/ 500 kbd
Data format	1 start bit, 8 data bits, 1 stop bit, no parity (fixed)
Electrical connections	1 x plug module plug with 9-pin D Sub socket (solder connection)
Operating voltage (according to IEC 364-4-41)	24 V DC ± 15 % (max. 500 mV ripple), current consumption max. 1.8 A (with output load)
Power consumption	Approx. 20 W (without load)
Housing	Aluminium die-cast
Protection class	Class 2 (to VDE 0106/IEC 1010-1), safety insulated
Weight (without installation accessories)	Approx. 4.5 kg

SwissRanger® SR4000™ Specifications

Imager Parameters (z)	**Value**	**Comment**
Illumination Wavelength	850 nm	Central wavelength
Optical filter	-	Bandpass / Glass substrate
Maximum Frame Rate	50 FPS	Camera setting dependent
Imager parameters (x,y)	Value	Comment
Pixel Array Size	176 (h) x 144 (v)	QCIF
Field of View	43.6° (h) x 34.6° (v) or 69° (h) x 56° (v)	Standard field of view cameras Wide field of view cameras
Pixel Pitch	40 µm	Horizontal and vertical
Angular Resolution	0.24° 0.39°	Standard field of view; central pixels Wide field of view; central pixels
Focus length / adjustment	10 mm 5.8 mm	Standard field of view cameras Wide field of view cameras Manually adjustable over operating range

Environmental	**Value**	**Comment**
External light disturbances	Designed for indoor use	Not to be used in direct sunlight
Operating Temperature	+10 °C to +50 °C (50 °F to 122 °F)	Housing temperature
Storage Temperature	-20 °C to +70 °C (-4 °F to 158 °F)	

Power Connections	**Value**	**Comment**
Electrical Power Requirements	12 V (-2%; +10%), maximum 1.0 A, (typical 0.8 A)	Power supply available from MESA
Trigger connector	Lumberg M8 Male 4-pin	Screw connector (on camera)
Power connector	Lumberg M8 Male 3-pin	Screw connector (on camera)

Software	**Value**	**Comment**
Software Drivers	Windows XP, Windows 7 (32-bit and 64-bit), Vista (32-bit and 64-bit), Linux 32-bit	
Software API	C, C++, Matlab	

Software features	**Value**	**Comment**
Modulation frequency selection	29/30/31 MHz or 14.5/15/15.5 MHz selectable	Depending on camera model
Acquisition mode	Continuous, Triggered	Trigger via Software or Hardware
Integration time	0.3 to 25.8 ms, steps of 0.1 ms	Selectable
Confidence Map	Measures quality of distance data, quality threshold to be set by user	

Data Output	**Value**	**Comment**
Spherical distance (Range)	0-65535 (16 Bit) <--> 0-5 m 0-65535 (16 Bit) <--> 0-10 m	@ 30 MHz modulation @ 15 MHz modulation Data output from camera without

		Cartesian coordinate transfer
Cartesian XYZ coordinates	x, y, z (m)	Up to 5 m distance @ 30 MHz modulation Up to 10 m distance @ 15 MHz modulation
Signal amplitude	0-65535 (16 Bit)	Value above 32767 indicates saturation
Converted grayscale Image	0-65535 (16 Bit)	Value above 32767 indicates saturation
Confidence Map	0-65535 (16 Bit)	Quality threshold to be set by user

Mechanical	**Value**	**Comment**
Dimensions	65 x 65 x 68 mm 65 x 65 x 76 mm	For USB cameras For Ethernet cameras Excludes the connectors
Case Material	Anodized Aluminum	
Color front housing	Black	
Color back cover	Red	
Window Material	Polycarbonate	Illumination cover
	Borofloat glass	Objective cover
Mounting Holes	4 x M4; 2 x 4H7; 1 x 1/4”	
Weight	470 g 510 g	For USB cameras For Ethernet cameras
Cooling	Passive, no fan	Camera always to be connected to a heat sink

Microsoft® LifeCam Cinema™ Specifications

Product Dimensions	
Webcam Length	2.20 inches (55.9 millimeters)
Webcam Width	1.81 inches (46.0 millimeters)
Webcam Depth/Height	1.58 inches (40.0 millimeters)
Webcam Weight	3.36 ounces (95.3 grams)
Webcam Cable Length	72.0 inches +6/-0 inches (1829 millimeters +152/-0 millimeters)

Compatibility and Localization	
Interface	Compatible with USB 2.0 High Speed specification
Operating Systems	Microsoft Windows® 7, Windows Vista®, and Windows XP Service Pack 2 or higher (excluding Windows XP 64-bit)

Imaging Features	
Sensor	CMOS sensor technology
Resolution	• Motion Video: 1280 x 720 pixels video • Still Image: Up to 5 megapixel (2880x1620 pixels, interpolated) photos*
Imaging Rate	Up to 30 frames per second
Field of View	73° diagonal field of view
Imaging Features	• Digital pan, tilt, and zoom • Auto focus, range from 6” to infinity • Automatic image adjustment with manual override

NI USB-6008 Specifications

Analog Input (not used in this study)

Analog Output (not used in this study)

Digital I/O (not used in this study)

External Voltage

+5 V output (200 mA maximum)
- Minimum .....+4.85 V
- Typical.....+5 V

+2.5 V output (1 mA maximum) .....+2.5 V
+2.5 V accuracy .....0.25% maximum
Reference temperature drift .....50 ppm/°C maximum

Event Counter

Number of counters 1
Resolution .....32 bits
Counter measurements.....Edge counting (falling-edge)
Counter direction .....Count up
Pull-up resistor.....4.7 kΩ to 5 V
Maximum input frequency.....5 MHz
Minimum high pulse width.....100 ns
Minimum low pulse width.....100 ns
Input high voltage .....2.0 V
Input low voltage .....0.8 V

Bus Interface

USB specification .....USB 2.0 full-speed
USB bus speed .....12 Mb/s

Power Requirements

USB
- 4.10 to 5.25 VDC
  - Typical.....80 mA
  - Maximum.....500 mA
- USB suspend
  - Typical.....300 µA
  - Maximum.....500 µA

APPENDIX B

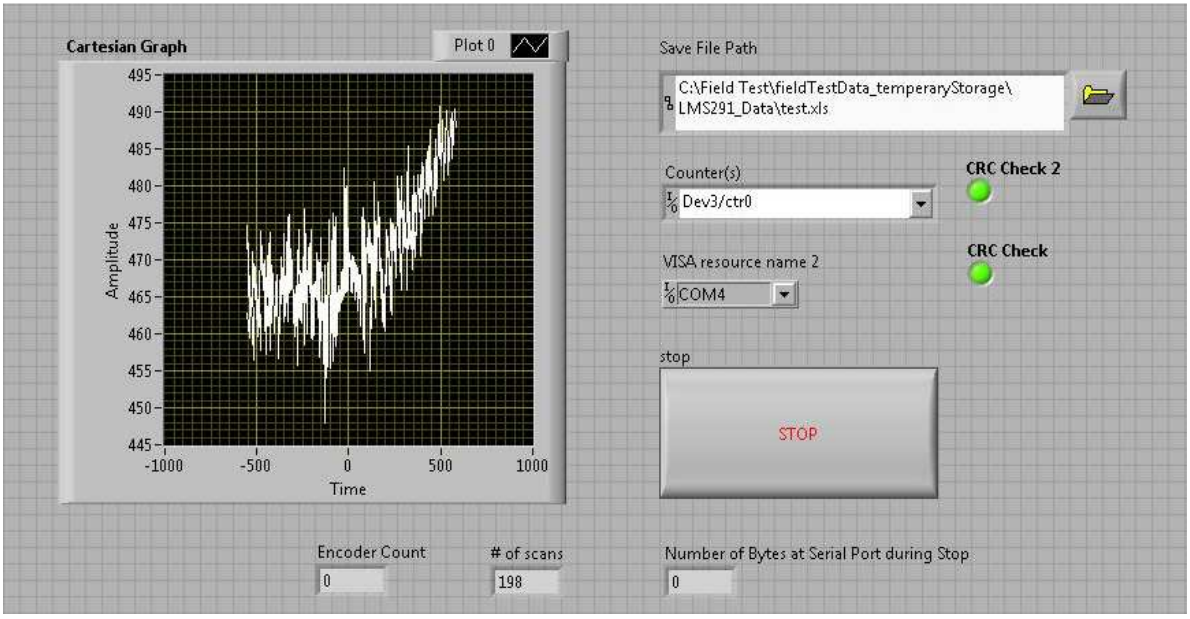
List of Programs

APPENDIX C

LabVIEW Program for Data Acquisition

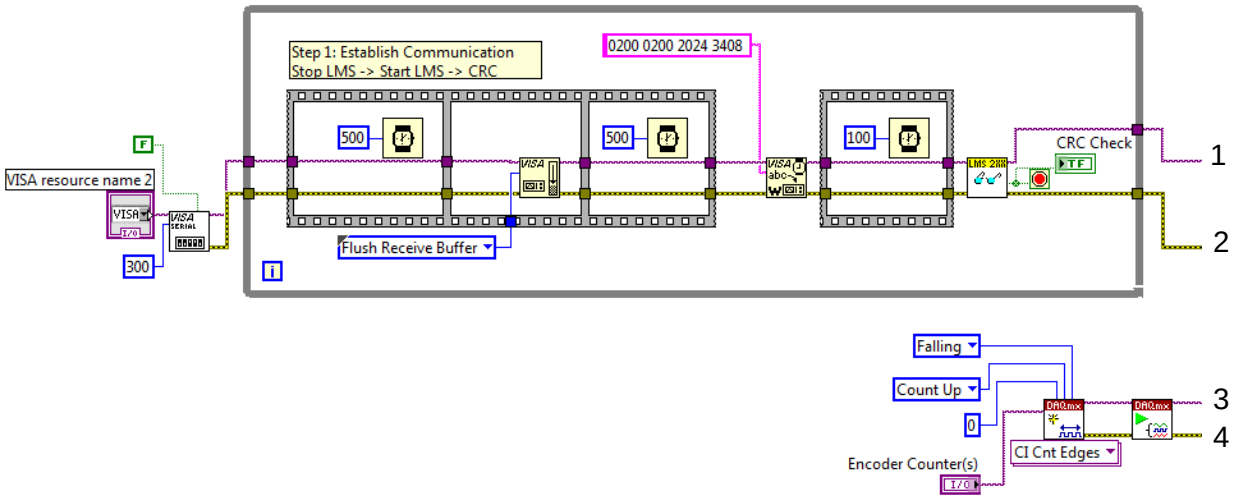
LabVIEW Program for Chapter IV (Improved Version for Chapter III)

Front Panel

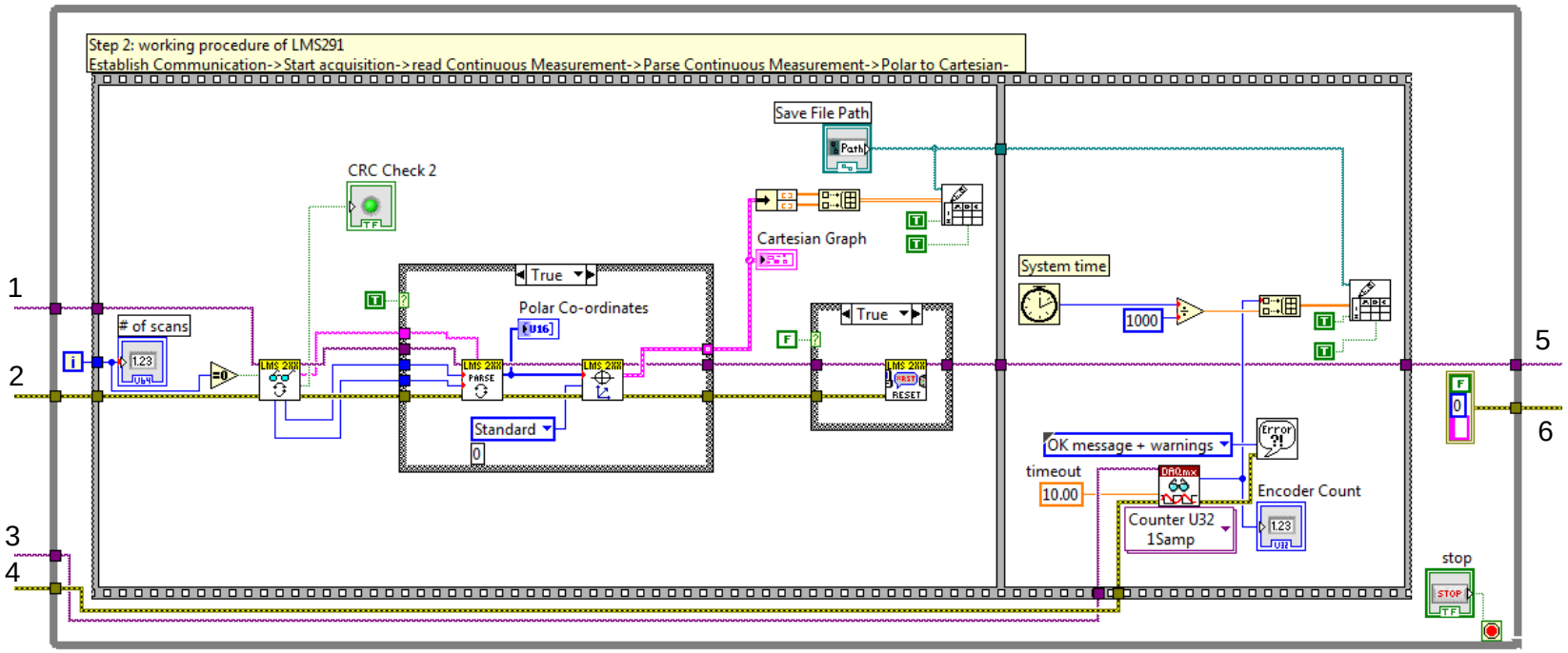


Block Diagram

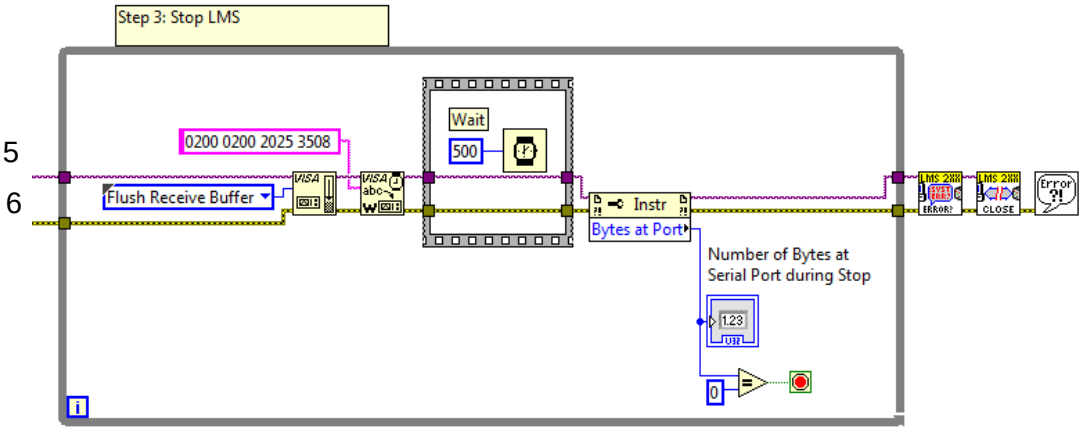
Part 3_1:



Part 3_2:

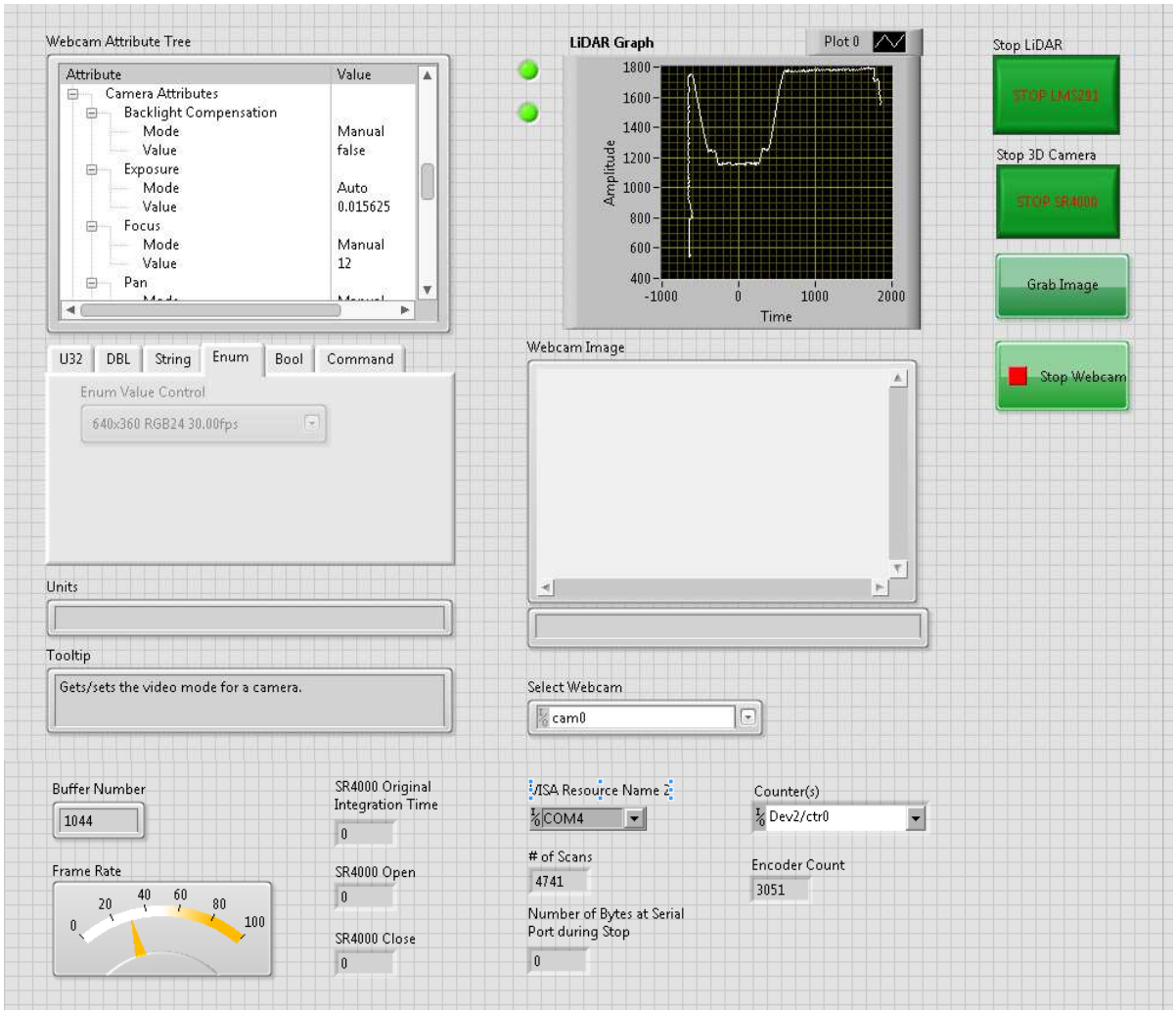

Step 2: working procedure of LMS291
Establish Communication->Start acquisition->read Continuous Measurement->Parse Continuous Measurement->Polar to Cartesian-
Save File Path
CRC Check 2
Cartesian Graph
True
Polar Co-ordinates
# of scans
Standard
System time
1000
OK message + warnings
timeout
10.00
Counter U32
1Samp
Encoder Count
stop
1
2
3
4
5
6

Part 3_3:


Step 3: Stop LMS
5
6
0200 0200 2025 3508
Flush Receive Buffer
VISA
VISA abc
Wait
500
Instr
Bytes at Port
Number of Bytes at Serial Port during Stop
1.23
U32
0
LMS 2RR ERROR?
LMS 2RR CLOSE
Error ?!
i

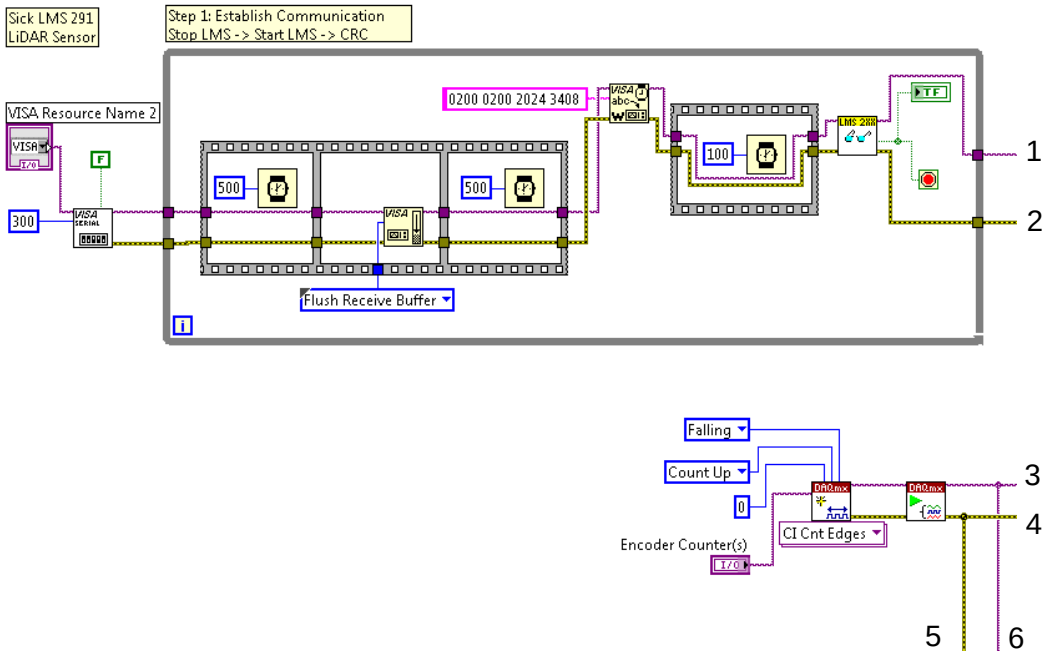
LabVIEW Program for Chapter V

Front Panel

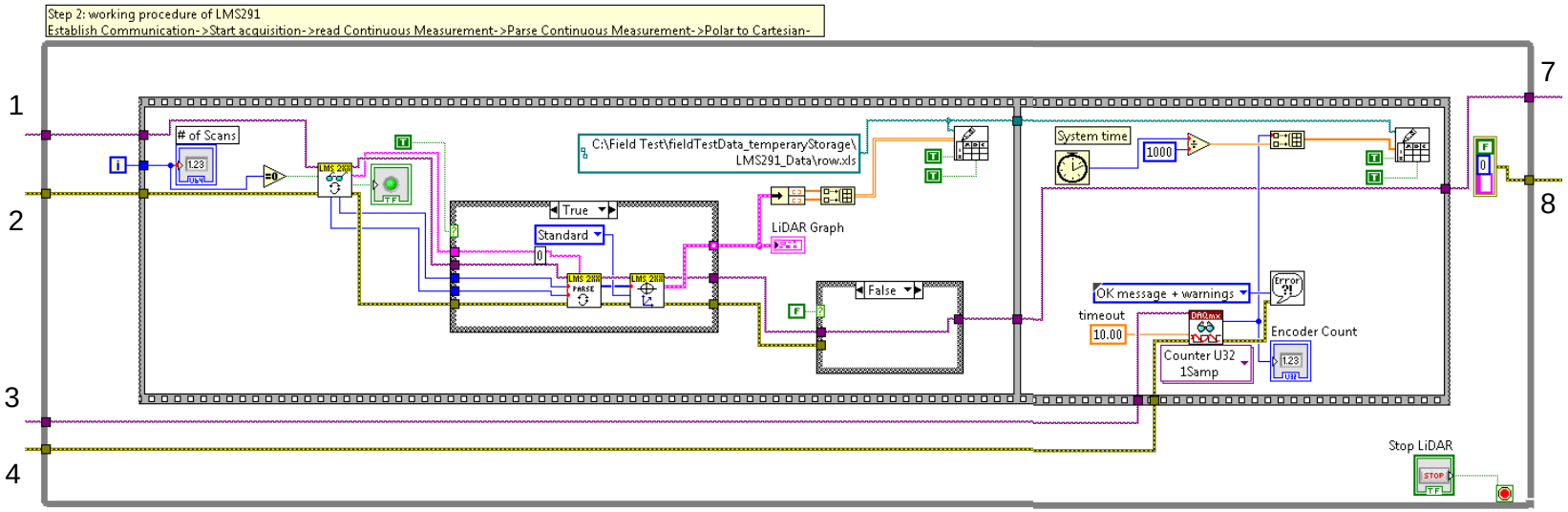


Block Diagram

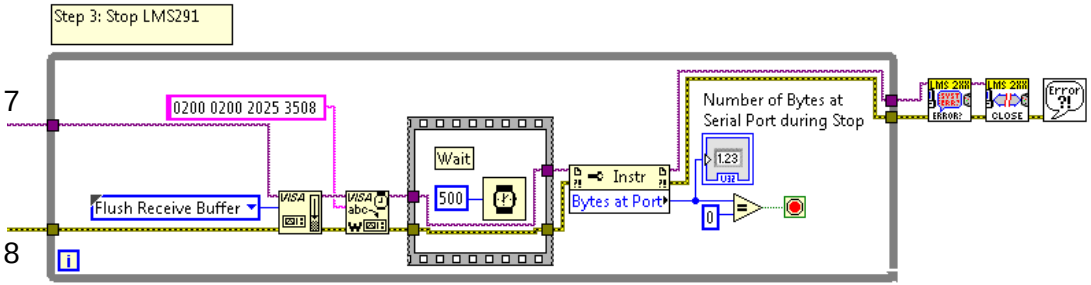
Part 8_1: Step 1. Establish communication with LiDAR sensor



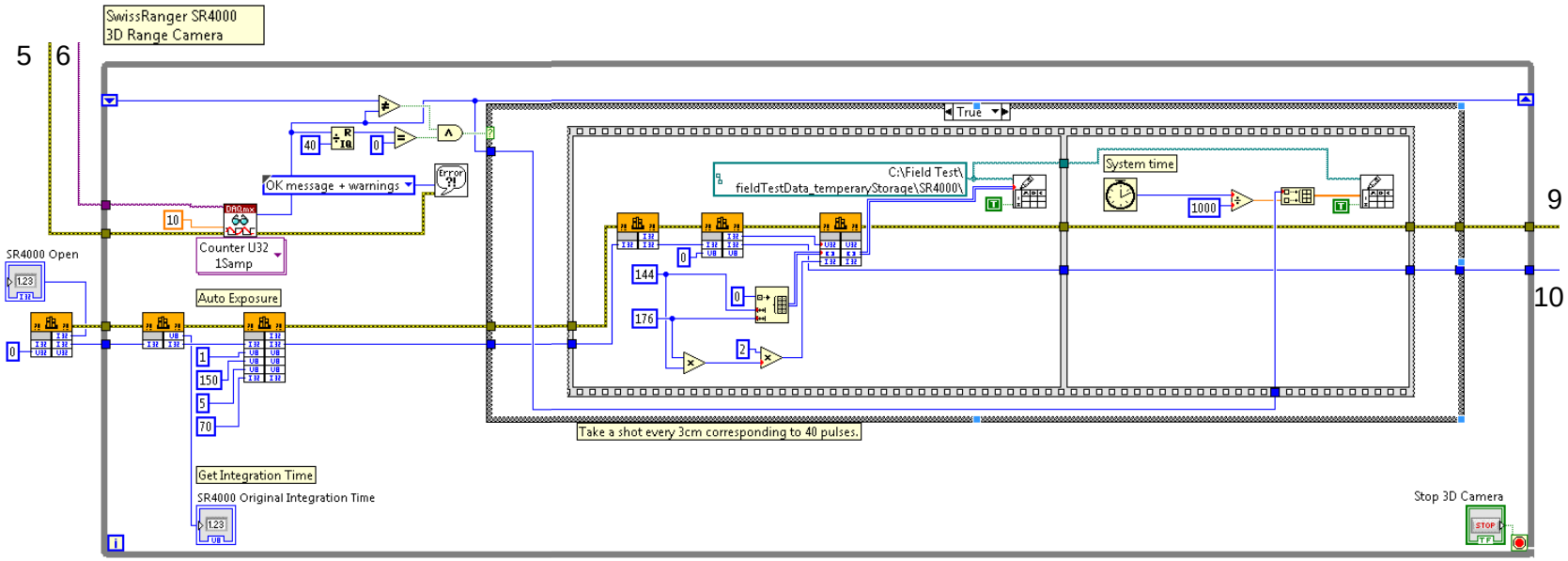
Part 8_2: Step 2. Start acquisition -> parse packages -> save LiDAR sensor's data



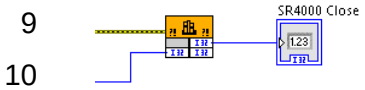
Part 8_3: Step 3. Stop LiDAR sensor



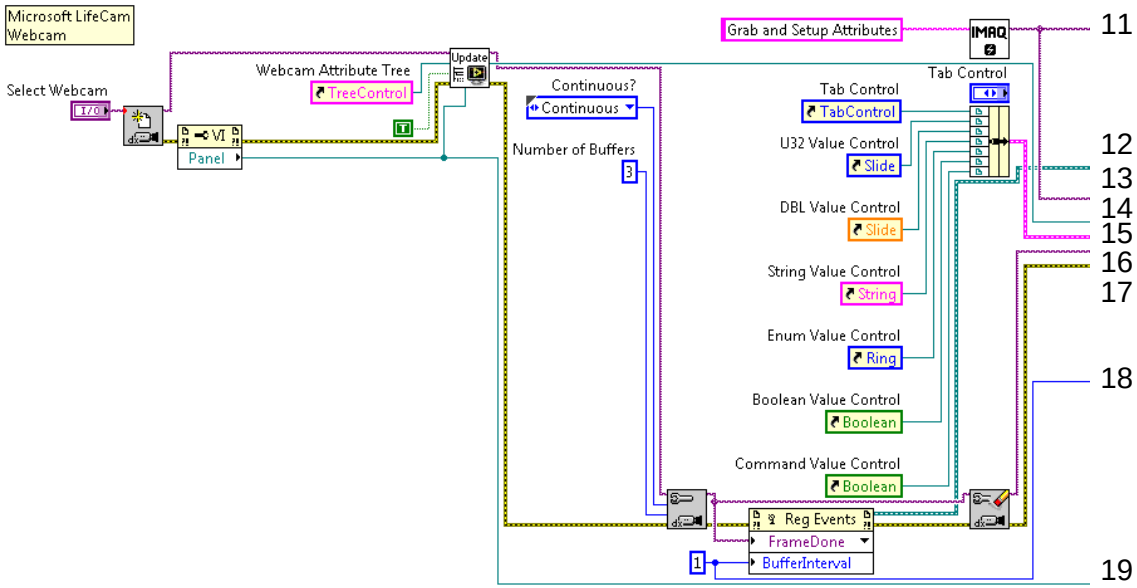
Part 8_4: Establish communication with 3D camera -> acquisition and save data



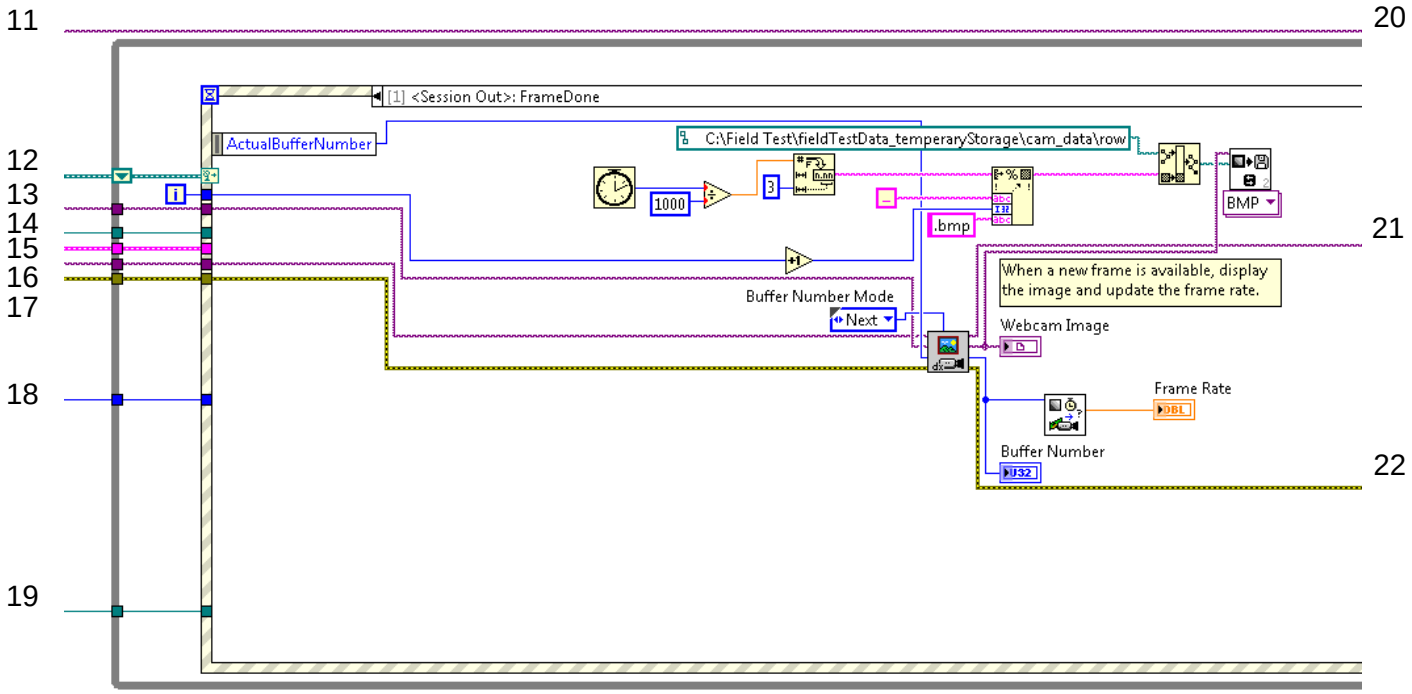
Part 8_5: Step 3. Stop 3D camera



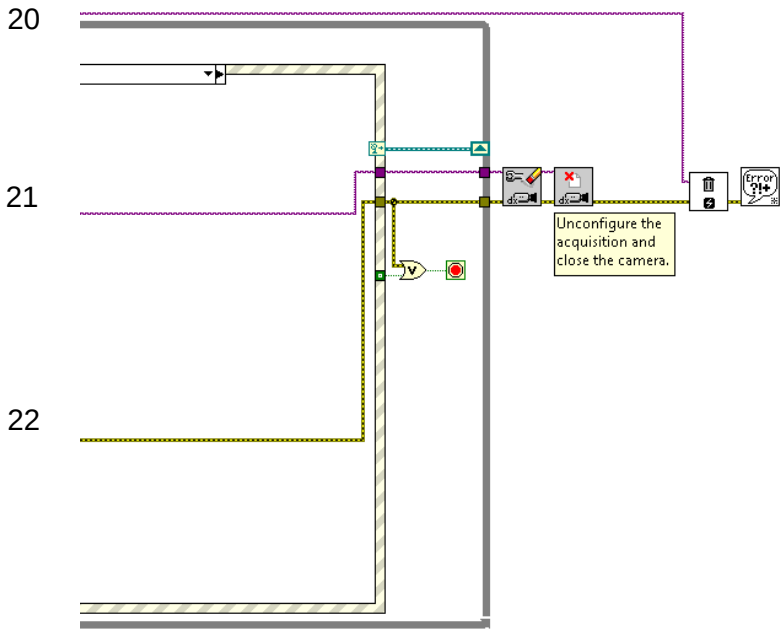
Part 8_6: Establish communication with webcam -> configure attributes



Part 8_7: Get and save images from webcam



Part 8_8: Close webcam



APPENDIX D

Selected MATLAB Code for Chapter IV

(Improved Version for Chapter III)

Main Function

```
% This code is used to parse the LMS291 data
collected for plant spacing measurement in 2012.
% Yeyin Shi, Oct 2012

close all;
clear all; clc;

global D_thres_AllScans valid_C_members_AllScans
core_C_AllScans span_C_AllScans;
global encoder stalkDiameter_AllStalks
groundCoor_scan;
global buf;
global num_stalk num_stalk_pending pending_count
pending_clusters;
global num_noise num_noise_pending
pending_noise_count pending_noise;
global thres;


encoder_countCoef = 0.77;  % Encoder reading
correction coefficient (mm/count);
ratio_n = [0.1 0.2 0.3 0.4 0.5];  % for variable
optimization
buf_array = [2 4 6 8 10 12];  % for variable optimization
OF = zeros(numel(buf_array), numel(ratio_n), 40);


for idx_row = 1:8
   for idx_trial = 1:5

      flag = 1;  % 'flag' indicating if this trial data was in
use or not, 1 – in use; 0 – not in use.

      switch idx_row

         %% Select the trial
         case 1
            groundTruth =
xlsread('D:\CornStalkDiameterProject\Data\Summer201
2\LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 1-4', 'F7:F31');
% row 1
            groundTruth_offset = 93;    % row 1
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\05232012_plots1to4_V8\laserScanner_5inchAboveRo
ots_05232012_plots1to4_V8_m', filename);

            switch idx_trial
               case 1
                  n_scans = 89.88;
                  encoder_offset = 33.26;
                  encoder_correctionCoef = 698/702.09;
               case 2
                  flag = 0;  % used as validation set
%                    n_scans = 61.63;
%                    encoder_offset = 35.11;
%                    encoder_correctionCoef =
698/703.01;
               case 3
                  n_scans = 48.79;
                  encoder_offset = 33.65;
                  encoder_correctionCoef = 698/703.55;
               case 4
                  n_scans = 59.06;
                  encoder_offset = 38.19;
                  encoder_correctionCoef = 698/703.24;
               case 5
                  n_scans = 51.36;
                  encoder_offset = 31.42;
                  encoder_correctionCoef = 698/704.78;
            end

         case 2
            groundTruth =
xlsread('D:\CornStalkDiameterProject\Data\Summer201
2\LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 1-4', 'F32:F56');
% row 2
            groundTruth_offset = 105;    % row 2
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\05232012_plots1to4_V8\laserScanner_5inchAboveRo
ots_05232012_plots1to4_V8_m', filename);

            switch idx_trial
               case 1
                  n_scans = 76.62;
                  encoder_offset = 37.11;
                  encoder_correctionCoef = 655/663.05;
               case 2
                  flag = 0;
               case 3
                  flag = 0;  % used as validation set
%                    n_scans = 71.15;
%                    encoder_offset = 39.04;
```

```
%                   encoder_correctionCoef =
655/661.35;
                case 4
                    n_scans = 79.36;
                    encoder_offset = 37.27;
                    encoder_correctionCoef = 655/663.59;
                case 5
                    flag = 0;
            end

        case 3
            groundTruth =
xlsread('D:\CornStalkDiameterProject\Data\Summer201
2\LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 1-4', 'F57:F81');
% row 3
            groundTruth_offset = 97;    % row 3
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\05232012_plots1to4_V8\laserScanner_5inchAboveRo
ots_05232012_plots1to4_V8_m', filename);

            switch idx_trial
                case 1
                    flag = 0;  % used as validation set
%                       n_scans = 58.86;
%                       encoder_offset = 20.10;
%                       encoder_correctionCoef =
670/673.37;
                case 2
                    n_scans = 80.26;
                    encoder_offset = 18.48;
                    encoder_correctionCoef = 670/674.83;
                case 3
                    n_scans = 61.53;
                    encoder_offset = 17.02;
                    encoder_correctionCoef = 670/675.68;
                case 4
                    n_scans = 72.23;
                    encoder_offset = 17.17;
                    encoder_correctionCoef = 670/678.14;
                case 5
                    flag = 0;  % used as validation set
%                       n_scans = 69.56;
%                       encoder_offset = 18.79;
%                       encoder_correctionCoef =
670/676.37;
            end

        case 4
            groundTruth =
xlsread('D:\CornStalkDiameterProject\Data\Summer201
2\LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 1-4',
'F82:F106');  % row 4
            groundTruth_offset = 105;    % row 4
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\05232012_plots1to4_V8\laserScanner_5inchAboveRo
ots_05232012_plots1to4_V8_m', filename);

            switch idx_trial
                case 1
                    n_scans = 75.28;
                    encoder_offset = 21.02;
                    encoder_correctionCoef = 500/506.66;
                case 2
                    n_scans = 75.28;
                    encoder_offset = 21.33;
                    encoder_correctionCoef = 500/507.05;
                case 3
                    n_scans = 71.70;
                    encoder_offset = 23.72;
                    encoder_correctionCoef = 500/502.66;
                case 4
                    flag = 0;  % used as validation set
%                       n_scans = 78.87;
%                       encoder_offset = 22.64;
%                       encoder_correctionCoef =
500/503.27;
                case 5
                    n_scans = 86.04;
                    encoder_offset = 22.10;
                    encoder_correctionCoef = 500/503.50;
            end

        case 5
            groundTruth = xlsread('LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 5-8', 'F9:F33');
% row 5
            groundTruth_offset = 110;    % row 5
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\06042012_plots5to8_V8\laserScanner_06042012_plot
s5to8_V8_5inAboveRoots', filename);

            switch idx_trial
                case 1
                    n_scans = 56.01;
                    encoder_offset = 24.87;      % Encoder
reading of the scan in which stalk #1 in the origin;
                    encoder_correctionCoef = 640/645.18;
% (Ground truth distance between stalk #1 and
#25)/(Encoder reading difference between stalk #1 and
#25);
                case 2
                    flag = 0;  % used as validation set
%                       n_scans = 61.62;
%                       encoder_offset = 25.87;
%                       encoder_correctionCoef =
640/646.03;
                case 3
                    n_scans = 58.81;
                    encoder_offset = 26.87;
                    encoder_correctionCoef = 640/644.18;
                case 4
                    n_scans = 50.41;
                    encoder_offset = 25.64;
                    encoder_correctionCoef = 640/650.26;
                case 5
                    n_scans = 50.41;
                    encoder_offset = 25.87;
                    encoder_correctionCoef = 640/645.88;
            end

        case 6
            groundTruth = xlsread('LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 5-8', 'F34:F58');
% row 6
            groundTruth_offset = 131;    % row 6, %
Ground truth location of stalk #1 (cm); row2:105;
row3:97; row4:105;
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
```

```
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\06042012_plots5to8_V8\laserScanner_06042012_plot
s5to8_V8_5inAboveRoots', filename);

            switch idx_trial
               case 1
                  flag = 0;
               case 2
                  n_scans = 52.81;
                  encoder_offset = 19.00;
                  encoder_correctionCoef = 611/612.71;
               case 3
                  flag = 0;
               case 4
                  n_scans = 46.94;
                  encoder_offset = 19.64;
                  encoder_correctionCoef = 611/611.92;
               case 5
                  flag = 0;  % used as validation set
%                    n_scans = 61.61;
%                    encoder_offset = 20.10;
%                    encoder_correctionCoef =
611/612.00;
            end

         case 7
            groundTruth = xlsread('LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 5-8', 'F59:F83');
% row 7
            groundTruth_offset = 110;    % row 7
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\06042012_plots5to8_V8\laserScanner_06042012_plot
s5to8_V8_5inAboveRoots', filename);

            switch idx_trial
               case 1
                  n_scans = 53.72;
                  encoder_offset = 17.09;
                  encoder_correctionCoef = 634/640.41;
               case 2
                  flag = 0;
               case 3
                  n_scans = 45.24;
                  encoder_offset = 14.55;
                  encoder_correctionCoef = 634/637.86;
               case 4
                  flag = 0;  % used as validation set
%                    n_scans = 50.89;
%                    encoder_offset = 16.17;
%                    encoder_correctionCoef =
634/639.72;
               case 5
                  flag = 0;  % used as test set
%                    n_scans = 56.54;
%                    encoder_offset = 16.94;
%                    encoder_correctionCoef =
634/638.64;
            end

         otherwise
            groundTruth = xlsread('LCB_stalkdiameter_1-
2012_CorrectedCompleted.xlsx', 'rows 5-8',
'F84:F108');  % row 8
            groundTruth_offset = 115;    % row 8
            filename = [num2str(idx_row) '_3in_'
num2str(idx_trial) '.xlsx'];
            filePattern =
fullfile('D:\CornStalkDiameterProject\Data\Summer2012
\06042012_plots5to8_V8\laserScanner_06042012_plot
s5to8_V8_5inAboveRoots', filename);

            switch idx_trial
               case 1
                  n_scans = 62.19;
                  encoder_offset = 16.02;
                  encoder_correctionCoef = 491/491.18;
               case 2
                  n_scans = 58.53;
                  encoder_offset = 16.48;
                  encoder_correctionCoef = 491/492.42;
               case 3
                  flag = 0;  % used as validation set
%                    n_scans = 51.21;
%                    encoder_offset = 15.55;
%                    encoder_correctionCoef =
491/492.57;
               case 4
                  n_scans = 62.19;
                  encoder_offset = 13.24;
                  encoder_correctionCoef = 491/496.34;
               case 5
                  n_scans = 62.19;
                  encoder_offset = 13.86;
                  encoder_correctionCoef = 491/496.19;
            end

      end
      groundTruth = groundTruth';


      if flag  % If this trial data was valid, process it.

         % Read in measured data file.
         raw_data = xlsread(filePattern);
         num_data = size(raw_data,1);
         thres_array = n_scans.*ratio_n;
         disp(['Row ' num2str(idx_row) ', Trial '
num2str(idx_trial) ':']);

         for i_buf = 1:numel(buf_array)
            for i_thres = 1:numel(thres_array)

               %% Set up global variables
               num_stalk = 0;  % how many identified
stalks
               num_stalk_pending = 0; % how many
pending stalks
               pending_count = zeros(1);  % how many
show-up scans for each pending stalk
               pending_clusters = cell(1);
               num_noise = 0;  % how many identified
noise
               num_noise_pending = 0;  % how many
pending noise
               pending_noise_count = zeros(1);  % how
many show-up scans for each pending noise
               pending_noise = {};
               thres = thres_array(i_thres);
               buf = buf_array(i_buf);
               i = 0;    % i points to the raw data array;
               j = 0;    % # of scans
               X = [];
               Y = [];
               encoder_raw = [];
               encoder = [];
               X_ground = [];
```

```
D_thres_AllScans = [];
valid_C_members_AllScans = [];
core_C_AllScans = [];
span_C_AllScans  = [];
encoder = [];
stalkDiameter_AllStalks = [];
groundCoor_scan = [];
num_stalk = 0;
num_stalk_pending = 0;
pending_count = [];
pending_clusters = [];
num_noise = 0;
num_noise_pending = 0;
pending_noise_count = [];
pending_noise = [];
dif_mean_location = [];
min_dif_mean_location = [];
idx_min_dif_mean_location = [];
l = [];
gt = groundTruth;
missCt_idx = [];
missCt = [];
addCt_idx = [];
addCt = [];
t_error = 0;
t_error_perc = 0;


%% Processing the data
while i <= (num_data-1)
   i = i+1;
   if ~isnan(raw_data(i,2))  % If this row is
not the encoder reading...
      if max(raw_data(i:i+400,2)) < 1800  %
If no noise, read this scan;
         j = j+1;  % # of valid scan;
         Y(:,j) = raw_data(i:i+400,2)/10;
         X(:,j) = raw_data(i:i+400,1)/10;
         encoder_raw(j) =
raw_data(i+401,1)/10;
         encoder(j) =
encoder_raw(j)*encoder_countCoef;  % Encoder
reading in cm of scan #;
         encoder(j) = (encoder(j)-
encoder_offset)*encoder_correctionCoef+groundTruth_
offset;  % Corrected encoder reading;
         X_ground(:,j) = X(:,j)+encoder(j);
         i = i+401;  % point to next scan;
         clustering_yeyin_V062013(j, X(:,j),
Y(:,j));
         registration_yeyin(j);
      else
         i = i+401;  % If there is noise,
eliminate this scan;
      end
   end
end


%% All location measurements for each
stalk.
location_AllStalks = [];
for i = 1:numel(groundCoor_scan)
   if numel(core_C_AllScans{i}) ~= 0
      groundCoor_scan{i}(:,3) =
core_C_AllScans{i}(:,3);
      for ii = 1:size(groundCoor_scan{i},1)
         if groundCoor_scan{i}(ii,3) ~= -6 &&
groundCoor_scan{i}(ii,3) ~= -7 &&
groundCoor_scan{i}(ii,3) ~= 0
            if numel(location_AllStalks) <
groundCoor_scan{i}(ii,3)  % A newly shown-up stalk...

location_AllStalks{groundCoor_scan{i}(ii,3)} = [i
groundCoor_scan{i}(ii,1:2)];
            else   % Has already shown-up
in previous scans...

location_AllStalks{groundCoor_scan{i}(ii,3)} =
[location_AllStalks{groundCoor_scan{i}(ii,3)}; i
groundCoor_scan{i}(ii,1:2)];
            end
         end
      end
   end
end

mean_location_AllStalks =
zeros(1,num_stalk);
std_location_AllStalks =
zeros(1,num_stalk);
for i = 1:num_stalk
   % Compute mean of the interquartile of
location measurements:
   tmp = sort(location_AllStalks{i}(:,2));  %
Sort location readings
   median_tmp = median(tmp);   %
Compute median
   Q1 =
median(tmp(find(tmp<median_tmp)));   % Compute first
quartile
   Q3 =
median(tmp(find(tmp>median_tmp)));   % Compute
third quartile
   mean_location_AllStalks(i) =
mean(tmp(find(tmp>=Q1 & tmp<=Q3)));

   std_location_AllStalks(i) =
std(tmp(find(tmp>=Q1 & tmp<=Q3)));
   clear tmp median_tmp Q1 Q3
end


%% Sort the stalk indices
mean_location_AllStalks_sorted = [];
idx_sort = [];
[mean_location_AllStalks_sorted idx_sort]
= sort(mean_location_AllStalks);
location_AllStalks_sorted = cell(1,
numel(location_AllStalks));
stalkDiameter_AllStalks_sorted = cell(1,
numel(stalkDiameter_AllStalks));
for i = 1:numel(idx_sort)
   if i == idx_sort(i)
      location_AllStalks_sorted{i} =
location_AllStalks{i};
      stalkDiameter_AllStalks_sorted{i} =
stalkDiameter_AllStalks{i};
   else
      location_AllStalks_sorted{idx_sort(i)}
= location_AllStalks{i};

stalkDiameter_AllStalks_sorted{idx_sort(i)} =
stalkDiameter_AllStalks{i};
   end
end


% Mean diameter of each stalk
```

```
            mean_stalkDiameter_AllStalks_sorted =
zeros(numel(stalkDiameter_AllStalks_sorted),1);
            for i =
1:numel(stalkDiameter_AllStalks_sorted)
                mean_stalkDiameter_AllStalks_sorted(i)
= mean(stalkDiameter_AllStalks_sorted{i});
            end



            %% Combine locations closer than 5cm to
each other to eliminate sheath interference:
            mean_location_AllStalks_sorted_combined
= mean_location_AllStalks_sorted;
            dif_mean_location =
zeros(numel(mean_location_AllStalks_sorted_combine
d)-1,1);
            for i =
2:numel(mean_location_AllStalks_sorted_combined)
                dif_mean_location(i-1) =
mean_location_AllStalks_sorted_combined(i)-
mean_location_AllStalks_sorted_combined(i-1);
            end
            [min_dif_mean_location
idx_min_dif_mean_location] = min(dif_mean_location);
            while min_dif_mean_location <= 5
                % which means the stalk
#idx_dif_mean_location(i) and stalk
#idx_dif_mean_location(i+1) are closer than 5cm

mean_location_AllStalks_sorted_combined(idx_min_dif
_mean_location) =
(mean_location_AllStalks_sorted_combined(idx_min_di
f_mean_location)+mean_location_AllStalks_sorted_co
mbined(idx_min_dif_mean_location+1))/2;

mean_location_AllStalks_sorted_combined(idx_min_dif
_mean_location+1) = [];

                dif_mean_location =
zeros(numel(mean_location_AllStalks_sorted_combine
d)-1,1); % reset
                for i =
2:numel(mean_location_AllStalks_sorted_combined)
                    dif_mean_location(i-1) =
mean_location_AllStalks_sorted_combined(i)-
mean_location_AllStalks_sorted_combined(i-1);
                end
                [min_dif_mean_location
idx_min_dif_mean_location] = min(dif_mean_location);
            end



            %% Error Calculation
            disp(['Result when thres='
num2str(ratio_n(i_thres)) 'n and buf=' num2str(buf) ':']);
            l =
mean_location_AllStalks_sorted_combined;
            tmp = find((l>=(gt(1)-10)) &
(l<=(gt(end)+10)));
            l = l(tmp);
            clear tmp;
            l = [l' zeros(numel(l),1)];
            gt = [gt' zeros(numel(gt),1)];
            for i = 1:size(l,1)
                tmp = find(gt>=(l(i,1)-10) &
gt<=(l(i,1)+10));
                if numel(tmp) == 0
                    l(i,2) = -4;    % '-4' means this is an
adding count
                elseif numel(tmp) == 1  % this might be
a correct count or a adding count
                    l(i,2) = tmp;
                    gt(tmp,2) = 6;
                else   % it is among two or more gt
stalks
                    % see which one is closer:
                    num_tmp = numel(tmp);
                    tmp1 = pdist([l(i,1) 0; gt(tmp,:)]);
                    [na closest] = min(tmp1(1:num_tmp));
                    l(i,2) = tmp(closest);
                    gt(tmp(closest),2) = 6;
                    clear tmp1 closest;
                end
                clear tmp;
            end

            % Display missing counts
            missCt_idx = find(gt(:,2)==0);
            missCt = numel(missCt_idx);
            disp(['There are ' num2str(missCt) ' missing
counts at locations:   ' num2str(gt(missCt_idx,1)')]);

            % Display adding counts
            for i = 1:size(gt,1)
                tmp = find(l(:,2)==i);
                if numel(tmp) > 1  % there is/are adding
count(s)
                    % see which measured location is
closer to the ground truth
                    num_tmp = numel(tmp);
                    tmp1 = pdist([gt(i,1) 0; l(tmp,1)
zeros(num_tmp,1)]);
                    [na closest] = min(tmp1(1:num_tmp));
                    tmp(closest) = [];
                    l(tmp,2) = zeros(numel(tmp),1);
                end
                clear tmp1 tmp closest;
            end
            addCt_idx = find(l(:,2)==0 | l(:,2)==-4);
            addCt = numel(addCt_idx);
            disp(['There are ' num2str(addCt) ' adding
counts at locations:   ' num2str(l(addCt_idx,1)')]);

            % Display total error
            t_error = missCt+addCt;
            t_error_perc = t_error/size(gt,1)*100;
            missCt_perc = missCt/size(gt,1)*100;
            addCt_perc = addCt/size(gt,1)*100;
            disp(['The total error are ' num2str(t_error) '
counts, ' num2str(t_error_perc) '%.']);

            % RMSE of locations
            ll = l(:,1);
            ll(addCt_idx) = [];
            gg = gt(:,1);
            gg(missCt_idx) = [];
            rmse = sqrt(sum((ll-gg).^2)/numel(ll));
            disp(['RMSE is ' num2str(rmse)]);
            if rmse > 2.54
                rmse_perc = (rmse-2.54)/2.54*100;
            else
                rmse_perc = 0;
            end

            % Value of the objective function
y=weight1*t_error_perc+weight2*(rmse/10)
            % %weight_t_error_perc = 1;
            weight_missCt_perc = 2;%1;
            weight_addCt_perc = 1;%2;
```

```
            weight_rmse = 0.75;%0.25;
            OF(i_buf, i_thres, 5*(idx_row-1)+idx_trial) =
weight_missCt_perc*missCt_perc+weight_addCt_perc*
addCt_perc+weight_rmse*rmse_perc;
            disp(['The value of the objective function is:
' num2str(OF(i_buf, i_thres, 5*(idx_row-1)+idx_trial))
'.']);


            %% Plot:
            %       figure; bar(l(:,1), ones(1,size(l,1)),
0.4, 'c', 'EdgeColor', 'c');
            %       hold on; bar(gt(:,1),
0.5.*ones(size(gt,1),1), 0.4, 'k');
            %       grid on; axis([0 800 0 1]);
xlabel('Location (cm)');  % row 5
            %       % grid on; axis([0 850 0 1]);
xlabel('Location (cm)'); % row 6
            %       % grid on; axis([0 850 0 1]);
xlabel('Location (cm)');  % row 7
            %       % grid on; axis([0 700 0 1]);
xlabel('Location (cm)');  % row 8
            %       title(['Ground Truth Locations &
Measured Locations of file ' filename ' at thres '
num2str(thres) ' and buf ' num2str(buf)]);

            %       figure;
bar(1:1:numel(mean_location_AllStalks),std_location_A
llStalks(1:numel(mean_location_AllStalks)));
            %       axis([0 35 0 10]); grid on;
            %       title(['Standard deviation of location
of file ' filename ' V8']);
            %       xlabel('Measured Stalk #');
ylabel('Standard deviation (cm)');
            %       figure; plot(1:1:numel(encoder),
encoder);
            %       title(['Modified encoder reading of '
filename ' V8']);
            %       axis equal; grid on; xlabel('Scan #');
ylabel('Encoder Readings (cm)');

            % Spacing calculation ----------------------
---------------------------------------
            % ll is the sensor measured locations
without the adding counts; gg is the ground truth
locations without the missing counts
            spacing_gt = zeros(size(gg,1)-1,1);
            spacing_sensor = zeros(size(ll,1)-1,1);
            for i_spacing = 1:(size(ll,1)-1)
                spacing_gt(i_spacing) =
gg(i_spacing+1)-gg(i_spacing);
                spacing_sensor(i_spacing) =
ll(i_spacing+1)-ll(i_spacing);
            end
            spacing_gt = spacing_gt';
            spacing_sensor = spacing_sensor';

            rmse_spacing =
sqrt(sum((spacing_sensor-
spacing_gt).^2)/numel(spacing_gt));
            disp(['RMSE of spacing measurement: '
num2str(rmse_spacing)]);
            disp('----------------------------------------------
--------------------------------');


        end
    end

else
        disp(['Row ' num2str(idx_row) ', Trial '
num2str(idx_trial) ' was invalid.']);
        disp('----------------------------------------------------
-----------------------');
    end
  end
end

OF_sum = sum(OF,3);


% %% Individual scan display for testing purpose:
% close all;
% for i_scan = 55:1:58
% %     figure, scatter(X_ground(:,i_scan), Y(:,i_scan));
%     figure, scatter(X(:,i_scan), Y(:,i_scan));
%     axis equal; grid on;
%     xlabel('Scan Line (cm)'); ylabel('Range (cm)');
%     title([num2str(filename) ' scan # '
num2str(i_scan)]);
% end
```

Selected MATLAB Code for Chapter V

LiDAR-and-RGB Approach

Main Function

```
% This code is the main function for processing the
webcam and laser data.
% Yeyin Shi, Sept 2013


close all;
clear all; clc;

%% Define global variables
global num_stalk dia_cm_AllStalks core_C_AllScans;
global encoder ind_laser;
global buf;
global stalkDiameter_AllStalks;



%% Read all RGB images in the folder
imageFolder = 'C:\StalkDiameterProject\Field Test
Data\Summer2013\06132013_V12\2_1_V12_webcam';

if ~isdir(imageFolder)
    errorMessage = sprintf('Error: The following folder
does not exist:\n%s', imageFolder);
    uiwait(warndlg(errorMessage));
    return;
end
filePattern = fullfile(imageFolder, '*.bmp');
listing=dir(filePattern);
nfiles = numel(listing); % Number of images in the
folder
filenames = cell(nfiles,1);
timeStamps_rgb = zeros(nfiles,1);
for i = 1:nfiles
    filenames{i} = listing(i).name;
    timeStamps_rgb(i) = str2double(filenames{i}(1:8));
end



% Sort timeStamps_rgb and filenames into ascending
order
timeStamps_rgb2 = timeStamps_rgb;
[timeStamps_rgb IX] = sort(timeStamps_rgb2);
clear timeStamps_rgb2;

filenames2 = filenames;
filenames = cell(numel(filenames2),1);
for i = 1:numel(filenames)
    filenames{i} = filenames2{IX(i)};
end



%% Extract each laser scan first (used for off-line
processing)
num_stalk = 0;  % how many identified stalks
buf = 6;

load raw_data_2_1_V12;
load dia_gt_2_V12;
load location_gt_2;
num_data = size(raw_data,1);
encoder_countCoef = 0.77;  % Encoder reading
correction coefficient (mm/count);
i = 0;
j = 0;

while i <= (num_data-1)
    i = i+1;  % pointer to each row of raw data
    if ~isnan(raw_data(i,2))  % If this row is not the
encoder reading...
        if max(raw_data(i:i+400,2)) < 1800  % If no noise,
read this scan;
            j = j+1;  % # of valid scan;
            Y(:,j) = raw_data(i:i+400,2)/10;
            X(:,j) = raw_data(i:i+400,1)/10;
            encoder_raw(j) = raw_data(i+401,1)/10;
            encoder(j) =
encoder_raw(j)*encoder_countCoef;  % Encoder
reading in cm of scan #;
            timeStamp_laser(j) = raw_data(i+402,1);  %
Time stamp of each scan
            X_ground(:,j) = X(:,j)+encoder(j);
            i = i+401;  % point to next scan;
            clustering_webcamLaser(j, X(:,j), Y(:,j));
        else
            i = i+401;  % If there is noise, eliminate this
scan;
        end
    end
end

theta_rgb = 62; % webcam's total field of view in degree
c = 2*(tan(theta_rgb/180*pi/2))/640; % coefficient used
later for diameter calculation



%% Laser and webcam matching
for i_rgb = 2:nfiles

    % Note: The time stamp of each webcam frame was
actually closer to the time
    % of the next frame been taken. So use the time
stamp of previous frame
    % as the time stamp of current frame.
    rgbImg_orig = imread(fullfile(imageFolder,
filenames{i_rgb-1}));
%     figure; imshow(rgbImg_orig); title(['Image '
filenames{i_rgb-1}]);
    rgbImg = rgbImg_orig;

    % Find corresponding laser scan with closest time
stamp
    ff = 1;  % flag indicating if a corresponding laser scan
can be found
    ind = find(
(timeStamp_laser>=timeStamps_rgb(i_rgb)-0.05) &
(timeStamp_laser<=timeStamps_rgb(i_rgb)+0.05) );
    if numel(ind) == 0
        ind = find(
(timeStamp_laser>=timeStamps_rgb(i_rgb)-0.1) &
(timeStamp_laser<=timeStamps_rgb(i_rgb)+0.1) );
        if numel(ind) == 0
            disp(['No laser scan was found close to image '
filenames{i_rgb-1} '. Processing next image...']);
            ff = 0;
        end
    end
```

```
   if ff  % if one or more corresponding laser scan was
found...
      % find a laser scan with closest time stamp:
      for i = 1:numel(ind)
         dif(i) = abs(timeStamp_laser(ind(i))-
timeStamps_rgb(i_rgb));
      end
      [min_dif idx_min_dif] = min(dif);
%        ind_laser = ind(idx_min_dif);   % ind_laser is the
index of laser scan corresponding to the rgb image
      ind_laser(i_rgb-1) = ind(idx_min_dif);
      clear dif min_dif idx_min_dif;
%        figure; scatter(X(:,ind_laser(i_rgb-1)),
Y(:,ind_laser(i_rgb-1)));
%        axis equal; grid on;
%        xlabel('Scan Line (cm)'); ylabel('Range (cm)');
%        title(['Laser scan #' num2str(ind_laser(i_rgb-1)) '
corresponding to RGB image ' filenames{i_rgb-1}]);



      %% Get stalk recognition results from the RGB
image:


      % Vegetative area segmentation:
      bwImg = zeros(360,640);
      for i = 1:360
         for j = 1:640
            R = rgbImg(i,j,1);
            G = rgbImg(i,j,2);
            B = rgbImg(i,j,3);
            if (R<=200) && (G<=200) && (B<=200)
               if R<G && B<G
                  bwImg(i,j) = 1;
               end
            end
         end
      end
%        figure; imshow(bwImg); title('Primary
Segmentation');
      img_current = bwImg;
      clear R G B i j;


      CC = bwconncomp(img_current);
      numPixels = cellfun(@numel,CC.PixelIdxList);
      [biggest,idx] = max(numPixels);
      tmp = zeros(size(bwImg));
      tmp(CC.PixelIdxList{idx}) = 1;
      img_current = tmp;
%        figure; imshow(img_current); title('Largest
connected area');
      clear CC numPixels biggest idx tmp;


      % Fill in small holes in the binary image:
      tmp = ones(size(img_current))-img_current;
      CC = bwconncomp(tmp,4);
      numPixels = cellfun(@numel,CC.PixelIdxList);
      hold_fill_thres = 100;
      idx = find(numPixels<=hold_fill_thres);
      for i = 1:numel(idx)
         tmp(CC.PixelIdxList{idx(i)})=0;
      end
      tmp = ones(size(tmp))-tmp;
      img_current = tmp;
%        figure; imshow(img_current); title(['Hole-filling
threshold ' num2str(hold_fill_thres)]);
      clear tmp CC numPixels idx;


      % Diameter calculation:
      thres_dia1 = 20;  % number of pixels corresponds
to minimum stalk diameter
      thres_dia2 = 80;
      if max(max(img_current))  % if there is potential
stalk object(s)
         sum_img = sum(img_current);
         % figure; plot(1:1:640, sum_img); title('Sum of
binary values along vertical axis');
         % grid on; axis([0 700 0 400]);
         % xlabel('Horizontal Dimension (pixel)');
         % ylabel('Sum along column');


         ind_pixel = find(sum_img>=360);  % Check if
the connected part(s) is vertically across the image


         if (numel(ind_pixel)>=thres_dia1)
            num_stalk_oneImg = 1;  %
num_stalk_oneImg is the number of stalks in an image
            dia = 1; % dia is the width of the valley or the
diameter of a stalk
            ind_eachStalk = {ind_pixel(1)};
            for j = 2:numel(ind_pixel)
               if ind_pixel(j) == ind_pixel(j-1)+1;  % if two
pixels are connected...
                  dia(num_stalk_oneImg) =
dia(num_stalk_oneImg)+1;  % count them as one stalk
                  ind_eachStalk{num_stalk_oneImg} =
[ind_eachStalk{num_stalk_oneImg} ind_pixel(j)];
               else  % if two pixels are not connected...
                  num_stalk_oneImg =
num_stalk_oneImg+1;  % treat it as another stalk
                  dia(num_stalk_oneImg) = 1;  % initialize
another stalk diameter counting
                  ind_eachStalk = [ind_eachStalk;
ind_pixel(j)];  % initialize another stalk cell
               end
            end


            % eliminate too small object(s) and object(s)
near the image edge
            num_stalk_oneImg2 = 0;
            ind_eachStalk2 = {};
            dia2 = [];
            for i = 1:num_stalk_oneImg
               % if an object is wide enough to be a
potential stalk:
               if (numel(ind_eachStalk{i}) >= thres_dia1)
&& (numel(ind_eachStalk{i}) <= thres_dia2)  % check
diameter
                  % if a stalk appears at or near the edge
of an image:
                  tmp =
find((ind_eachStalk{i}<=4)|(ind_eachStalk{i}>=636));
                  if numel(tmp) == 0
                     if num_stalk_oneImg2 == 0
                        num_stalk_oneImg2 = 1;
                     else
                        num_stalk_oneImg2 =
num_stalk_oneImg2+1;
                     end
                     ind_eachStalk2{num_stalk_oneImg2}
= ind_eachStalk{i};
                     dia2(num_stalk_oneImg2) = dia(i);
                  end
               end
```

```
        end
        num_stalk_oneImg = num_stalk_oneImg2;
        ind_eachStalk = ind_eachStalk2;
        dia = dia2;  % 'dia' the pixel width of each
stalk
        clear tmp num_stalk_oneImg2
ind_eachStalk2 dia2;

        if num_stalk_oneImg~=0

            location_rgb =
zeros(numel(ind_eachStalk),1);
            for i = 1:numel(ind_eachStalk)
                location_rgb(i) =
mean(ind_eachStalk{i});
            end
            % Use the rgb image to calculate the
diameter; using the laser scan to get the distance the
stalk from the sensor.



            % Get the clustering results here
            core_currentScan =
core_C_AllScans{ind_laser(i_rgb-1)};  % Cluster
locations in the corresponding laser scan
            j = 0;  % Initialize stalk count in this image
            dia_cm_currentImg = [];

            for i = 1:size(core_currentScan,1)

                tmp_x = core_currentScan(i,1);
                tmp_y = core_currentScan(i,2);
                tmp_theta = atan(tmp_x/tmp_y)/pi*180;
                if abs(tmp_theta)<=(66/2)
                    % Eliminate this cluster if it is located
larger than 66 degrees which would definitely out of the
webcam's field of view

                    [M_pixelInRgb M M_idx] =
sensorMatching(tmp_x,tmp_y);
                    % M_pixelInRgb is the pixel index in
RGB image;
                    % M is the laserPositionInRgbImg;
                    % M_idx is the data point index in the
laser scan

                    if numel(M) ~= 0

                        % 70 are the buffered search area
                        if (M(M_idx)-70>=1) &&
(M(M_idx)+70)<=640
                            tmp_matchIdx = find(
(location_rgb>=M(M_idx)-70) &
(location_rgb<=M(M_idx)+70) );
                        elseif (M(M_idx)-70<1)
                            tmp_matchIdx = find(
(location_rgb>=1) & (location_rgb<=M(M_idx)+70) );
                        else
                            tmp_matchIdx = find(
(location_rgb>=M(M_idx)-70) & (location_rgb<=640) );
                        end

                        if numel(tmp_matchIdx)~=0 % If a
corresponding stalk in the RGB image can be found...

                            % If more than one
corresponding stalk were found in the RGB image,
                            % keep one closer to the laser
estimated location
                            if numel(tmp_matchIdx)>1
                                tmp1 = M_pixelInRgb;
                                for i_tmp =
1:numel(tmp_matchIdx)
                                    tmp2 =
location_rgb(tmp_matchIdx(i_tmp));
                                    tmp_dif(i_tmp) = abs(tmp1-
tmp2);
                                end
                                [tmp_dif_min tmp_idx] =
min(tmp_dif);
                                tmp_matchIdx =
tmp_matchIdx(tmp_idx);
                            end
                            clear tmp1 tmp2 tmp_dif
tmp_dif_min tmp_idx;

                            j = j+1; % Increments stalk count;
                            % Then tmp_y is the
distance/sqrt(3)*2 of the stalk at
location_rgb(tmp_matchIdx) in the RGB image:
                            dis = tmp_y/2*sqrt(3);
                            dia_cm(tmp_matchIdx) =
c*(dia(tmp_matchIdx)+5)*dis*cos(0.85/180*pi); %
/cos(0.85/180*pi)was the incline angle correction



                            % 'dia_cm_currentImg' stores all
the information about recognized stalks in current RGB
image and laser scan:
                            dia_cm_currentImg(j,1:2) =
core_currentScan(i,1:2); % column 1:2 stores verified
stalk locations in laser scan;
                            dia_cm_currentImg(j,3) =
M_pixelInRgb; % column 3 stores estimated stalk
locations in rgb image from laser scan;
                            dia_cm_currentImg(j,4) =
location_rgb(tmp_matchIdx); % column 4 stores
recognized stalk locations in rgb image;
                            dia_cm_currentImg(j,5) =
tmp_matchIdx; % column 5 stores which stalk in rgb
image matched to the cluster in laser scan;
                            dia_cm_currentImg(j,6) =
dia_cm(tmp_matchIdx); % column 6 stores stalk
diameters

                        end

                    end
                    clear M_pixelInRgb M M_idx
tmp_matchIdx;

                end
                clear tmp_x tmp_y tmp_theta;

            end
            clear location_rgb core_currentScan
num_stalk_oneImg ind_eachStalk dia_cm;

            % if two clusters in laser scan correspond
to the same stalk in rgb image, eliminate the one further
away from the stalk location in rgb image
            if numel(dia_cm_currentImg) ~= 0
                if size(dia_cm_currentImg,1)>1
                    i = 1;
                    while i<=size(dia_cm_currentImg,1)
```

```
                    tmp_sameStalk =
find(dia_cm_currentImg(:,5)==dia_cm_currentImg(i,5));
                    if numel(tmp_sameStalk)>1 %
which means there are multiple clusters corresponded
to one stalk...
                        [tmp_dif tmp_closest] =
min(abs(dia_cm_currentImg(tmp_sameStalk,3)-
dia_cm_currentImg(tmp_sameStalk,4)));
                        tmp_sameStalk(tmp_closest) =
[];
                        for ii = 1:numel(tmp_sameStalk)

dia_cm_currentImg(tmp_sameStalk(numel(tmp_sameS
talk)-ii+1),:) = []; % Eliminate multiple clusters.
                        end
                    end
                    i = i+1;
                end
            end
            clear i ii tmp_sameStalk tmp_dif
tmp_closest;


            % Mark the estimated position on RGB
image:
            figure; imshow(rgbImg_orig);
title(['Image ' filenames{i_rgb-1}]);
            for i = 1:size(dia_cm_currentImg,1)
                x_plot =
ones(360,1).*dia_cm_currentImg(i,3);
                y_plot = 1:1:360;
                hold on; plot(x_plot, y_plot, 'r');
            end



            % Assign results to the global variable:
            if ~exist('dia_cm_AllStalks','var')
                dia_cm_AllStalks = cell(i_rgb-1,1);
                dia_cm_AllStalks{i_rgb-1} =
zeros(1,7);
            end
            if numel(dia_cm_AllStalks)<i_rgb-1
                dia_cm_AllStalks = [dia_cm_AllStalks;
cell(i_rgb-1-numel(dia_cm_AllStalks),1)];
                dia_cm_AllStalks{i_rgb-1} =
zeros(size(dia_cm_currentImg,1),6);
            end


            dia_cm_AllStalks{i_rgb-1}(:,1:2) =
dia_cm_currentImg(:,1:2); % dia_cm_AllStalks{}(:,1:2)
stores verified stalk locations in laser scan;
            dia_cm_AllStalks{i_rgb-1}(:,3) =
dia_cm_currentImg(:,3); % column 3 stores estimated
stalk locations in rgb image from laser scan;
            dia_cm_AllStalks{i_rgb-1}(:,4) =
dia_cm_currentImg(:,4); % column 4 stores recognized
stalk locations in rgb image;
            dia_cm_AllStalks{i_rgb-1}(:,5) =
dia_cm_currentImg(:,6); % column 5 stores stalk
diameters;
            % dia_cm_AllStalks{}(:,6) will be used
later for registration purpose.
            % dia_cm_AllStalks{}(:,7) will be the
ground coordinate locations.
        end

        clear dia_cm_currentImg;

    end
```

```
        end

    end

end


%% Register RGB images based on the encoder
readings and the feature (stalk locations) in each image
for i = 1:numel(dia_cm_AllStalks)
    registration_webcamLaser(i);
end



%% Location calculation
location_AllStalks = [];
for i = 1:numel(dia_cm_AllStalks)
    if numel(dia_cm_AllStalks{i}) ~= 0
        for ii = 1:size(dia_cm_AllStalks{i},1)
            if dia_cm_AllStalks{i}(ii,6) ~= -6 &&
dia_cm_AllStalks{i}(ii,6) ~= -7 &&
dia_cm_AllStalks{i}(ii,6) ~= 0
                if numel(location_AllStalks) <
dia_cm_AllStalks{i}(ii,6)  % A newly shown-up stalk...
                    location_AllStalks{dia_cm_AllStalks{i}(ii,6)}
= [i dia_cm_AllStalks{i}(ii,7)];
                else   % Has already shown-up in previous
scans...
                    location_AllStalks{dia_cm_AllStalks{i}(ii,6)}
= [location_AllStalks{dia_cm_AllStalks{i}(ii,6)}; i
dia_cm_AllStalks{i}(ii,7)];
                end
            end
        end
    end
end
mean_location_AllStalks = zeros(1, num_stalk);
for i = 1:num_stalk
    mean_location_AllStalks(i) =
mean(location_AllStalks{i}(:,2));  % Compute mean of
locations for each stalk;
    % %std_location_AllStalks(i) =
std(location_AllStalks{i}(:,2));  % Compute standard
deviation of locations for each stalk;
end

load location_gt_2;
location_gt = location_gt-
(location_gt(1))+min(mean_location_AllStalks);
length_gt = max(location_gt)-min(location_gt);
length_encoder = max(mean_location_AllStalks)-
min(mean_location_AllStalks);
encoder_ratio = length_gt/length_encoder;
mean_location_AllStalks1 = (mean_location_AllStalks-
min(mean_location_AllStalks)).*encoder_ratio+min(me
an_location_AllStalks);

figure; bar(mean_location_AllStalks1,
ones(1,numel(mean_location_AllStalks1)), 0.4, 'c',
'EdgeColor', 'c');
figure; bar(location_gt, 0.5.*ones(1, numel(location_gt)),
0.4, 'k');
```

```
figure; bar(mean_location_AllStalks1,
ones(1,numel(mean_location_AllStalks1)), 0.4, 'c',
'EdgeColor', 'c');
hold on; bar(location_gt, 0.5.*ones(1,
numel(location_gt)), 0.4, 'k');
title('Ground Truth Locations & Measured Locations');
% rmse_location =
sqrt(sum((mean_location_AllStalks1-
location_gt).^2)/numel(mean_location_AllStalks))



%% Diameter calculation
load dia_gt_2_V12;
n_tmp = numel(stalkDiameter_AllStalks);
count = [];
non_stalk = [];
n_stalk = 0;
thres_count = 10;
dia_cm_mean = [];

for i = 1:n_tmp
    count(i) = numel(stalkDiameter_AllStalks{i});
     if count(i) < thres_count
         non_stalk = [non_stalk i];
     else
        n_stalk = n_stalk+1;
        dia_cm_mean(n_stalk) =
mean(stalkDiameter_AllStalks{i})*10;
     end
end
% figure; bar(1:1:n_tmp,count); grid on; title('Number of
images of each stalk');
clear n_tmp;

figure;
scatter(1:1:numel(dia_cm_mean),dia_gt,'g','filled');
axis([0 25 0 50]); grid on;
hold on;
scatter(1:1:numel(dia_cm_mean),dia_cm_mean, 'k', '*');
xlabel('Stalk Index'); ylabel('Diameter (mm)');
title('Row 2 rep 2 at V12');

% rmse = sqrt(sum((dia_cm_mean-
dia_gt').^2)/numel(dia_cm_mean))

% ---------------------------------------------------------------------
--------
%% Display individual scan for testing purpose
close all;
for i_test = 1:5:20
    % Display RGB image
    rgbImg_orig = imread(fullfile(imageFolder,
filenames{i_test}));
    figure; imshow(rgbImg_orig); title(['#' num2str(i_test) '
' 'Image ' filenames{i_test}]);

    % Display laser scan
    figure; scatter(X(:,ind_laser(i_test)),
Y(:,ind_laser(i_test)));
    % %figure; scatter(X(:,i_test), Y(:,i_test));
    axis equal; grid on;
    xlabel('Scan Line (cm)'); ylabel('Range (cm)');
    title(['Laser scan #' num2str(ind_laser(i_test)) '
corresponding to RGB image ' filenames{i_test}]);
%     title(['Laser scan #' num2str(i_test)]);
    x = 0:0.1:40;
    y = x.*(tan(59/180*pi))+5.5;
    hold on; plot(x,y,'r');
    x = -40:0.1:0;
    y = x.*(tan(121/180*pi))+5.5;
    hold on; plot(x,y,'r');
end
```

3D Range Imaging Approach

Main Function

```
close all;
clear all; clc;


%% Global variables definition
global Info1 Info2;
global num_stalk;
global buf;
global encoder;
global stalkDiameter_AllStalks;

num_stalk = 0;
buf = 15;
stalkDiameter_AllStalks = {};

c = 2*tan(43.6/2*pi/180)/176;

%% Distortion correction
loading_calib;

raw = xlsread('C:\StalkDiameterProject\Field Test
Data\Summer2013\06132013_V12\06132013_V12_sr4
000\2_1_V12.xlsx');
rows = size(raw,1);
num_img = rows/(144+1);
img_orig = zeros(144,176,num_img);
img_rec = zeros(144,176,num_img);
encoder = zeros(1,num_img);
timeStamp = zeros(num_img);
i = 1;

while (145*(i-1)+1) <= rows
    img_orig(:,:,i) = raw(145*(i-1)+1:145*(i-1)+144,:);
    encoder(i) = raw(145*(i-1)+145,1)/10;
    timeStamp(i) = raw(145*(i-1)+145,2);
    phy = bitshift(img_orig(:,:,i), -2);
    img_orig(:,:,i) = phy/(2^14)*500;
    % % figure; imagesc(img_orig(:,:,i)); axis off;
title('Raw distance image');
    img_rec(:,:,i) = rect(img_orig(:,:,i),eye(3),fc,cc,kc,KK);
% img_rec contains images after distortion correction
    % % figure; imagesc(img_rec(:,:,i)); axis off;
title('Undistorted distance image');
    save('UndistortedImg.mat', 'img_rec');
    i = i+1;
end
img = img_rec;
% figure; plot(1:1:numel(encoder),encoder);

%% Thresholding
img_thres = zeros(144,176,num_img);
img_bw = zeros(144,176,num_img);
thres1 = 25;  % 30 is the minimum distance between
sensor and plant row; add 5 tolerance
thres2 = 51;  % 46 is the maximum distance between
sensor and plant row; add 5 tolerance
for i = 1:num_img
```

```
    tmp = img(:,:,i);  % temporarily stores the thresholded
image
    tmp1 = ones(144,176); % temporarily stores the
binary image
    for ii = 1:144
        for jj = 1:176
            if (tmp(ii,jj) > thres2) || (tmp(ii,jj) < thres1)
                tmp(ii,jj) = 500;
                tmp1(ii,jj) = 0;
            end
        end
    end
    img_thres(:,:,i) = tmp;  % img_thres contains images
after thresholding
    img_bw(:,:,i) = tmp1;
end
% % img = img_thres;  % 'img' stores all the
thresholded images
img = img_bw;  % 'img' stores all the binary images

%% Find the connected part(s) in an image
Info1 = cell(num_img,1);
Info2 = cell(num_img,1);
size_thres = 1500;  % A connected part needs to be
larger than 1500 pixels in order to be valid.
for i_img = 1:num_img

    clear img_current img_current2 CC numPixels
biggest idx;
    img_current = img(:,:,i_img);
    CC = bwconncomp(img_current);
    numPixels = cellfun(@numel,CC.PixelIdxList);
    [biggest,idx] = max(numPixels);
    compo = {};
    img_current2 = zeros(size(img_current));
    while biggest >= size_thres
        tmp = zeros(size(img_current));
        tmp(CC.PixelIdxList{idx}) = 1;
        tmp1 = tmp;
        sum_tmp1 = sum(tmp1,2);
        % figure; plot(1:1:144, sum_tmp1);
        min_sum_tmp1 = min(sum_tmp1(10:130));
        if min_sum_tmp1 > 0
            compo = [compo; CC.PixelIdxList{idx}];  %
'compo' contains all large enough connected areas
            img_current2(compo{end}) = 1;
        end
        numPixels(idx) = 1;  % Delete the largest part from
current 'numPixels'
        [biggest,idx] = max(numPixels);
    end
    clear tmp tmp1 sum_tmp1 min_sum_tmp1;
    % figure; imagesc(img_current2); axis equal; axis([0
176 0 144]);

    % Convert 'compo' to Cartesian coordinates
'compo_C'.

    % Fill in small holes in the binary image:
    img_current3 = ones(144,176)-img_current2;
    CC_3 = bwconncomp(img_current3,4);
    numPixels_3 = cellfun(@numel,CC_3.PixelIdxList);
    hold_fill_thres = 200;
    idx_3 = find(numPixels_3<=hold_fill_thres);
    for i = 1:numel(idx_3)
        img_current3(CC_3.PixelIdxList{idx_3(i)})=0;
    end
    % % figure; imagesc(img_current3); title(['Hole-filling
threshold ' num2str(hold_fill_thres)]);
    img_current2 = ones(144,176)-img_current3;
    % figure; imagesc(img_current2);

    % Detect stalk object:
    if max(max(img_current2))  % if there is potential
stalk object(s)
        sum_img = sum(img_current2);
        % figure; plot(1:1:176, sum_img); title('Sum of
binary values along vertical axis');
        % grid on; axis([0 180 0 150]); axis equal;
        % xlabel('Horizontal Axis (pixel)');
        % ylabel('Sum along column');
        % hold on; plot(1:1:176, ones(1,176).*72);

        % ind = find(sum_img>=72);
        ind = find(sum_img>=144);  % Check if the
connected part(s) is vertically across the image

        thres_dia1 = 10;  % number of pixels corresponds
to minimum stalk diameter
        thres_dia2 = 30;
        if (numel(ind)>=thres_dia1) % % &&
(numel(ind)<=thres_dia2)
            num_stalk_oneImg = 1;  % num_stalk_oneImg
is the number of stalks in an image
            j = 1;
            dia = [1]; % dia is the width of the valley or the
diameter of a stalk
            ind_eachStalk = {ind(1)};
            for j = 2:numel(ind)
                if ind(j) == ind(j-1)+1;  % if two pixels are
connected...
                    dia(num_stalk_oneImg) =
dia(num_stalk_oneImg)+1;  % count them as one stalk
                    ind_eachStalk{num_stalk_oneImg} =
[ind_eachStalk{num_stalk_oneImg} ind(j)];
                else  % if two pixels are not connected...
                    num_stalk_oneImg =
num_stalk_oneImg+1;  % treat it as another stalk
                    dia(num_stalk_oneImg) = 1;  % initialize
another stalk diameter counting
                    ind_eachStalk = [ind_eachStalk; ind(j)];  %
initialize another stalk cell
                end
            end

            % eliminate too small object(s) and object(s)
near the image edge
            num_stalk_oneImg2 = 0;
            ind_eachStalk2 = {};
            dia2 = [];
            for i = 1:num_stalk_oneImg
                % if an object is wide enough to be a
potential stalk:
                if (numel(ind_eachStalk{i}) >= thres_dia1) &&
(numel(ind_eachStalk{i}) <= thres_dia2)  % check
diameter
                    % if a stalk appears near the edge of an
image:
                    clear tmp;
                    tmp =
find((ind_eachStalk{i}<=4)|(ind_eachStalk{i}>=172));
                    if numel(tmp) == 0

                        if num_stalk_oneImg2 == 0
                            num_stalk_oneImg2 = 1;
                        else
                            num_stalk_oneImg2 =
num_stalk_oneImg2+1;
                        end
```

```
                ind_eachStalk2{num_stalk_oneImg2} =
ind_eachStalk{i};
                dia2(num_stalk_oneImg2) = dia(i);
              end
            end
        end
        num_stalk_oneImg = num_stalk_oneImg2;
        ind_eachStalk = ind_eachStalk2;
        dia = dia2;
        clear num_stalk_oneImg2 ind_eachStalk2 dia2;

        % Correct diameter estimation based on the
distance reading
        tmp = img_thres(:,:,i_img);
        for i = 1:num_stalk_oneImg
          tmp_sum = 0;
          tmp_idx = 0;
          for ii = 1:numel(ind_eachStalk{i})
            for jj = 1:144
              % Because the step of hole fill-in, those
noise pixels
              % need to be excluded from the
distance calculation
              % here:
              if tmp(jj,ind_eachStalk{i}(ii)) ~= 500
                tmp_sum =
tmp_sum+tmp(jj,ind_eachStalk{i}(ii));
                tmp_idx = tmp_idx+1;
              end
            end
          end
          d = tmp_sum/tmp_idx;
          dia_c(i) = c*dia(i)*d*10;  % 'dia_c' is the
corrected diameter in mm according to distance reading
'd'
        end
        clear tmp tmp_sum tmp_idx ii jj;

        % sort ind_eachStalk
        locale_stalk = zeros(num_stalk_oneImg,1);
        for i = 1:num_stalk_oneImg
          locale_stalk(i) = mean(ind_eachStalk{i});  %
location of each stalk in the image coordinates
        end

        [tmp, IX] = sort(locale_stalk,'descend');
        info1 = cell(num_stalk_oneImg,1);
        info2 = zeros(num_stalk_oneImg,4);
        for i = 1:num_stalk_oneImg
          info1{i} = ind_eachStalk{IX(i)};  % horizontal
pixel indices of all members in a stalk object
          info2(i,1) = dia_c(IX(i));  % width of each stalk
object
          info2(i,2) = locale_stalk(IX(i));  % location (in
pixels) of each stalk in the image
          info2(i,3) = 0;
          info2(i,4) = c*info2(i,2)*d+encoder(i_img);
        end


        Info1{i_img} = info1;
        Info2{i_img} = info2;
        clear tmp d dia dia_c info1 info2 locale_stalk;
        clear ind_eachStalk num_stalk_oneImg;
        clear IX sum_img ind;
        clear CC numPixels biggest idx compo
img_current2;

      end
    end

end

%% Register with previous image
for i = 1:numel(Info2)
  registration_sr4000(i);
end

%% Location and spacing measurement
location_AllStalks = [];
for i = 1:numel(Info2)
  if size(Info2{i},1) ~= 0
    for ii = 1:size(Info2{i},1)
      if Info2{i}(ii,3) ~= -6 && Info2{i}(ii,3) ~= -7 &&
Info2{i}(ii,3) ~= 0
        if numel(location_AllStalks) < Info2{i}(ii,3)  %
A newly shown-up stalk...
          location_AllStalks{Info2{i}(ii,3)} = [i
Info2{i}(ii,4)];
        else   % Has already shown-up in previous
scans...
          location_AllStalks{Info2{i}(ii,3)} =
[location_AllStalks{Info2{i}(ii,3)}; i Info2{i}(ii,4)];
        end
      end
    end
  end
end
mean_location_AllStalks = zeros(1, num_stalk);
for i = 1:num_stalk
  mean_location_AllStalks(i) =
mean(location_AllStalks{i}(:,2));  % Compute mean of
locations for each stalk;
end

load location_gt_2;

location_gt = location_gt-
(location_gt(1))+min(mean_location_AllStalks);
length_gt = max(location_gt)-min(location_gt);
length_encoder = max(mean_location_AllStalks)-
min(mean_location_AllStalks);
encoder_ratio = length_gt/length_encoder;
mean_location_AllStalks1 = (mean_location_AllStalks-
min(mean_location_AllStalks)).*encoder_ratio+min(me
an_location_AllStalks);

figure; bar(mean_location_AllStalks,
ones(1,numel(mean_location_AllStalks)), 0.4, 'c',
'EdgeColor', 'c');
figure; bar(location_gt, 0.5.*ones(1, numel(location_gt)),
0.4, 'k');
figure; bar(mean_location_AllStalks1,
ones(1,numel(mean_location_AllStalks1)), 0.4, 'c',
'EdgeColor', 'c');
hold on; bar(location_gt, 0.5.*ones(1,
numel(location_gt)), 0.4, 'k');
title('Ground Truth Locations & Measured Locations');
rmse_location = sqrt(sum((mean_location_AllStalks1-
location_gt).^2)/numel(mean_location_AllStalks))


%% Diameter estimation
for i = 1:numel(stalkDiameter_AllStalks(:))
  stalkDiameter_mean(i) =
mean(stalkDiameter_AllStalks{i});
end

load dia_gt_2_V12;
```

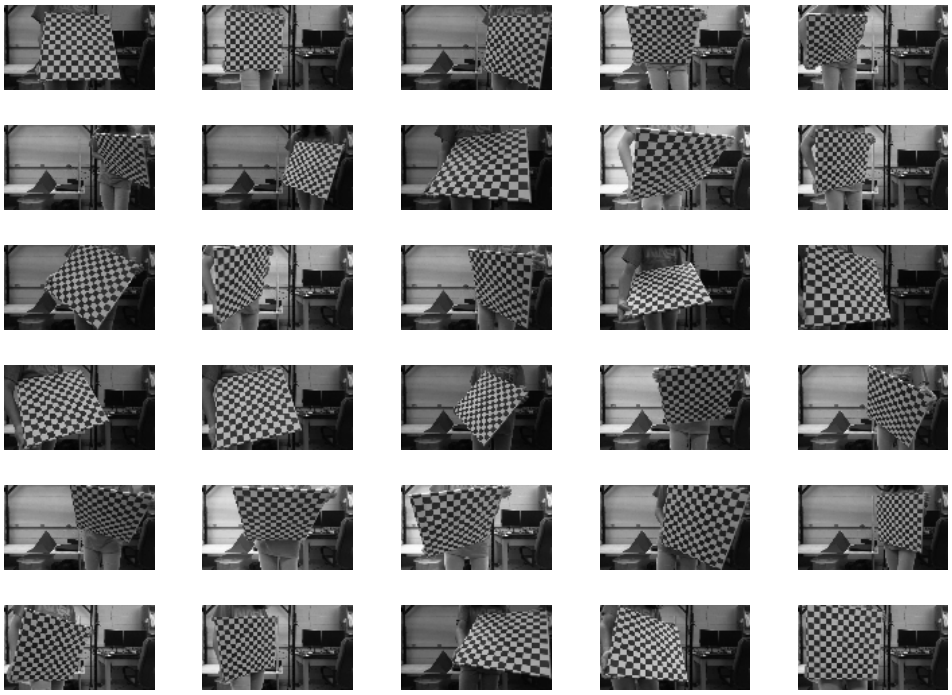
```
figure;
scatter(1:1:numel(stalkDiameter_mean),stalkDiameter_
mean,'filled');
axis([0 25 0 50]); grid on;
hold on; scatter(1:1:numel(dia_gt),dia_gt,'g','filled');
xlabel('Stalk Index'); ylabel('Diameter (mm)');
title('Row 3 rep 3 at V12');
rmse = sqrt(sum((stalkDiameter_mean-
dia_gt).^2)/numel(dia_gt))

% ------------------------------------------------------------------
%% Display individual image for testing purpose
% close all;
% for i = 1:
%     figure; imagesc(img_thres(:,:,i));
% %     figure; imagesc(img_rec(:,:,i));
% %     figure; imagesc(img(:,:,i));
% %     axis off;
%     title(['Image #' num2str(i) ' stalk #']);
% end
```

APPENDIX E

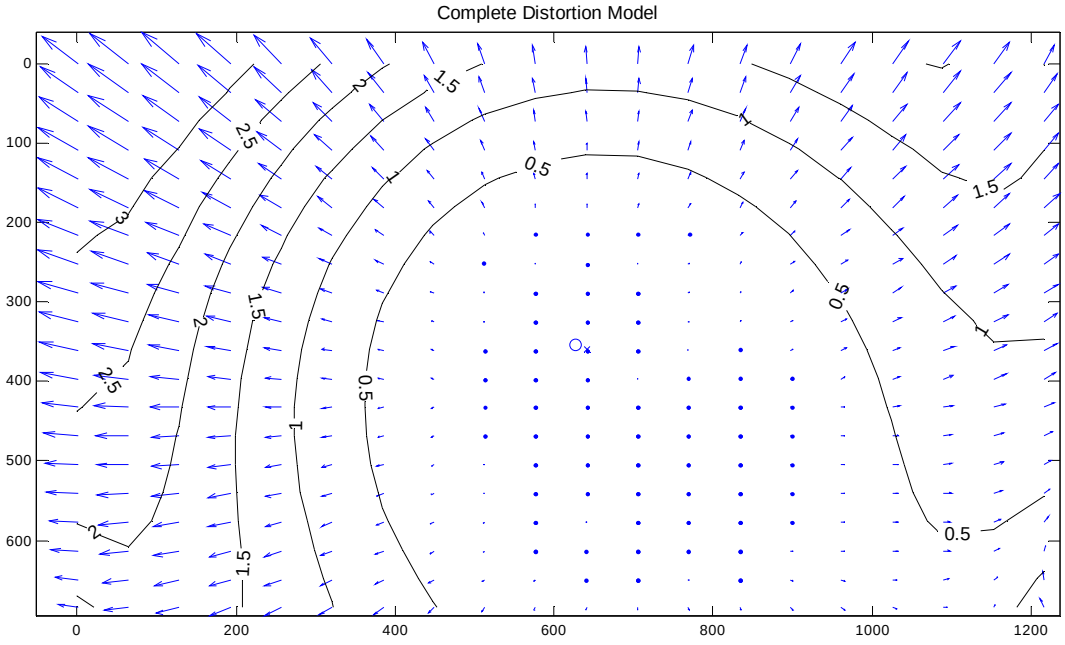
Webcam Distortion Correction Result

(a)



(b)

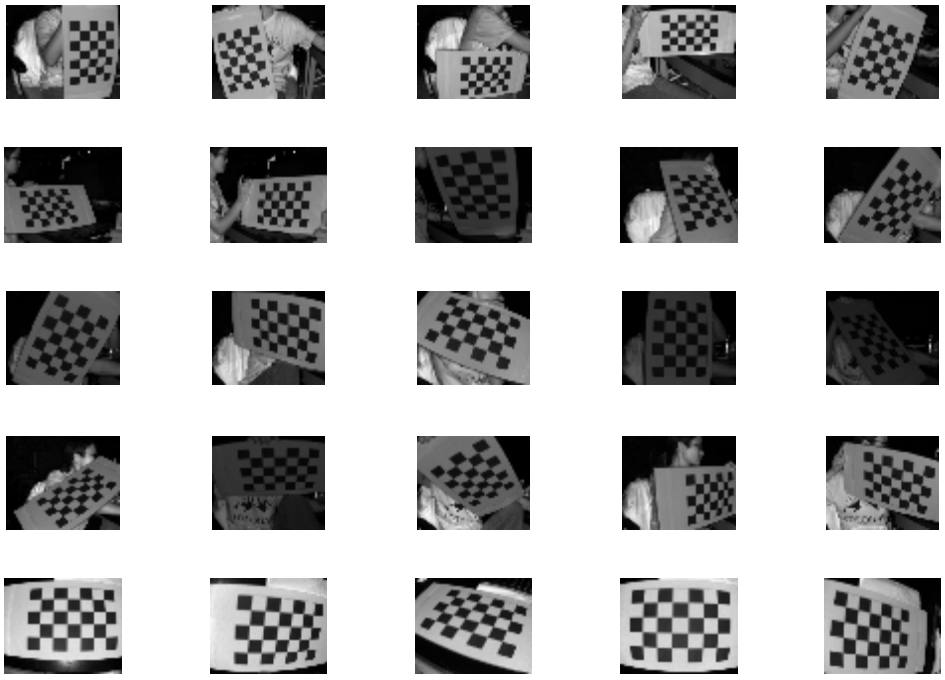
Fig. 47 Webcam's images before (a) and after (b) distortion correction.



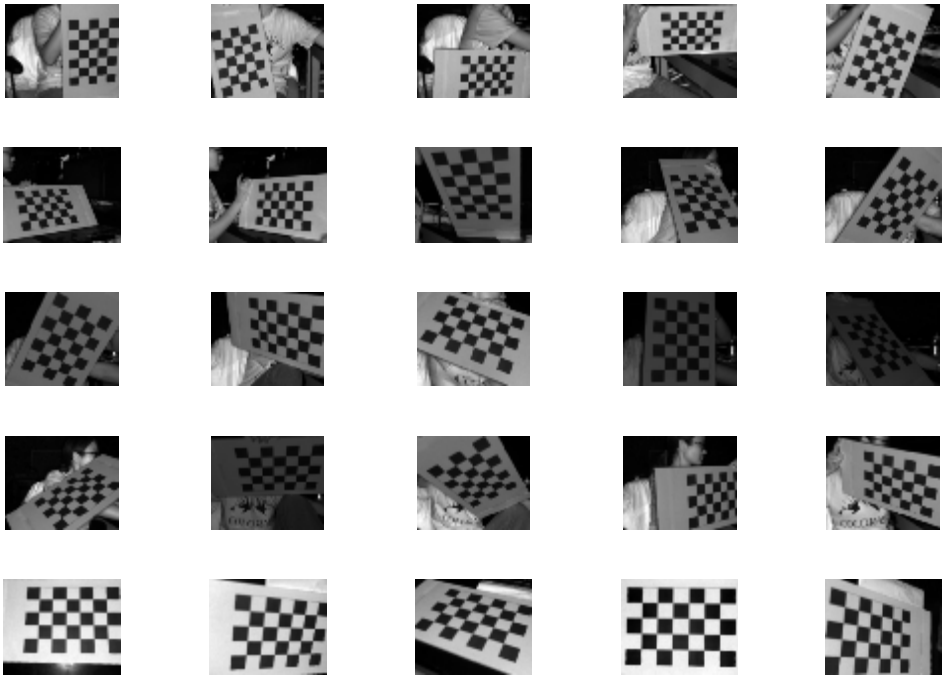
Pixel error	= [0.2427, 0.2419]	
Focal Length	= (990.123, 1000.77)	+/- [0.5251, 0.7]
Principal Point	= (625.226, 353.405)	+/- [1.269, 1.025]
Skew	= 0	+/- 0
Radial coefficients	= (0.02089, -0.0356, 0)	+/- [0.002412, 0.007014, 0]
Tangential coefficients	= (-0.001433, -0.0008385)	+/- [0.0003761, 0.0004636]

Fig. 48 Complete distortion model of the webcam with camera parameters.

3D Range Camera Distortion Correction Result

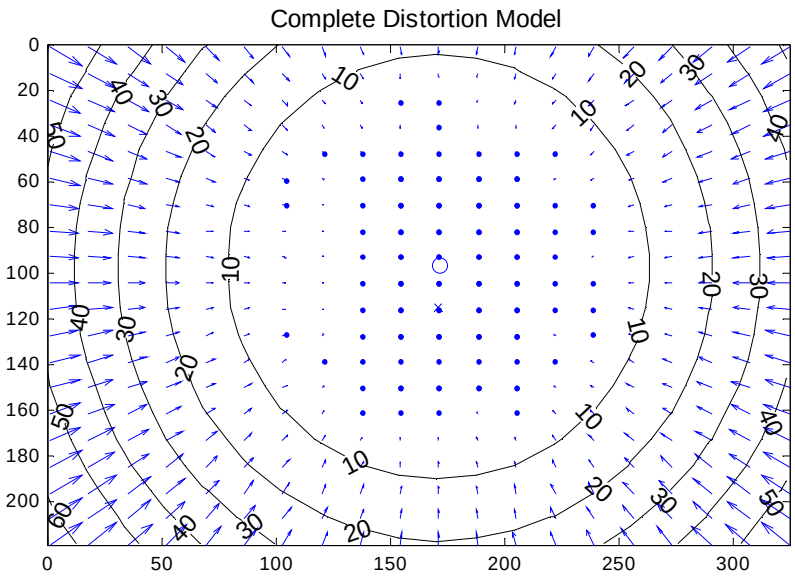


(a)



(b)

Fig. 49 Amplitude images from the 3D range camera before (a) and after (b) distortion correction.



Pixel error	= [0.1201, 0.1169]	
Focal Length	= (250.2, 250.149)	+/- [1.192, 1.275]
Principal Point	= (171.061, 95.9839)	+/- [2.147, 1.933]
Skew	= 0	+/- 0
Radial coefficients	= (-0.8638, 0.6088, 0)	+/- [0.02152, 0.1276, 0]
Tangential coefficients	= (0.0036, 0.001379)	+/- [0.004059, 0.004426]

Fig. 50 Complete distortion model of the 3D range camera with camera parameters.

VITA

Yeyin Shi

Candidate for the Degree of

Doctor of Philosophy

Thesis: CORN PLANT LOCATION, SPACING AND STALK DIAMETER MEASUREMENT USING MACHINE VISION TECHNOLOGIES

Major Field: Biosystems and Agricultural Engineering

Biographical:

Education:

Completed the requirements for the Doctor of Philosophy in Biosystems and Agricultural Engineering at Oklahoma State University, Stillwater, Oklahoma in May, 2014.

Completed the requirements for the Master of Science in Biosystems and Agricultural Engineering at at Oklahoma State University, Stillwater, Oklahoma in December, 2009.

Completed the requirements for the Bachelor of Science in Mechanical Engineering at Nanjing Forestry University, Nanjing, Jiangsu, China in 2007.

Experience:

Employed as a research assistant by the Department of Biosystems and Agricultural Engineering, Oklahoma State University, from August 2007 to December 2013.

Professional Memberships:

American Society of Agricultural and Biological Engineers
Association of Overseas Chinese Agricultural, Biological and Food Engineers
Alpha Epsilon, Honor Society of Outstanding Biological and Agricultural Engineers