Applications of Low-Cost Computer Vision for Agricultural Implement Feedback and Control

by

Trevor Stanhope

Presented to the Department of Bioresource Engineering

In partial fulfillment of the requirements for

For the Degree of Master of Science

McGill University, Macdonald Campus

Montreal, Québec, Canada

April 15th, 2016

ABSTRACT

Applications of computer vision in agriculture have significant potential for improving the precision and efficiency of several field operations, such as weed control. Cultivation is commonly employed for non-chemical weed control by organic farmers or as a supplementary control method for herbicide-resistant weeds. Cultivation requires precise hitch steering systems in order to compensate for the independent motion of the implement. Conventional systems utilize mechanical guiding rods for row detection, but guiding rods are contact sensors and perform poorly during the earliest stages of crop growth when cultivation is often critical. Modern systems based on real-time kinematic global navigation satellite systems (RTK-GNSS) have been developed which are highly accurate but require a secondary receiver on the implement and are often prohibitively expensive for small-scale operations. Therefore, a low-cost computer vision system was developed as a non-contact sensor to supplement guiding rods for early-season, inter-row cultivation. The computer vision system was interfaced with a Sukup Auto-Guide electro-hydraulic hitch steering system. Two cameras were mounted to the cultivator tool-bar in-line with crop rows to obtain semi-orthogonal video streams of the plants passing beneath the implement. Python and OpenCV were used to develop an algorithm for detecting the offset of the crop rows and to adjust the hydraulic steering accordingly via PID control. The computer vision system was compared against guiding rods at travel speeds of 6, 8, 10, and 12 km/h in corn and soybean fields under varying ambient light conditions and differing crop stages. The computer vision system significantly outperformed the guiding rods when crop plants were less than 15 cm and there was no significant difference in performance between the two sensing techniques when the crops were larger.

In addition to row detection, the same computer vision system was used for robust, real-time implement visual tracking. Implement-mounted cameras have the potential to provide machinery feedback, i.e. travel speed and direction, during field operations where accurate tracking data may be used for adjusting operational parameters, such as the responsiveness of a hydraulic steering system. Several feature-descriptor algorithms were evaluated using six different surfaces (gravel, asphalt, turf grass, seedlings, corn residue, and pasture) at travel speeds between 1 and 5 m/s. When comparing visual tracking with RTK-GNSS, ORB with CLAHE pre-processing (CLORB) and 1NN cross-checking was the most robust with respect to real-time applications. For 95% of measurements, CLORB achieved errors under 0.23 m/s. Of the feature-descriptors which achieved acceptable accuracy, only CLORB was capable of operating in real-time (25 Hz), whereas USURF, SURF and SIFT were only capable of 15 Hz or less.

RÉSUMÉ

Les applications de la vision par ordinateur ont le potentiel d'améliorer plusieurs opérations dans le domaine de l'agriculture, tel que le contrôle des mauvaises herbes. Une de ces méthodes consiste à gérer ces plantes indésirables en les cultivant ; une méthode utilisée par les agriculteurs biologiques et leur contre-parti conventionnel le font quand ils font face à des mauvaises herbes résistantes aux herbicides. Cette méthode demande le contrôle précis de la direction de l'attelage pour compenser pour le mouvement indépendant de celui-ci. Les systèmes conventionnels vont utiliser des tiges pour détecter les rangées mécaniquement. Par contre, ces tiges ont tendance à mal performer durant les phases de croissance initiales des plantes – un moment critique dans lequel les mauvaises herbes doivent être désherbées. Les systèmes plus modernes qui utilisent la navigation globale par système satellite kinematic à temps réel, fonctionne avec exactitude, mais demandent un receveur secondaire placé sur l'outillage, attaché à l'attelage. Le coût de ce dernier rend cette technologie inaccessible aux opérations à plus petites échelles. Un système de vision par ordinateur (aussi appeler vision numérique), sans contact et à bas prix, a été créée à cette effet. Elle a comme but de désherber entres les rangées, tôt dans la saison, là où les tiges fonctionnent moins. Cette vision numérique est connectée à un système de pilotage électrohydraulique Sukup Auto-Guide. Deux caméras ont été placées sur la barre à outils de l'équipement de désherbage afin d'obtenir un image semi-orthogonal des plants qui passent dessous. Le langage programmatique Python et OpenCV ont été utilisés afin de développer un algorithme pour la détection de la compensation des rangées et ensuite, ajuster le système pilotage par PID. Ce système de vision numérique a été mis au défi contre les tiges à des vitesses de 6, 8, 10, et 12 km/h dans des champs de maïs et de soya, sous des conditions de lumières ambiantes variées et à différentes étapes de la pousse. Le système de vision numérique surpasse la performance des tiges dans tous les cas où les plantes étaient moins de 15cm. Dépassées cela, il n'y avait pas de différences significatives entre les deux technologies.

En plus du système de détection des rangées, ce même système de vision numérique a servi pour le suivi de plusieurs paramètres tels que la direction et la vitesse durant les opérations. Ces données peuvent servir à des ajustements tel que la vitesse de réaction du système de pilotage hydraulique. Plusieurs algorithmes d'analyse des traits particuliers ont été évalués sur un total de six surfaces différentes (gravier, asphalte, gazon, semis, résidus de maïs, et pâturage) à une vitesse entre 1 et 5 m/s. En comparant le suivi avec RTK-GNSS, ORB avec CLAHE, prétraitement (CLORB) et une vérification croisée par 1NN, était la solution la plus robuste par rapport aux applications à temps réel. Pour 95% des mesures, CLORB a performé avec une précision de 0.23 m/s. Tous les systèmes numériques atteignent une précision similaire, mais CLORB a été testé en temps réel (25 Hz), tandis que USURF, SURF et SIFT étaient seulement efficaces sous moins de 15 Hz.

ACKNOWLEDGMENTS

I want to extend my gratitude to Agri-Fusion 2000, Inc. for their financial support and special thanks to their personnel, Jofroi Desperrier Roux, Michel Gauthier, and Tristan Généreux, for their invaluable insights when developing the cultivator guidance system. Additionally, I wish to thank Richard Raynault and Marc-Antoine Baril of Ferme Bonneterre for additional support and allowing me to test an updated version of the system on their fields.

Dr. Viacheslav Adamchuk has been an incredible influence on me and because of him I plan to continue to work in developing Precision Agriculture technologies. I have much to thank him for, but most of all I appreciate the opportunities he has provided me. His guidance has helped me become both a better scientist and a better engineer.

Partial funding of this research was provided through the National Science and Engineering Research of Canada (NSERC) Engage program and I want to thank Jean Cantin of the Québec Ministry of Agriculture, Fisheries and Food (MAPAQ) for his encouragement.

Lastly, I wish to thank my colleagues in the Precision Agriculture and Sensor Systems research team who have assisted me with my work: Antoine Pouliot, Florian Reumont, Yue Su, Frédéric René-Laforest, and Brett Bennett.

I dedicate this work to both of my grand-dads

FORMAT OF THESIS

This thesis is constructed as an assembly of two independent projects on the use of computer vision in agriculture. Both parts have been prepared for publication as journal articles. Following the general introduction and literature review, Chapter 3 describes the development and evaluation of the inter-row cultivator steering system, and is intended for submission to the journal of *Applied Engineering in Agriculture*. Subsequently, Chapter 4 compares several feature-descriptor algorithms and their application in visual tracking of agricultural implements, and is intended for submission to the journal of *Computers and Electronics in Agriculture*. General conclusions and appendices of supplemental materials complete this thesis.

TABLE OF CONTENTS

CONTRIBUTION OF AUTHORS

Dr. Viacheslav Adamchuk supervised the research discussed in Chapter 3 and Chapter 4, and served as the primary editor of this thesis. Jofroi Desperrier Roux provided technical insights with respect to the practical design considerations and field testing of the computer vision cultivator guidance system discussed in Chapter 3. The Résumé was translated to French by Florian Reumont, Ian Burelle, and Amanda Boatswain-Jacques.

LIST OF TABLES

LIST OF FIGURES

LIST OF DEFINITIONS

BMP	Best Management Practices
BRIEF	Binary Robust Independent Elementary Features
CANBUS	Controller Area Network Bus
CMOS	Complementary Metal-Oxide Semiconductor
DAQ	Data Acquisition
DOF	Degrees of Freedom
DoG	Difference of Gaussian
FAST	Features from Accelerated Segment Test
GNSS	Global Navigation and Satellite System
GPIO	General Purpose Input and Output
GPS	Global Positioning System
HSV	Hue, Saturation, and Value
IMU	Inertial Measurement Unit
IR	Infra-Red
LoG	Laplacian of Gaussian
MOSFET	Metal-Oxide-Semiconductor Field-Effect Transistor
NMEA	National Marine Electronics Association
ORB	Oriented FAST and Rotated BRIEF
PID	Proportional-Integral-Derivative
PWM	Pulse-Width Modulation
RGB	Red, Green, and Blue
RANSAC	Random Sample Consensus
RTK	Real-Time Kinematic
SURF	Speeded-Up Robust Features
SIFT	Scale Invariant Feature Transform
VDC	Volts of Direct Current
VRT	Variable Rate Technology
VSLAM	Visual Simultaneous Localization and Mapping

Chapter 1: Introduction

Agricultural technology and food production systems have progressed rapidly over the last several decades in response to rising global populations and standards of living. Information technologies for farm management, such as real-time kinematic global navigation satellite systems (RTK-GNSS), have contributed significantly to decision making and improving operational efficiencies. Concurrently, greater environmental awareness has led to an interest in better management of fertilizers and pesticides to address soil and water quality concerns (Matson, 1997). Adoption of precision agriculture technologies, especially variable-rate technologies (VRT) and digital imaging, has the potential to cost-effectively improve farm operations like fertilizer application, weed control, and guidance systems. By providing tools which allow for implements to respond in real-time to their conditions, precision agriculture technologies can greatly increase the degree of control farmers have over their fields (Robertson, 2007).

Demand for high-precision guidance and mapping in agriculture has resulted in tractor systems pioneering the way in vehicle guidance systems have achieved steering accuracy of less than 2.5 cm and full fly-by-wire control. Following closely behind tractors, implements have become the next target for precision agriculture technologies. Although auto-steer for tractors has been widely adopted (Erickson, 2015), agricultural technologies exhibit sequential adoption trends and farmers have been slower to adopt implement technologies, e.g. VRT or yield monitors (Winstead, 2010). Unlike tractors, which for the most part are all-purpose machines, implements are used for specific tasks within a small window during the season. As such, demonstrating the cost-effectiveness of incorporating control systems into implements is a limiting factor in their adoption. Mission-critical field implements, e.g. sprayers, seeders, and cultivators, are likely to adopt precision agriculture technologies which utilize sensor systems to improve control (Erickson, 2015).

As independent machines from the tractor, implements directly interact with the crop and soil and are subject to unique dynamics which can have negative effects on their performance (Cowell, 1966). According to Heraud et al. (2009), wandering of an implement can be caused by both asymmetric geometry of the implement and asymmetric forces acting upon it, e.g. due to sloped terrain. Compensating for these effects increases the stress of tractor operation, and steering accuracy has been shown to decrease dramatically as extra demands are placed on drivers (Kaminaka, 1981). Incorporating Proportional-Integral-Derivative (PID) control for steerable hitches can limit wandering to satisfactory levels, e.g. within ±5 cm (Heraud, 2009). Overall, supervised guidance systems using RTK-GNSS or computer vision are useful because they allow for increased travel speeds and implement widths without placing additional stress on operators (Wilson, 2000).

For many precision agriculture systems, e.g. hitch steering compensation, it is necessary for control systems to have real-time feedback from sensors on the implement. Ideally, an implement should be self-sufficient and as independent as possible from the tractor, relying only on the tractor for power and hydraulics. However, networking sensors between the tractor and implement, e.g. via a Controller Area Network Bus (CANBUS), can be used for advanced management strategies (Darr, 2012). In a CANBUS, embedded controllers throughout the tractor are able to transmit sensor information over a shared data connection without a host computer. With respect to agriculture and forestry tractors, this communication protocol is standardized by ISO 11783 (ISO, 2014). Unfortunately, CANBUS systems are not available on older tractors and thus, implements which feature self-sufficient control and sensor systems can afford the farmers with greater flexibility. The cost of compartmentalizing implement control systems must be weighed against the need for reliability and serviceability. A problem with this approach is excessive complexity of sensor systems. Key technological components for implement functionality should not be overly specialized or prohibitively expensive to replace.

The main challenge when designing embedded systems for real-time implement feedback and control is minimizing the system complexity by employing sensors which are versatile, yet low-cost and easily serviced. As such, precision implement design is a ripe environment for applications of computer vision. Among current technologies, digital imaging is arguably the most powerful single-sensor platform and the cost of cameras has decreased dramatically. A 640 x 480 pixel Red-Green-Blue (RGB) camera provides a detail-rich myriad of different types of information including: texture, color, motion, and objects. By employing digital imaging on agricultural platforms, control systems can achieve a high-degree of sensor versatility for relatively low cost, maintenance, and sensor calibration (Scarramuzza, 2008). However, embedded systems, such as those in agricultural control systems, have limited memory and computational power. Therefore, when incorporating digital imaging for real-time feedback in embedded applications, it is necessary to consider optimizing for resource-constrained environments (Kopetz, 2011).

The goal of this thesis was to explore applications of computer vision for real-time feedback and control of agricultural implements. Specific objectives were: 1) to develop and evaluate a low-cost computer vision system for automatic guidance of inter-row cultivators at early stages of crop growth, and 2) to develop and evaluate a visual tracking algorithm for real-time detection of travel-over-ground velocity which could be integrated into the same computer vision system used for cultivator guidance.

Chapter 2: Literature Review

In precision agriculture, demand for advanced data acquisition and control has resulted in significant interest in the development of computer vision systems. Digital cameras are relatively inexpensive and can provide very detailed information, e.g. a simple 640 x 480 RGB image contains 3 color channels with 307200 spatially-structured 8-bit data points. As such, digital imagery can be employed for extracting information on texture, structure, color, and motion. Examples of computer vision in agricultural applications include soil composition analysis (Sofou, 2005), nutrient deficiency (Xu, 2011), and disease detection (Sankaran, 2010). Camera systems, and computer vision sensors systems in general, have potential to provide farmers with greater control and monitoring capabilities (Lee, 2010). Recently, some commercial systems have incorporated computer vision techniques for guidance and control of field equipment. Examples of successful commercial products which have entered the industry in the past decade are presented in Figure 2.1, such as the Garford[1] InRow cultivator (Garford Farm Machinery Ltd, Frognall, United Kingdom), the CLAAS AutoFill and CAMPilot (AGROCOM Verwaltungs GMBH, Bielefeld, Germany), and the VineScout guidance system for small tractors in orchards and vineyards (Clemens GMBH & Co KG, Wittlich, Germany).

Figure 2.1 Clemens VineScout (top left), Garford InRow (top right), CLAAS AutoFill (bottom left), and CLAAS CAMPilot (bottom right). Adapted from Möller (2010).

[1] Mention of a trade name, proprietary product, or company name is for presentation clarity and does not imply endorsement by the authors, or McGill University, nor does it imply exclusion of other products that may also be suitable.

Although the information provided by computer vision systems is incredibly versatile, the primary limitation for their practical application is the computational power required for data processing of image matrices. This limitation is especially apparent for real-time video analysis. For example, a 640 x 480 color camera running at 25 frames/s produces greater than 1 MB of data every second. Fortunately, not only have digital cameras become prolific, but the cost of microprocessors and microcontrollers has also steadily declined, especially with respect to architectures such as Intel Atom processors (Intel Corp., Santa Clara, California, USA), Atmel ATmega microcontrollers (Atmel Corp., San Jose, California, USA), and ARM Cortex processors (ARM Ltd., Cambridge, England, UK). Similarly, motherboards and embedded systems have continued to decrease in size (Figure 2.2) while continuing to provide a greater degree of features like on-board General Purpose Input/Output (GPIO).

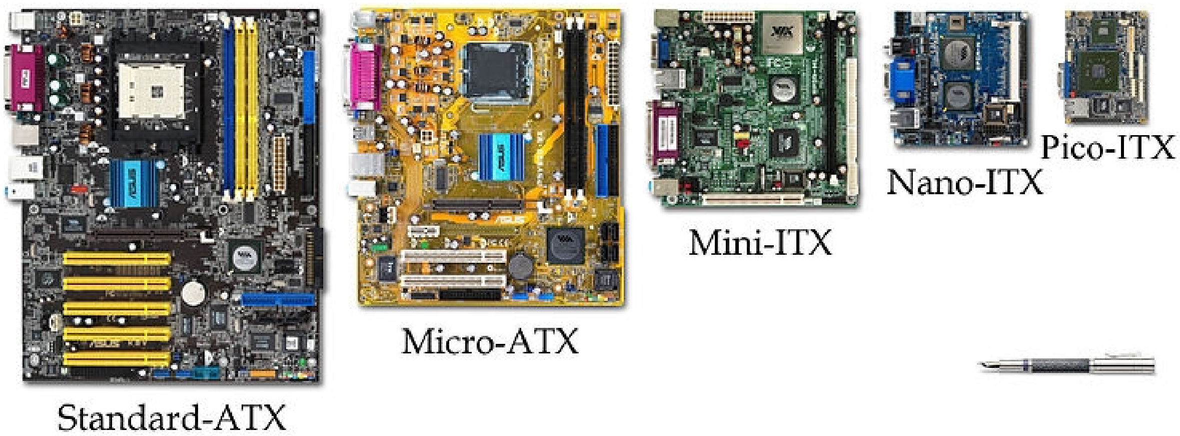

Figure 2.2 Comparison of motherboard form factors. Reprinted with permission by the VIA Gallery (2007) under the Creative Commons License 2.0.

The proliferation of low-cost cameras and embedded systems has coincided with a greater focus in agriculture on innovative technological solutions to complex problems. By integrating computer vision into precision agriculture systems, a single camera can provide information-rich sensor feedback which can be employed to reduce stress on tractor operators and improve the precision of the job performed. Overall, computer vision's versatility makes it a valuable tool with numerous potential applications for addressing challenges in precision agriculture.

2.2 Crop Cultivation and Implement Guidance

The emergence of herbicide resistant weeds, such as glyphosate-resistance in the United States, and increasing public interest in organic food products have led to a greater adoption of non-chemical pest control mechanisms (Livingston, 2015). Since 2008, there has been a 72% increase in sales from organic farming

(USDA, 2015). In addition, current better management practices (BMPs) for weed control recommend incorporating mechanical cultivation and cover crops to mitigate the effects of herbicide resistance (Norsworthy, 2012). Although BMPs for managing herbicide resistance weeds have not been universally accepted, farmers who rely on glyphosate for a large proportion of acreage are rapidly adopting BMPs (Dong, 2016). As non-chemical weed control increases in popularity, it is therefore important to reconsider technological solutions which were waylaid in favor of herbicides.

As an alternative to chemical weed management, cover crops and mechanical cultivation practices are commonly employed in organic farming operations as ecological alternatives (Damian, 2011). Historically, effective non-chemical weed control required a large agriculture workforce to manually control weeds. Mechanized weeding implements, commonly referred to as cultivators, are an effective weed control method which consist of a tractor drawn tool-bar equipped with tillage tools for cutting and burying weeds at the soil surface before their root zones can become established (Jones, 1996). Examples of tools commonly used for cultivation include coulters, chisels, harrows, brushes and tines (Froud-Williams, 1983). However, the weed population, size, and growth habit influences the tools used and the depth of covering required in order to prevent re-emergence (Baerveldt, 1999). Manual weeding by hand is strenuous and physically demanding work (Chatizwa, 1997), but requires no initial cost of equipment and therefore, it is practical for small areas (Hansen, 2004). The practice of manual weeding remains the dominant method for organic weed control in developing nations where there is abundant local labor in the agriculture sector, such as Sub-Saharan Africa and China (Mrema, 2008). However, in developed nations with smaller agricultural workforces, hiring laborers for manual field work is prohibitively expensive, except for specialty crops (Taylor, 2012).

Inter-row brush cultivators are often considered the best for horticulture, but the choice of tools and the frequency of their use depends on the morphology of the crop and the weeds (Bond, 2001). Inter-row cultivation specifically refers to cultivation of the bare region of soil between crop rows where weeds often become established. Similarly, intra-row cultivation refers to the practice of cultivating between consecutive plants in a row (Tillett, 2008). Both methods require precise lateral positioning of the implement in order to maximize the cultivated area without causing damage to the crops: For example, inter-row cultivators typically operate with an error tolerance of only ±5 cm and even minor path deviations can result in damaged crops. Due to their relative mechanical simplicity, mechanized inter-row cultivation is common for large-acreage, close-spaced row crops, e.g. soy and corn. Conversely, mechanized intra-row cultivation is most valuable for vegetable crop producers, e.g. tomatoes, where weeding between plants is necessary and manual weeding is labor intensive (Pérez-Ruíz, 2014). Due to the mechanical complexity of intra-row cultivator implements, their use is often restricted to operational speeds less than 4.0 km/h (Tillett, 2008).

Although mechanical cultivator implements are widely used in organic agriculture, hand-weeding is the most effective way for eradicating troublesome patches of weeds in small-scale operations (Marshall, 1992) and is occasionally used after mechanical inter-row weeding to remove any remaining weeds (Ionescu, 1996). Mechanical weed control is a viable option only within a certain range of soil conditions, the amplitude of which varies upon soil type and implement used (Bowman, 1997). For example, excessive soil moisture impedes field workability and may delay mechanical weed control until the crop is too high or the weeds are too well established (Bàrberi, 2002).

To compensate for the movement of the cultivator implement relative to the tractor, systems use sensors to detect the lateral offset of the crop row and subsequently control a hydraulic steering system. A conventional method for detecting the crop rows involves guiding rods which are mounted to a rotary potentiometer (Figure 2.3). The guiding rods contact the crop stems and the resulting voltage is used to estimate the angular position and by extension, the lateral offset of the crop row. The reference signal is fed to a hydraulic-solenoid controller which adjusts the steering mechanism, e.g. a double-acting hydraulic cylinder, to adjust the implement's position accordingly. Although guiding rods are rugged and provide proficient accuracy, the rods perform poorly with seedlings (Thwacker, 1996). During the early stages of growth (e.g. < 15 cm) when cultivation is often critical, seedlings are not sufficiently large enough for the guiding rods to contact the plants properly, and ultimately this can result in poor tracking and damaged crops. When using guiding rods early in season, tractor operators are often forced to travel at low speeds, e.g. less than 4 km/h, and to manually control the hydraulics while driving.

Figure 2.3 Mechanical guiding rods for crop row detection. The guiding rods are designed to track the position of the crop row by mechanical contact with the stems of the plans.

Several non-contact methods have also been developed for row detection, e.g. RTK-GNSS receivers and computer vision. RTK-GNSS offers high accuracy and is commonly available on North American farms. Several hitch manufacturers have adopted RTK-GNSS for implement guidance, e.g. the ProTrakker 500DB (ProTrakker, Odebolt, Iowa, USA) and the Orthman GPS Tracker IV (Orthmann Manufacturing Inc., Lexington, Nebraska, USA). The relative positions of the implement and tractor are dynamic, e.g. for pull-type and articulated-hitch implements, and therefore, a secondary receiver is needed on the implement itself. Although many organic farmers are equipped with RTK-GNSS for tractor auto-steering, equipping each active tractor with an additional receiver, i.e. two per cultivator, is prohibitively expensive for many small-scale producers. As such, the adoption of RTK-GNSS for hitch steering systems has been relatively low as compared to mechanical guiding rods.

As an alternative to guiding rods and RTK-GNSS, computer vision has the potential to be a non-contact, precise, and relatively low-cost method for row detection. Research has explored a variety of imaging methods, such as RGB color cameras and more advanced multispectral imaging systems (Slaughter, 2008). Computer vision techniques have been shown to be capable of detecting the lateral offset of crop rows relative to the implement with a high degree of reliability and accuracy (Tillett, 1991). Similarly, computer vision offset detection has been successfully employed as a component of a high-precision testing system used to verify the performance of tractor auto-steer with a resolution of 2 mm on concrete (Easterly, 2010). By integrating computer vision systems into agricultural platforms, guidance precision can be increased which allows the implement to clean a greater region between crop rows. Applications of computer vision have demonstrated that it can be a reliable method for row detection in various crops, such as sugar beet (Tillett, 2002) and cereals (Hague, 2006), and is effective for feedback and control of agricultural implements (Slaughter, 1999). Several different methodologies have been proposed for identifying the relative position of the crop rows, including stereovision mapping (Kise, 2005), implementation of Hough Line Transform (Rovira-Más, 2005), and band-pass filters (Hague 2001), among others. However, the development of a low-cost system which is capable of analyzing images from basic cameras in diverse ambient lighting conditions and is compatible with a variety of hitch steering systems could further increase adoptability of computer vision row detection.

2.3 Motion Estimation

With respect to the precise nature of guiding agricultural implements, control systems benefit from real-time estimates of the implement's motion vectors. Vector information has numerous applications for improving the precision of agricultural operations. Examples include metering on seeders and transplanters (Griepentrog, 2005), true groundspeed for slip estimation (Thansandote, 1977), and tracking direction of a tractor (Kise, 2005). Several different techniques exist for acquiring vector data, each of which have advantages and disadvantages depending on the application environment (e.g. orchard, row crop), sampling rate, and the level of precision and accuracy required.

In most contexts, motion vectors can typically be acquired using RTK-GNSS receivers. The advantage of RTK-GNSS is that not only does it provide accurate tracking information (i.e. heading and ground speed), but also provides the geospatial location of the receiver. Due to its accuracy and versatility, RTK-GNSS has been widely adopted in agriculture for applications such as auto-steering (Erickson, 2016), auto-steer via RTK-GNSS has been shown to be capable of 1 cm accuracy (Gan-Mor, 2007). Compared to the alternatives, RTK-GNSS receivers provide excellent accuracy, but require additional infrastructure like local base stations which increases their cost of implementation (Figure 2.4). Non-differential GPS systems are exponentially lower-cost than RTK-GNSS and have been shown to be sufficiently accurate for steady-state speed estimation (Keskin, 2006). However, low-cost GPS devices have a low polling frequency, typically less than 1 Hz, and have been shown be capable of only 0.2 m/s accuracy for 45% of measurements during non-steady-state trials (Witte, 2004). Additionally, low-cost receivers can exhibit inconsistent accuracy depending on direction of travel, for example, Ehsani et al. (2002) found that cross-track error for several differential GPS receivers was higher for the North-South direction compared to East-West.

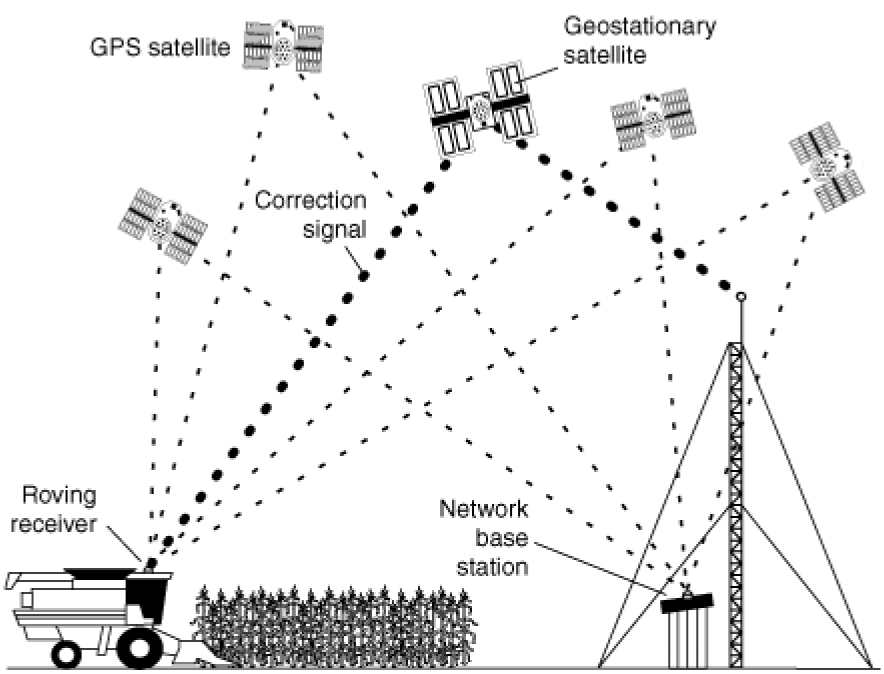

Figure 2.4 Diagram of an RTK-GNSS agricultural network. Reprinted with permission from Grisso et al. (2009).

Common mechanical alternatives to GNSS for detecting motion include rotary encoders in contact with the soil surface, commonly known as fifth-wheels, and transmission-based odometers. Fifth-wheels and odometers are advantageous because they are low-cost and provide a sufficient degree of precision for many applications like seed metering. However, due to high wheel slip in many agricultural applications, odometers often demonstrate relatively poor accuracy. Although fifth-wheels address this by using encoders with minimal rolling resistance, they nevertheless are contact sensors and are prone to slippage errors on some surfaces, such as freshly tilled soil (Tompkins, 1988). Additionally, fifth-wheels provide limited information compared to other techniques, like RTK-GNSS, because they are only capable of 2 degrees-of-freedom (DOF), i.e. acceleration and velocity along a single axis.

To address issues inherent to contact sensors, a non-contact method has been proposed based on the principle of Doppler-shift. Speed detection with Doppler-shift utilizes an electromagnetic transmitter and receiver (Figure 2.5). A wave is emitted and by observing the shift in the reflected wave off of the ground travel speed can be determined (Lhomme-Desages, 2009). With respect to agricultural applications, the Doppler-shift method has been shown to be viable for non-contact slip detection of tractors on several agricultural surfaces, such as tilled soil and pasture (Thansandote, 1977). An initial limitation of early Doppler-shift systems was the unidirectionality of a single transmitter-receiver, but a method for bi-directional speed detection from a single transceiver was demonstrated successfully by Imou et al. (2001).

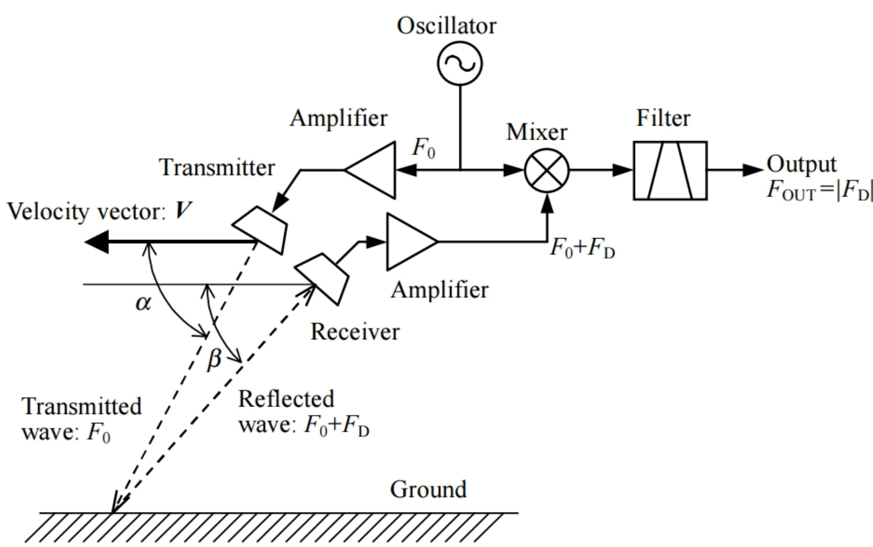

Figure 2.5 Unidirectional Doppler shift vehicle-mounted ground speed sensor. Reprinted from Imou et al. (2001).

Another non-contact method for motion feedback commonly used in robotics are Inertial Measurement Units (IMUs). IMUs are electronic devices that measure a body's specific force and angular rate using a combination of multi-axis, micro-electromechanical system (MEMS) accelerometers and gyroscopes. For example, an IMU with a tri-axis gyroscope and tri-axis accelerometer can provide 6 DOF. However, IMUs cannot directly measure velocity, but instead provide high precision axial and rotational acceleration at a high sampling rates. By integrating acceleration and gyroscopic data over discrete time-steps, IMUs can be used to estimate a vehicle's velocity and position by dead reckoning. A major disadvantage of using IMU-based dead reckoning for navigation is accumulated error, also known as drift, and Abbe error, i.e. the magnification of angular error over distance. Therefore, IMUs are not typically employed on their own to estimate vehicle speed, but in sensor fusion as part of an Integrated Positioning System (IPS), i.e. with a non-differential GPS. In an IPS, the IMU provides feedback at a very fast rate (e.g. 50 Hz), whereas the slower GPS device is used to continually correct for IMU drift errors. This technique has been demonstrated to be effective by Sukkarieh et al. (1999), Yi et al. (2007), and Guo et al. (2003 and 2008). Due to compact size and low power consumption of non-differential GPS and IMU chipsets, IPS has experienced near ubiquitous adoption in embedded systems, especially mobile phones.

With respect to versatility, visual tracking is particularly well suited for vector estimation on agricultural vehicles. Visual tracking is the process of determining the movement of an object by analyzing consecutive images produced by camera(s). It is advantageous because it provides a high sampling rate (>20 Hz), is non-contact, and the information provided can be used for a wide range of alternate applications. With respect to robotics, visual tracking is essential for many applications, including visual odometry (Nistér, 2006) and visual Simultaneous Localization and Mapping (VSLAM) (Davison, 2007), especially for applications without reliable satellite coverage, e.g. forests. Visual odometry refers to the tracking of a non-stationary platform, e.g. robot or vehicle using an on-board camera, whereas VSLAM refers to constructing a map of an environment while simultaneously keeping track of the vehicle's relative location. However, visual tracking with respect to real-time applications requires the detection algorithm to be not only precise, but robust, i.e. tolerant of varying environments, and computationally efficient.

Early evaluation of visual tracking for agriculture applications was conducted by Stone et al. (1992). A system consisting of a single camera, also referred to as a monocular system, was developed with the camera oriented vertically to the test surface (Figure 2.6). Estimation of movement between two consecutive frames was achieved by minimizing the displaced frame difference (DFD). Results were generally successful and achieved 2.2% error for an average travel speed of 0.47 m/s on bare soil. However, the system displayed poor tolerance for rotation, side-slip (1.5% of image length), and variations in lighting intensity (Stone, 1992).

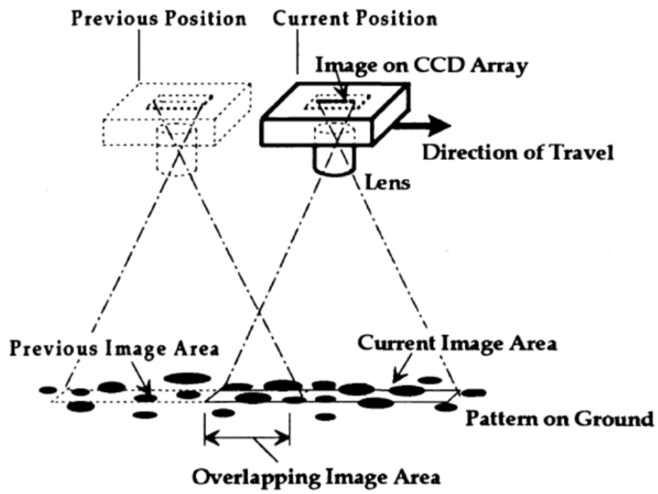

Figure 2.6 Illustration of a monocular visual tracking system. Reprinted from Stone et al. (1992).

In the past decade, visual tracking has seen significant advancements for autonomous applications with much of the emphasis on robustness. In a modern improvement upon Stone et al. (1992), Joos et al. (2010) developed a system for vector detection of a vehicle traveling at highway speeds using a low-cost sensor which captured a 30 mm × 30 mm image at up to 5000 frames-per-second. Consecutive frames were matched using an optical flow estimation based on least-squares which achieved ~5% error at 18 m/s when compared against a fifth-wheel sensor. Similar methods for frame matching via optical flow, such as kernel-based mean-shift visual tracking (Comaniciu, 2000) or contour-based conditional density propagation (Isard, 1998) have also been successfully implemented. However, many modern applications of visual tracking have shifted the focus to feature-based visual tracking due to the greater degree robustness offered by such algorithms (Gauglitz, 2011). Feature-based visual tracking is a highly versatile approach which attempts to mimic human pattern recognition and tracking by identifying distinct features in an image. Since features are defined throughout the image plane, comparison between two images allows for up to 6 degrees-of-freedom (DOF) by determining the homography between the two frames with techniques such as Random Sample Consensus (RANSAC).

The earliest feature detection algorithm was proposed by Harris et al. (1988) and is commonly known as the Harris Corner Detector. The Harris Corner Detector is a relatively simple algorithm which is an excellent introduction to the concept of feature detection. For an image, the difference in pixel intensity for a displacement of (*u,v*) in all directions about each point (*x,y*) is calculated:

$$E(u,v)=\sum_{x,y} \underbrace{w(x,y)}_{\text{window function}} \cdot \left[\underbrace{I(x+u,y+v)}_{\text{shifted intensity}}-\underbrace{I(x,y)}_{\text{intensity}}\right]^2 \tag{2.1}$$

where *I()* is the intensity function, *w()* is the window function, (*x,y*) is the considered point, and (*u,v*) is the displacement.

An advantage of Harris corners, apart from computational efficiency, is rotation-invariance, i.e. if the image is rotated the same corners will be detected. However, Harris corners are sensitive to scale, e.g. if the image depth changes, the effective size of features in the image are scaled. As such, Harris corners are not scale-invariant. Modern algorithms, such as SIFT (Lowe, 2004), SURF (Bay, 2008), and ORB (Rublee, 2011), have been developed which are capable of reliably matching features between frames regardless of changes in lighting, rotation, and scale.

When determining motion from video, knowledge of the camera model used by the visual tracking system is essential when transforming vectors from pixels to true distance. Camera models for visual tracking methods can be categorized into two groups: monocular or stereovision. Monocular methods rely on the images from a single camera and are simple and low-cost, but are limited by variable subject depth (distance from lens to imaging plane): if the subject depth changes, the relative size of each pixel will also change. As such, without a complementary method for depth estimation, monocular methods are effectively limited to only 4 DOF. However, devices such as LIDAR and RGB-D cameras, also known as time-of-flight (ToF) cameras, can be used for depth estimation of monocular systems (Zhang, 2015), albeit such devices increase the cost of the tracking system. Alternatively, stereovision methods utilize two cameras which are displaced horizontally from each other by a distance known as the baseline separation. This configuration is similar to human binocular vision and allows a system perceive relative depth by generating a 3D disparity map (Tippetts, 2016). As such, stereovision has been shown to be a robust method for visual tracking in off-road applications. Nistér et al. (2006) developed a robust stereovision system for VSLAM with Harris Corner Detection for a full-sized, autonomous off-road vehicle. Two cameras were used with a baseline separation of 28 cm. Findings showed that the error distribution for visual estimates is non-Gaussian, and therefore RANSAC is necessary for a robust stereovision system. However, RANSAC operations limited the system to a processing rate of 13 Hz. Stereovision has also been tested on several small-scale robotics systems, including a robot intended for use in a sugar beet field (Ericson, 2008) and VSLAM (Howard, 2008). Although it did not incorporate visual tracking, an advanced stereovision system for crop height detection and row guidance was developed for tractors by Kise et al. (2005). The system was integrated via hydraulic power-steering of a tractor via serial bus to control the steering using Pulse-Width Modulation (PWM), and the guidance system achieved a root mean square error (RMSE) of 3 cm lateral error deviation from the true path at 8 km/h (Kise, 2005).

When comparing monocular or stereovision systems it is important to consider the computational resources available for the target application, e.g. processing power and memory. For example, since stereovision determines the depth of the image it often provides better accuracy on irregular terrain (Nistér, 2006). However, determining depth from two images requires lens distortion correction prior to generating the disparity map (Bradski, 2008), and both operations add computational complexity. As such, there is a trade-off between accuracy and speed which must be considered. Even between stereovision algorithms Tippetts et al. (2016) have demonstrated that there is a significant trade-off between accuracy and speed. Similarly, the same trade-off has been observed in feature detection and descriptor algorithms which are necessary for visual tracking in both monocular and stereovision systems (Hartmann,2013). With respect to agricultural applications, a visual tracking system for mature crops, e.g. pesticide sprayer for corn, will have very different requirements compared to a system intended for early-season or post-harvest field conditions, e.g. cultivator for seedlings.

The benefit of visual tracking is that the technique can be incorporated for motion vector estimation as a secondary use of the camera. As long as the camera model is known and sufficient computational power is available, visual tracking via an implement-mounted camera system can be performed in addition to other imaging operations. As such, computer vision systems have excellent potential to be robust and low-cost, and can be adapted to a variety of complex tasks. Sequential trends in the commercial adoption of agricultural technologies indicate that before a system can be considered viable, continued field trials are necessary to demonstrate their performance and practicality. Due to the limitations of low-cost embedded systems, further research is needed to assess which visual tracking methodologies can offer the best degree of performance with respect to both accuracy and computational efficiency of agricultural computer vision systems.

Chapter 3: Computer Vision Cultivator Guidance

3.1 Introduction

To address the limitations of guiding rods during early-season inter-row cultivation, a computer vision system was developed to interface with a common hydraulic hitch steering system. The challenges inherent to designing a computer vision cultivation system include: 1) differentiating between the crop row, soil, and weeds, a process known as crop segmentation, 2) estimating lateral error based on the camera perspective, and 3) guidance correction of the cultivator implement via the actuation of a hitch steering system.

3.1.1 Crop Segmentation

With respect to the segmentation of green vegetation (e.g., for crop row tracking or weed detection), a significant amount of research has been focused on developing robust color indices. Using unfiltered RGB images from conventional digital cameras is not employed due to the high inter-correlation between the three color channels (Brivot, 1996). Research has shown that imaging a combination of bands in the visible and infra-red (IR) spectrum produces reliable results due to the high reflectance of green and IR (Slaughter, 2008). However, this approach requires specialized camera systems. Therefore, if using RGB data, images should be transformed to an alternate color index which is more advantageous for plant segmentation. Specialized color indices have been developed for agricultural applications, such as Excess Green (ExG) (Woebbecke, 1995) or the Vegetative Index (VEG) (Hague, 2006). Standard indices such as Hue-Saturation-Value (HSV) and normalized RGB (Figure 3.1) have also been used successfully for plant segmentation (Moorthy, 2015).

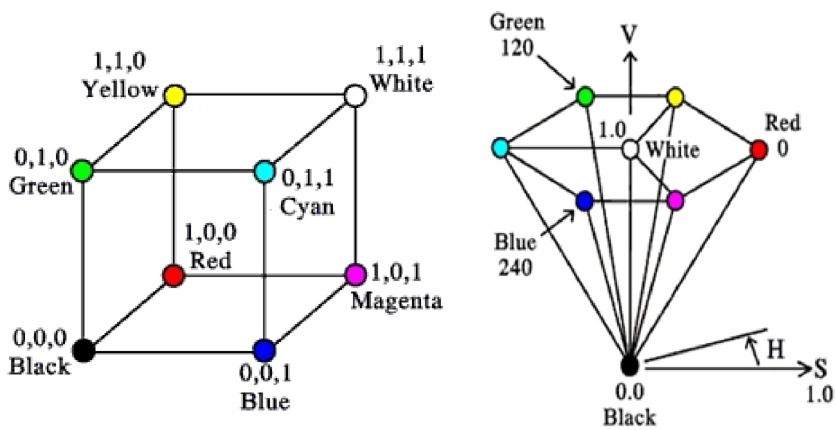

Figure 3.1 Graphical representation of the RGB (left) and HSV (right) color-spaces. Reprinted with permission by Otto (2000).

During the process of crop detection, varying ambient light is one of the major limiting factors for successfully implementing computer vision plant segmentation. Variations in ambient light occur naturally with changes in weather and time of day, and can dramatically change the appearance of crop foliage in a digital image with respect to texture and color. Additionally, non-uniform lighting intensity, e.g. shadows, are also a major concern. Non-uniform lighting results in irregular variance in the intensity of pixels and increases the complexity of segmenting plants using color indices. This necessitates implementing adaptive procedures for crop segmentation where constant and uniform algorithm parameters do not perform adequately due to varying illumination (McCarthy, 2010). Therefore, any robust segmentation algorithm must have the ability to dynamically change thresholding parameters under different lighting conditions to provide consistent performance.

Thresholding techniques proposed for segmenting images include dynamic thresholding methods (Rovira, 2005), Otsu-based thresholding methods (Meyer, 2008), and statistical mean-based segmentation (Guijarro, 2011). Thresholding has been shown to be both computationally efficient and functional, but these methods generally assume that the histogram of the image is bimodal, i.e. that the vegetation and the background belong to two different brightness regions (Meyer, 2008). Although thresholding reduces error caused by varying ambient lighting, such methods may exhibit lower performance for non-uniform illumination conditions within the same image (Tian, 1998). To address this concern, research efforts have focused on developing adaptive algorithms for vegetation segmentation, for example, the Environmentally Adaptive Segmentation Algorithm (EASA) (Tian, 1998), or the mean-shift-based learning procedure proposed by Zheng et al. (2009). Although these methods perform well for changing conditions, they add computational complexity. Recent improvements, such as a the Naive Bayes learner proposed by Moorthy et al. (2015), have reached similar levels of accuracy compared to the state-of-the-art EASA, yet are capable of faster processing times and lower memory usage.

3.1.2 Camera Perspective

After successfully segmenting plants within the image, the lateral offset of the crop row must be determined. Methods for estimating the lateral offset can be grouped into two classes based on whether the camera's angle of inclination is either equal to zero, or greater than zero; these classes are referred to as vertical or oblique, respectfully (Figure 3.2).

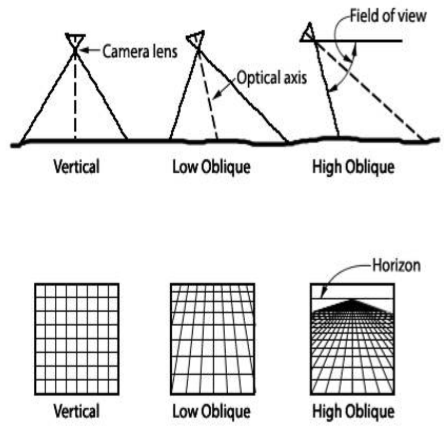

Figure 3.2 Oblique and vertical viewing angles. Reprinted from Wolf et al. (2000).

Vertical methods, also known as orthogonal perspective, rely on a camera system which faces vertically downward and is directly aligned with a single crop row of the cultivator. An early, yet effective, approach proposed by Olsen et al. (1995) for detecting the lateral offset of the crop relies on taking the sum of the pixel elements gray values in the direction of travel. The resulting curve represents the likelihood of the row's position for each x-index within the image. To isolate the most probable offset, two separate methods were compared: 1) a least squares regression of a sinusoidal wave, and 2) a Fourier Fast Transform low-pass filter. Both filtration methods achieved an error of 10 mm in cereals, but performed poorly on sugar beets due to their characteristically large leaf volume. In a similar study by Slaughter et al. (1999), an algorithm for detecting the lateral offset of the row using individual segmentation of plants in the image was proposed. For each plant, a histogram of the intensities was calculated which was then used to find the median offset of each plant. If a plant's median was significantly different than the other plants in the image, it was considered a weed and disregarded. The row offset was then calculated based on the medians of the remaining plants. This method was tested on lettuce and tomatoes for use with a band sprayer operating at 8 km/h, and achieved a standard error of 9 mm and 95th percentile error of 12 mm.

Conversely, oblique perspective methods rely on a camera with a positive angle of inclination and thus multiple rows are present in the field of view. One approach for row estimation with an oblique view utilizes the Hough Line Transform (HLT). In a study by (Pla, 1997), a system using HLT performed with an average error of 18 mm when detecting linear rows in cauliflower and sugar beets. A similar oblique perspective approach using a band-pass filter proposed by Hague et al. (2001) utilized prior knowledge of the spacing of the crop rows of cereals and beets. Supported by the British Beet Research Organization, the system was capable of 3 cm precision at speeds of up to 10 km/h. The project was considered highly successful and was commercialized in 2001 by Garford Farm Machinery (Frognall, Deeping St-James, Peterborough, England, UK) under the name RoboCrop™.

When comparing the vertical and oblique perspective methods, both have advantages and disadvantages. The oblique approach is less sensitive to missing plants and high weed density due to the greater number of visible rows. However, oblique methods rely on prior knowledge of crop spacing and linear rows with low curvature. Additionally, lens distortion is an issue for oblique systems due to the greater subject distance and orientation of the camera. To compensate for increased subject distance, oblique methods require a higher resolution camera, which in turn results in greater computational requirements and costs.

Comparatively, orthogonal systems optimize resolution, in pixels-per-centimeter, and require only basic calibration. However, the reduced field of view for orthogonal systems is a concern when there are significant gaps in the crop rows or heavy weed coverage. To address the issues inherent to the orthogonal approach, a system with two or more cameras may provide sufficient redundancy. Additionally, cameras can be oriented width-wise to the crop row to increase the effective field of view along the direction of travel (Slaughter, 1999).

3.1.3 Hitch Steering Systems

Vehicle steering has received tremendous interest and tractor steering systems in modern tractors have achieved precision of less 2.5 cm. However, due to additional dynamics which act on the implement, such as soil forces and topography, solely relying on tractor steering is not applicable in all environments. Therefore, row sensor feedback is often used in conjunction with a hitch steering system. Several different varieties of hitch steering systems exist, including parallelogram hitches, side-shift (push) hitches, pivoting (articulated) hitches, and rotating stabilizers (Figure 3.3). Actuated hitch systems which rely on disc-steering, e.g. pivoting hitch or rotating stabilizers, and side-shift systems have both seen commercial success (Thacker, 1995).

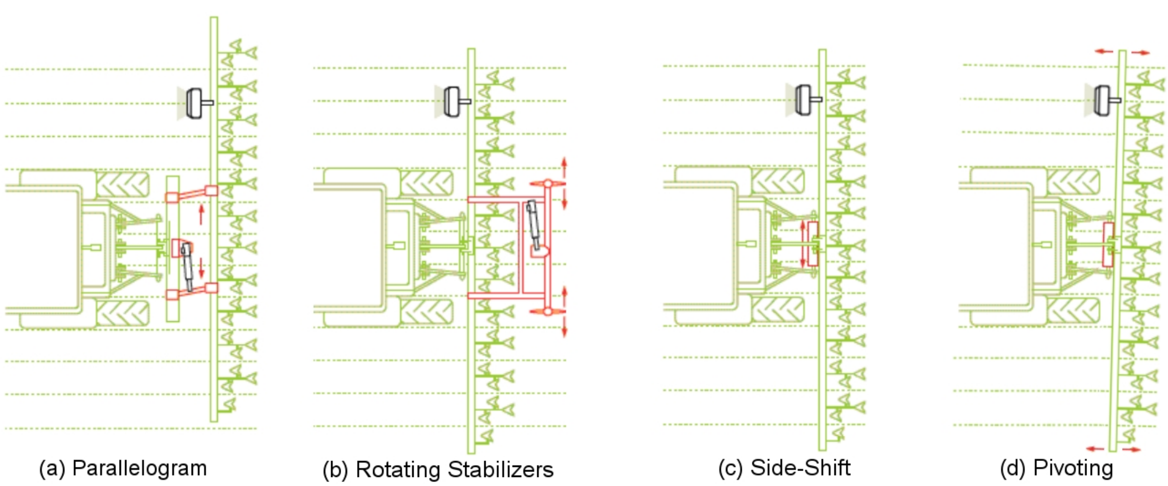

Figure 3.3 Common styles of hydraulic hitch steering systems. Adapted from Möller (2010).

For light cultivation, disc-steer and side-shift style control have been demonstrated to be effective at speeds under 8 km/h and on flat terrain (Kocher, 2000). However, side-shift and parallelogram systems have been observed to cause problems when the implement is configured for heavier cultivation, e.g. for deep harrows and chisels, due to “jumping”, i.e. the effect of the hitch shifting the tractor. Correcting this problem requires either removing tools or increasing the size of the tractor, both of which may not be practical. Alternatively, disc-steer systems are steered by the lateral force generated by soil resistance on the cultivator discs causing the cultivator to track toward the path of least resistance, e.g. much like that of a rudder on a sailboat. Due to this intuitive mechanism for adjusting the tracking direction of an implement, disc-steer platforms are often preferred by farmers for deeper cultivation practices. However, since disc-steer systems rely on soil resistance, they have been known to exhibit reduced performance in very sandy soils.

3.1.4 Objective

The **objective** of this study was to develop a low-cost and computationally efficient control system to retrofit an electro-hydraulic inter-row cultivator guidance system with computer vision row detection. To be considered effective for commercial use, the lateral error of the guidance system must achieve a 95th percentile lateral error within ±5 cm for travel speeds from 6 to 12 km/h on soy and corn crops from 5 to 20 cm in height.

3.2 Materials and Methods

For field evaluation, a twelve row Hiniker heavy cultivator (Hiniker, Co., Mankato, Minnesota, USA) was equipped with a Sukup Auto-Guide hydraulic steering system (Sukup, Sheffield, Iowa, USA) and drawn by a Fendt Vario 850 (AGCO GmbH, Duluth, Georgia, USA). The cultivator tool-bar was configured for a row spacing of 30 inches and equipped with finger weeders, disc-harrows, and wide tine plows. A computer vision guidance system was developed to interface with the hydraulic solenoid controller and serve as an alternate row detection sensor to the mechanical guiding rods (Figure 3.4). The remaining components of the hydraulic guidance system, including the hitch control module, center-pivot potentiometer (i.e. hitch position sensor), and hydraulic solenoids were left unmodified. This configuration allowed the tractor operator to easily switch between the two modes of row detection between trials during field testing.

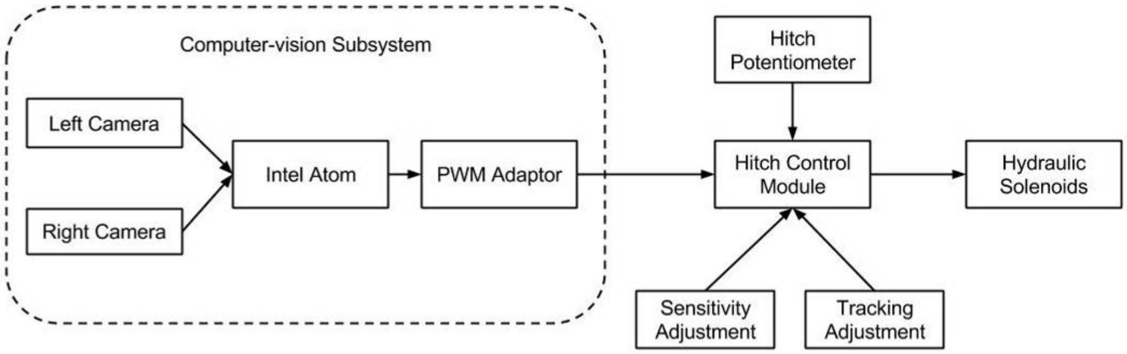

Figure 3.4 Diagram of the computer vision system integrated with hydraulic controller.

Images of the plants passing beneath the cultivator were acquired with two weather-proof cameras: 1/3" Red-Green-Blue (RGB) complementary metal-oxide semiconductor (CMOS) sensor, and aperture of 2.4 (F), and 6.0 mm focal length (Shenzhen Yufei Technology Co., Shenzhen, China). The cameras were attached to the cultivator tool-bar via specially designed brackets which allowed for both lateral and vertical adjustments. In a compromise between the orthogonal and perspective methods, the cameras were mounted at a low-oblique perspective of 30° inclination from vertical, with the wide image axis aligned with the direction of travel, and at a subject depth of 1.0 m (Figure 3.5). This approach provided an extended longitudinal field of view, a resolution of approximately 1 mm/px, and relatively low tangential distortion. In order for the system to tolerate regions of high weed density or gaps in the crop rows, the cameras were installed on the 3rd from center rows of the cultivator tool-bar.

Figure 3.5 Test tractor and adjustable mounting system for camera and guiding rods.

An embedded Linux controller was developed based on the Debian 7.8 operating system and optimized for the dual-core 1.8 GHz Intel Atom D525 Pineview architecture (Intel Corp., Santa Clara, California, USA). The system's hardware consisted of a NC9MGL-525 Mini-ITX motherboard (Jetway Computer Corp., Newark, California, USA) equipped with an 8 GB solid-state drive (SSD) (Mushkin, Inc., Englewood, California, USA) and 2 GB of rapid-access memory (RAM) (Micron Technology Inc., Boise, Idaho, USA). A run-time application and image processing server was developed predominantly using C++, the Python programming language (v.2.7.6) (Python Software Foundation, Welmington, Delaware, USA), and the OpenCV libraries (v.2.4.5) (Itseez, Inc., San Francisco, California, USA). The application functioned as a local server for performing computer vision analysis, handling peripheral devices, and displaying a graphical interface to the tractor operator (Figure 3.6).

Figure 3.6 Graphical front-end for tractor operator. This interface provided a real-time estimate of the row offset and a means for the driver to visually confirm proper tracking of tractor.

Later versions were configured to host a wireless ad-hoc network which allowed operators to tune system parameters via a mobile application. To generate the output voltage signal to the hydraulic control system, an ATmega328P microcontroller (Atmel Corp., San Jose, California, USA) was implemented as an 8-bit pulse-width modulation (PWM) generator with a 490 Hz switching frequency. The microcontroller was controlled via the Universal Serial Bus (USB) as a serial peripheral device with the microprocessor serving as the USB host. This robust platform proved to be a fault-tolerant, cost-effective system with sufficient computing power for real-time analysis of the images from the two cameras.

3.2.1 Plant Segmentation

In order to detect the crop rows, images were captured from the two CMOS cameras at a frame-rate of 25 frames/s. As a pre-processing step, both cameras were configured to manually downscale to a resolution of 320 × 240 pixels. After capturing each image, the image matrix was transformed from the Red-Green-Blue (RGB) color-space to the Hue-Saturation-Value (HSV) color-space in order to simplify color analysis and reduce the complexity of applying band-pass image filters:

$$
H_{i,j} = \begin{cases} 60 \cdot \dfrac{G_{i,j} - B_{i,j}}{V_{i,j} - \min\left(R_{i,i}, G_{i,j}, B_{i,j}\right)} & \text{if } V_{i,j} = R_{i,j} \\ 120 + 60 \cdot \dfrac{B_{i,j} - R_{i,j}}{V_{i,j} - \min\left(R_{i,i}, G_{i,j}, B_{i,j}\right)} & \text{if } V_{i,j} = G_{i,j} \\ 240 + 60 \cdot \dfrac{R_{i,j} - G_{i,j}}{V_{i,j} - \min\left(R_{i,i}, G_{i,j}, B_{i,j}\right)} & \text{if } V_{i,j} = B_{i,j} \end{cases} \tag{3.1}
$$

$$
S_{i,j} = \begin{cases} \dfrac{V_{i,j} - \min\left(R_{i,i}, G_{i,j}, B_{i,j}\right)}{V_{i,j}} & \text{if } V_{i,j} \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.2}
$$

$$
V_{i,j} = \max\left(R_{i,i}, G_{i,j}, B_{i,j}\right) \tag{3.3}
$$

After transforming the image to the HSV color-space, a band-pass plant detection filter (BPPD) was applied to isolate pixels which could represent plant foliage using a fast percentile filtering method similar to the method discussed by Duin et al. (1986). The BPPD filter selects for pixels with hue from yellow-green to blue-green, saturation within percentile bounds, and values (i.e. brightness) between the extremes of under- and over-exposed. Thresholding values were determined empirically using a training set of sample images in varying light and crop conditions. During this process, it was observed that the cameras experienced significant blue-shifting of crop foliage in very bright or low light, so the upper threshold for hue was set well into the cyan-blue region. This change did not have any noticeable negative impact on performance due to the relative absence of blue-tones in soil.

$$
P_n(C) = A_{nN} + \left(nN \bmod 1\right) \cdot \left(A_{\lfloor nN \rfloor + 1} - A_{\lfloor nN \rfloor}\right) \tag{3.4}
$$

where n is the percentile of the sorted array A of length N.

$$
BPPD_{i,j} = \begin{cases} 1 & \text{if } H_{\min} \le H_{i,j} \le H_{\max} \wedge P_a(S) \le S_{i,j} \le P_b(S) \wedge P_{\alpha}(V) \le V_{i,j} \le P_{\beta}(V) \\ 0 & \text{otherwise} \end{cases} \tag{3.5}
$$

where H_{min} = 45 (yellow-green), H_{max} = 105 (blue-green), a was the lower saturation percentile (33rd percentile), b was the upper saturation percentile (99th percentile), α was the lower value percentile (15th), and β was the upper value percentile (95th).

The BPPD filter utilized the linear interpolation percentile function (Equation 3.4) to calculate the upper and lower thresholds of the value and saturation channels. This approach eliminated the need for static limits and reduced false-positive classification of pixels under varying lighting conditions. As a final post-processing step, a morphological opening with a 3 by 3 elliptical Gaussian kernel was applied to the BPPD mask to minimize any remaining noise while preserving the structure of crop foliage (Figure 3.6). This three step process was found to be computationally non-intensive, yet produced sufficient segmentation in diverse lighting conditions. Notably, the percentile-based band-pass filters of saturation and intensity produced reliable masks for the majority of the worst-case scenarios of poor exposure and shadows (Figure 3.6).

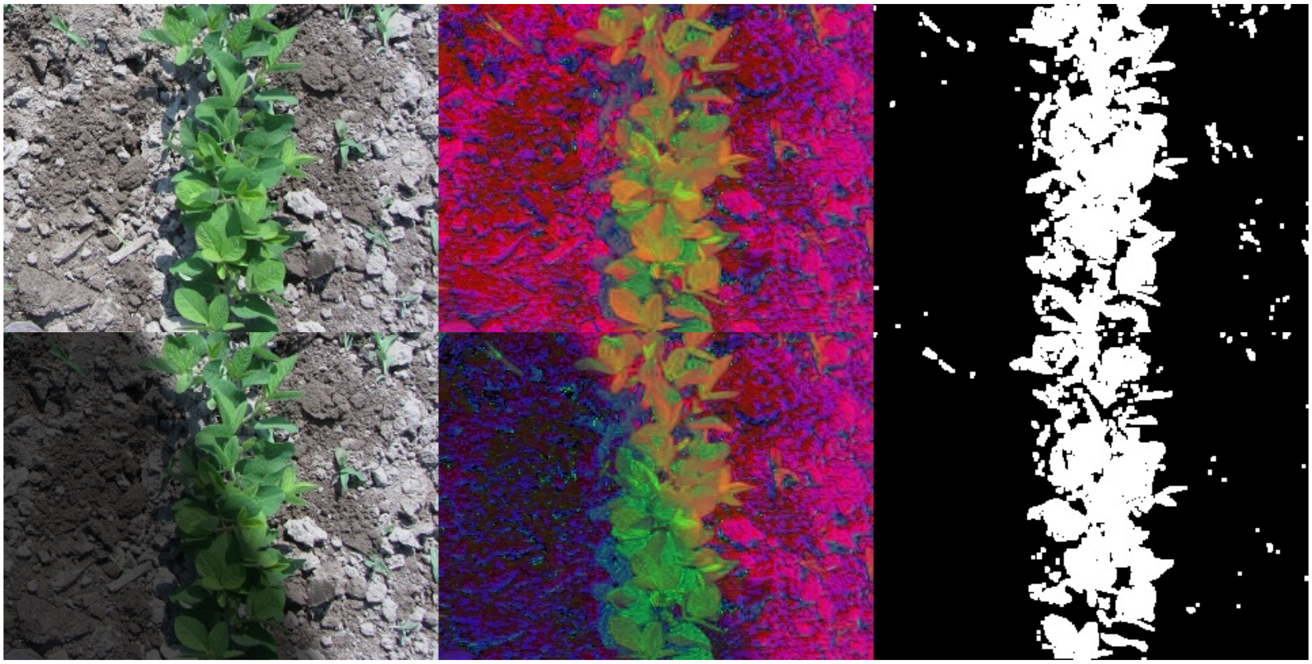

Figure 3.6 BPPD algorithm applied to image with diffuse lighting (above) and non-uniform lighting (below).

3.2.2 Row Estimation

Due to the non-Gaussian distribution of crop foliage in an image, a two-camera weighted histogram filter (referred to here as 2CWH) was used to estimate the lateral offset of the crop row. First, the column summation of the BPPD mask (M) was calculated in the direction of travel resulting in an array (*C*) representing the lateral distribution of plant foliage within the image:

$$C_i = \sum_{j=0}^{H} M_{i,j} \tag{3.7}$$

where *H* is the height of the image (pixels), and M is the binary matrix produced by the BPPD filter.

Indices of the array with low values suggest bare-soil, moderate values suggest sparsely distributed weeds, and higher values suggest presence of the crop row due to the longitudinal alignment of the plant foliage. Using this distribution, the centroid of the crop row was estimated by applying a high-pass percentile threshold to select for indices which demonstrated significantly greater longitudinal alignment than others. For the camera resolution of 1 mm/px and 480 mm field-of-view, the 95^{th} percentile corresponded to a width of 26 mm, i.e. 1 inch rounded to a resolution of 2 mm:

$$p_i = \begin{cases} i & \text{if } C_i \geq P_{\text{¥}}\left(C\right) \\ 0 & \text{otherwise} \end{cases} \tag{3.8}$$

where P is the percentile function, and y is the percentile threshold (95th percentile).

The resulting array (p) consists of all indices of the image which are most likely to represent the crop row. The estimated lateral of the crop row for each camera was estimated by taking the weighted centroid of the probable indices, where the weight of each index was the normalized value of its corresponding column summation:

$$x = \frac{\sum_{i=0}^{N} p_i \cdot C_i}{\sum_{i=0}^{N} C_{p_i}} - \frac{W}{2} \tag{3.9}$$

where N is the number of elements in p, and x is the position of the estimated centroid (in pixels).

To compensate for errors in the detection process inherent to single camera systems, the row centroid estimation process was repeated for each image, producing two column summation arrays (C_1 and C_2) and two estimated centroids of lateral offsets (x_1 and x_2). After calculating the row centroid for each camera, the magnitude of the column summation values for each lateral offset were compared to determine the final estimated offset of the crop row:

$$e_t = \begin{cases} \text{mean}(x_1, x_2) & \text{if } \left|x_1 - x_2\right| \leq \varepsilon \\ \text{argmax}\left(\max\left(C_1\left(x_1\right), C_2\left(x_2\right)\right)\right) & \text{otherwise} \end{cases} \tag{3.10}$$

where ε is the maximum acceptable error tolerance (30 pixels), and e_t is the estimated offset error at time t.

This method for row detection prioritized estimations from the two cameras which were in agreement. In the event of a significant difference between the two estimates, the dominant centroid was assumed to be the best estimate of the lateral offset. The 2CHF approach provided an effective means for reducing errors due to weeds and row gaps which was computationally non-expensive (Figure 3.7).

Figure 3.7 Row estimation by histogram thresholding. The red dots indicate the value of the column summation array and the green line represents the resulting estimate of the crop row centroid. The position of the green line relative to the crop row illustrates the effectiveness of the 2CHF method for determining the center of the crop row.

For performance evaluation, the lateral error of the crop row in pixels was converted to centimeters based on the mounting configuration of the camera. For cameras at a subject depth of 1.0 m measured from the lens to the image centroid, the horizontal field-of-view was determined empirically to be approximately 48 cm when measured laterally along the center-line. For an image width of 240 px, this resulted in a resolution of 0.2 cm/px. Due to the semi-orthogonal perspective of the camera and narrow region of interest (±25 px from center), tangential distortion and radial distortion was assumed to be insignificant along the lateral axis.

3.2.3 Electro-Hydraulic Control

Hydraulic steering of the cultivator was achieved by actuating two 0.75 m stabilizers mounted 1.65 m behind the cultivator tool-bar and spaced at a distance of 1.48 m apart (Figure 3.8). The hydraulic actuation of the stabilizers had a ±75° angular range of motion and a max angular velocity of 22°/s. The two stabilizers were actuated via a hysteresis-style hydraulic-solenoid controller. Actuation of the hydraulic system was determined by the voltage differential between the feedback signal from the row sensor, either the guiding rods or computer vision system, and a rotary potentiometer mounted to the active mechanism of the hitch (Figure 3.4).

Figure 3.8 Rotating stabilizers of cultivator steering system. Both stabilizers move in tandem and are actuated by a single hydraulic cylinder.

For the computer vision module, signal conditioning of the output was implemented based on a discrete Proportional-Integral-Derivative (PID) feed-back controller, where the tuning coefficients were initially chosen to provide similar response to that of the guiding rods and were subsequently modified by trial and error:

$$u(t) = K_P e_t + \frac{K_I}{N}\sum_{i=0}^{N}\left(e_{t-i}\right) + \frac{K_D}{M}\sum_{j=0}^{M}\left(e_{t-j} - e_{t-j-1}\right) \tag{3.12}$$

where u is the output value, e is the array of estimated offset values stored in memory, K_P = 1.0, K_I = 4.0, K_D = 0.5, N = 16 is the number of integral samples, M = 8 is the number of differential samples,

The output value was transmitted as character commands at a polling frequency of 16 Hz to the PWM controller via a weatherproof (IP68) universal serial bus (USB) interface. The microcontroller used in this study operated at a logic-level of 5.0 VDC and therefore was capable of generating a 0.0 - 5.0 VDC PWM output signal. However, the input range of the Sukup Auto-Guide system was observed to be 0.10 V to 8.0 V with a fixed supply of 9.7 Volts. To account for this discrepancy in voltage levels between the systems, a metal-oxide semiconductor field-effect transistor (MOSFET) logic-level converter circuit (Figure 3.9) was implemented to scale the PWM signal from 5.0 VDC to 9.7 VDC.

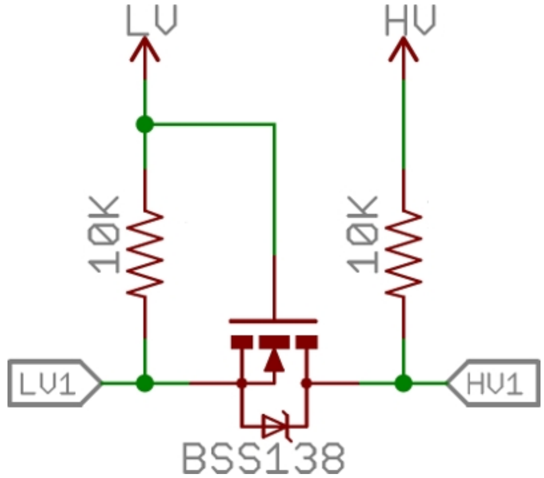

Figure 3.9 MOSFET logic level converter circuit for adjusting voltage levels between the PWM adaptor and the hydraulic hitch controller.

The output voltage corresponding to a given bit-value can be calculated with the following conversion:

$$w = \begin{Bmatrix} 0 & \text{if } u \le 0 \\ 2^{R-1} & \text{if } u \ge 2^{R-1} \\ \left(u + 2^{R-2}\right) \cdot \left(\dfrac{V_{\max}}{V_{HV}} + \dfrac{V_{\min}}{V_{LV}} \right) & \text{otherwise} \end{Bmatrix} \tag{3.13}$$

$$V_{out} = \frac{w}{2^{R-1}} \cdot V_{HV} + V_{min} \tag{3.14}$$

where V_{min} = 0.10, V_{max} = 8.00, V_{HV} = 9.70, V_{LV} = 5.0

This circuit configuration allowed for a resolution of 200 bits and enabled the computer vision subsystem to interface with hydraulic controllers of differing voltage requirements. Notably, the Sukup Auto-Guide was an analog controller, whereas not all data-acquisition systems (DAQ) are intended for reading PWM signals. In the event of interfacing with a digital hydraulic controller which does not support a PWM input signal, digital-to-analog conversion can be achieved with a low-pass filter (Alter, 2006). Due to the very low impedance requirements of DAQ systems, recommended values for resistance and capacitance are 1.5 Ω and 470 μF, respectively. This configuration results in a first-order system with a time constant of 0.0007 seconds, cutoff frequency of 226 Hz, and a gain of -5.93 dB at 490 Hz, and therefore causes minimal bandwidth-loss for the PWM signal.

3.2.4 Camera Calibration

Before conducting each set of field trials, the following camera calibration procedure was followed to ensure proper alignment of the cameras: 1) the cultivator was aligned with the crop rows which was verified by measurements with a tape measure at the working tools; 2) lateral adjustments were made to the camera bracket to ensure the vertical center-line of each camera was aligned with the crop row; 3) vertical adjustments made to the camera bracket to ensure a subject depth of 1.0 m when measured in a direct line-of-sight from the camera lens to the soil surface. In addition to setup of the camera bracket, all tractor hydraulics were set to their default settings. The hydraulic controller settings were adjusted via the hydraulic control modules user interface which provided both sensitivity and tracking adjustment inputs (Figure 3.10). The sensitivity adjustment effectively mapped the output range between the sensor voltage and radial resolution of the stabilizers, with a range of 1 to 10 resulting in a linear relationship from 7.7°/V to 18.8°/V, respectively. Similarly, the tracking adjustment offset the zero position of the stabilizers, with a scale of -3 to +3 corresponding to -25° to +25°, respectively. Therefore, at the beginning of each set of trials the following settings were ensured: 1) the sensitivity was set to 10 out of 10, and 2) the tracking adjustment was set to 0.

Figure 3.10 Control module for the Auto-Guide hitch steering system (Sukup, Sheffield, Iowa, USA).

3.2.5 Field Trials

Field tests of the system took place over the summer of 2014 from June to August on straight-drilled corn and soybean crops. Only organic crops were considered in this study. No pesticides were applied to the fields and varying degrees of weed coverage were present during the trials. All fields used for testing were maintained by Agri-Fusion 2000, Inc. (St-Polycarpe, Québec, Canada), a 2500 hectare organic operation. Trials consisted of randomized, single passes across the field from headland to headland. For each pass, five sample points representative of the crop stage were selected and the height of crops at each point was estimated by measuring the distance from the soil surface to the tallest leaflet using a tape measure. The average of the five samples was used to determine the height classification in 5 cm groups, e.g. 0 - 5 cm and 5 - 10 cm, etc. In order to determine the reliability of the two systems at differing speeds of cultivation, trials were conducted at four approximate travel speeds: 6, 8, 10, and 12 km/h. Throughout each trial the following data was logged to file: 1) the lateral error estimated by the computer vision system, 2) RTK-GNSS coordinates, 3) PWM voltage output, and 4) time which the image was captured to within 1 ms. Figure 3.11 presents the estimated lateral error for two different trials of the same crop, stage and travel speed.

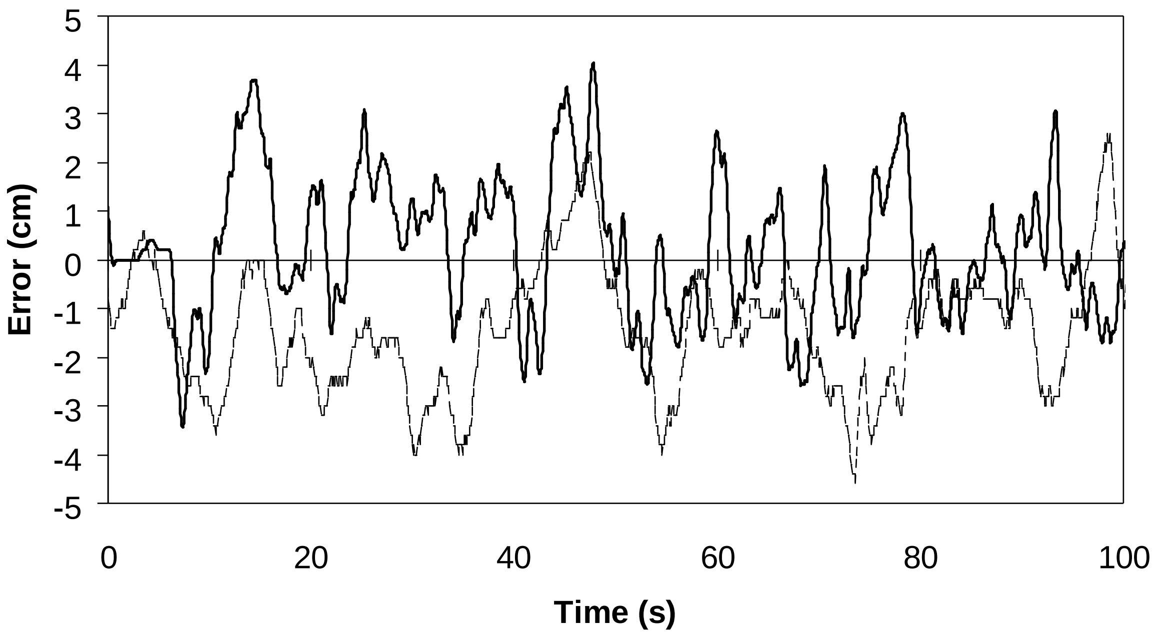

Figure 3.11 Two computer vision trials (10 - 15 cm soy at 8 km/h) conducted on different fields.

Prior to each trial, both the tractor and cultivator implement were aligned with the crop row to the best of the operator's ability. Once aligned, either the computer vision (CV) or guiding rod (GR) sensor were connected as the control input to the hydraulic module (Figure 3.10). Once ready, the system logger was started, the travel speed was set via the automatic speed controller of the vehicle, and the tractor was engaged into gear. During all trials, the tractor was manually operated by a professional driver without the aid of auto-steering. Due to farm management restrictions with respect to crop health and best management practices for cultivation, some combinations of crop stage and travel speed were not tested extensively, e.g. travel speed of 12 km/h was not tested at the <10 cm height stage due to the high probability of causing excessive damage to seedlings.

3.3 Results

Table 3.1 presents a summary of all trials conducted, including duration of the trial and estimated crop height. As can be seen, there is noticeably lower performance with respect to both RMSE and 95th percentile error for guiding rods when used on crops at the earlier stages of growth.

Table 3.1 Summary of all trials

Treatment	Crop Stage	Height (cm)	Speed (km/h)	Duration (s)	RMSE (cm)	95th Percentile (cm)
Computer Vision (CV)	Corn (>15 cm)	20	6	55	1.7	3.0
		15	8	81	4.3	5.2
		15	8	112	1.5	3.0
		25	8	140	1.9	3.6
		20	10	189	1.7	3.4
		20	10	225	1.7	3.0
		20	12	193	2.2	4.0
	Soy (<10 cm)	5	6	334	0.5	0.6
		5	6	149	0.9	1.2
		5	6	431	0.6	1.0
		10	6	476	1.5	2.6
		10	6	375	1.3	2.5
		10	6	71	1.7	3.8
		10	8	80	2.1	4.0
		10	10	61	2.1	3.8
		10	10	50	1.4	2.4
	Soy (10 - 15 cm)	15	6	313	1.0	1.8
		15	8	360	1.8	2.8
		15	8	313	1.6	2.8
		15	8	139	3.6	7.0
		15	10	380	1.5	3.0
		15	12	418	1.7	2.8
	Soy (>15 cm)	25	8	437	2.5	3.4
		20	10	360	2.1	3.0
		20	12	353	1.7	3.0
		25	12	98	2.5	3.6
		25	12	298	2.8	5.4
		20	12	205	1.9	3.4
Guiding Rods (GR)	Corn (>15 cm)	20	6	167	1.6	3.0
		20	6	338	3.7	6.8
		15	8	195	2.0	3.8
		15	8	211	1.7	3.0
		25	10	274	2.0	3.8
		20	12	128	2.3	4.4
	Soy (<10 cm)	10	8	84	3.4	7.0
		10	6	58	2.6	4.6
		10	8	84	3.5	5.6
		10	6	76	3.5	6.6
		10	10	101	5.0	10.2
	Soy (10 - 15 cm)	15	8	140	3.2	5.8
		15	8	100	4.2	7.0
		15	10	145	2.9	6.0
		15	10	138	4.6	8.4
		15	12	96	5.0	9.8
	Soy (>15 cm)	20	6	556	2.0	4.0
		20	6	625	1.0	2.4
		20	10	344	1.5	2.4
		25	10	156	1.3	2.8
		25	12	107	1.7	3.0
		25	12	146	2.8	6.2
		25	12	104	1.4	2.6
		25	12	153	1.9	3.0

Table 3.2 presents 95th percentile and RMSE values for trials grouped into four crop-stages: Soy <10 cm, Soy 10-15 cm, Soy >15 cm, and Corn >15 cm. With respect to RMSE, both methods for row detection performed sufficiently for field use, with an average RMSE of 2.88 cm and 1.90 cm for all trials using computer vision and guiding rod systems, respectively. However, since the guidance error within a given trial can be said to be non-parametric, the 95th percentile should be considered as a more reliable metric of the system performance with respect to agricultural guidance system standards (ISO, 2012).

Table 3.2 RMSE and 95th Percentile with respect to crop species and height

	RMSE (cm)				95th Percentile (cm)			
Treatment	Soy (<10 cm)	Soy (10 - 15 cm)	Soy (>15 cm)	Corn (>15 cm)	Soy (<10 cm)	Soy (10 - 15 cm)	Soy (>15 cm)	Corn (>15 cm)
CV	1.36	1.87	2.22	2.15	2.43	3.37	3.63	3.60
GR	3.61	3.98	1.70	2.22	6.80†	7.40†	3.30	4.13

† exceeds the acceptable lateral error for cultivation (±5.0 cm), indicating possible damage to crops

Table 3.3 presents the pair-wise t-test of 95th percentile errors grouped by crop-stage with respect to the row sensor used. As can be seen, there was a significant decrease in error when using the guiding rods for crops greater than 15 cm in height.

Table 3.3 Pair-wise t-test of 95th percentile errors for computer vision (CV) compared to guiding rods (GR)

		CV			GR				
Crop	Stage (cm)	Mean (cm)	Std error (cm)	Trials	Mean (cm)	Std error (cm)	Trials	Mean diff. (cm)	p-value
Soy	< 10	2.87	1.08	7	6.8	2.12	5	-3.93	0.009*
Soy	10 - 15	3.37	1.83	6	7.4	1.69	5	-4.03	0.010*
Soy	> 15	3.63	0.90	6	3.3	1.28	8	0.33	0.297
Corn	> 15	3.60	0.80	7	4.1	1.41	6	-0.53	0.225

* significant at a confidence of α = 0.05

Table 3.4 presents the results of multiple-comparison analysis by Tukey's Honest Significant Difference (HSD) test with respect to the row sensor used. When disregarding crop type and grouping trials by height alone, the two row detection methods showed no significant difference at the 10 - 15 cm crop height.

Table 3.4 Tukey HSD multiple comparison of 95th percentiles (irrespective of crop species)

	Group 1		Group 2				
	Treatment	Stage (cm)	Treatment	Stage (cm)	Mean diff. (cm)	Std error (cm)	p-value
CV vs. CV	CV	< 10	CV	10 - 15	0.78	0.30	0.454
	CV	< 10	CV	> 15	0.72	0.28	0.459
	CV	10 - 15	CV	> 15	-0.06	0.29	0.900
CV vs. GR	CV	< 10	GR	< 10	2.26	0.34	0.001*
	CV	< 10	GR	10 - 15	1.81	0.30	0.001*
	CV	< 10	GR	> 15	0.62	0.28	0.603
	CV	10 - 15	GR	< 10	1.48	0.35	0.052
	CV	10 - 15	GR	10 - 15	1.03	0.31	0.197**
	CV	10 - 15	GR	> 15	-0.16	0.29	0.900
	CV	> 15	GR	< 10	1.54	0.33	0.024*
	CV	> 15	GR	10 - 15	1.09	0.29	0.100
	CV	> 15	GR	> 15	-0.10	0.26	0.900
GR vs. GR	GR	< 10	GR	10 - 15	-0.45	0.35	0.900
	GR	< 10	GR	> 15	-1.64	0.33	0.014*
	GR	10 - 15	GR	> 15	-1.19	0.29	0.057

* significant at a confidence of α = 0.05

** not significant when classified by crop height alone (see Table 3.3)

3.4 Discussion

Overall, the guiding rods exhibited significantly greater 95th percentile error relative to the computer vision system for the <10 cm and 10 - 15 cm crop stages of soy (Table 3.1). However, the accuracy of the guiding rods increased dramatically as the plants matured, resulting in a significant decline in both 95th percentile and RMSE (Table 3.2). Conversely, the computer vision system demonstrated a slight increase in 95th percentile error as crop height increased. This effect is possibly attributable to the greater crop foliage area of mature plants reducing the precision of row centroid estimation. However, the correlation between crop height and 95th percentile of the computer vision system was determined to not be statistically significant (Table 3.4).

Linear regression of 95th percentile values revealed that the performance of the guiding rod system decreased as a function of travel speed for the <10 cm and 10 - 15 cm stages (Figure 3.12). Conversely, the computer vision system did not exhibit the same effect for the tested range of 6 km/h to 12 km/h. These results suggest that in order to achieve a 95th percentile error of less than 5.0 cm when using guiding rods with underdeveloped crops it is necessary to reduce the speed of cultivation. This conclusion corresponds to the reported experiences of tractor operators when using the guiding rod systems with smaller plants. The positive correlation between 95th percentile error and travel speed is attributable to the increased probability of the guiding rods losing the crop row when traveling at speeds in excess of 8 km/h. Due to the mechanical action of row estimation via guiding rods, higher speed cultivation causes the rods to exert greater stress on underdeveloped plants, thus resulting in poor tracking and damage to the seedlings. Therefore, it can be asserted that the computer vision method for row detection is a more robust solution than mechanical guiding rods with respect to early season cultivation.

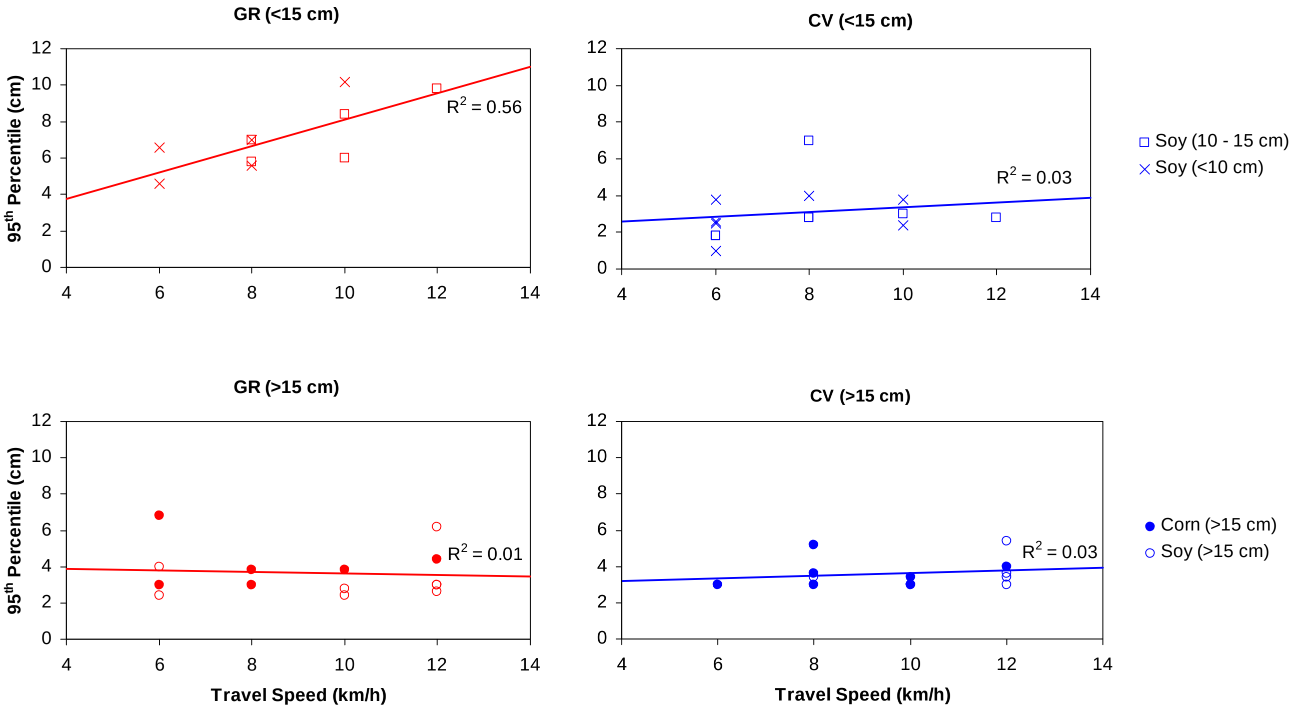

Figure 3.12 Guiding rod (GR) and computer vision (CV) 95th percentile errors with respect to travel speed and grouped by crop height.

3.4.2 Future Improvements

For this study, row offsets were determined by analyzing the plant foliage mask in the direction of travel with a relatively small subject distance and cameras mounted at a low-oblique perspective (Figure 3.2). Due to the low angle of inclination, close subject depth, small region of interest, and orthogonal method for row estimation it was not necessary to incorporate tangential or radial distortion correction. However, for alternate usage cases, e.g. cameras aligned at the mid-point between rows or at subject depths greater than 1.0 m, it would be necessary for tangential and radial distortion correction to be implemented (Figure 3.13).

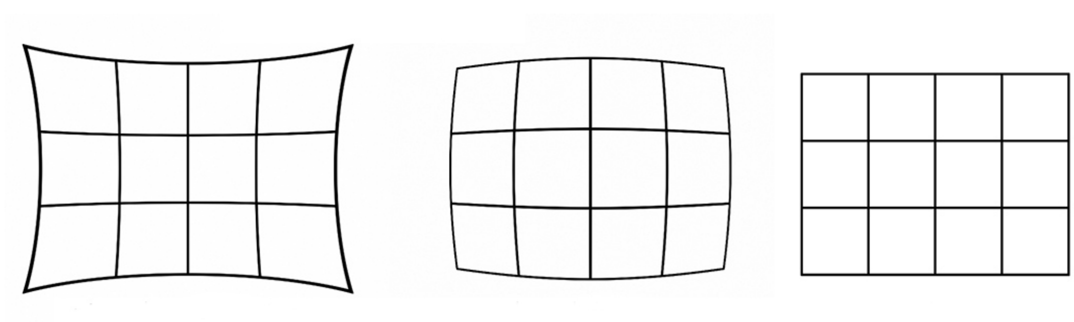

Figure 3.13 Illustration of radial lens distortion. Concave distortion (left) is commonly referred to as pincushion distortion, whereas concave distortion (center) is commonly known as barrel distortion.

Due to the variable nature of how computer vision row detection systems are installed, a versatile method for in-field camera calibration, such as that demonstrated by Lee et al. (2009), may be well suited to address distortion correction. The method utilizes a checkerboard pattern which is placed in the camera's field of view. An image is captured and the positions of the corners are identified. Using the known size of the squares, the tangential and radial distortion coefficients (P_n and K_n, respectively) can be determined empirically. Once the distortion coefficients are known, each newly captured image can be rectilinearized using the Brown-Conrady model of distortion correction (Brown, 1966):

$$
\begin{aligned}
x_d &= x_u \cdot \left(1 + K_1 r^2 + K_2 r^4 + \ldots\right) + \left(P_2 \cdot \left(r^2 + 2x_u^2\right) + 2P_1 x_u y_u\right) \cdot \left(1 + P_3 r^2 + P_4 r^4 + \ldots\right) \\
y_d &= y_u \cdot \left(1 + K_1 r^2 + K_2 r^4 + \ldots\right) + \left(P_2 \cdot \left(r^2 + 2y_u^2\right) + 2P_1 x_u y_u\right) \cdot \left(1 + P_3 r^2 + P_4 r^4 + \ldots\right) \\
r &= \sqrt{\left(x_u - x_c\right)^2 + \left(y_u - y_c\right)^2}
\end{aligned}
\tag{3.15}
$$

where (x_d, y_d) is the distorted image point as project on image plane using lens, (x_u, y_u) undistorted image point project by ideal pin-hole camera, (x_c, y_c) is the distortion center, K_n is nth radial distortion coefficient, P_n is the nth tangential distortion coefficient.

With respect to the style of hydraulic hitch, this study was restricted to a rotating-stabilizer guidance system where the cultivator tool-bar was parallel to the tractor axle at all times. When cultivating straight-drilled rows, the cameras were effectively aligned with the direction of travel. As such, the proposed method for row detection discussed in this paper is only appropriate for both rotating stabilizer and center-shift steering systems where the camera's field of view is in-line with the crop row under most conditions. However, several commercially available cultivator guidance systems are based on pivoting hitch and parallelogram style systems (Figure 3.3). Preliminary testing of the row detection method proposed in this work has been undertaken for the AcuraTrak 3G pivoting hitch (Sunco Manufacturing, Inc., North Platte, Nebraska, USA) with generally positive results.

In pivoting hitch systems, the cultivator tool-bar is rotated about a central pivot-point to guide the implement in the desired direction by generating lateral adjustment forces on cultivator discs. This design can eliminate the "tail-out" effect which is caused when the tractor is following a curved path (Thacker, 1996). However, pivoting hitch steering systems may rotate the cultivator tool-bar as much as ±10°. For cameras mounted to the tool-bar, this pivoting action effectively changes the apparent orientation and lateral error of the crop row. To compensate for this effect, knowledge of the camera's position relative to the pivot point is required as well as the camera's instantaneous orientation relative to the direction of travel. Although the cultivator orientation may be determined via sensor feedback from the hitch itself (e.g. via a rotary encoder), an approach which utilizes the information already provided by a camera is an attractive option. Methods such the Hough Line Transform or feature-based visual tracking of the images (i.e. keypoint tracking) may prove to be robust solutions which do not require additional sensors and would therefore be system-agnostic. For example, by tracking the motion of the ground moving beneath the cultivator, the orientation of the camera relative to the crop rows can be estimated and subsequently compensated for (Appendix C).

Due to the inherent nature of agriculture control systems, numerous uncontrollable variables exist which can have a detrimental effect on system performance, e.g. soil conditions, terrain slope, hydraulic pressure, and travel speed. These dynamic external factors can ultimately impact the performance of the system. As seen previously in Figure 3.11, two computer vision trials which were conducted on the same crop-stage and travel speed but on different fields demonstrated noticeably different behaviors, specifically with respect to bias and response time.

To account for non-linearity and variability inherent to implement guidance, an adaptive control system would be a viable solution to reduce the need for regular re-calibration (e.g., when switching between a disc-steer and side-shift hitch system), human intervention, or additional feedback sensors on implements, e.g. fifth wheels or tilt-sensors. Reinforcement learning techniques, such as Q-learning (Watkins, 1992), are well suited to this style of control system and have produced successful results for robotics and automotive applications with variable load requirements, such as DC motor control (Aziz, 2015) and active suspension systems (Chiou, 2012). The Q-learning algorithm is primarily intended for applications which require uncalibrated control of non-linear, multiple-input, multiple-output systems. Q-learning is based on the principal of a reward mechanism, i.e. for a given action the resulting behavior of a system can be automatically classified by the system and subsequently rewarded or penalized. Actions which result in positive behavior for a given state of the system (e.g., adjusting gain offsets due to bias) are incentivized and those which resulted in negative behavior (e.g., high gains resulting in overshooting) are penalized. Ultimately, the learning process produces a non-linear response matrix which adapts to the current working environment of the system.

A particularly interesting subset of modern control systems, known as fuzzy logic controllers (FLCs), have been investigated for facilitating PID tuning (Visioli, 1999) and for systems which adaptively self-tune their parameters (Güzelkaya, 2003). Unlike classical PID controllers, FLC systems incorporate fuzzy logic either to classify the behavior of a system or to adjust gain coefficients in order to optimize performance (Boubertakh, 2010). Q-learning has been demonstrated to be a viable complement to FLCs in non-linear, complex systems, such as active automotive suspension systems (Chiou, 2012) and variable-load DC motor control (Aziz, 2015). Applications of adaptive controllers in agriculture have the potential to improve performance and reduce calibration when working with varying implement configurations and field conditions.

3.5 Conclusions

The dual-camera BPPD row estimation method proved to be an effective method for plant segmentation in various lighting and crop conditions, including in the presence of weed coverage. Using computer vision as sensor input for a rotating-stabilizer hydraulic steering system significantly improved the precision of inter-row cultivation at the early stages of crop growth. The computer vision system outperformed the guiding rods at the <10 cm stages of soy and corn, with a mean improvement of 2.26 cm. When controlling for different crop species, the computer vision system outperformed the guiding rods at the <10 cm and 10 - 15 cm stages of soy, with a mean differential of 3.93 and 4.03 cm, respectively. However, at the later stages of growth, i.e. soy and corn >15 cm, there was no significant difference in performance between the two guidance systems. Thus, we conclude that computer vision row sensors can be incorporated as low-cost embedded systems to improve the precision of critical early-season cultivation operations, but that mechanical guiding rods continue to be a robust and sufficient method for detection in the later stages of cultivation.

Chapter 4. Feature-based Visual Tracking

4.1 Introduction

Visual tracking has the potential to serve as valuable sensory feedback for agricultural platforms which utilize digital imaging. Possible applications include adaptive responsiveness for control systems which may require adjusting operational parameters, e.g. to compensate for overshooting or time solenoid activation on cultivator hitches or sectional sprayers, respectively. Alternatively, visual tracking may be employed to correct for poorly aligned cameras in the case of side-shift and rotating stabilizer hitch systems, or mitigate row detection error due to the motion of a pivoting-hitch steering system. As such, visual tracking has a variety of possible applications, but first it is important to verify whether this approach is reliable enough for use in agricultural systems which are off-road and in the presence of significant ground coverage.

In order to identify the motion vectors of an agricultural platform, such as a cultivator implement, the visual tracking algorithm must be capable of finding and matching distinctive features between images regardless of varying agricultural surfaces and travel speeds. Examples of distinct features include blobs, corners, and edges. A distinctive feature can generalized as a sub-region of an image which exhibits maximum variation when the window of consideration is shifted in all directions about the point. The process of searching for such keypoints is known as Feature Detection. After a set distinct features have been located in an image, next the region around each feature must be described. The process of generating descriptive parameters for a feature is known as Feature Description. Applications of visual tracking have commonly used histogram-based feature detectors and descriptors, like SIFT and SURF, but recently introduced binary descriptors, such as ORB, have been shown to offer similar performance at lower computational cost (Hartmann, 2013). Once a set of features and their descriptors have been generated for both images, a clustering algorithms, such as k-Nearest Neighbors can be employed to find matches between the two sets. By calculating the displacement of keypoint pairs between images and the precise time differential between each image, the motion vectors of objects (or the ground surface, in the case of visual tracking) can be estimated.

4.1.1 SIFT Feature-Descriptor

The Scale-Invariant Feature Transform (SIFT), is feature detector and descriptor algorithm proposed by Lowe (2004), was a significant improvement over previous methods and remains one of the most popular algorithms due to its excellent accuracy and reliability. SIFT uses the Laplacian of Gaussian (LoG) which acts as a blob detector for finding distinct blobs of different sizes by varying a scaling parameter (σ). LoG searches for the local maxima at varying scales which provides a list of (x,y,σ) values. However, LoG is computationally expensive, so the SIFT algorithm detects local extrema of the image after applying a Difference of Gaussians (DoG) filter as an approximation of LoG (Equation 4.1). DoG is calculated by obtaining the difference of Gaussian blurring of an image with two different σ-values, i.e for σ and $K\sigma$. This process is conducted for different octaves of the image in a Gaussian Pyramid. This approach allows SIFT to perform well regardless of significant image noise compared to other detectors (Rosten, 2010).

$$DoG_{k,\pm}(x,y) = G(x,y,k\pm) - G(x,y,\pm) = \frac{1}{2\circledR(k\pm)^2}\cdot e^{-\left(x^2+y^2\right)/2(k\pm)^2} - \frac{1}{2\circledR(\pm)^2}\cdot e^{-\left(x^2+y^2\right)/2(\pm)^2} \tag{4.1}$$

SIFT utilizes a Taylor series expansion of the scale-space to achieve more accurate location of extrema, but if the pixel intensity at a particular extrema is below a contrast threshold value, given as 0.03 by Lowe (2004), it is rejected. An attribute of DoG is that edges exhibit a greater extrema response than other features, e.g. blobs or corners, so edges are discarded using a concept similar to the Harris Corner Detector: A 2 × 2 Hessian matrix (H) is used to compute the principal curvature of each feature and if one eigenvalue is greater then the other by an edge threshold, given as 10 by Lowe (2004), then the keypoint is discarded.

For feature description, a 16 × 16 neighborhood around each feature is broken into 16 sub-neighborhoods of 4 × 4 pixels. For each sub-neighborhood, the orientation histogram is calculated for 8 bins to produce a feature descriptor vector with 128 values. Lastly, several metrics are applied to each keypoint in order to eliminate any low-contrast keypoints and edge keypoints and only keep a predetermined number (*N*) of the best keypoints (x,y) and their corresponding 128-dimensional feature descriptor vectors.

4.1.2 SURF Feature-Descriptor

Speeded-Up Robust Features (SURF) was developed as an improvement upon the SIFT algorithm in order to provide similar accuracy but with significantly faster computational speeds (Bay, 2008). Like SIFT, SURF is a scale and rotation-invariant feature detector and descriptor. Convolution with the Gaussian second-order derivatives in SIFT via LoG is computationally expensive, so SURF offers improved computational speed over SIFT by approximating LoG with box filters and using pre-computed integral images (Figure 4.1). SURF is capable of almost real-time computation without loss in performance which represents an important advantage for many computer vision applications (Bay, 2008).

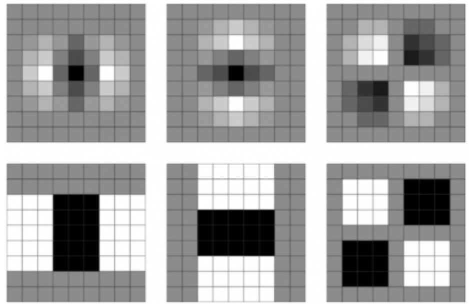

Figure 4.1 Gaussian filters and their corresponding box-filters (σ = 1.2) used by the SURF algorithm.

For feature detection, SURF uses the fast Hessian detector to locate possible keypoints (Equation 4.2). Keypoints with candidates scores (Equation 4.3) less than a pre-determined value, referred to as the Hessian threshold, are rejected, and Hessian thresholds from 500 to 2000 are typical. Next, SURF uses the sums of Haar wavelet responses about each keypoint (Equation 4.4) to generate feature descriptors. This method demonstrates excellent performance compared to other state-of-the-art methods (Bay, 2008). A variant of SURF, unknown as Upright SURF (U-SURF) uses the same methodology for feature-detection but does not calculate the orientation of each keypoint by convolution of two Haar wavelets and saves significant processing time and is robust up to ±15° (Bay, 2008). A square region about each keypoint is split into 4 × 4 pixel sub-regions and a feature vector is applied:

$$H(x,y,\pm)=\begin{bmatrix} \frac{\partial^2}{\partial x^2}G(\pm)\cdot I(x,y) & \frac{\partial}{\partial x}\frac{\partial}{\partial y}G(\pm)\cdot I(x,y) \\ \frac{\partial}{\partial x}\frac{\partial}{\partial y}G(\pm)\cdot I(x,y) & \frac{\partial^2}{\partial y^2}G(\pm)\cdot I(x,y) \end{bmatrix} \cong \begin{bmatrix} L_{xx}(x,y,\pm) & L_{yx}(x,y,\pm) \\ L_{xy}(x,y,\pm) & L_{yy}(x,y,\pm) \end{bmatrix} \tag{4.2}$$

$$c(x,y,\pm)=D_{xx}(\pm)\cdot D_{yy}(\pm)-(0.9D_{xy}(\pm))^2 \approx \det[H(x,y,\pm)] \tag{4.3}$$

$$\left[\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\right] \tag{4.4}$$

where *H* is the Hessian matrix, *G* is the Gaussian function, *I(x,y)* is the intensity of the pixel at (x,y), D_{xx}, D_{yy}, and D_{xy} are 9 × 9 box-filter approximations of the Gaussian function for σ = 1.2 (Figure 4.1), and L_{xx}, L_{xy}, and L_{yy} are the second-order derivatives of the grayscale image.

In order to add more distinctiveness to each feature, the SURF feature-descriptor can be computed for an extended 128 dimensions. The sums of d_x and $|d_x|$ are computed separately for $d_y < 0$ and $d_y \geq 0$. Similarly, the sums of d_y and $|d_y|$ are split up according to the sign of d_x. This doubles the length of the descriptor and doesn't add much computational complexity (Bay, 2008). Another important improvement in SURF is the use of sign of Laplacian about each keypoint. This process adds no computational cost since it is already computed during detection. The sign of the Laplacian distinguishes bright blobs on dark backgrounds from the reverse situation. This is useful in the matching stage, as features will only be matched if they share the same type of contrast.

4.1.3 FAST Features

A primary limitation of visual tracking with feature-descriptor algorithms is computational requirement which is restrictive for real-time applications. With an emphasis on the need for a real-time feature detector, Rosten et al. (2006) developed Features from Accelerated Segment Test (FAST). FAST uses discretized circles around a candidate point (p) and each point is classified as a corner if there exists a contiguous arc of at least *n*-pixels with intensities either above or below a brightness threshold (t). This version of FAST is commonly referred to as FAST-*n*, and typically an arc length of 9 or 12 points and a brightness threshold of 31 are used (Rosten, 2006). A high-speed feature scoring method, known as the FAST-score, was proposed to as a corner test in order to exclude the large number of non-corners produced by FAST (Equation 4.5). If p is a corner, then at least three of these must all be brighter than $I_p + t$ *or* darker than $I_p - t$, where I_p is the pixel intensity. If neither of these is the case, then p cannot be a corner. The full segment test criterion can then be applied to the passed candidates by examining all pixels in the circle.

$$C(p) = \max\left\{ \sum_{q \in S_+} \left| I_q - I_p \right| - t, \sum_{q \in S_-} \left| I_q - I_p \right| - t \right\} \tag{4.5}$$

The FAST detector exhibits high speed and performance, but has a few weaknesses:

1. Unlike SIFT or SURF, FAST features do not have an orientation component
2. Features which may have been sufficiently distinctive can be eliminated by the FAST-score
3. The corner detector efficiency depends on the order of tests applied
4. Multiple features are often detected adjacent to one another

To address these concerns, continued work has been conducted to improve the repeatability of FAST. A machine-learning approach was proposed by Rosten et al. (2010) to improve the segmentation testing procedure. Based on their results, the improved method (FAST-ER) exhibited the best repeatability followed closely by FAST-9, whereas FAST-12 performed significantly worse. With respect to computational speed, FAST-9 performed significantly faster than FAST-ER, which operated at 188 MPix/s and 75.4 MPix/s, respectively (Rosten, 2010). As such, despite exhibiting slightly lower repeatability, FAST-9 is commonly preferred to FAST-ER due to its beneficial compromise between repeatability and computational speed.

4.1.4 BRIEF Descriptors

Both SIFT and SURF use 128-dimensional floating point vectors for description which requires 512 bytes for each feature. However, creating such a vector for thousands of features takes a lot of memory which is not feasible for resource-constrained applications, such as embedded systems. In response to this concern, Binary Robust Independent Elementary Features (BRIEF) was developed for producing high-speed feature descriptors (Calonder, 2010). BRIEF provides a shortcut to find the binary strings directly without finding the descriptors. BRIEF assigns an area of smoothened image patch of 31 × 31 pixels around a feature point and selects a set of n_d location pairs (*x,y*). Next, several intensity comparisons (known as τ-tests) are performed on each location (Equation 4.6). The τ-tests are repeated for each pair producing a n_d-bit string descriptor (Equation 4.7). Due to their computational simplicity, binary descriptors have significantly higher descriptor generation speeds compared to alternate methods.

$$\tau(p;x,y)=\begin{cases}1 & p(x)<p(y)\\ 0 & p(x)\geq p(y)\end{cases} \tag{4.6}$$

$$f_n(p)=\sum_{1\leq i\leq n}^{n=256} 2^{i-1}\tau(p;x,y) \tag{4.7}$$

4.1.5 ORB Feature-Descriptor

Although very computationally efficient, BRIEF is extremely sensitive to in-plane rotation (Kulkarni, 2013). Oriented FAST and Rotated BRIEF (ORB) was developed by Rublee et al. (2011) as a real-time, rotation-invariant, feature-descriptor algorithm with comparable performance to SIFT and SURF. The first major improvement included in ORB is the assignment of orientations for each FAST keypoint, known as Oriented FAST (oFAST). First, the FAST feature detector is used to find keypoints, and then the Harris corner measure is applied to find the *N* best points. The Harris-score is used in place of the FAST-score to improve reliability, albeit at a slight cost to speed. Next, in the FAST corner detection stage, the orientation of the keypoint is determined using the primary moment method of the intensity centroid for a 15 × 15 region around each keypoint. This process provides each keypoint with an orientation at relatively low computational cost (Rublee, 2011).

The second major improvement in ORB is the inclusion of the Rotation-Aware BRIEF (rBRIEF) algorithm, a binary-descriptor based on BRIEF, i.e. each element in the feature descriptor array is either 0 or 1. ORB “steers” BRIEF descriptors according to the orientation of the keypoints, a technique referred to as rBRIEF by Rublee et al. (2011). rBRIEF computes the intensity weighted centroid of the patch located at the center of each corner. The direction of the vector from the corner point to the centroid gives the orientation. To improve the rotation invariance, moments are computed in x and y in a circular region around eahc keypoint of radius (*r*), where *r* is known as the patch size. For any set of feature descriptors consisting of n_d binary tests at a point (x_i, y_i), a 2 × n matrix is defined (*S*) which contains the coordinates of these pixels. Then, using the orientation (θ) of the region, the rotation matrix is computed and subsequently applied to *S* in order to obtain the rotated version (S_θ). ORB discretizes the angle to increments of *2π/30* and constructs a lookup table of pre-computed BRIEF patterns. As long as the keypoint orientation *θ* is consistent across views, the correct set of points S_θ will be used to compute its descriptor.

An important benefit of BRIEF is the descriptors have a large variance and a mean near 0.5 which results in more discriminative features. However, once oriented along the keypoint direction with rBRIEF, this property is lost and the variance is distributed. To resolve this, ORB uses a greedy search (Cormen, 1990) among all possible binary tests to find the ones that have the highest variance:

$$g_n(p,^{-}) = f_n(p) \,|\, (x_i, y_i) \in S_{-} \tag{4.8}$$

Implementations of ORB has been developed for system-on-a-chip (SoC) systems specifically for embedded applications which have reported speeds of 18 ms per 640 × 480 pixel image (Lee, 2013). As such, ORB is a good choice for low-power devices, i.e. systems with restricted computational capacity. Monocular systems, which only capture a single image at a time compared to two for stereovision, have significantly lower computational requirements, and ORB has been shown to be a reliable feature-descriptor for monocular VSLAM (Mur-Artal, 2015). However, since ORB uses corner detection and binary-descriptors, it may exhibit reduced repeatability when employed on surfaces with a high number of similar and poorly defined edges, e.g. homogenous surfaces with low feature distinctiveness like concrete or asphalt.

4.1.6 CLAHE

Histogram equalization is a pre-processing technique for adjusting pixel intensities in order to enhance the contrast of images and edge definition. However, global histogram equalization can cause degradation of some features. To address this, adaptive histogram equalization (AHE) partitions the image into equally sized rectangular tiles, e.g. 8 × 8 tiles is a common choice (Zuiderveld, 1994). For each tile, the sub-histogram is calculated and used to equalize each image tile independently. However, AHE is still prone to over-amplification of noise in relatively homogenous sub-regions of an image.

To address noise-amplification, contrast limited adaptive histogram equalization (CLAHE) was proposed by

Pizer et al. (1990). CLAHE uses the slope of the transformation function in the neighborhood of a given pixel to perform contrast amplification. The amplification is proportional to the slope of the neighborhood's cumulative distribution function (CDF), i.e. the value of the histogram for the given pixel. CLAHE limits over-amplification by clipping the histogram at a predefined value, known as the clip limit, before computing the CDF, thus reducing the slope of the transform function (Figure 4.2). The clip limit of the sub-histogram depends on the normalization of the histogram and thereby on the size of the neighborhood region. Lastly, interpolation allows a significant improvement in efficiency without compromising the quality of the result (Pizer, 1987).

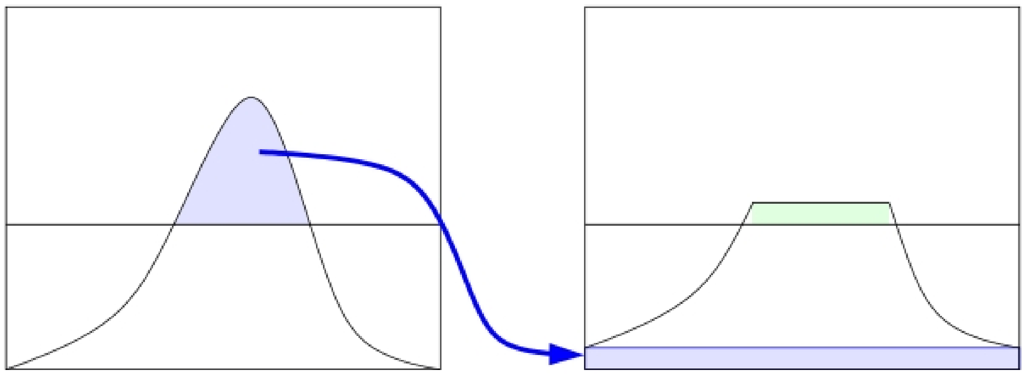

Figure 4.2 CLAHE histogram value redistribution. For each tile, values in the histogram of intensities is clipped at the clip limit and the clipped volume is uniformly distributed throughout the tile.

As a pre-processing technique, CLAHE has a positive effect on edge enhancement (Figure 4.3), but is somewhat computationally expensive because it requires computing a histogram and CDF for each of the 8 × 8 tiles in the image. Reza (2004) proposed a system level implementation of CLAHE to optimize speed without precision loss in resource-constrained applications. CLAHE has been demonstrated to be an effective technique for real-time contrast enhancement (Yadav, 2014). With respect to feature description, CLAHE has been included in the histogram-binary combined corner enhancement (HBCCE) algorithm, which improved the repeatability of the Binary Robust Invariant Scalable Keypoints (BRISK) detector, a binary detector similar to FAST (Leutenegger, 2011), from 10% to 40% with neglible reduction in speed (El Harraj, 2015).

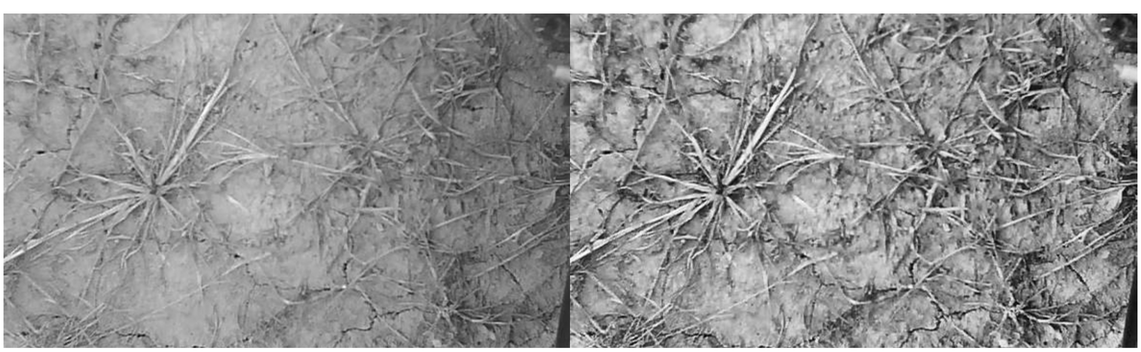

Figure 4.3 Comparison of grayscale (left) and CLAHE (right) images. CLAHE can greatly improve image contrast without degrading features for a relatively small computational cost.

4.1.7 kNN Matching

The *k*-Nearest Neighbors algorithm (*k*NN) is a non-parametric method used for classification and regression (Altman, 1992). kNN is one of the simplest machine-learning algorithms available for supervised learning. The idea is to search for the closest match of the test data in the feature space. With respect to feature matching of images, kNN is used to find the k-nearest neighbors of each feature in one image to the features of a second image. A commonly used distance metric for continuous variables is Euclidean distance, known as the L^2 norm. However, features descriptors produced by binary descriptor algorithms, i.e. BRIEF and rBRIEF, are discrete variables, so instead of Euclidean distance the Hamming norm must be used (Hamming, 1950),. This provides a significant improvement in kNN computational speed because calculating the Hamming norm only requires applying the XOR and bit count which are very fast operations on modern microprocessors with Streaming SIMD Extensions (SSE) instructions (Fog, 2004).

Applications of the kNN algorithm in computer vision commonly search for either one or two nearest neighbors, referred to as 1NN and 2NN, respectively. For example, 2NN produces two matches for each feature. Lowe (2004) proposed a widely used method, known as the ratio-test, to filter for outliers when using 2NN matching. For each feature in the first image, the two closest matches are calculated in the second. Then, the distance metric of each match is compared, and the match is only accepted if the ratio between the two distances is less than some threshold, e.g. a threshold of 0.7 is typical. With respect to 1NN, calculating a single neighbor can result in poor performance as a match will be found for every feature in the first image regardless of how dissimilar the feature descriptors of the two keypoints are. For example, if a feature was detected in the first image, but the feature was absent in the second image, 1NN will still try to find a nearest neighbor for the orphaned feature. To address this, 1NN is often used with cross-checking. Cross-checking is based on the principle that the distance from one keypoint (A) to another keypoint (B) is not necessarily an invertible statement, i.e. B may be the closest point to A, but A may not be the closest point to B. Cross-checking computes 1NN for both directions, A-to-B and B-to-A, and a match is only accepted if A and B are closest for both directions (Figure 4.4).

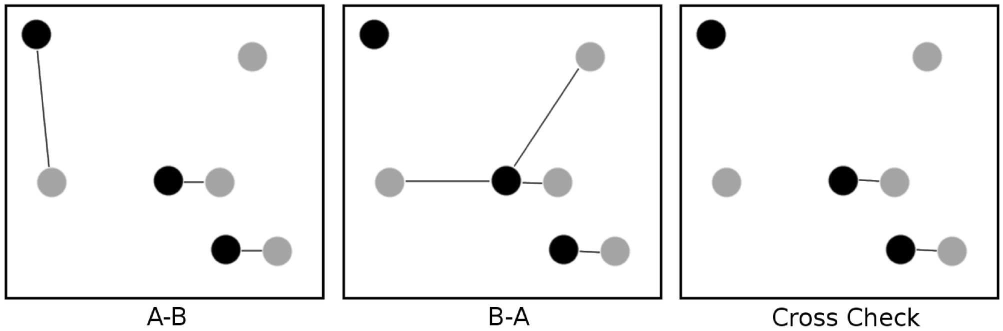

Figure 4.4 Matching with 1NN and cross-checking. Cross-checking eliminates matches between the two sets of features which are only detected when calculating the nearest neighbor for a particular direction. Feature pairs which are both each other's nearest neighbor are accepted.

4.1.8 Objective

The **objective** of this study was to compare the efficiency of five feature-descriptor algorithms for monocular visual tracking: SURF, U-SURF, SIFT, ORB, ORB with CLAHE pre-processing (referred to as CLORB). Motion vectors of an agricultural vehicle were determined using visual tracking for speeds between 1 and 5 m/s over six different surfaces. Specifically, these algorithms were to be analyzed in terms of their reliability, computational efficiency, precision, and susceptibility to increased errors on specific travel surfaces.

4.2 Materials and Methods

A weatherproof (IP65) camera (Shenzhen Yufei Technology Co., Shenzhen, China) with a resolution of 640 × 480 pixels was mounted to the front of a John Deere 850D Gator (John Deere, Illinois, USA) at an orthogonal perspective relative to the ground surface and a height of 1.0 m (Figure 4.5). The camera had an aperture of 2.4 (F), a 6 mm focal point (f), 1/3" color CMOS sensor with a 4:3 aspect ratio. In order to maximize the overlap between consecutive images at higher travel speeds, the camera was oriented width-wise such that the 640 pixel axis of the images were aligned with the direction of travel. The camera was capable of automatic white-balance adjustment and had a frame capture rate of 25 Hz. The lens of the camera was not infrared (IR) filtered and the camera enclosure was equipped with 24 infrared LEDs. The camera was rated to a minimum illumination of 0.5 lux and could be used effectively in a wide range of ambient lighting conditions.

Figure 4.5 Test vehicle and mounting bracket for camera. The camera mounting system was fixed to the crash bar of the vehicle via a U-bracket.

After installation of the camera bracket, 10 calibration images were taken of a 9 × 7 checkerboard pattern in order to determine the camera distortion model similar to the approach by Lee et al. (2009). Corner detection was conducted on each image to find the geometric position of the 48 distinct corners (Figure 4.6). For each image, the corner points were used to fit a radial and tangential distortion model ((Brown, 1966):

$$\begin{aligned} x_d &= x_u \cdot \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^4\right) + \left(p_2 \cdot \left(r^2 + 2x_u^2\right) + 2p_1 x_u y_u\right) \\ y_d &= y_u \cdot \left(1 + k_1 r^2 + k_2 r^4 + k_3 r^4\right) + \left(p_1 \cdot \left(r^2 + 2y_u^2\right) + 2p_2 x_u y_u\right) \\ r &= \sqrt{\left(x_u - x_c\right)^2 + \left(y_u - y_c\right)^2} \end{aligned} \tag{4.9}$$

where radial terms k_1, k_2, and k_3 are -3.21×10^{1}, -6.03×10^{-3}, and -3.22×10^{-3}, respectively, and tangential terms p_1 and p_2 are -7.13×10^{-2} and 2.41×10^{-7}, respectively.

The distance coefficients of each image were averaged to generate the distance coefficients used for radial and tangential distortion correction (1). Using the known dimensions of the checkerboard squares (30 mm by 30 mm), an image resolution of 1.0 mm/px was achieved with this camera configuration.

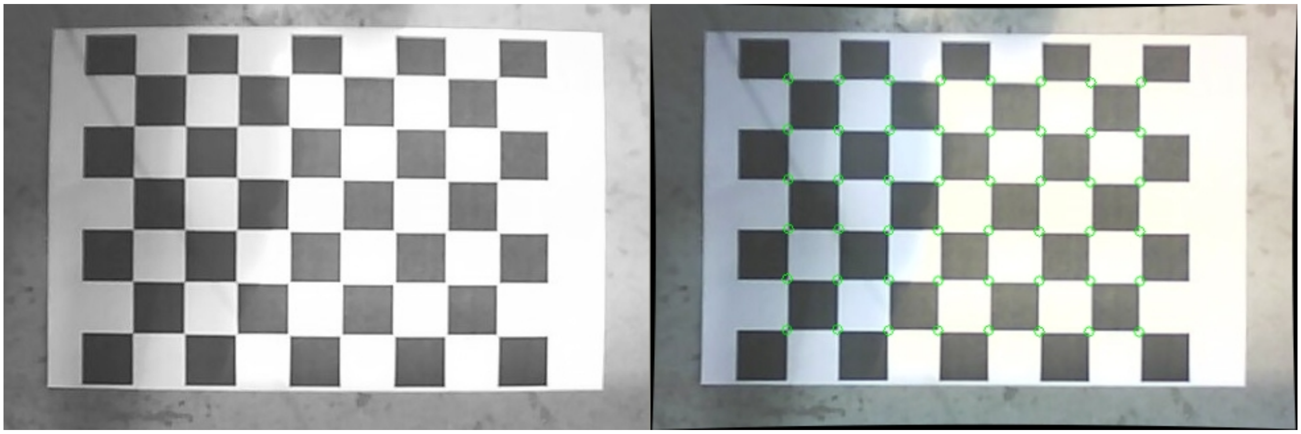

Figure 4.6 Distortion correction of camera using the checkerboard technique. Although it is visually difficult to distinguish the effects of distortion correction, this procedure dramatically improved the accuracy of keypoint displacement calculations.

A Trimble AgGPS 542 real-time kinematic global navigation and satellite system (RTK-GNSS) rover and GA530 receiver (Trimble Navigation Limited, Sunnyvale, California, USA) capable of acquiring both L1 and L2 frequencies were mounted to the test vehicle. The rover was configured to produce NMEA-0183 data strings for vector track and speed over ground (VTG) and time, position, and satellite fix (GGA) at the maximum supported rate of 20 Hz. The accompanying base station was positioned in an open area and all trials were conducted within 2 km of the base station. The application was optimized to run on a 2.8 GHz Intel i7 VMC3501-K (Nexcomm, Taipei, Taiwan). An application was developed using the Python programming language (version 2.7.11) (Python Software Foundation, Wilmington, Delaware, USA) and OpenCV (version 2.4.9) (Itseez, Nizhny Novgorod, Russian Federation) to capture videos with low-latency and produce accompanying GPS logs with high precision time-stamps (1 μs precision) corresponding to each frame.

As specified by ISO 12188-2, the visual tracking system was evaluated from 1 m/s to 5 m/s (ISO, 2012) on six (6) surfaces: asphalt, gravel, turf grass, seedlings (< V2), corn residue, and pasture (Figure 4.7). For each trial, the logging system and RTK-GNSS were activated. Once an accurate satellite signal was acquired, the vehicle was engaged into gear and the accelerator was steadily applied until 5 m/s was reached. Although the vehicle was manually operated, during each trial the operator attempted to maintain an average acceleration of 0.25 m/s^2 for a target trial length of approximately 45 s. For each surface type, five replicates were conducted. For each trial, the approximate height of the crop coverage relative to the ground surface was estimated. This resulted in a total dataset of 32699 images with corresponding RTK-GNSS point and vector data.

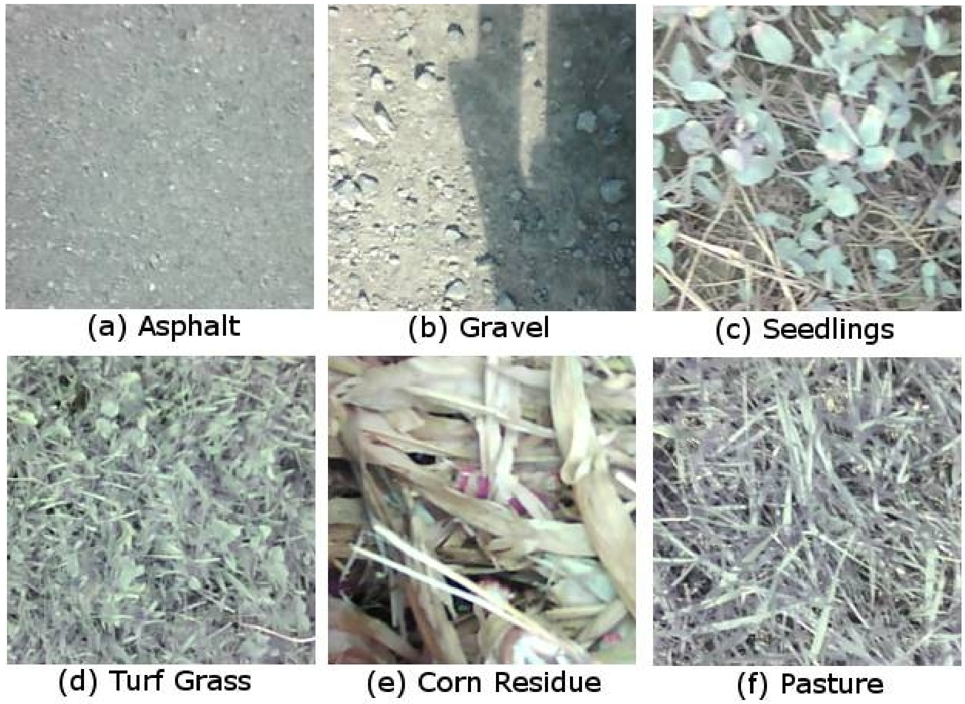

Figure 4.7 Sample images of the six testing surfaces considered for this study.

Five feature-descriptor algorithms were evaluated: SURF, U-SURF, ORB, ORB with CLAHE pre-processing (CLORB), and SIFT. Additionally, for each feature-descriptor algorithm, both 1NN with cross-checking and 2NN with the ratio-test (threshold of 0.7) were evaluated, for a total of 10 algorithm configurations. Post-processing of the collected video data was computed in a single thread on a 3.1 GHz Intel Xeon CPU E31225 (Intel Corp., Santa Clara, California, USA) to simulate performance close to that of high-end embedded systems. Processing rates of each algorithm were calculated by comparing real-time clock readings to 1 µs precision at the beginning and end of each iteration

The visual tracking algorithm consisted of an iterative process consisting of 9 sequential operations (Figure 4.8). First, the next RGB image was captured and converted to gray-scale:

$$\mathrm{Gray}_{\mathrm{i,j}} = 0.299 \cdot R_{i,j} + 0.587 \cdot G_{i,j} + 0.114 \cdot B_{i,j} \tag{4.10}$$

In the case of ORB, a version was tested which included an additional pre-processing step of CLAHE on the gray-scale image. Next, lens distortion was corrected for (Equation 4.9). One of the feature-descriptor algorithms was then used to detect keypoints for an image, and then computing feature descriptors for kNN-matching the keypoints of the current image to the keypoints of the previous image stored in memory. The set of matches produced by kNN was then converted to polar vectors before a histogram vector filter was applied to eliminate outliers. Lastly, the median angle and displacement was calculated from the set of all remaining inlier vectors.

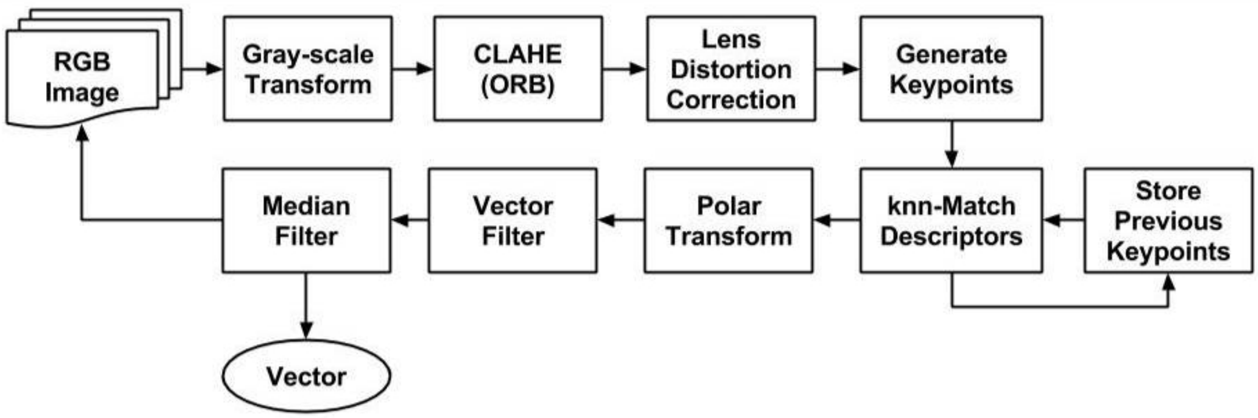

Figure 4.8 Process diagram of visual tracking algorithm. Note: The CLAHE pre-processing step was only applied in the case of CLORB.

With respect to SURF and U-SURF, common parameters used in the literature and by Bay et al. (2008) were employed, specifically: four (4) Gaussian pyramid octaves and two (2) images within each octave of a Gaussian pyramid. Based on the literature, kNN matching of SURF floating-point descriptors was conducted using the L^2 norm distance metric. With respect to SIFT, algorithm parameters similar to those used by Lowe (2004) were applied: the number of octave layers was set to three (3), and the contrast and edge thresholds were set to 0.04 and 10, respectively. As per the literature, kNN matching of SIFT floating-point descriptors was conducted using the L^2 norm distance metric. With respect to ORB, two variants were evaluated: ORB and ORB with CLAHE pre-processing (CLORB). Both algorithms were configured to use FAST-9 with an edge threshold of 31, a patch size of 31, eight (8) pyramid levels, and two (2) randomized pairs for rBRIEF feature description. As per Rublee et al. (2011), kNN matching of ORB binary descriptors was conducted using the Hamming norm distance metric. In the case of CLORB, CLAHE with a clip limit of 2 and a grid size of 8 × 8 was used for a total of 64 tiles per image.

After finding matching keypoints between two consecutive frames using either the 2NN ratio-test or 1NN cross-checking, each pair of keypoints (x_1,y_1) and (x_2,y_2) was transformed to a vector in 2D polar-coordinates:

$$P(x_1, y_1, x_2, y_2) = \begin{pmatrix} r \\ \theta \end{pmatrix} = \begin{pmatrix} \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \\ \frac{180}{\pi} \operatorname{atan2}((x_2 - x_1), (y_2 - y_1)) \end{pmatrix} \tag{4.11}$$

where (x_1, y_1) and (x_2, y_2) are the locations of the keypoint in the first and second image, and atan2 is the quadrant-aware arctan function (Kerrisk, n.d.).

In the event of image pairs with no keypoint matches detected, a value of Not-a-Number (NaN) was used. Pairs of images without matches, i.e. NaN values, were disregarded from further analysis. However NaN values were considered when calculating inlier-outlier ratios.

Despite including cross-checking or the ratio-test to filter kNN matching, errors can still occur and it is necessary to filter the set of all possible vectors to remove any false-positive matches (Strasdat, 2010). To simplify this process with respect to visual tracking of orthogonal images, the set of motion vectors was assumed to be unimodal, i.e. it was assumed that the field-of-view was dominated by keypoints sharing a dominant motion vector. To identify this dominant vector, a computationally inexpensive histogram filter was employed (Equation 4.12). First, a 360-bin histogram (H) of the integer-rounded θ-values was generated. Next, the discrete second-order gradient (Equation 4.13) of the histogram cumulative summation (Equation 4.14) was used to calculate an array of differences (D). The central vector angle corresponding to the maximum value of the gradient array was found (θ_C) using the argmax() function (Equation 4.15). The resulting best estimate for the motion vector (v_m) between two consecutive frames was taken to be the median and mean of the sets of all vector radii (R_k) and all vector angles (Θ_k), respectively, whose angle of orientation within a tolerance of 1˚ of the θ_C:

$$\begin{array}{ll}
1. & H = \text{histogram}\left(\lfloor \Theta \rfloor, \text{bins} = 360 \right) \\
2. & D = \text{gradient}\left(\text{cumsum}\left(H \right), \text{order} = 2 \right) \\
3. & {}^{-}{}_{C} = \text{argmax}\left(D \right) \\
4. & k = \left\{ i \mid i \in {}^{-}{}_{C} + \pounds \le \Theta_i \le {}^{-}{}_{C} + \pounds \right\} \\
5. & v = \begin{pmatrix} f \cdot \text{median}\left(R_k\right) \\ \text{median}\left(\Theta_k\right) \end{pmatrix}
\end{array} \tag{4.12}$$

$$\underset{x \in S \subseteq X}{\text{gradient}}\, f(x) \;\; := \;\; \left\{ f\left(x_{i+1}\right) - f\left(x_i\right) \mid i \in [0, N] \right\} \tag{4.13}$$

$$\underset{x \in S \subseteq X}{\text{cumsum}}\, f(x) \;\; := \;\; \left\{ \sum_{i=0}^{n} f\left(x_i\right) \mid n \in [0, N] \right\} \tag{4.14}$$

$$\underset{x \in S \subseteq X}{\text{argmax}}\, f(x) \;\; := \;\; \left\{ x \mid x \in S \wedge \forall\, y \in S : f(y) \le f(x) \right\} \tag{4.15}$$

where Θ is the set of all vector angles, R is the set of all vector radii, θ_C is the central vector angle about which to accept vectors as valid, ε is the tolerance window (in degrees), and f is the frame rate of the video-stream (25 Hz).

This methodology proved to be effective for determining the dominant motion vector, even in the case of persistent shadows or non-moving objects (e.g., vehicle frame) in the field of view. This filter only required 2 ms per 640 × 480 image compared to 7 ms for RANSAC. Figure 4.9 shows an example of the histogram filter rejecting outlier vectors.

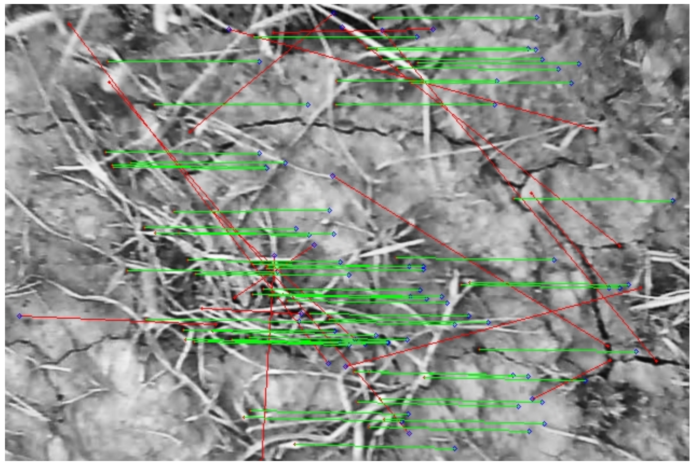

Figure 4.9 Keypoint matching and outlier vector rejection. Outlier vectors (red) which do not follow the dominant direction and rate of motion are rejected.

For subsequent data analysis and multiple comparisons between trials, tracking values produced by the RTK-GNSS receiver were smoothed using a 5-point 1st-order Savitsky-Golay filter to reduce vibration effects. Similarly, vector estimates from feature-based visual tracking were smoothed using a 5-point moving median filter instead of Savitsky-Golay filtering due to the non-parametric nature of error distribution. In order to determine the efficiency of visual tracking irrespective of vehicle dynamics, a Kalman filter was considered for this study.

4.3 Results

Figure 4.10 presents a comparison between two trials, the first on asphalt and the second on corn residue. In general, the performance of the visual tracking methods was similar to that of RTK-GNSS. All of the feature-descriptor methods exhibited better performance on surfaces which were homogeneous (e.g. asphalt) compared to heterogeneous surfaces (e.g. corn residue).

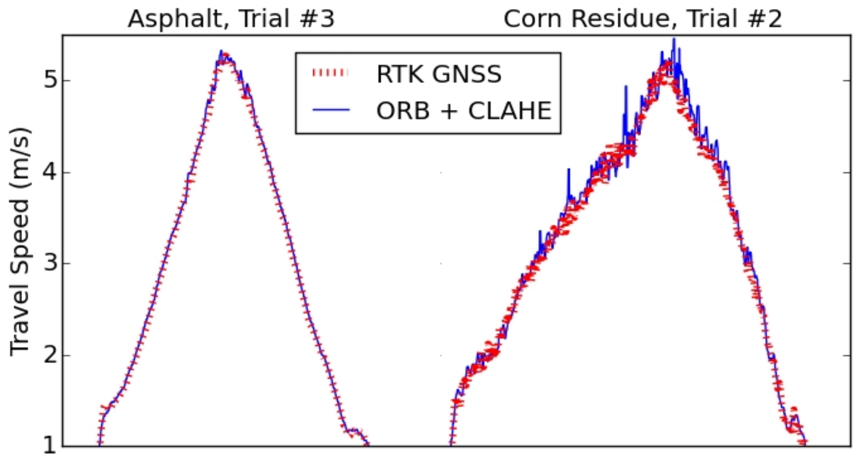

Figure 4.10 Examples of trials which produced good (left) and bad (right) results for visual tracking.

Figure 4.11 presents the normalized frequency distribution of errors for visual tracking relative to RTK-GNSS data. Two curves are presented: 1) uncompensated ground speed, and 2) ground speeds which were corrected using a constant conversion factor based on surface height:

$$Crop\ Height\ Factor = \frac{D - h^P}{D} \tag{4.16}$$

where P is the zoom power (1.0918), D is the ideal subject depth (100 cm), and h is the crop height (in cm).

As can be seen, the gravel, asphalt, and seedling surfaces showed negligible difference in spread and bias between the uncompensated and compensated methods. However, turf grass, pasture, and corn residue had noticeably greater bias and spread for uncompensated estimates. In the case of turf grass, depth compensation corrected both of these forms of error, but for pasture and corn residue depth compensation only was able to reduce bias.

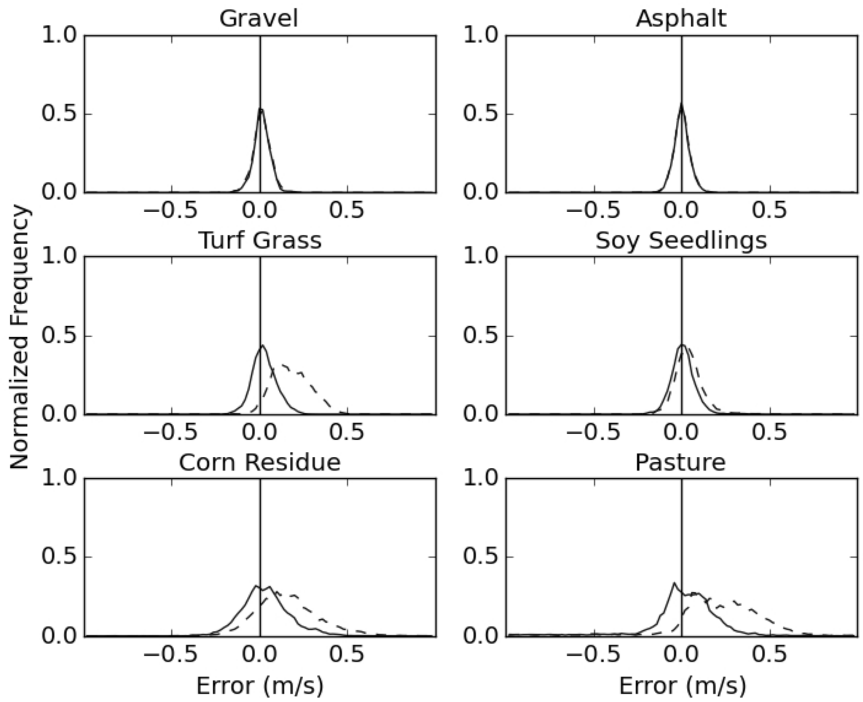

Figure 4.11 Normalized frequency of errors for uncalibrated (solid) and depth-compensated (dashed). As the height of the surface coverage increased, visual tracking consistently over estimated ground speed if uncompensated, but once a uniform scalar was applied errors were normally distributed about 0.

Figures 4.12, 4.13, and 4.14 illustrate the effect that varying the keypoint detection threshold had 95th percentile errors for each algorithm by speeds grouped by 1 m/s intervals, i.e. from 1 - 2 m/s, 2 - 3 m/s, 3 - 4 m/s and 4 - 5 m/s. Groups were calculated by considering the visual tracking estimates with respect to their corresponding RTK-GNSS speed values, whereas frames with NaN values for visual tracking estimates were disregarded. Each graph presents a range of keypoint detection thresholds which illustrate the trade-off between accuracy and computational speed.

With respect to the ORB variants (Figure 4.13), CLORB with the 2NN ratio-test performed best overall with respect to computational speed and precision. ORB exhibited a severe decrease in accuracy at speeds greater than 2 m/s, although this effect was reduced for 2NN with the ratio-test compared to 1NN with cross-checking for the same keypoint detection thresholds.

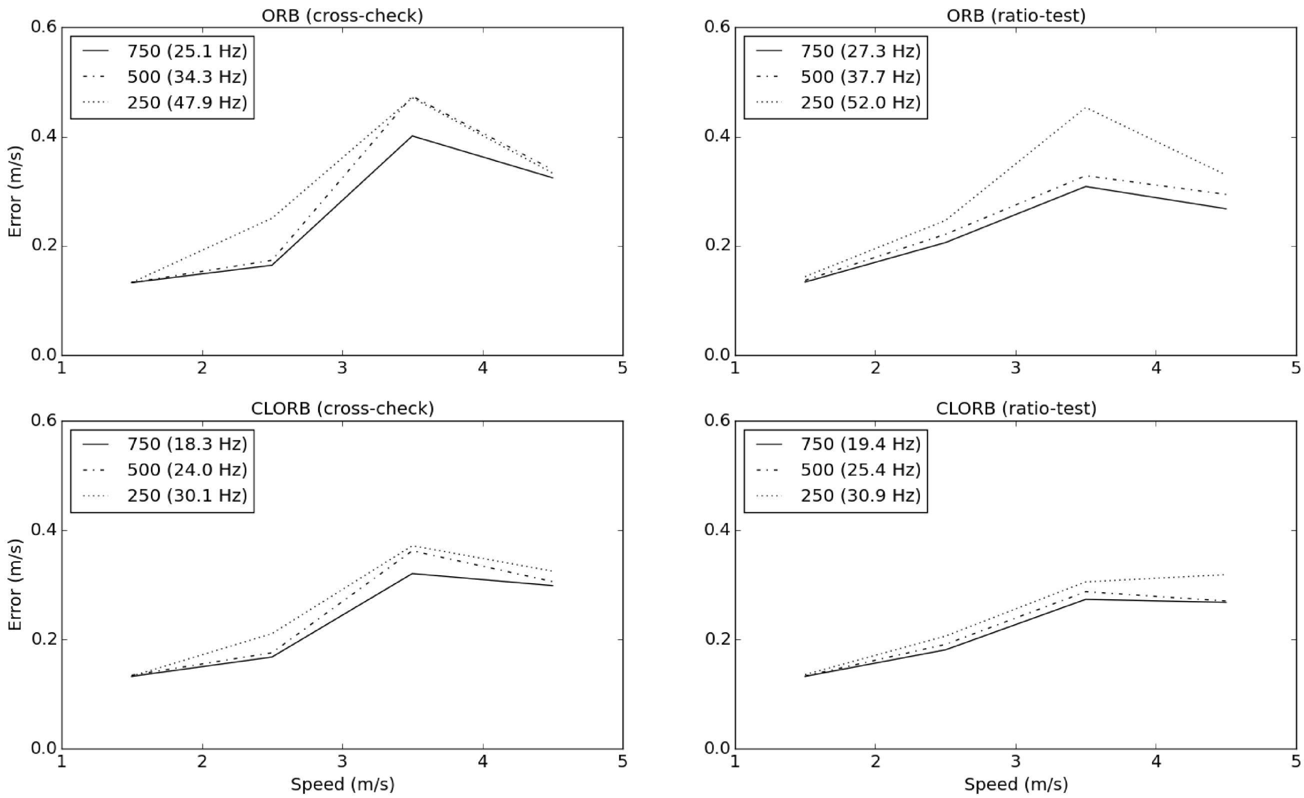

Figure 4.12 The 95th percentile error of ORB and CLORB with respect to travel speed and keypoint detection threshold. As travel speed increased, ORB decreased in accuracy considerably more so than CLORB. Additionally, despite reduced computational speed for CLORB, when comparing results for similar processing times CLORB exhibited lower error than ORB.

With respect to the SURF variants, both U-SURF and SURF were evaluated at three Hessian thresholds. Figure 4.13 demonstrates that U-SURF was both more computationally efficient and, unexpectedly, precise compared to SURF. In general, U-SURF exhibited less variation with respect to the keypoint detection threshold. The kNN matching methodology did not have an observable effect with respect to U-SURF, but the performance of SURF did improve when using 1NN cross-checking compared to the 2NN ratio-test.

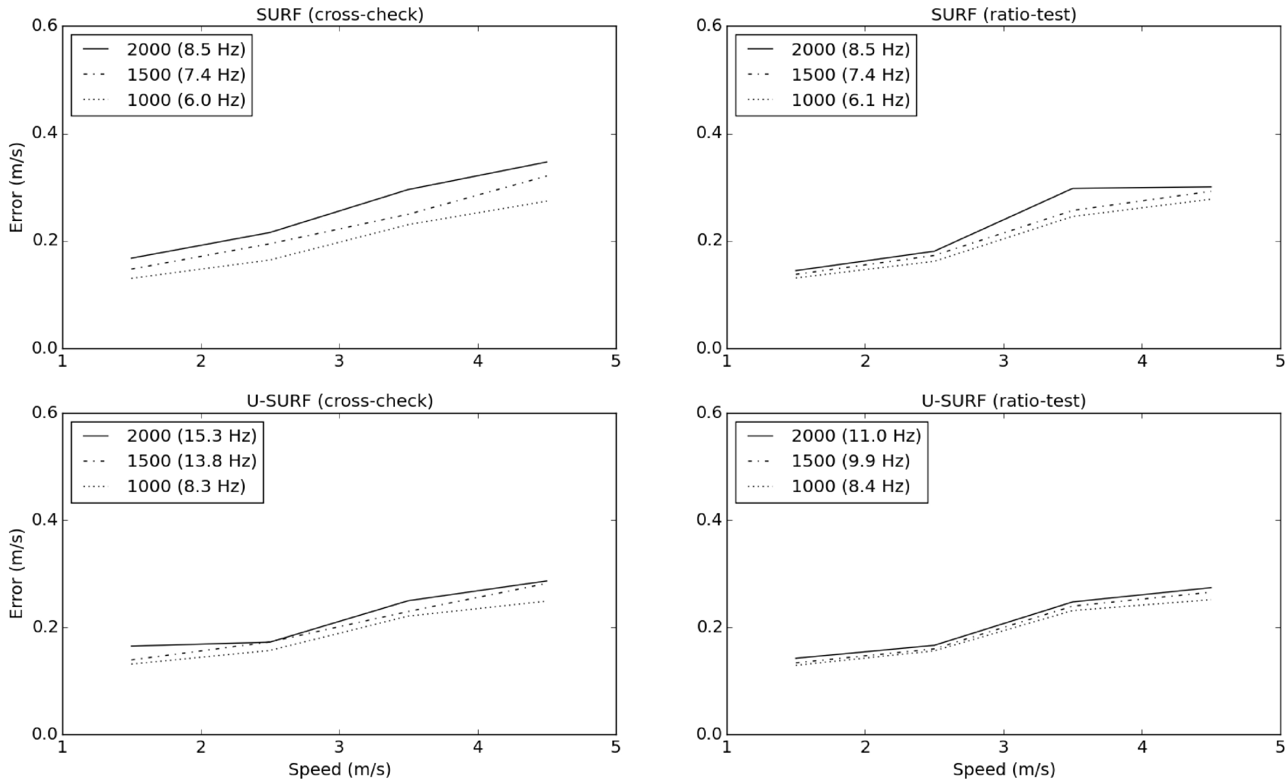

Figure 4.13 The 95th percentile error of SURF and U-SURF with respect to travel speed and Hessian threshold. U-SURF ran considerably faster than SURF, yet exhibited similar levels of accuracy.

With respect to the SIFT variants, Figure 4.14 demonstrates that SIFT had excellent precision regardless of the keypoint detection threshold or kNN matching methodology, albeit at a high computational cost.

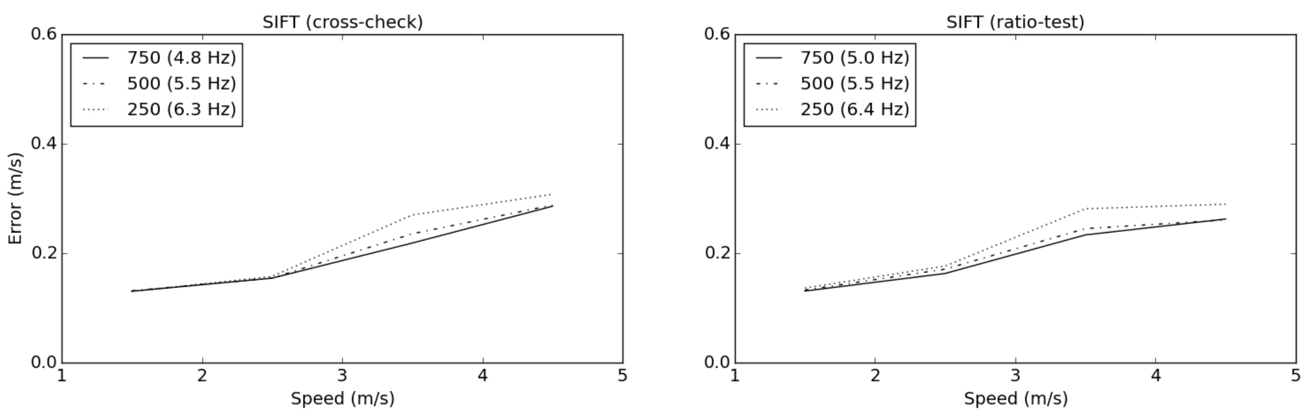

Figure 4.14 The 95th percentile error for SIFT with respect to travel speed and keypoint detection threshold. SIFT exhibited excellent accuracy for both 1NN with cross-checking and 2NN with the ratio-test, but had the slowest computational times of all algorithms tested regardless of threshold.

Table 4.1 shows a summary of 95th percentile values and the average processing time for each combination of feature-descriptor and kNN matching method. It can be seen that incorporating CLAHE as a pre-processing step for ORB reduced the 95th percentile error, notably by 0.08 m/s and 0.04 m/s of 3 - 4 m/s for cross-checking and ratio-test, respectively. SIFT was best overall, but had the slowest computational speed.

Table 4.1 Summary of visual tracking 95th percentile error for all testing surfaces

Algorithm	Process Rate (Hz)	95th Percentile Error (m/s)				
		Total	1 - 2 m/s	2 - 3 m/s	3 - 4 m/s	4 - 5 m/s
$SURF_{2000}$ (cross-check)	8.7	0.29	0.17	0.22	0.30	0.35
$SURF_{2000}$ (ratio-test)	8.7	0.25	0.14	0.18	0.30	0.30
U-$SURF_{2000}$ (cross-check)	15.1	0.24	0.16	0.17	0.25	0.29
U-$SURF_{2000}$ (ratio-test)	10.9	0.22	0.14	0.17	0.25	0.27
ORB_{500} (cross-check)	34.2	0.25	0.13	0.16	0.40	0.32
ORB_{500} (ratio-test)	37.6	0.23	0.13	0.21	0.31	0.27
$CLORB_{500}$ (cross-check)	24.0	0.24	0.13	0.17	0.32	0.30
$CLORB_{500}$ (ratio-test)	25.3	0.23	0.13	0.18	0.27	0.27
$SIFT_{500}$ (cross-check)	5.5	0.21	0.13	0.15	0.22	0.29
$SIFT_{500}$ (ratio-test)	5.4	0.21	0.13	0.16	0.23	0.26

Note: this table presents results for keypoint detection thresholds which exhibited the best overall performance.

Table 4.2 shows a summary of RMSE values and average number of inliers produced per image for each combination of feature-descriptor and kNN matching method. Incorporating CLAHE as a pre-processing step for ORB had no significant effect on RMSE, but increased the average number of inlier keypoints for 1NN cross-checking by 4%. As in Table 4.1, SIFT performed best overall, but, with the exception of the 4 - 5 m/s speed range where it exhibited similar performance to the other algorithms.

Table 4.2 Summary of visual tracking RMSE for all testing surfaces

Algorithm	Average Number of Inliers	RMSE (m/s)				
		Total	1 - 2 m/s	2 - 3 m/s	3 - 4 m/s	4 - 5 m/s
$SURF_{2000}$ (cross-check)	253.3	0.114	0.115	0.102	0.118	0.121
$SURF_{2000}$ (ratio-test)	91.5	0.101	0.095	0.098	0.108	0.102
U-$SURF_{2000}$ (cross-check)	236.2	0.099	0.124	0.080	0.097	0.098
U-$SURF_{2000}$ (ratio-test)	124.6	0.088	0.079	0.079	0.093	0.095
ORB_{500} (cross-check)	435.6	0.092	0.061	0.067	0.125	0.105
ORB_{500} (ratio-test)	160.7	0.097	0.064	0.118	0.113	0.090
$CLORB_{500}$ (cross-check)	454.9	0.088	0.057	0.068	0.120	0.010
$CLORB_{500}$ (ratio-test)	163.5	0.087	0.062	0.088	0.101	0.093
$SIFT_{500}$ (cross-check)	439.6	0.070	0.048	0.055	0.074	0.093
$SIFT_{500}$ (ratio-test)	201.6	0.074	0.050	0.063	0.085	0.090

Note: this table presents results for keypoint detection thresholds which exhibited the best overall performance

The results of ordinary least-squares linear regression of a linear model with no intercept for all surfaces are presented in Table 4.3 for each combination of the feature-descriptor algorithms. ORB performed the worst overall, but CLORB exhibited significantly better performance similar to that of the U-SURF, SURF and SIFT algorithms. Although CLORB was slower than ORB, this was balanced by its improved estimation accuracy.

Table 4.3 Linear regression best-fit parameters

Algorithm	Slope	r-value
$SURF_{2000}$ (cross-check)	1.002	0.989
$SURF_{2000}$ (ratio-test)	0.990	0.975[†]
$U\text{-}SURF_{2000}$ (cross-check)	1.005	0.988
$U\text{-}SURF_{2000}$ (ratio-test)	0.988	0.971[†]
ORB_{500} (cross-check)	0.975	0.964[†]
ORB_{500} (ratio-test)	0.967	0.953[†]
$CLORB_{500}$ (cross-check)	0.993	0.985
$CLORB_{500}$ (ratio-test)	0.993	0.980
$SIFT_{500}$ (cross-check)	1.011	0.997
$SIFT_{500}$ (ratio-test)	1.009	0.992

[†] significantly different from 1:1

Figure 4.15 presents the probability plots by ordered quantiles. This method of graphical analysis helps visualize that for all surfaces, with the exception of pasture, errors were normally distributed. However, some feature-descriptor algorithms had greater normality on asphalt than others, specifically SIFT and CLORB.

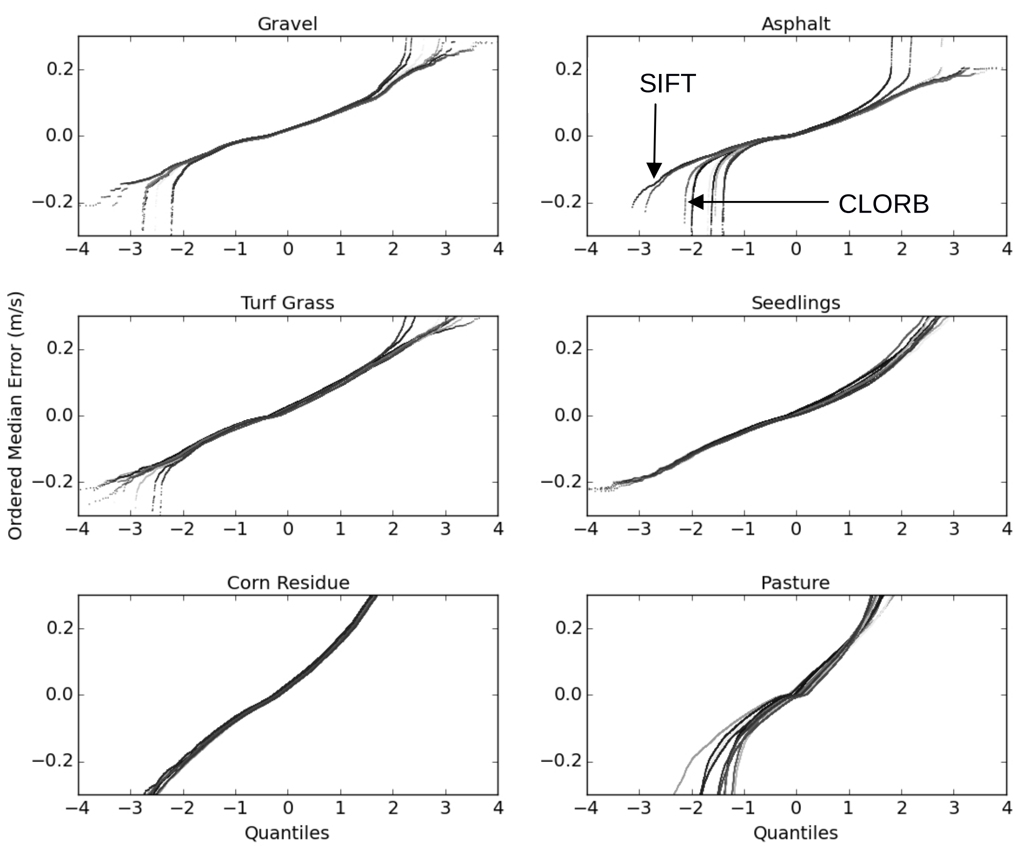

Figure 4.15 Probability plots by surface type. Among the surfaces tested, asphalt and pasture resulted in the greatest disparities in accuracy between the different algorithms.

In order to visualize the acceleration effects, Figure 4.16 presents the percent error of the CLORB method compared to RTK-GNSS. Acceleration effects can be seen for the more homogenous surfaces of gravel, asphalt, turf grass and seedlings. During acceleration, the visual tracking algorithm appears to lead the RTK-GNSS by up to 20%, whereas for deceleration, this phenomenon was significantly less pronounced.

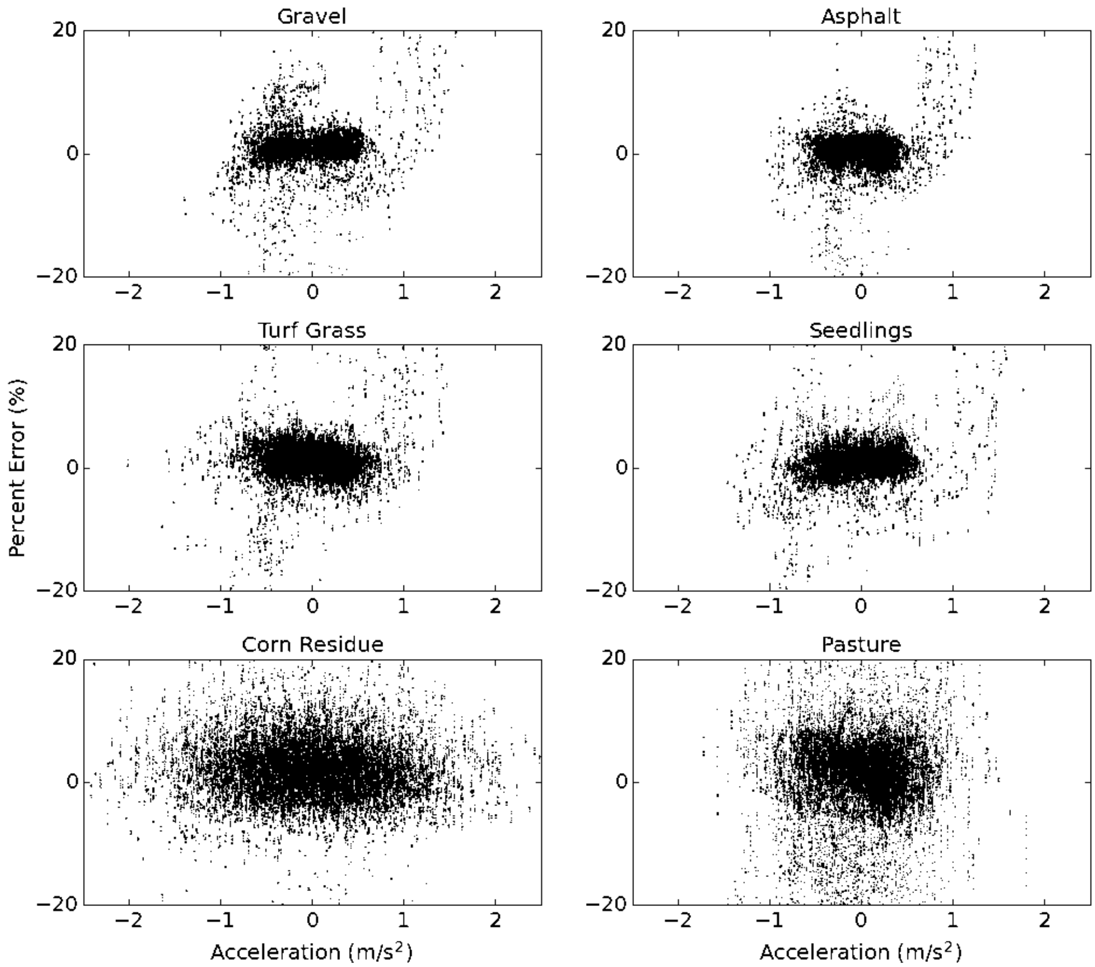

Figure 4.16 Effect of acceleration on discrepancy between unfiltered RTK and CLORB speed estimates.

Figure 4.17 presents inlier-outlier ratios as a function of travel speed, i.e. the final number of keypoints detected after the fast histogram filter compared to the initial number keypoints detected by kNN. As can be seen, all of the algorithms tested demonstrated a slight downward trend as the speed increased. 1NN cross-check produced greater inlier-outlier ratios methods with less variance compared to 2NN ratio-test. The ORB and SIFT methods had less variability that the SURF variants for both kNN methods. As can be seen, 2NN with the ratio-test often results in consecutive images with very few or no keypoint matches, e.g. detected keypoint pairs do not have distance ratios which satisfy the ratio threshold. This effect is especially noticeable for SURF and U-SURF at higher travel speeds.

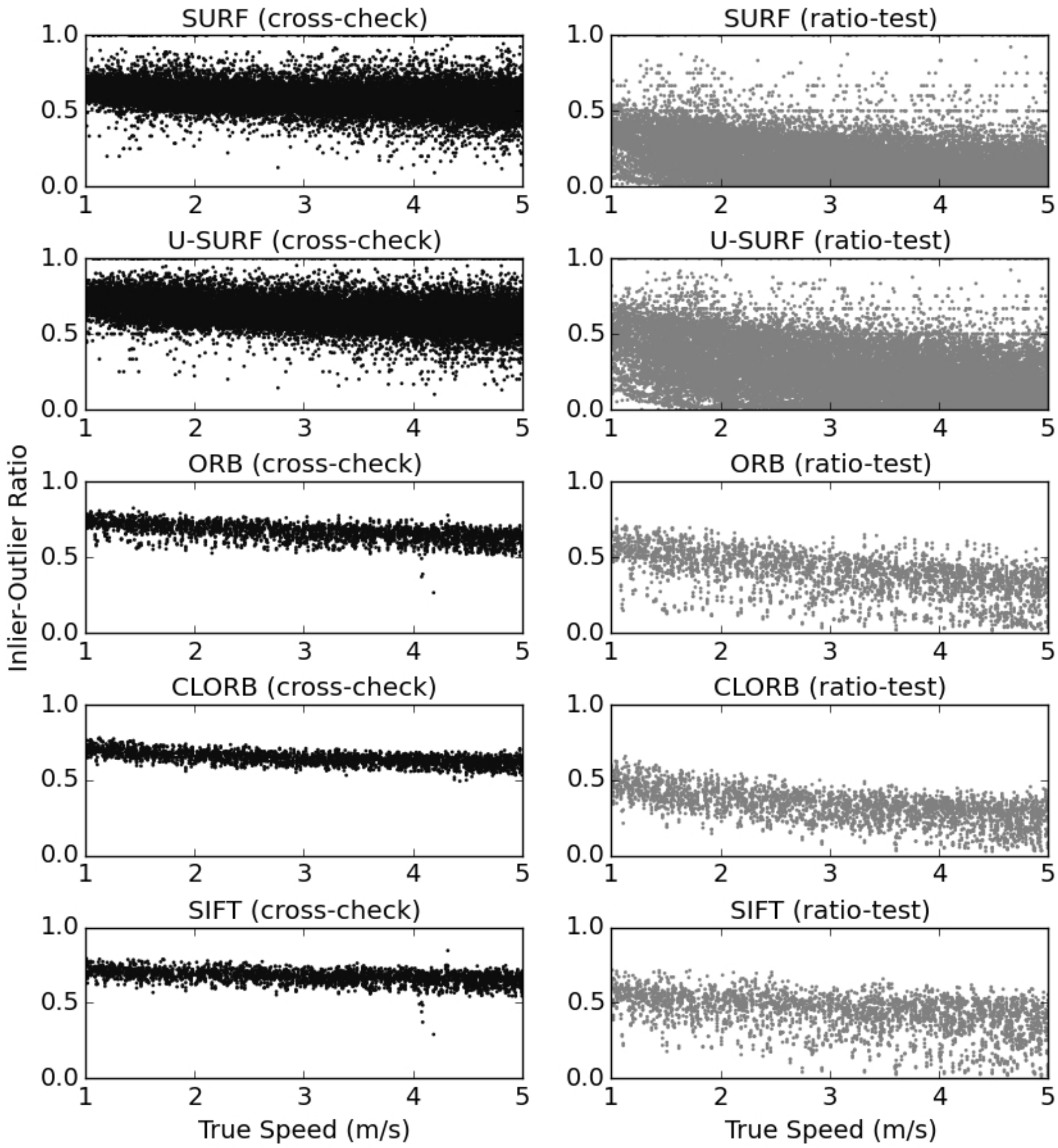

Figure 4.17 Inlier-outlier ratio after vector filtering with respect to RTK-GNSS travel speed.

Figure 4.18 presents Tukey HSD multiple comparison plots with 95% family confidence interval for each algorithm with respect to surface. Visual tracking on pasture and asphalt exhibited noticeable difference intervals for SURF, U-SURF and ORB for both kNN methods, however due to variance within the samples this was not considered significant. CLORB and SIFT had considerably smaller differentials for asphalt compared SURF, U-SURF and ORB in addition to lower variances, and therefore pasture can be considered to be significantly worse than other surfaces with respect to 95th percentile error of SIFT and CLORB. Both SURF and U-SURF showed slight improvement when using cross-checking compared to the ratio-test, whereas ORB decreased slightly in performance.

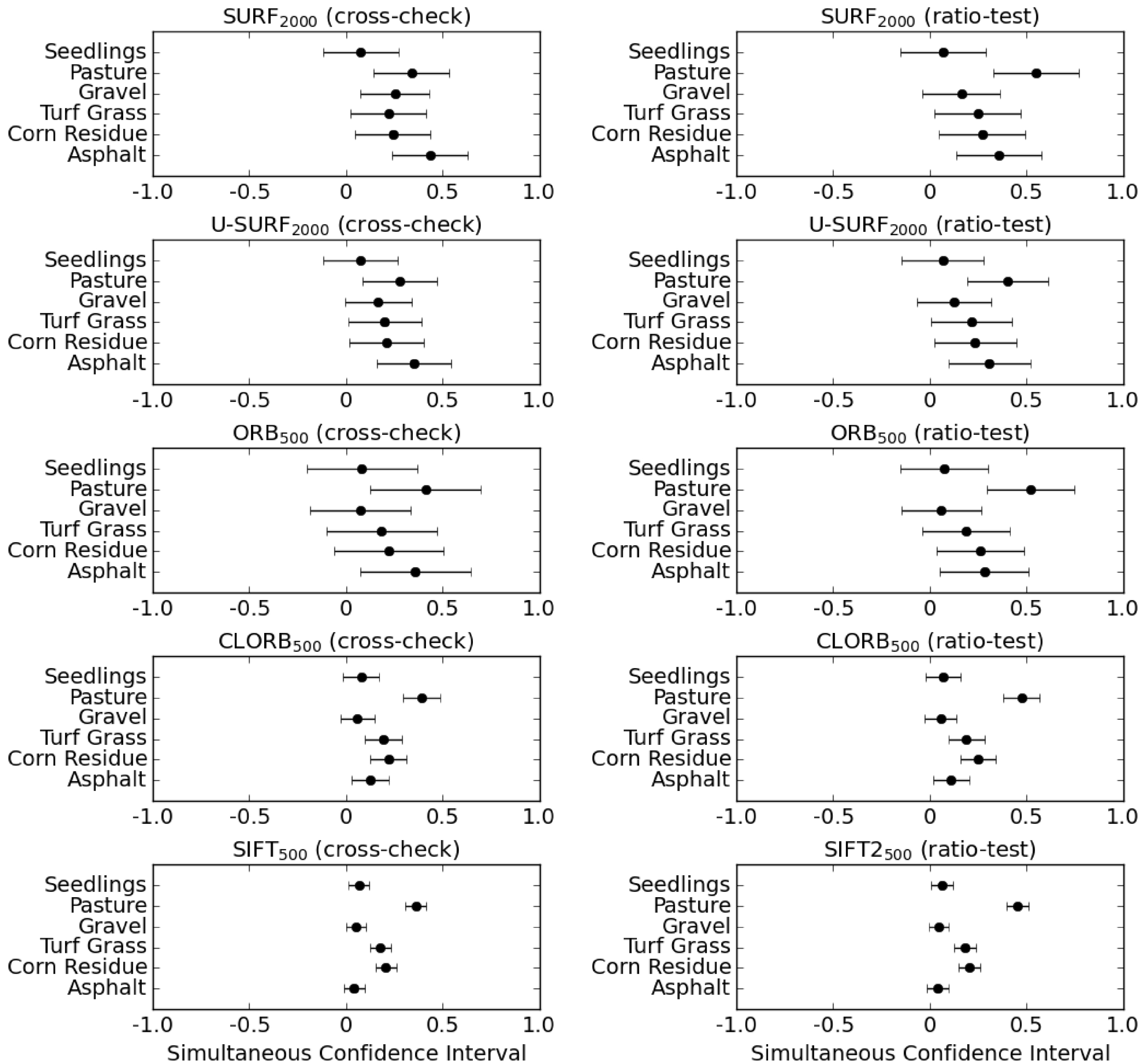

Figure 4.18 Tukey HSD multiple comparison of RMSE values by algorithm and surface, α = 0.05.

4.4 Discussion

Among the surfaces tested, pasture had the worst performance, followed by corn residue (Figure 4.18). This effect is likely attributable to the greater heterogeneity of the pasture and corn residue surfaces compared to asphalt, gravel, seedlings and turf grass. All of the algorithms tested exhibited an increase in 95th percentile error with increasing travel speed (Table 4.1 and Figures 4.12 to 4.14). Although the ORB algorithm was computationally efficient, it demonstrated reduced accuracy for speeds from 3 to 5 m/s compared to the other algorithms (Table 4.1). U-SURF not only outperformed SURF, but was computationally faster by 50% (Table 4.1). SIFT and CLORB exhibited the best overall precision and this difference was most noticeable on asphalt

(Figure 4.18). However, SIFT also had the slowest computational time (Table 4.1). Incorporating CLAHE as a pre-processing step prior to the ORB algorithm significantly improved the fit of zero intercept linear regression (Table 4.3) and reduced estimation error for asphalt in particular (Figure 4.18). This effect is likely due to the enhanced contrast improving the distinctiveness of features on asphalt which is a challenging surface for feature detection and description . Despite the increased computational cost, CLORB was still capable of real-time performance of approximately 25 Hz (Table 4.1). Overall, 1NN with cross-checking increased the inlier-outlier ratios (Figure 4.17) and the total number of inliers detected (Table 4.2) for all of the algorithms with a neglible decrease in computational speed (Table 4.1). With respect to 1NN with cross-checking and 2NN with the ratio-test, there was no significant difference between the accuracy of the two methods with respect to the imaging surface (Figure 4.18). In general, visual tracking on turf grass, seedlings, gravel and asphalt exhibited lower error and less variability compared to pasture and corn residue (Figure 4.11). Notably, visual tracking may have been estimating speed closer to real-time than RTK-GNSS during fast acceleration (>1 m/s), as can be seen in Figure 4.16 where CLORB over-estimated RTK-GNSS speed when accelerating and under-estimated RTK-GNSS speed when decelerating.

Based on these results, incorporating monocular visual tracking into agricultural sensing and control systems may be suitable for many field operations. Visual tracking was particularly effective on surfaces with relatively homogeneous subject depth, such as seedlings / soil, gravel, asphalt, or low weeds / grasses Therefore, visual tracking may be viable for motion feedback on implements like row crop cultivation, seeders, or strip tillage, or sprayers. However, without a method for depth compensation, monocular systems are only capable of moderate accuracy 0.2 - 0.3 m/s (95th percentile) on less homogenous surfaces, such as mature soy or pasture grass. For most operations, this accuracy is considered acceptable, and if computer vision is already used by the system (e.g., in some sectional sprayers) visual tracking may be implemented, effectively reducing the system’s cost by eliminating the need for additional GNSS receivers or fifth-wheels.

4.4.1 Future Research

With respect to cultivator guidance systems, visual tracking may prove to be valuable sensory feedback for adaptive controllers. Although travel speed was found to not have a significant effect on performance of a rotating-stabilizer hitch steering, visual tracking speed estimation may prove to be useful for adjusting responsiveness of other hitch steering systems, such as pivoting or parallelogram hitches. Similarly, challenges inherent to pivoting hitch steering systems, specifically the varying orientation of the camera with respect to the crop row, may be easily mitigated with orientation compensation based on visual tracking (Appendix C).

Additionally, classical PID control is not universally appropriate for implement systems due to the dynamic nature of their use. Further research into the development of an adaptive controller which utilizes visual tracking for reinforcement learning would be an excellent project to synthesize the concepts discussed in this research.

Monocular visual tracking may be a viable component of sensor fusion systems, specifically for depth estimation (Appendix D). Current methodologies for canopy height detection often utilize ultrasonic (Gil, 2007), LIDAR (Llorens, 2011), or stereovision systems. However, for tractors already equipped with RTK-GNSS, by comparing visual tracking and satellite data via a CANBUS networked system may be a viable method for estimating crop canopy height. Essentially, the error between the estimated speed of the visual tracking system and RTK-GNSS is proportional to crop height by the camera depth compensation model (Equation 4.16). Therefore, this approach be sufficient for crop height estimation without ultrasonic or laser distance sensors. Such a system may be easily integrated with existing monocular vision systems for sectional sprayer control (Tian, 2002). Ideally, the versatility of computer vision in a complex control system could be tested by conducting a cost-benefit analysis of boom height control on a sectional sprayer for a combination of RTK-GNSS, ultrasonic, LIDAR, stereovision, and monocular sensor systems.

4.5 Conclusion

The SIFT, SURF, U-SURF and CLORB algorithms all displayed similar accuracy regardless of surface, whereas ORB suffered a serious decrease in accuracy for speeds greater than 2 m/s, particularly for 1NN with cross-checking. U-SURF not only outperformed SURF, but was computationally faster by approximately 50%. Variants of the SIFT algorithm exhibited the best overall precision, which was most noticeable on asphalt. However, incorporating CLAHE pre-processing prior to the ORB algorithm with cross-checking greatly improved reliability, and despite the increased computational cost, was still capable of real-time performance (e.g. >25 Hz). Comparisons between the performance of ORB and CLORB for similar computational times show that CLAHE had an overall positive effect on performance. Overall, there was a general relationship between precision and degree complexity of feature-description. For example, SIFT exhibited the best precision, but also significantly slower computational time. The errors for each feature-descriptor were fit to a linear model with an intercept of zero using ordinary least squares regression. All of the algorithms exhibited a very close correlation to RTK-GNSS values with the exception of ORB. These results suggest that agricultural applications which require real-time vector estimation with low-cost cameras should utilize ORB with CLAHE due to the good compromise between precision and computational speed. However, applications which require high accuracy vector estimations and do not have strict time constraints should continue to use U-SURF due to its excellent performance on all surfaces and at all travel speeds. For tractor performance testing on concrete and asphalt (both of which are very homogeneous surfaces for visual tracking), SIFT had the best performance due to DoG's ability to find distinctive keypoints regardless of noise and homogeneity. When comparing visual tracking against RTK-GNSS, ORB with CLAHE pre-processing (CLORB) and 1NN cross-checking was the most robust with respect to real-time applications. CLORB achieved 95th percentile error of 0.23 m/s and faster processing times than RTK-GNSS. These results suggest that agricultural applications which require real-time vector estimation with low-cost cameras might consider CLORB due to the algorithm's beneficial compromise between precision, reliability, and computational speed.

Chapter 5: General Conclusions

This research has focused on uses of computer vision for implement feedback and control systems, specifically with respect to cultivator guidance systems. Organic agriculture and best management practices for pesticide-resistance include mechanical cultivation as an important method for non-chemical weed control. However, due to the time-sensitive nature of mechanical cultivation, it is essential that implements can be used early in the season and achieve the maximum possible rate of weed removal. This requires implement guidance systems which are non-contact and sufficiently reliable to achieve 95th percentile errors of less than 5 cm for travel speeds up to 12 km/h.

Computer vision systems have the potential to be versatile sensor platforms which provide a wealth of information for precision agriculture control systems. As the processing power of embedded systems increases and their cost decreases, it is now possible to analyze dense sensor data from digital imagery in real-time if proper algorithms and techniques are used. Applications for computer vision agriculture not only exist for tractors and field robotics, but also for mobile devices as tools for farmers. As the adoption of precision agriculture technologies increases and the possibility of autonomous tractors becomes a reality, computer vision will play a key role in the toolkit of sensor systems because of its versatility.

Due to the large amount of data in digital images, a single camera mounted to an agricultural implement can provide valuable feedback for more than a single purpose. In this work, an implement-mounted camera system was tested to detect the lateral error of the cultivator relative to the crop row. Two cameras were mounted to a cultivator implement tool-bar with the cameras positioned at a low-oblique perspective and 1.0 m above the soil surface. A band-pass plant detection method (BPPD) was used for segmentation of the crops and the weighted centroid of the lateral offset was estimated by a fast histogram percentile filter. A PID control signal was generated which was used as guidance feedback for a hydraulic hitch steering system. This method was successfully tested in field trials of corn and soy bean and significantly improved cultivator guidance on crops less than 15 cm in height compared to conventional mechanical row detectors. As a result of this work, computer vision guidance systems retrofitted to onto conventional hitch steering systems can extend the period of time during the season that farmers can perform inter-row cultivation. Detection of seedlings was not possible with mechanical sensors and impractical for RTK-GNSS due to high infrastructure costs, but the computer vision system was capable of reliably detecting the crop row offset of seedlings <10 cm in height even in the presence of weeds. Overall, the computer vision system achieved a 95th percentile error of less then 4.0 cm and outperformed the mechanical guiding rods at the <10 cm and 10 - 15 cm growth stages.

In addition to crop row detection, an evaluation of several feature detection and description algorithms was conducted to determine whether feature-based visual tracking could be reliably integrated with a implement-mounted monocular camera system. A visual tracking algorithm was developed and compared against high-accuracy RTK-GNSS data on six different surfaces (gravel, asphalt, turf grass, seedlings, corn residue, and pasture) for travel speeds between 1 and 5 m/s using several of the state-of-the-art feature detection and description algorithms. Based on this analysis, incorporating CLAHE as a pre-processing step to the ORB algorithm can improve accuracy and reliability on various agricultural surfaces with minimal loss in computational speed. Overall, SIFT was found to be the most accurate feature detection and description algorithm, but this accuracy came at a trade-off in computational speed. With respect to real-time visual tracking, CLORB offered the best comprise between accuracy (95th percentile of 0.23 m/s) and computational speed (25 Hz). Monocular feature-based visual tracking providing excellent accuracy on asphalt, gravel, turf grass, and seedlings, but was unreliable on highly heterogeneous surfaces such as tall pasture grass or heavy post-harvest corn residue.

Based on these findings, visual tracking could be used to provide both ground speed and tracking direction information in real-time for some agricultural operations. This vector data may prove to be useful for adjusting the responsiveness of systems such as cultivator hitches or sectional sprayers. Since visual tracking provides 4 degrees-of-freedom, implement vector data may also be applied to correct for poorly aligned cameras in the case of side-shift and rotating stabilizer hitch systems, mitigate rotational error caused by the motion of pivoting-hitch systems, or complement RTK-GNSS to enhance precision during non-steady-state movement.

Of course, incorporating computer vision systems into agriculture is not without limitations. Hardware costs for sufficiently powerful embedded systems limit many applications of computer vision due to the shear complexity of image analysis techniques. Fortunately, as the quality and processing power of embedded systems continue to improve it is becoming more economical for farm operations to consider the benefits which implement-mounted cameras offer. Coupled with adaptive control systems, embedded applications of computer vision can not only be used for autonomous guidance systems, but also to gather information on crop and soil conditions and provide farmers with greater control and monitoring capabilities. In conclusion, computer vision systems have incredible potential for feedback and control in precision agricultural due to their versatility and low-cost.

LIST OF REFERENCES

Aziz, K. A., El-Bardini, M., & El-Rabaie, N. M. (2015). Embedded Adaptive Fuzzy Controller Based on Reinforcement Learning for DC Motor with Flexible Shaft. *Arabian Journal for Science and Engineering*, 40(8), 2389-2406.

Alter, D. M. (2006). Using PWM output as a digital-to-analog converter on a TMS320F280X digital signal controller. Texas Instruments, Dallas, TX, Application Report SPAA88.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3), 175-185.

Baerveldt, S., & Ascard, J. (1999). Effect of soil cover on weeds. *Biological Agriculture & Horticulture*, 17(2), 101-111.

Bàrberi, P. (2002). Weed management in organic agriculture: are we addressing the right issues?. *Weed Research*, 42(3), 177-193.

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3), 346-359.

Bond, W., & Grundy, A.C. (2001). Non - chemical weed management in organic farming systems. *Weed Research*, 41(5), 383-405.

Boubertakh, H., Tadjine, M. Glorennec, P.Y., & Labiod, S. (2010). Tuning fuzzy PD and PI controllers using reinforcement learning. *ISA Transactions*, 49(4), 543-551.

Bowman, G. (1997). Steel in the field: a farmer's guide to weed management tools. Sustainable Agriculture Network handbook series (USA). no. 2.

Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Inc.

Brivot, R., & Marchant, J. A. (1996). Segmentation of plants and weeds for a precision crop protection robot using infrared images. *IEEE Proceedings - Vision, Image and Signal Processing*, 143(2), 118-124.

Brown, D.C. (1966). Decentering distortion of lenses. *Photogrammetric Engineering*, 32(3), 444–462.

Chiou, J.S., Tsai, S.H. & Lui, M.T. (2012). A PSO-based adaptive fuzzy PID-controller. *Simulation Modeling Practice and Theory*, 26, 49-59.

Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). Brief: Binary robust independent elementary features. In *11th European Conference on Computer Vision*, (pp. 778-792).

Chatizwa, I. (1997). Mechanical weed control: the case of hand weeders. *Brighton Crop Protection Conference on Weeds*, 203-208.

Comaniciu, D., Ramesh, V., & Meer, P. (2000). Real-time tracking of non-rigid objects using mean shift. In *Computer Vision and Pattern Recognition, Proceedings, IEEE Conference on* (Vol. 2, pp. 142-149).

Cormen, T. H. Leiserson, C. E., Stein, C., & Rivest, R. (1990). Greedy Algorithms. In *Introduction to Algorithms* (pp. 329). MIT Press.

Cowell, P. A., & Makanjuola, G. A. (1966). The lateral dynamic behaviour of tractor-mounted implements, with particular reference to the three-point linkage. *Journal of Agricultural Engineering Research*, 11(3), 152-169.

Damian, A. (2011). Weed control in maize crop by mechanical and manual management practices. *Scientific papers USAMV Bucuresti, Series A: Agronomy*, 262-268.

Dong, F., Mitchell, P. D., Hurley, T. M., & Frisvold, G. B. (2016). Quantifying Adoption Intensity for Weed-Resistance Management Practices and Its Determinants among US Soybean, Corn, and Cotton Farmers. *Journal of Agricultural and Resource Economics*, 41(1), 42-61.

Duin, R. P., Haringa, H., & Zeelen, R. (1986). Fast percentile filtering. *Pattern Recognition Letters*, 4(4), 269-272.

Ehsani, M. R., Sullivan, M. D., Zimmerman, T. L., & Stombaugh, T. (2003). Evaluating the dynamic accuracy of low-cost GPS receivers. In *2003 ASAE Annual Meeting* (pp 1).

El Harraj, A., & Raissouni, N. (2015, January). Histo-binary combined corner enhancement (HBCCE) improved algorithm for image processing corner detection. In *Proceedings of the International Conference on Image Processing, Computer Vision, and Pattern Recognition (IPCV)*, 95-101.

Ericson, E., & Astrand, B. (2008). Visual Odometry System for Agricultural Field Robots. In *Proceedings of the World Congress on Engineering and Computer Science*.

Erickson, B., & Widmar, D. A. (2015). Precision Agricultural Services Dealership Survey Results. *Sponsored By CropLife Magazine And The Center For Food And Agricultural Business. Purdue University*.

Fog, A. (2004). Optimizing software in C++: An optimization guide for Windows, Linux and Mac platforms.

Gan-Mor, S., Clark, R. L., & Upchurch, B. L. (2007). Implement lateral position accuracy under RTK-GPS tractor guidance. *Computers and Electronics in Agriculture,* 59(1)*,* 31-38.

Gauglitz, S., Hèollerer, T., & Turk, M. (2011). Evaluation of Interest Point Detectors and Feature Descriptors for Visual Tracking. *International Journal of Computer Vision,* 94(3), 335-360.

Gil, E., Escolà, A., Rosell, J. R., Planas, S., & Val, L. (2007). Variable rate application of plant protection products in vineyard using ultrasonic sensors. *Crop Protection*, 26(8), 1287-1297.

Griepentrog, H. W., Skou, P. T., Soriano, J. F., & Blackmore, B. S. (2005). Design of a seeder to achieve highly uniform sowing patterns. In *Proceedings of the 5th European Conference on Precision Agriculture* (pp. 675-682).

Grisso, R., Alley, M., & Heatwole, C. (2009, May 1). Precision Farming Tools: Global Positioning System (GPS). Retrieved April 12, 2016, from http://pubs.ext.vt.edu/442/442-503/442-503.html

Guijarro, M., Pajares, G., Riomoros, I., Herrera, P. J., Burgos-Artizzu, X. P., & Ribeiro, A. (2011). Automatic segmentation of relevant textures in agricultural images. *Computers and Electronics in Agriculture*, 75(1), 75-83.

Güzelkaya, M., Eksin, I., & Yeşil, E. (2003). Self-tuning of PID-type fuzzy logic controller coefficients via relative rate observer. *Engineering Applications of Artificial Intelligence*, 16(3), 227-236.

Guo, L. S., & Feng, L. (2003). A low-cost integrated positioning system of GPS and inertial sensors for autonomous agricultural vehicles. In 2003 ASAE Annual Meeting (p. 1). American Society of Agricultural and Biological Engineers.

Guo, L.-S., & Zhang, Q. (2008, January). A low-cost integrated positioning system for autonomous off-highway vehicles. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering,* 222(11), 1997-2009.

Hague, T., & Tillett, N. D. (2001). A bandpass filter-based approach to crop row location and tracking. *Mechatronics*, 11(1), 1-12.

Hague, T., Tillett, N. D., & Wheeler, H. (2006). Automated crop and weed monitoring in widely spaced cereals. *Precision Agriculture,* 7(1), 21-32.

Hamming, R. W. (1950). Error detecting and error correcting codes. *Bell System Technical Journal*, *29*(2), 147-160.

Hansen, P. K., Kristoffersen, P., & Kristensen, K. (2004). Strategies for non-chemical weed control on public paved areas in Denmark. *Pest Management Science*, 60(6), 600-604.

Harris, C., & Stephens, M. (1988, August). A combined corner and edge detector. In *Alvey Vision Conference* (pp. 50).

Hartmann, J. M., Klussendorff, J. H., & Maehle, E. (2013, September). A comparison of feature descriptors for visual SLAM. In *Mobile Robots (ECMR), 2013 European Conference on* (pp. 56-61)..

Heraud, J. A., & Lange, A. F. (2009). Agricultural automatic vehicle guidance from horses to GPS: How we got here, and where we are going. *American Society of Agricultural and Biological Engineers*.

Howard, A. (2008, September). Real-time stereo visual odometry for autonomous ground vehicles. In *Intelligent Robots and Systems, 2008 IEEE/RSJ International Conference on* (pp. 3946-3952).

Ionescu, N.E., Perianu, A., Popescu, A., Sarpe, N., & Roibu, C. (1996). Weed control in corn and soybean crops by mechanical and manual management practices*. Annales ANPP* (France).

Isard, M., & Blake, A. (1998). Condensation-conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1), 5-28.

ISO. (2012). ISO 12188-2: Tractors and machinery for agriculture and forestry -- Test procedures for positioning and guidance systems in agriculture -- Part 2: Testing of satellite-based auto-guidance systems during straight and level travel. Geneva, Switzerland: ISO.

ISO (2014). ISO 11783: Tractors and machinery for agriculture and forestry -- Serial control and communications data network. Geneva, Switzerland: ISO.

Jones, P.A., Blair, A.M., & Orson, J.H. (1996). Mechanical damage to kill weeds. *International Weed Control Congress, Copenhagen.*

Joos, M., Ziegler, J., & Stiller, C. (2010, June). Low-cost sensors for image based measurement of 2D velocity and yaw rate. In *Intelligent Vehicles Symposium (IV), 2010 IEEE* (pp. 658-662).

Kaminaka, M. S., Rehkugler, G. E., & Gunkel, W. W. (1981). Visual monitoring in a simulated agricultural machinery operation. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 23(2), 165-173.

Kerrisk, M. (n.d.). Atan2 - Linux User's Manual. Retrieved April 12, 2016, from http://linux.die.net/man/3/atan2

Keskin, M., & Say, S. M. (2006). Feasibility of low-cost GPS receivers for ground speed measurement. *Computers and Electronics in Agriculture*, 54(1), 36-43.

Kise, M., Zhang, Q., & Más, F. R. (2005). A stereovision-based crop row detection method for tractor-automated guidance. *Biosystems Engineering*, 90(4), 357-367.

Kocher, M.F., Smith, M.B., Grisso, R.D., & Young, L.J. (1994, December). Performance Comparison for Two Types of Implement Guidance Systems. *Paper No.94-3574, ASAE International Winter Meeting* (pp. 21). ASAE.

Kocher, M. F., Smith, M. B., Grisso, R. D., & Bashford, L. L. (2000). Performance tests of three-point mounted implement guidance systems. *Applied Engineering in Agriculture*, 16(6), 565-603.

Kopetz, H. (2011). Real-time systems: design principles for distributed embedded applications. Springer Science & Business Media.

Kulkarni, A.V., Jagtap, J.S., Harpale, V.K. (2013). Object recognition with ORB and its implementation on FPGA. *International Journal of Advanced Computer Research*, 3(3), 164.

Lee, S. H., Lee, S. K., & Choi, J. S. (2009). Correction of radial distortion using a planar checkerboard pattern and its image. *IEEE Transactions on Consumer Electronics*, 55(1), 27-33.

Lee, W. S., Alchanatis, V., Yang, C., Hirafuji, M., Moshou, D., & Li, C. (2010). Sensing technologies for precision specialty crop production. *Computers and Electronics in Agriculture*, *74*(1), 2-33.

Lee, K.Y., & Byun, K. J. (2013). A hardware design of optimized orb algorithm with reduced hardware cost. *Advanced Science and Technology Letters,* 43, 58-62.

Leutenegger, S., Chli, M., & Siegwart, R. Y. (2011, November). BRISK: Binary robust invariant scalable keypoints. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 2548-2555).

Lowe, D. (2004). Distinctive image features from local scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91-110.

Lhomme-Desages, D. (2009). Doppler-based ground speed sensor fusion and slip control for a wheeled rover. *IEEE/ASME Transactions on Mechatronics*, 14(4), 484-492.

Livingston, M., Fernandez-Cornejo, J., Unger, J., Osteen, C., Schimmelpfennig, D., Park, T., & Lambert, D. M. (2015). The economics of glyphosate resistance management in corn and soybean production. *Economic Research Report No. ERR-184, US Department of Agriculture*.

Llorens, J., Gil, E., & Llop, J. (2011). Ultrasonic and LIDAR sensors for electronic canopy characterization in vineyards: Advances to improve pesticide application methods. *Sensors*, 11(2), 2177-2194.

Marshall, T. (1992). Weed control in organic farming systems. *Proceedings 1st International Weed Control Congress*, 311-314.

Matson, P. A., Parton, W. J., Power, A. G., & Swift, M. J. (1997). Agricultural intensification and ecosystem properties. *Science*, *277*(5325), 504-509.

McCarthy, C. L., Hancock, N. H., & Raine, S. R. (2010). Applied machine vision of plants: a review with implications for field deployment in automated farming operations. *Intelligent Service Robotics*, *3*(4), 209-217.

Mrema, G. C., Baker, D., & Kahan, D. (2008). Agricultural mechanization in sub-Saharan Africa: time for a new look. FAO.

Meyer, G. E., Mehta, T., Kocher, M. F., Mortensen, D. A., & Samal, A. (1998). Textural imaging and discriminant analysis for distinguishing weeds for spot spraying. *Transactions of the ASAE*, 41(4), 1189-1197.

Meyer, G. E., & Neto, J. C. (2008). Verification of color vegetation indices for automated crop imaging applications. *Computers and Electronics in Agriculture*, 63(2), 282-293.

Möller, J. (2010, November). Computer vision – A versatile technology in automation of agriculture machinery. *International Agriculture and Garden Machinery Exhibition (EIMA) Bologna, 21st Annual Meeting*. Retrieved February 28, 2016, from http://www.clubofbologna.org/ew/documents/KNR_Moeller.pdf.

Moorthy, S., Parvathi, K. M., Boigelot, B., & Mercatoris, B. (2015, July). Effective segmentation of green vegetation for resource-constrained real-time applications. In *10th European Conference on Precision Agriculture.*

Mur-Artal, R., Montiel, J. M. M., & Tardos, J. D. (2015). ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Transactions on Robotics*, 31(5), 1147-1163.

Nistér, D., Naroditsky, O., & Bergen, J. (2006). Visual odometry for ground vehicle applications. *Journal of Field Robotics, 23(*1), 3-20.

Norsworthy, J. K., Ward, S. M., Shaw, D. R., Llewellyn, R. S., Nichols, R. L., Webster, T. M., & Witt, W. W. (2012). Reducing the risks of herbicide resistance: best management practices and recommendations. *Weed Science*, 60(1), 31-62.

Olsen, H. J. (1995). Determination of row position in small-grain crops by analysis of video images. *Computers and Electronics in Agriculture*, 12(2), 147-162.

Otto, G. (2000, August 29). Color in Computer Graphics. Retrieved March 1, 2016, from http://viz.aset.psu.edu/gho/sem_notes/color_2d/html/primary_systems.html

Pla, F., Sanchiz, J. M., Marchant, J. A., & Brivot, R. (1997). Building perspective models to guide a row crop navigation vehicle. *Image and Vision Computing*, 15(6), 465-473.

Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., Romeny, B., Zimmerman, J. B., & Zuiderveld, K. (1987). Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3), 355-368.

Pérez-Ruíz, M., Slaughter, D. C., Fathallah, F. A., Gliever, C. J., & Miller, B. J. (2014). Co-robotic intra-row weed control system. *Biosystems Engineering*,126(1), 45-55.

Pizer, S. M., Johnston, R. E., Ericksen, J. P., Yankaskas, B. C., & Muller, K. E. (1990). Contrast-limited adaptive histogram equalization. In *Publications by IEEE*.

Reza, A. M. (2004). Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology*, 38(1), 35-44.

Robati, J., Navid, H., Rezaei, M., & Ghiasi, A. R. (2012). Automatic guidance of an agricultural tractor along with the side-shift control of the attached row crop cultivator. *Journal of Agricultural Science and Technology*, 2(1), 151-158.

Rosten, E., & Drummond, T. (2006). Machine learning for high-speed corner detection. In *2006 European Conference on Computer Vision* (pp. 430-443).

Rosten, E., Porter, R., & Drummond, T. (2010). Faster and better: A machine learning approach to corner detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *32*(1), 105-119.

Rovira-Más, F., Zhang, Q., Reid, J. F., & Will, J. D. (2005). Hough-transform-based vision algorithm for crop row detection of an automated agricultural vehicle. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 219(8), 999-1010.

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011, November). ORB: an efficient alternative to SIFT or SURF. In *Computer Vision (ICCV), 2011 IEEE International Conference on* (pp. 2564-2571).

Sankaran, S., Mishra, A., Ehsani, R., & Davis, C. (2010). A review of advanced techniques for detecting plant diseases. *Computers and Electronics in Agriculture*, *72*(1), 1-13.

Scaramuzza, D. & Siefwart, R. (2008). Appearance-Guided Monocular Omnidirectional Visual Odometry for Outdoor Ground Vehicles. *IEEE Transactions on Robotics*, 24(5), 1015-1026.

Slaughter, D. C., Chen, P., & Curley, R. G. (1999). Vision guided precision cultivation. *Precision Agriculture*, 1(2), 199-217.

Slaughter, D. C., Giles, D. K., Fennimore, S. A., & Smith, R. F. (2008). Multispectral machine vision identification of lettuce and weed seedlings for automated weed control. *Weed Technology*, 22(2), 378-384.

Sofou, A., Evangelopoulos, G., & Maragos, P. (2005). Soil image segmentation and texture analysis: a computer vision approach. *Geoscience and Remote Sensing Letters, IEEE*, *2*(4), 394-398.

Stone, M. L., & Kranzler, G. A. (1992). Image-based ground velocity measurement. *Transactions of the ASAE*, *35*(5), 1729-1735.

Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F. (1999). A high integrity IMU/GPS navigation loop for autonomous land vehicle applications. *Robotics and Automation, IEEE Transactions on*, *15*(3), 572-578.

Strasdat, H., Montiel, J. M. M., & Davison, A. J. (2010, May). Real-time monocular SLAM: Why filter?. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on* (pp. 2657-2664).

Suzuki, S. (1985). Topological structure analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1), 32-46.

Szeliski, R. (2010). Computer vision: algorithms and applications. Springer Science and Business Media.

Taylor, J. E., Charlton, D., & Yúnez-Naude, A. (2012). The end of farm labor abundance. *Applied Economic Perspectives and Policy*, *34*(4), 587-598.

Thacker, G. W., & Coates, W. E. (1995a). Precision Guidance Techniques to Reduce Weed Competition and Production Costs in Cotton. Cotton: A College of Agriculture Report.

Thacker, G. W. and W.E. Coates. (1995b). Practical Considerations of Precision Guidance and Weed Control in Cotton. Cotton, A College.

Thacker, G. W., & Coates, W. E. (1996). How the quick hitch guidance systems work and their practical applications. Cotton: A College of Agriculture Report.

Thansandote, A., Stuchly, S. S., Mladek, J., Townsend, J. S., & Shlosser, H. (1977). A new slip monitor for traction equipment. *Transactions of the ASAE*, 20(5), 851-856.

Tian, L. F., & Slaughter, D. C. (1998). Environmentally adaptive segmentation algorithm for outdoor image segmentation. *Computers and Electronics in Agriculture*, 21(3), 153-168.

Tian, L. (2002). Development of a sensor-based precision herbicide application system. *Computers and Electronics in Agriculture*, 36(2), 133-149.

Tillett, N. D. (1991). Automatic guidance sensors for agricultural field machines: a review. *Journal of Agricultural Engineering Research*, 50, 167-187.

Tillett, N. D., Hague, T., & Miles, S. J. (2002). Inter-row vision guidance for mechanical weed control in sugar beet. *Computers and Electronics in Agriculture*, 33(3), 163-177.

Tillett, N. D., Hague, T., Grundy, A. C., & Dedousis, A. P. (2008). Mechanical within-row weed control for transplanted crops using computer vision. *Biosystems Engineering*, 99(2), 171-178.

Tippetts, B., Lee, D. J., Lillywhite, K., & Archibald, J. (2016, January). Review of stereo vision algorithms and their suitability for resource-limited systems. *Journal of Real-Time Image Processing*, 11(1), 5-25.

Tompkins, F. D., Hart, W. E., Freeland, R. S., Wilkerson, J. B., & Wilhelm, L. R. (1988). Comparison of tractor ground speed measurement techniques. *Transactions of the ASAE*, 31(2), 369-0374.

VIA Gallery. (2007, September 11). VIA Mini-ITX Form Factor Comparison. Retrieved April 12, 2016, from https://www.flickr.com/photos/15932083@N05/2123297741/

Visioli, A. (1999). Fuzzy logic based set-point weight tuning of PID controllers. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 29(6), 587-592.

Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine learning*, 8(4), 279-292.

Wilson, J. N. (2000). Guidance of agricultural vehicles—a historical perspective. *Computers and Electronics in Agriculture*, 25(1), 3-9.

Winstead, A. T., Norwood, S. H., Griffin, T. W., Runge, M., Adrian, A. M., Fulton, J., & Kelton, J. (2010). Adoption and use of precision agriculture technologies by practitioners. In *Proceedings of the 10th International Conference of Precision Agriculture.*

Witte, T. H., & Wilson, A. M. (2004). Accuracy of non-differential GPS for the determination of speed over ground. *Journal of Biomechanics*, 37(12), 1891-1898.

Woebbecke, D. M., Meyer, G. E., Von Bargen, K., & Mortensen, D. A. (1995). Color indices for weed identification under various soil, residue, and lighting conditions. *Transactions of the ASAE*, 38(1), 259-269.

Wolf, P. R. & Dewitt, B. A. (2000). Elements of Photogrammetry: with applications in GIS (Vol. 3). New York: McGraw-Hill.

Xu, G., Zhang, F., Shah, S. G., Ye, Y., & Mao, H. (2011). Use of leaf color images to identify nitrogen and potassium deficient tomatoes. *Pattern Recognition Letters*, 32(11), 1584-1590.

Yadav, G., Maheshwari, S., & Agarwal, A. (2014, September). Contrast limited adaptive histogram equalization based enhancement for real time video system. In *Advances in Computing, Communications and Informatics (ICACCI), 2014 IEEE International Conference on* (pp. 2392-2397).

Yi, J., Zhang, J., Song, D., & Jayasuriya, S. (2007, October). IMU-based localization and slip estimation for skid-steered mobile robots. In *Intelligent Robots and Systems, 2007 IEEE/RSJ International Conference on* (pp. 2845-2850).

Zhang, J., Kaess, M., & Singh, S. (2015). Real-time method for depth enhanced visual odometry. *Autonomous Robots*, 1-13.

Zheng, L., Zhang, J., & Wang, Q. (2009). Mean-shift-based color segmentation of images containing green vegetation. *Computers and Electronics in Agriculture*, 65(1), 93-98.

APPENDIX A: Code

Please see CD-ROM

APPENDIX B: Experimental Data

Please see CD-ROM

APPENDIX C: Orientation Compensation

Computer vision systems have been successfully employed on inter-row cultivation for row detection, with a particular emphasis on side-shift hitches (Robati, 2012). However, pivoting-hitch systems, such as the Sunco AcuraTrak (Sunco Manufacturing, Inc., North Platte, Nebraska, USA), pose a specific set of challenges for crop row estimation via imaging systems. If the cameras are mounted to the tool-bar, the pivoting action of the hitch effectively changes the orientation of the cameras relative to the crop row (Figure C.1). For example, a camera which is positioned 200 cm laterally and 50 cm longitudinally from the pivot point would have approximately 2 cm of positional error due to pivoting only 5°. Therefore, the computer vision detection process will overestimate or underestimate the lateral error (depending on configuration and tracking direction) which may negatively impact the steering system.

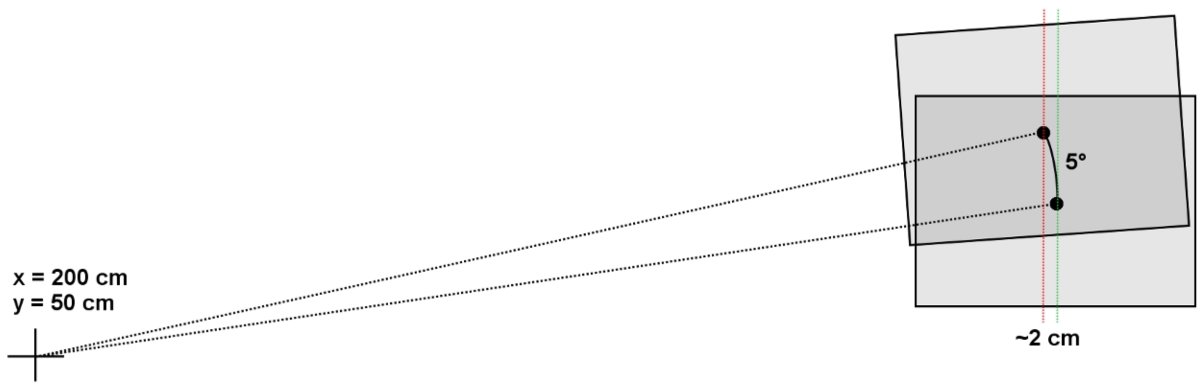

Figure C.1 Diagram of pivoting-hitch induced camera error.

Fortunately, rotational error due to the motion of a pivoting-hitch can be mitigated by employing rotation compensation using the motion vector information provided by visual tracking. Assuming that the longitudinal component of the implement's velocity in the direction of the crop row (V_y) is much greater than the rate of lateral adjustment (V_x), the small angle approximation applies and the orientation of the image can be corrected by applying a rotational matrix transform about the hitch's pivot point:

$$M = \begin{bmatrix} \sim & & (1-\sim)\cdot x_p - \quad \cdot y_p \\ - & \sim & \cdot x_p + (1-\sim)\cdot y_p \end{bmatrix} \tag{C.1}$$

where $\alpha = \cos(\theta)$, $\beta = \sin(\theta)$, and (x_p , y_p) is the pivot-point about which the image will be rotated (mm).

However, for applications with relatively a small variation in orientation, or embedded systems with limited processing power, this additional transformation may not be practical to compute. For example, the row estimation method discussed in Chapter 3, the errors are relatively small for rotational changes of less than

±10° which is the typical range of motion for pivoting-hitch systems, e.g. Sunco AcuraTrak has a pivoting angle of ±5°. Therefore, if the orientation change is small, the positional error of the image centroid relative to the true position of the row may be approximated with the following equation:

$$\begin{aligned} \pounds_x &= x_p - \sqrt{x_p^2 - y_p^2} \cdot \sin\left(\bar{\ } + {}^{\circ} - \mathrm{atan2}\left(y_p, x_p\right)\right) \\ \pounds_y &= y_p - \sqrt{x_p^2 - y_p^2} \cdot \cos\left(\bar{\ } + {}^{\circ} - \mathrm{atan2}\left(y_p, x_p\right)\right) \end{aligned} \tag{C.2}$$

where x_p is the lateral distance from the camera to the pivot point, y_p is the longitudinal distance from the camera to the pivot point, ρ is the roll of the camera (assumed to be 0°), and θ is the instantaneous orientation of the camera relative to the direction of travel.

APPENDIX D: Crop Height Estimation

A possible application of monocular visual tracking system is for inferring crop height, i.e. estimating the subject depth from the camera lens to the imaging surface. Crop height, although not particularly relevant to cultivator guidance systems, is very valuable for pesticide application (Gil, 2007). By maintaining a consistent height of a sprayer system above the crop canopy, greater uniformity of pesticide application can be achieved (Llorens, 2011). If a complementary method for speed detection is available, e.g. RTK-GNSS or fifth-wheel, the relative error of visual tracking estimate, which is a function of subject depth, can be used to calculate to infer the crop height.

This principle was tested in controlled conditions using a laboratory test-bench. A 3D-printed bracket was constructed to hold two red laser diodes (650 nm) spaced 80 mm apart and oriented along the line of sight of the camera (Figure D.1). Although more complex configurations could be employed to reduce errors due to surface heterogeneity and reflectance, e.g. diode patterns consisting of three or more diodes of differing colors, the simple two diode pattern used for calibration was considered sufficient as a proof-of-concept.

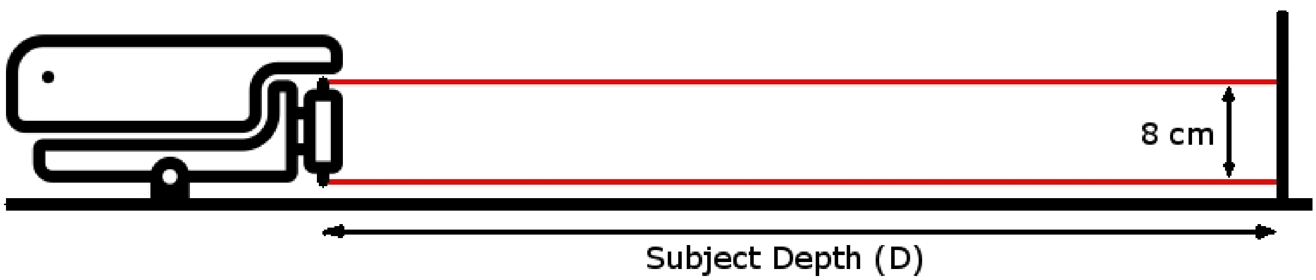

Figure D.1 Depth estimation test setup.

Eighteen distances were tested at 5 cm intervals from 15 cm to 100 cm, i.e. the assumed true depth of the camera model. Five replicates were taken for each depth. For each replicate, an RGB image was captured and subsequently transformed to the HSV color-space. Once in the HSV color-space, a band-pass filter was employed to create a mask of bright red pixels. Lastly, the mask was remapped to correct for any radial distortion inherent to the camera lens (Equation 4.10), and elliptical morphological opening was used to smooth the mask. To calculate the centroid of each dot with sub-pixel precision, the contours were detected using the algorithm proposed by Suzuki et al. (1985) was applied to the mask. Lower computational time and more consist results were achieved by limiting the contour search area to ±50 px of the horizontal axis and rejecting any circles whose centroids were greater than ±2 px from the center-line. Lastly, the distance between the two centroids was calculated. Table D.1 presents the results of the camera depth calibration process.

Table D.1 Results of subject depth calibration test

Depth (cm)	Mean Dot Spacing (px)	Standard Deviation (mm)	Depth Correction Factor (px/mm)
100	80.0	0.13	1.00
95	84.0	0.19	1.05
90	88.6	0.21	1.11
85	93.4	0.25	1.17
80	98.1	0.15	1.23
75	103.4	0.29	1.29
70	111.5	0.39	1.39
65	119.1	0.51	1.49
60	127.0	0.45	1.59
55	137.0	0.61	1.71
50	150.1	0.55	1.88
45	166.3	0.56	2.08
40	185.0	0.45	2.31
35	209.6	0.75	2.62
30	241.0	0.94	3.01
25	286.4	0.68	3.58
20	352.6	0.95	4.41
15	451.9	1.02	5.65

Using this data, the Levenberg-Marquardt method of non-linear least squares was used to fit a power function model which achieved a coefficient of determination (R^2) of 0.9999 (Figure D.2) for an exponential distortion factor of -1.0918 px/px. As such, for any given speed estimate from a monocular visual tracking system and true ground speed from RTK-GNSS, the height of the crop canopy can be estimated with the following equation:

$$h = D\left(1-\left(\frac{v_{CV}}{v_{GPS}}\right)^{P}\right) \tag{D.1}$$

where D is the default height of the camera (100 cm), v_{CV} is the speed estimated by visual tracking (m/s), v_{GPS} is the speed estimated by RTK-GNSS (m/s), and P is the exponential distortion factor (-1.0918 px/px).

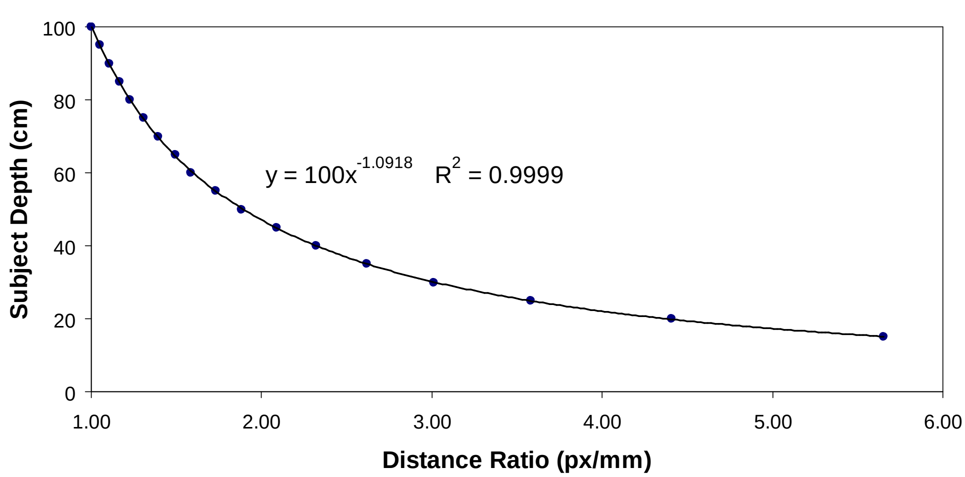

Figure D.2 Subject depth to imaging surface as a function of the pixel-per-millimeter resolution.

Although not formally tested in this thesis, future research is needed to verify if this lost-cost laser-diode approach provides sufficient precision and robustness for in-situ crop height compensation. Compared to LIDAR and stereovision, this method may be satisfactory for depth compensation for monocular systems. An ideal experimental setup would compare the monocular system to three common depth detection methodologies: stereovision, ultrasonic, and laser distance. Additionally, instead of a conventional tractor, the camera should be mounted to a sprayer at a height of 2.0 m (typical ground clearance of a boom sprayer). This configuration would allow the different sensor systems to be evaluated later in the growing season and with crop conditions which are more widely representative of those during mid-season pesticide application.