

# **TROUGH-BASED LOCALIZATION FOR AUTONOMOUS AGRICULTURAL ROBOTS**

by

Piyanun Ruangurai

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of  
Doctor of Engineering in Microelectronics and Embedded Systems

Examination Committee    Dr. Mongkol Ekpanyapong (Chairperson)  
Prof. Matthew N. Dailey  
Prof. Manukid Parnichkun

External Examiner:    Prof. Pedro Melo-Pinto  
Department of Engineering  
University of de Tras-os-Montes e Alto Douro  
Vila Real, Portugal

Nationality:    Thai

Previous Degree:    Master of Engineering in Microelectronics  
and Embedded Systems  
Asian Institute of Technology  
Pathumthani, Thailand

Scholarship Donor:    Rajamangala University of Technology Phra  
Nakhon, Thailand - Royal Thai Government  
- AIT Fellowship

Asian Institute of Technology  
School of Engineering and Technology  
Thailand  
May 2022

## **AUTHOR'S DECLARATION**

I, Piyanun Ruangurai, declare that the research work tractor ried out for this dissertation was under the regulations of the Asian Institute of Technology. The work presented in it is my own and has been generated by me as the result of my original research, and if external sources were used, such sources have been cited. It is original and has not been submitted to any other institution to obtain another degree or qualification. This is a true copy of the dissertation, including final revisions.

Date: 22 April 2022

Name: Piyanun Ruangurai

Signature:

## **ACKNOWLEDGMENTS**

Firstly, I would like to thank Dr. Mongkol Ekpanyapong, who is my thesis Chairperson, Prof. Matthew N Dailey advises me, and Prof. Manukid Parnichkun recommends my work. All of them gave me a chance to study at AIT and carried me on with my research. They also encourage, suggest, and support me a lot.

I appreciate thanking all of my awesome friends, Mr. Chatchai Pruetong, Mr. Jednipat Moonrinta, Mr. Chaiyaporn Silawatchananai, and all of my friends from the Mechatronics laboratory and AI center. I strongly say that this research cannot be complete without their bits of help, suggestions, and encouragement.

Lastly, I was encouraged and supported by my family for a long journey thus, I would like to say thank you to all of my beloved.

## ABSTRACT

The navigation task is one of the most difficult tasks for an intelligent robot because of these three reasons: “Where am I?”, “Where am I going?” and “How do I get there.” For precision agriculture, high precision and accuracy equipments are required for path tracking but they are expensive. Thus, this research proposes to improve the localization method based on machine vision in order to obtain low cost and data richness. In this work, the direct seeding of rice tractor with a machine vision system is applied as the pivot study. During traveling along the paddy field, the nearby wheel track row is the reference for guidance and the control variables are the orientation and the lateral distance between a tractor and a nearby row. Three algorithms: Principal Components Analysis (PCA), Hough transform (HOUGH), and Random Sample Consensus (RANSAC) is used to estimate the trough and the performance comparison has been studied. The experimental results show that PCA and HOUGH methods are suitable to estimate the trough distance. A novel log-likelihood objective function is proposed to improve the accuracy of the trough estimate method. It provides the optimized trough position compared with the ground truth. Our optimization with the PCA method has the average errors between trough position and ground truth as 28.4 and 28.1 millimeters in two experiments in the year 2019/2 and 2020/1 respectively.

**Keywords:** Agricultural robot, Image processing, Rice weeding, Trough detection

# CONTENTS

	<b>Page</b>
<b>ACKNOWLEDGMENTS</b>	<b>iii</b>
<b>ABSTRACT</b>	<b>iv</b>
<b>CONTENTS</b>	<b>v</b>
<b>LIST OF TABLES</b>	<b>vi</b>
<b>LIST OF FIGURES</b>	<b>vii</b>
<b>LIST OF ABBREVIATIONS</b>	<b>ix</b>
<b>CHAPTER 1 INTRODUCTION</b>	<b>1</b>
1.1 Background of the Study	1
1.2 Statement of the Problem	2
1.3 Objective of Thesis	2
1.4 Scopes and Limitations	2
<b>CHAPTER 2 LITERATURE REVIEW</b>	<b>3</b>
2.1 Autonomous Agriculture Systems	3
2.2 Machine Vision Systems in Agriculture	6
2.3 Trend of Robotic in Agriculture	12
<b>CHAPTER 3 METHODOLOGY</b>	<b>15</b>
3.1 Overall System	15
3.2 Hardware Design	16
3.3 Algorithms	20
<b>CHAPTER 4 RESULT</b>	<b>37</b>
4.1 Simulation Result of Using GPS Tracking Based on EKF	37
4.2 Experiment Result Tracking Based on GPS without EKF in Paddy Field	39
4.3 Localization Base on Machine Vision System	43
<b>CHAPTER 5 CONCLUSION AND FUTURE WORK</b>	<b>49</b>
5.1 Conclusion	49
5.2 Future Work	49
<b>REFERENCE</b>	<b>51</b>

## LIST OF TABLES

<b>Tables</b>	<b>Page</b>
Table 2.1 Comparison of Guidance Systems	4
Table 2.2 Comparison of Tracking and Control Systems	5
Table 2.3 Comparison of Machine Vision Systems in Fruit Detection	7
Table 2.4 Comparison of Machine Vision Systems in Guidance Row Detection	10
Table 3.1 Pseudocode for Trough Merging Step	23
Table 3.2 Pseudocode for Best Trough Selection	24
Table 3.3 Pseudocode for PCA Trough Estimation	25
Table 3.4 Pseudocode for RANSAC Trough Estimation	26
Table 3.5 Pseudocode for Hough Transform Trough Estimation	27
Table 3.6 Pseudocode of Maximum likelihood Guidance Distance and Orientation Estimation	32
Table 4.1 Trough Detection Results	44
Table 4.2 Summary of Guidance Row Estimation Results in Experiments 2019/2 and 2020/1	45
Table 4.3 Difference between Each Line Fitting Method and the Optimized Line Fitting Method, Over Experiments 2019/2 and 2020/1	47
Table 4.4 Time Consumption of Each Method Over Experiments 2019/2 and 2020/1	48

## LIST OF FIGURES

<b>Figures</b>	<b>Page</b>
Figure 2.1 Robot Farming System	13
Figure 3.1 Overall System of Autonomous Tractor for Direct Seeding of Rice	15
Figure 3.2 Commercial Tractor Integrating with a Rear Rice Planting System	16
Figure 3.3 DC Motor for Steering Control	17
Figure 3.4 Stepping Motor for Velocity Control	17
Figure 3.5 Sensors	18
Figure 3.6 Microcontroller	19
Figure 3.7 Motor Drivers	20
Figure 3.8 Block Diagram of Vision-based Trough Localization	21
Figure 3.9 Sample Results of Best Trough Selection Process	23
Figure 3.10 Guidance Row Orientation and Distance Estimation	25
Figure 3.11 Relationship between Trough's Depth and Assumed Probability of Obtaining White and Black Pixels	31
Figure 3.12 Path Planning	34
Figure 3.13 Parameters in the World Coordinate and the Tractor Coordinate	34
Figure 3.14 Block Diagram of the Control System for Path Tracking	35
Figure 4.1 Simulink Model of Generating Circle Path and GPS Position	37
Figure 4.2 Result of Generate Circle Path and GPS Position	38
Figure 4.3 Result on Simulink Model of Tracking GPS with EKF	39
Figure 4.4 Tracking Using GPS and Steering Controller Based Pure Pursuit for Estimation Orientation	40
Figure 4.5 The Straight Line Result of Tracking Based Pure Pursuit for Estimating Orientation	40
Figure 4.6 Tracking Using GPS and Steering Controller Based Pure Pursuit and Look-ahead for Estimating Orientation	41
Figure 4.7 Result of Tracking Using GPS and Steering Controller Based Pure Pursuit and Look-ahead for Estimating Orientation	41
Figure 4.8 Fuzzy Rule	42
Figure 4.9 Result of U-turn Path Tracking Based Fuzzy Self-adjustment for PD	43

## LIST OF FIGURES

<b>Figures</b>	<b>Page</b>
Figure 4.10 The Ground Truth of Guidance Row Distance and Orientation Using Markers	44
Figure 4.11 Example Estimates of Guidance Row Orientation and Lateral Distance	45
Figure 4.12 Guidance Row Estimation Results of Experiments 2019/2 and 2020/1	46



## LIST OF ABBREVIATIONS

$P_1$	= previous way point $[X_{n-1}, Y_{n-1}]$ with respect to world coordinate
$P_2$	= forward way point $[X_n, Y_n]$ with respect to world coordinate
$P_T$	= current tractor's position $[X_T, Y_T]$ with respect to world coordinate
$P_{LT}$	= look-ahead point design $[X_{LT}, Y_{LT}]$ with respect to world coordinate
$\theta_T$	= angular deviation of the forward way point with respect to the north direction
$\theta$	= bearing of the tractor orientation respect to north direction of the world coordinate
$\theta_P$	= angular deviation of the guidance row with respect to the north direction
$\theta_d$	= desired yaw orientation
$\theta_{d_{new}}$	= desired look-ahead orientation
$\theta^+$	= angular deviation of look-ahead direction with respect to the forward way point direction
$\theta_{L_t}$	= angular deviation of look-ahead direction with respect to guidance row distance
$\theta_{P_t}$	= angle between tractor to forward way point direction and guidance row
$\phi$	= guidance row orientation
$d$	= guidance row distance
$L_t$	= look-ahead distance
$d_T$	= distance from tractor to forward way point
$k$	= gain
$\delta$	= tractor's steering
$V$	= tractor's velocity
$\mathbf{P}$	= homogeneous camera matrix
$\mathbf{P}^+$	= pseudo-invert of homogeneous camera matrix
$\mathbf{K}$	= intrinsic matrix
$\mathbf{R}$	= pose – rotation matrix
$\mathbf{t}$	= translation matrix
$\mathbf{X}$	= homogeneous representation of a tractor plane system $[X_t \ Y_t \ Z_t \ 1]^T$

- x** = homogeneous representation of an image plane  $[u \ v \ 1]^T$
- H** = homogeneous coordinate matrix map between the ground coordinate and the image point
- $t_c$  = camera translation matrix
- $Z_c$  = height of camera
- $h$  = trough depth
- $\sigma$  = standard deviation of trough's width
- $\beta$  = variable parameter

# CHAPTER 1

## INTRODUCTION

### 1.1 Background of the Study

While Thailand has become one of the largest rice exporters in the world, most Thai agricultural activities still depend on human labor with limited technology enhancement resulting in low productivity. The importance of using autonomous systems has been raised by some recent studies to improve agricultural production. In autonomous agricultural systems, we needed to have two major subsystems: the autonomous vehicle that employs navigation guidance and control algorithms, and the autonomous add-on equipment that involve a set of applications to solve crop operation problems such as planting, weeding, fertilizing, and harvesting.

Many researchers have also emphasized the importance of using multiple sensors fusion in autonomous vehicle systems to improve accuracy and reduce uncertainty by merging information from multiple sensors including real-time kinematic-global positioning systems (RTK-GPS), inertial measurement units (IMU), and machine vision. Sensor fusion algorithm has been widely adopted for an autonomous agricultural vehicle today to enhance guidance system, localization system, and control system (Astrand et al., Bakker et al., Gan-Mor et al., Nieuwenhuizen et al. and . Perez-Ruiz et al.). However, the cost of those multiple sensors is expensive and they require sufficient accuracy for precision farming. Other researchers have emphasized the autonomous add-on equipment, such as inter-row and intra-row weeding, fertilizing, harvesting, and planting (Carballido et al., Peruzzi et al, Reiser et al, Vieri et al.).

Our work is focused on autonomous tractor for rice planting. Although Puddled-transplanted rice (CT-TPR) has been a conventional method of rice planting in Asia (Rashid et al., 2009) for several decades, the drawback of CT-TPR is that it requires more water and labor than the Dry and Wet direct seeding of rice (DSR) process (Kumar and Ladha, 2011). DSR process is more likely to increase and DSR planting machines are widely available in the market today. Hence, this work adapts the DSR tractor as a baseline machine with add-on equipment to be used as one of the core parts of the proposed system and emphasizes on improving row-guidance system.

## **1.2 Statement of the Problem**

The navigation task is one of the most difficult tasks for an intelligent robot because it needs to answer three questions, “Where am I?”, “Where am I going?” and “How do I get there?” (Leonard and Durrant-Whyte, 1991). It needs to perform localization, path planning, and control tracking to answer all these three questions. High precision and accuracy of travel distance are mandatory for the agriculture work thus many previous researchers have been using multiple sensors fusion such as RTK-GPS, LiDAR, compass and odometer, etc. with some filters. However, the high precision sensor cost is very high and not affordable for Thai farmers. The other problem is RTK-GPS will have information loss when satellites are occluded. The solution is to use low price sensor such as a camera instead of RTK-GPS. Moreover, the guidance and localization system need to have a reliable algorithm in order to get potential applications of autonomous rice vehicles.

## **1.3 Objective of Thesis**

The main objectives of this thesis are to design algorithms for an autonomous guidance and localization system. They are described as below.

1. To develop an automatic guidance system
2. To get high inter-row precision by using machine vision to detect trough made from the tractor wheel’s trace
3. To optimize the localization position

## **1.4 Scopes and Limitations**

The main scopes and limitations of this thesis are described below.

1. Using square/rectangle wet field
2. Controlling water level in the field
3. Estimating trough parameters from offline

## CHAPTER 2

### LITERATURE REVIEW

In this chapter, the autonomous agriculture system researchers are reviewed. The topics are related to guidance system, localization system, tracking and control system, and add-on machine equipment in agriculture.

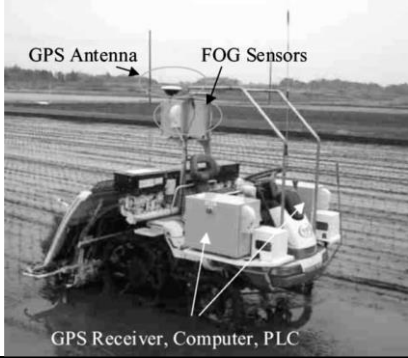
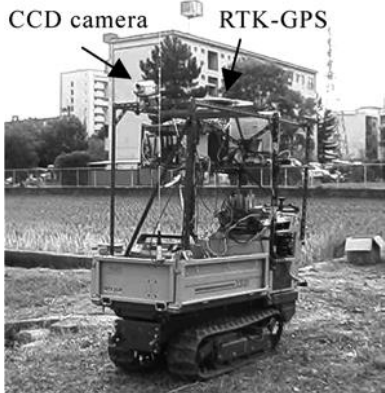
#### **2.1 Autonomous Agriculture Systems**

An autonomous tractor in agriculture is an intelligent tractor, which emphasizes the enhancement of navigation or guidance and tracking control system. The purpose of using the autonomous tractor is to achieve high efficiency. The autonomous implementation in agriculture is the development of robot applications, such as planting, weeding, fertilizing, and harvesting to improve crop operation. Nagasaka et al. (2004) used autonomous guidance for rice transplanting. In this research, the guidance of a tractor is based on fiber-optic gyroscope (FOG) and real-time global position system (RT-GPS) fusion data. Morimoto et al. (2005) develop a prototype of an autonomous farm transport robot. Most of this research paper was used machine vision to detect intersection guidance markers (IGM) using Hough transform at intersection road. Rotary encoders were installed as guidance sensors and used dead reckoning for turning on the intersection. All guidance systems mentioned above are shown in Table 2.1.

The tracking control system in tractor such as Backmanvet al. (2012), they were controlling lateral position of the tractor. Two algorithms were applied in the tracking part. The first algorithm is the Nonlinear Model Predictive Control (NMPC), and the second algorithm is the Target Point algorithm (TP). The algorithms' results were compared. The Extended Kalman Filter (EKF) was integrated into the algorithm. It was used to smooth the signal fusion between GPS, IMU, and 2D laser to recognize adjacent driveline. The best result was achieved by using the Nonlinear Model Predictive model. Moreover, Kraus et al. (2013) emphasized controlling the tractor in uncertain environments such as different soil conditions by implementing Moving Horizon Estimation (MHE) to estimate the wheel's slip and orientation of the tractor. The result of estimation was fed to NMPC to provide a wheel velocity and a steering rate.

**Table 2.1**

*Comparison of Guidance Systems*




Researcher group	Guidance systems	Applications
Nagasaka et al. (2004)	<p>FOG+RTGPS</p> 	Transplanting
Morimoto et al. (2005)	<p>Machine vision based IGM guidance</p> 	Transportation tractor

Han et al. (2015) introduced the path-tracking for a guidance tillage tractor in a paddy field based on a dynamic model of tractor, tire slip, and side force from the soil. The algorithm used a look-ahead in tracking and setting the offset distance for updating a new waypoint defined by limiting the boundary offset (LBO). It was tested in many different parameters of look-ahead distance, LBO, velocity, and Interval at the curved path for finding the optimal result. The error on a straight line was highest error when the tractor was following a curve and tried to come back to the straight line in the next row. Moreover, the soil's moisture content parameter was another effect on the tracking error. Kayaca et al. (2015) introduced the kinematic controller for desiring velocity and yaw. After that, velocity was controlled based on the Proportional-Integral-Derivative (PID) controller. Model Predictive Control (MPC) was set up to control yaw dynamics which is calculated based on nonlinear least squares frequency domain system

identification. EKF was also used to estimate the position, velocity, and orientation of the tractor. The tracking and control systems mentioned above are illustrated in Table 2.2.

**Table 2.2**

*Comparison of Tracking and Control Systems*

Researcher group	Tracking and control systems
J. Backman (2012)	<p>NMPC</p> 
Kraus et al. (2013)	<p>MHE+NMPC</p> 
Han et al. (2015)	<p>Look-ahead</p> 

PID, MPC

Kayaca et al. (2015)



## 2.2 Machine Vision Systems in Agriculture


Machine vision systems are becoming popularly used for agricultural related tasks to meet high precision demand and rich data with a low-cost sensor. Nowadays, the systems are used in many applications. Robots, tractors, or drones which autonomous platforms normally use machine vision. For example, the research area in green fruit detection of Chaivivatrakul and Dailey (2014) proposed a method for improving yield detection of bitter melon and pineapple fruits based on texture information acquired from the machine vision system. The method had main five steps: features and descriptors, feature classification, fruit point mapping, morphological closing, and region extraction. They used well-known methods of features and descriptors such as SIFT Harris, ORB, SIFT, SURF, Mesh, and IORB. The result shows that ORB+SURF128 and Harris+SURF128 are the best features and descriptors for pineapple and bitter melon respectively. The classification step uses a SVMs classifier. Fruit points are used to generate the location of the candidate as a binary image. Next, the regions of the candidate can be merged by using morphological closing with a disc or ellipse shape and then extracting large positive fruit regions as fruit detection. In addition, Li et al. (2016) introduced an approach for the automatic prediction of green citrus based on a vision sensor that uses fast normalized cross correlation (FNCC) for counting immature green citrus fruit. First, FNCC is used to detect the location of fruit based on feature and texture. The result of using FNCC still has some false positives thus RBH color is used to filter out. Moreover, the results from the previous filter intersect with the results of fruit detection based on circular Hough transform (CHT)



and image background removal based RBH. The purpose of using CHT is to detect the shape of citrus which is a circle. Finally, they filter out again by using the feature of smooth and entropy. Relevant research, Qureshi et al. (2017) created two new approaches to detect green mangos fruit in mango tree canopies images; the first perform texture-based dense segmentation algorithm using machine learning, and another is elliptical shape-based model and color algorithm. The comparison of fruit detection is shown in Table 2.3.

**Table 2.3**

*Comparison of Machine Vision Systems in Fruit Detection*

Researcher group	Applications
Chaivivatrakul and Dailey (2014)	<p data-bbox="592 869 1070 913">Pineapple and bitter melon detection</p> 

Li et al. (2016)

Green citrus fruit yield detection

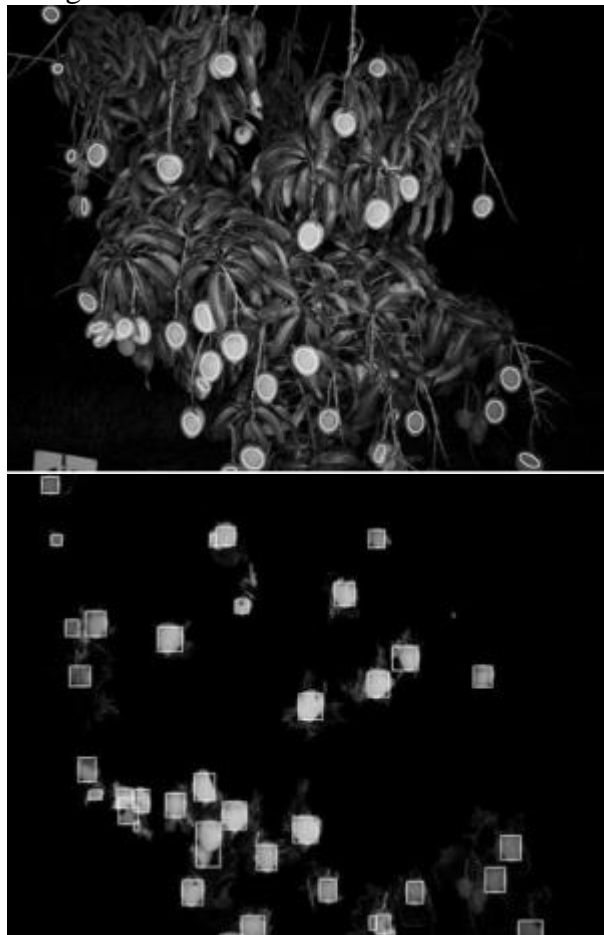


\* red color circuit is final result.  
blue color circuit is intersection result.

---

Qureshi et al. (2017)

Mango detection



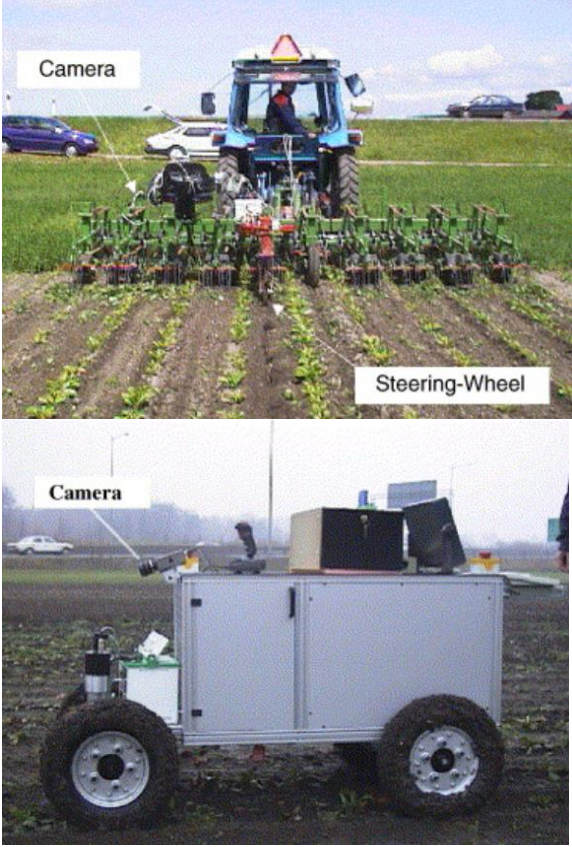
\*top and bottom images are color and smoothness filtering,  
and texture-based dense segmentation respectively.

---

Many studies are developing machine vision systems algorithms to improve the guidance of autonomous robot. For instance, Astrand and Baerveldt (2005) proposed recognition of plant rows using a Hough transform and adapted the plants' size in purpose to get data more than one row. The experiment was set in two detect on rape and sugar beets. The result shows that the model can be able guidance machine in the applications of cultivator and guidance following of mobile robot, Leemans and Destain (2006). They used a machine vision system for seed row localization in the sugar beet field. This research estimated guidance row distance from detecting furrow row which made from special disc, and seed row. The researchers found that seed row estimation based on the Hough transform is better than the estimation of furrow rows. After that Leemans and Destain (2007) applied their seed row localization algorithm on the sugar beet seed drill machine with connected to the tractor. Then, they controlled the lateral of the seed drill machine based on the information of their algorithm. Xue et al. (2012) applied variable field-of-view (FOV) machine vision on an automation mobile robot to navigate guidance between two corn crop rows and the controller of the mobile robot using a fuzzy logic controller to follow guidance crop row. In addition, Jiang et al. (2015) applied a new approach for multiple crop row detection for guidance an agricultural robot using machine vision systems. Their method had five main steps: grayscale image transformation, binary image using Otsu's algorithm, candidate of center point prediction using multiple ROIs to divide the scene horizontally based on geometric constraints on the inter-row space, real center point confirmation, and crop row detection based on linear least square method (LSM). Another research group is Kanagasingham et al. (2020). The researchers improved an autonomous crop-weeding robot based on a machine vision system. The advantage of using machine vision was to estimate rice crop rows as line guidance for tracking. Then fusion with global data from compass, and GPS for path routing. The comparison of machine vision systems in guidance row detection shows in Table 2.4.

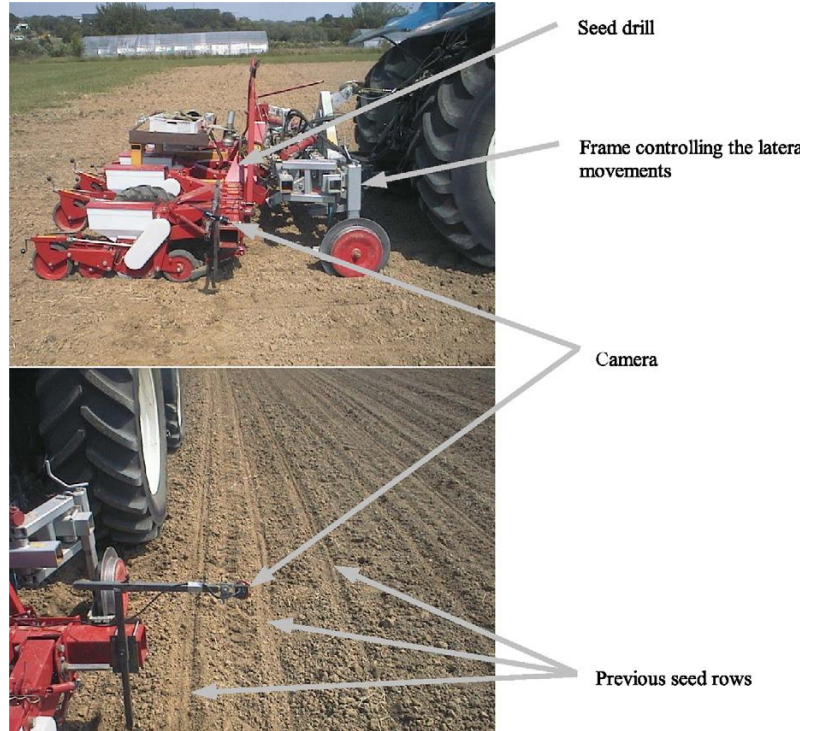
**Table 2.4**

*Comparison of Machine Vision Systems in Guidance Row Detection*

Researcher group	Applications
Astrand and Baerveldt (2005)	<p data-bbox="595 461 1038 499">Cultivator and guidance following</p>  <p data-bbox="595 1339 1378 1406">*top and bottom images are rape detection for cultivator and sugar beet detection for following plant row respectively</p>

Leemans and Destain, (2006, 2007).


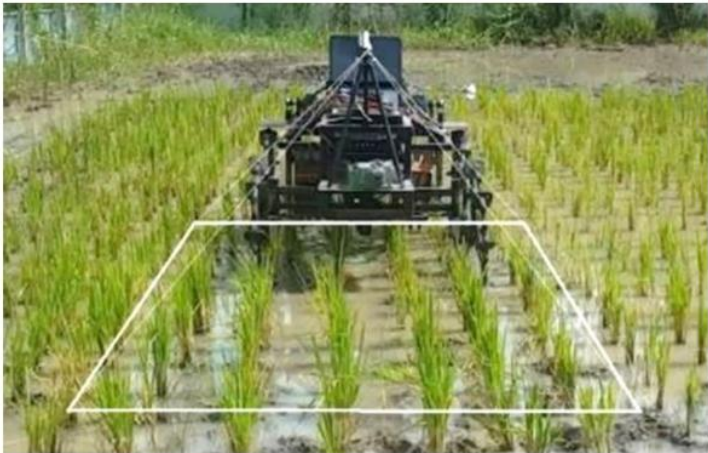
Sugar seed drill



Xue et al. (2012)

Corn crop row guidance



Researcher group	Applications
Jiang et al. (2015)	<p data-bbox="595 282 1015 320">Crop row detection for guidance</p> 
Kanagasingham et al. (2020)	<p data-bbox="595 781 1297 819">Machine vision based rice crop row for weeding robot</p> 

### 2.3 Trend of Robotic in Agriculture

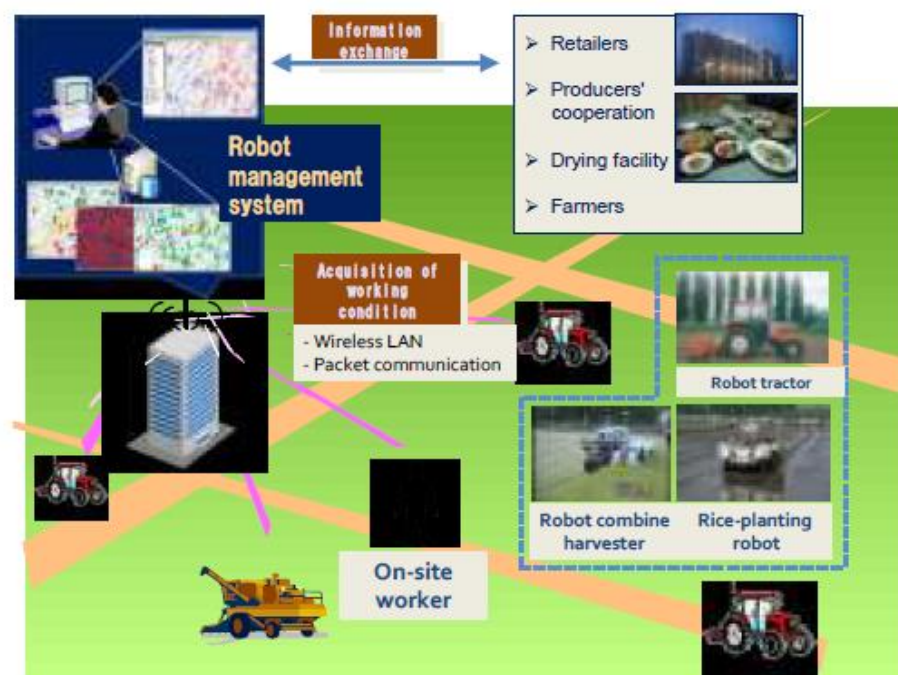
Agricultural tractors are useful for multiple purposes including planting, weeding, spraying, sprouting, fertilization, and harvesting. With the rapid advances in technology, autonomous tractors are becoming increasingly important for increasing labor efficiency and labor shortage problems. The important factors that affect the design of an autonomous agricultural tractor are primarily cost and efficiency. An autonomous agricultural tractor requires precise sensors and actuators, but many sensors used nowadays, such as real-time kinematic-global positioning system (RTK-GPS) sensors, are precise and accurate enough but quite expensive and inconvenient to use. Moreover, there has an issue in some applications in which GPS information is not close enough to the trajectory point. A new challenge of researchers nowadays is to develop a sensor to be accurate information to control system at low cost. At the same

time, machine vision is emerging to obtain besides low cost and data richness, vision sensors give precise local or relative information, which complements global sensors such as GPS.

In addition, full automation in farming is the goal of researchers. For example, Noguchi and Barawid (2011) were set up on electronic robots and standard tractors which were used in the plant system until the harvest system. They tried to reduce cost by using low-cost sensors and using LAN network to communicate between operator and robot in purpose to manage farm system. The result of this system was shown that robots could follow the navigation map accurately and it can operate via LAN network communication in real-time.

**Figure 2.1**

*Robot Farming System (Reprinted from Noboru Noguchi, 2011)*



From all previous mentioned, the researchers found that a low-cost automatic guidance system needs to be developed. The aim of this paper is the development of a low-cost automatic guidance system for direct rice seeding. To control an autonomous vehicle, at least two parameters, steering, and velocity must be controlled. To control these parameters precisely, the guidance or feedback needs to be precise. This study deals with the problem of obtaining the tractor's guidance signal based on machine vision.

The idea is to use the previous line made by the tractor's wheel during the direct seeding of rice to generate the guidance signal for the next row. Machine vision is used to measure the distance and angular deviation of the tractor with respect to the guidance row.

There are several key differences between our approach and the previous state-of-the-art research. Leemans and colleagues find the best filter for use in the Hough transform to track a furrow row or seed row in a dry field. Our aim is to find a method to detect guidance rows in a wet and muddy field. The guidance line made by the tractor in the mud during direct rice seeding is very noisy, with large clumps of displaced soil.



## CHAPTER 3

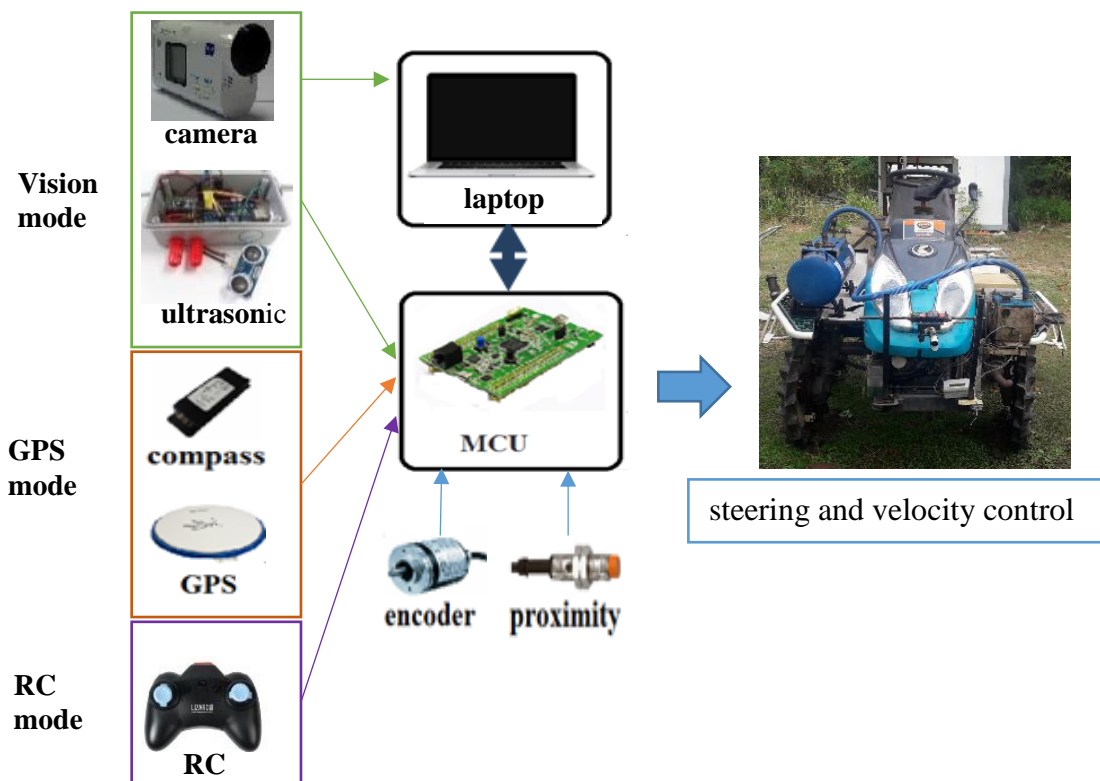
### METHODOLOGY

This chapter describes the overall systems of the developed autonomous tractor for direct seeding of rice. The hardware design and methodologies are mentioned in this section. There are three parts: guidance and localization algorithms, path planning, and path tracking control. Figure 3.1 shows the system diagram of the autonomous tractor for direct seeding of rice that modified from the commercial tractor.

#### 3.1 Overall System

**Figure 3.1**

*Overall System of Autonomous Tractor for Direct Seeding of Rice*



There are three operation modes of the autonomous tractor: manual operation, autonomous guidance based on GPS, and autonomous guidance based on vision. In manual operation, the user can maneuver the steering wheel, acceleration, and braking pedal by driving or using a remote control (RC). In autonomous self-guidance mode,

GPS and compass are used for localization and orientation. Then the tractor can steer the heading for the path tracking. The last operation mode is autonomous guidance based on the vision which is the main topic of this research. The steering and velocity control of the tractor is handled by the STM32F4 Discovery board where the tractor command from either laptop or RC signal. A DC motor is installed and coupled with the steering wheel. The steering angle is measured by using the incremental encoder. The pedal for acceleration and braking is attached to a stepper motor. The wheels' velocity is sensed by using proximity sensors installed at both left and right driven wheels. Information feed to the feedback control loop in order to provide precise control of the tractor.

### **3.2 Hardware Design**

The tractor used in this project is modified from a 17 horsepower commercial tractor integrating the rice planting system shown in Figure 3.2. The tractor origin coordinate system is set at the origin position (0, 0, 0), which is the center of the tractor's projection onto the ground. The actuators and sensors are attached to the tractor.

**Figure 3.2**

*Commercial Tractor Integrating with a Rear Rice Planting System*



#### **3.2.1 Actuator**

In this work, a 250W 24 VDC motor is attached to control steering through the gear transmission as shown in Figure 3.3. The incremental encoder is used to measure the

steering angle. Two limit switches are also attached to detect the most left and right steer, including the initialization. The user can detach the transmission from the steering wheel when it operates in manual driving mode. To control the velocity of the tractor, a stepper motor is attached to press the accelerator similar to what a human does, as shown in Figure 3.4. One limit switch is used as the home position, then the tractor does not move.

**Figure 3.3**

*DC Motor for Steering Control*



**Figure 3.4**

*Stepping Motor for Velocity Control*



### **3.2.2 Sensors**

NovAtel-OEM628 GPS is used in purpose to get a position of the tractor in latitude and longitude. The information of the orientation tractor comes from an HMR3000 compass

bearing respect to the north. KOYO rotary encoder is attached at steering to obtain the current steering angle. Another sensor is ifm-IF5646 proximities which are attached at each rear wheel. It sends the data to calculate the distance traveling and velocity of the tractor which counts the round per minute by using a special disc. The Sony action camera model hdr-as200vr attached to the gimbal aims to use for detecting guidance lines, which were made from the previous tractor's track. The position of the camera is in front of the tractor at a height of 85 cm and a pitch of 0.26 radians referenced to the ground. The gimbal helps to stabilize the pitch and row of the camera because those parameters change significantly as the tractor moves through the paddy field. Thus, HC-SR04 ultrasonic is attached to the tractor to measure height between camera and ground. All hardware is shown in Figure 3.5.

**Figure 3.5**

*Sensors*



GPS (NovAtel-OEM628)



Camera(hdr- as200vr)



Compass (HMR3000)



Proximity (ifm-IF5646)



Ultrasonic (HC-SR04)



Rotary Encoder (TRD-N1000-RZ)



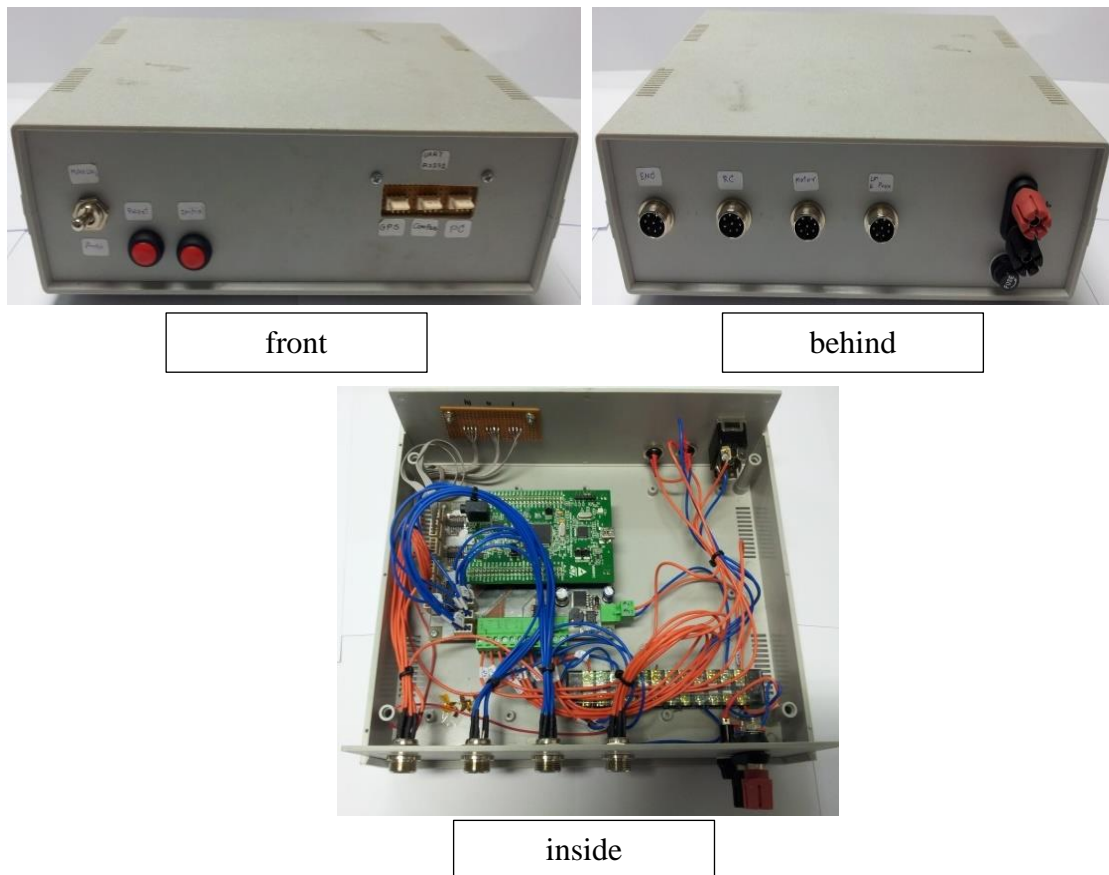
Camera with gimbal

### 3.2.3 Embedded Electronic Device

The stm32 microcontroller is used as the main controller for path planning, receiving sensor data for localization, and trajectory tracking control via close loop feedback. The desire of this controller is shown in Figure 3.6. The stepping and DC drivers are shown in Figure 3.7. Moreover, MATLAB Simulink toolboxes with the Waijung block set library are used to write the code. A laptop is used as a second controller. It uses for interfacing with a user and estimating local trough. The C++ language with OpenCV library is used for processing images.

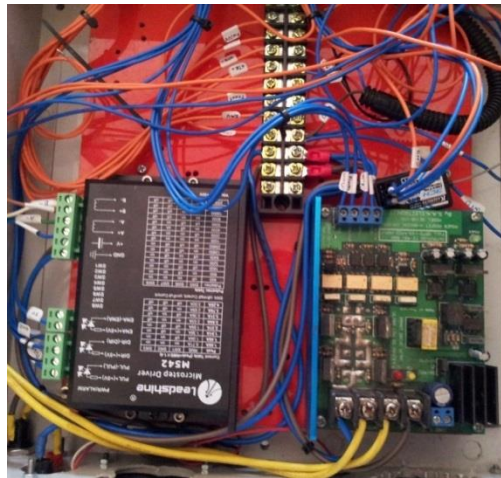
**Figure 3.6**

#### *Microcontroller*



**Figure 3.7**

*Motor Drivers*



### **3.3 Algorithms**

This research has three main parts; the first part is guidance and localization where the current tractor position. The second part is path planning which is the waypoints for tractor traveling. The third part is path tracking and controlling the tractor closed to the waypoints.

#### ***3.3.1 Guidance and Localization***

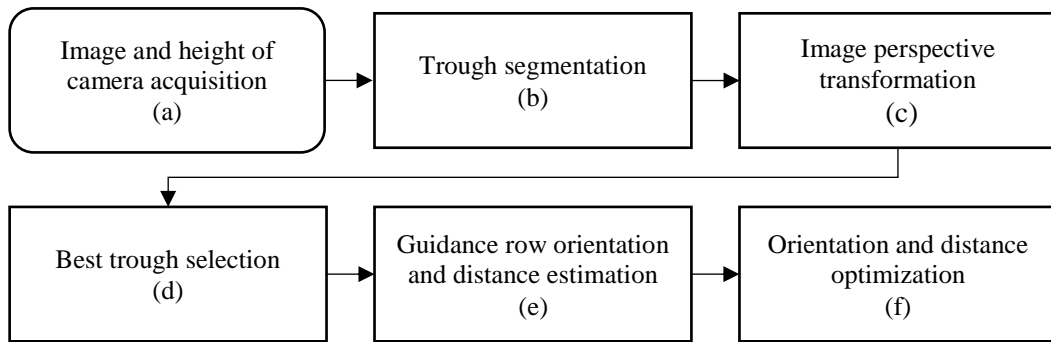
In this research, two approaches are applied: global localization based on GPS, and local localization, which emphasize a topic of this research, using image processing to find trough localization based on the previous wheel's track. Moreover, local trough localization has adopted an optimization method to help reduce estimation error.

**3.3.1.1 GPS and Compass System.** The application uses GPS for locating latitude and longitude of tractor and uses with digital compass for estimating the current orientation of tractor. First, received data from GPS with ten iterations then average latitude and longitude. After that, the position information is converted to the initial world coordinate as position (0, 0). In addition, the initial orientation comes from the compass concerning the North direction.

**3.3.1.2 Vision System.** The localization technique adopted in this research is image processing to detect trough, which is made from the tractor's wheels of the previous row. The guidance line generated by the tractor in the mud during traveling for rice planting is very noisy, with large clumps of displaced soil. In the experiment, RGB images are acquired at 30 frames per second with 1920x1080 pixels resolution. Block diagram overall of vision-based trough localization shows in Figure 3.8.

**Figure 3.8**

*Block Diagram of Vision-based Trough Localization*



**3.3.1.2.1 Image and Height of Camera Acquisition.** First, the RGB image and the current height of the camera are acquired. The RGB image where image resolution 1920x1080 pixels is fed into the next step.

**3.3.1.2.2 Trough Segmentation.** Guidance row segmentation is applied in this step. Image is converted to grayscale and undistorted. We use OpenCV calibration software for estimating camera parameters. Its use compensates for an undistorted image. After that, the image is reduced from 1920x1080 pixels to 1920x730. This step aims to reduce computation time. After that, Gaussian blurring and median filters are used followed by adaptive thresholding for separating background from the trough.

**3.3.1.2.3 Image Perspective Transformation.** In this step, the image perspective transformation is transformed using a homogeneous camera matrix (P) which consist of intrinsic matrix (K) and extrinsic parameters (pose - rotation 'R', and translation 't'), to get a perpendicular bird's-eye view The formula of camera shows

$$\mathbf{x}_{3 \times 1} \propto (\mathbf{K}_{3 \times 3} [\mathbf{R}_{3 \times 3} \mid \mathbf{t}_{3 \times 1}]) \mathbf{X}_{4 \times 1}, \quad (1)$$

Where  $\mathbf{X}$  is a homogeneous representation of a tractor plane system ( $[X_t \ Y_t \ Z_t \ 1]^T$ ) and  $\mathbf{x}$  is a homogeneous representation of an image plane ( $[u \ v \ 1]^T$ ). In the experiment, the tractor travels along the paddy field thus the only translation of the camera has an effect ( $\mathbf{t}_c = [X_c \ Y_c \ Z_c]^T = -\mathbf{R}^T \mathbf{t}$ ), the height of the camera is changed ( $Z_c$ ) because the gimbal helps to stabilize the rotation. The height of camera comes from an ultrasonic sensor. We also assume that the ground is flat ( $Z_r=0$ ). Thus, an instantaneous estimate of the homogeneous coordinate matrix ( $\mathbf{H}$ ) is map between the ground coordinate and the image point of the bird's eye-view coordinate. We can write

$$\begin{aligned} \mathbf{x}_{3 \times 1} &= \mathbf{H}_{3 \times 3} \mathbf{X}_{3 \times 1} \\ \mathbf{X}_{3 \times 1} &= \mathbf{H}_{3 \times 3}^{-1} \mathbf{x}_{3 \times 1} \end{aligned} \quad (2)$$

**3.3.1.2.4 Best Trough Selection.** After the image perspective transform as mentioned before, best trough selection is the next step. In this step, the image has at least one object thus we need to classify the object to be the candidate of the trough. The objects in which the center of the moment is less than or equal to 50 pixels in perspective view, will be considered as the same trough. Table 3.1 shows the detail of pseudocode for trough merging. After we get candidates for the trough, the best trough is selected for tractor tracks. The pseudocode for best trough selection is shown in Table 3.2. Starting with the three largest areas of trough candidates are selected. Then, if there have more than three candidates, we will compute the ratio between major and minor axis length. If the minor length or the ratio of each candidate is less than or equal to the conditions which are 3 and 40 pixels respectively, they will be removed from the trough candidates. Moreover, if the major length has more than or equal to 215 pixels, they are also removed. Currently, if trough candidates still have three candidates, the smallest area will be removed from the candidate. But if the trough candidates have two candidates, then select the one that has an area more than 1.5 times of each other but if it is not in that condition, the trough candidate which has closed to the tractor will be selected as the best trough. The sample of best trough selection shows in Figure 3.9.



**Table 3.1**

*Pseudocode for Trough Merging Step*

---

Input: Binary bird's eye view image  $\mathbf{I}$  containing candidate trough connected components, threshold distance  $\delta$  (50 in our experiments).

Output: Integer label image with merged connected components for candidate troughs.

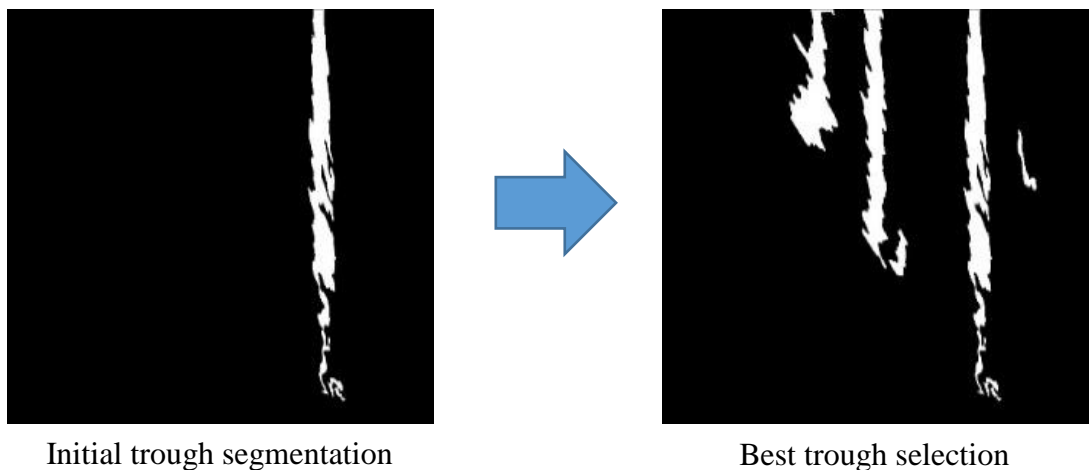
---

```
 $\mathbf{C} \leftarrow$  connected components ( $\mathbf{I}$ )
For each  $c_i \in \mathbf{C}$  do
    Compute x-coordinate of center of mass  $m_i$  for  $c_i$ 
end for
sort  $(c_i, m_i)$  pairs in ascending  $m_i$  order
For  $i \in 1..|\mathbf{C}|-1$  do
    If  $|m_i - m_{i+1}| \leq \delta$ 
        Mark  $c_i$  and  $c_{i+1}$  for merging
    end if
end for
Return image with distinct label for each merged set of
components.
```

---

**Figure 3.9**

*Sample Results of Best Trough Selection Process*



**Table 3.2***Pseudocode for Best Trough Selection*


---

Input: Image **I** labeling pixels of each connected component from previous step, minimum minor length  $l_u$  (40 pixels), maximum minor length  $l_e$  (215 pixels), area  $A$  (20,000 pixels), and ratio between major length and minor length  $R$  (3).

Output: Binary image with selected trough connected component(s).

---

```

C ← three largest connected components in I
w ← image width
c0 ← largest element of C
For each ci ∈ C
    compute major axis length mi and minor axis length ni
    ri ← mi/ni
    If ni ≤ lu or ni ≥ le or ri ≤ R
        remove ci from C
if |C| = 3
    remove smallest component from C
if |C| = 2
    c1 ← largest component of C
    c2 ← second largest component of C
    |c1| ← size area of largest component of C
    |c2| ← size area of second largest component of C
    x1 ← horizontal center of mass of c1
    x2 ← horizontal center of mass of c2
    if |c1| > 1.5|c2| or |x1 - w/2| < |x2 - w/2|
        return c1
    else
        return c2
return c0

```

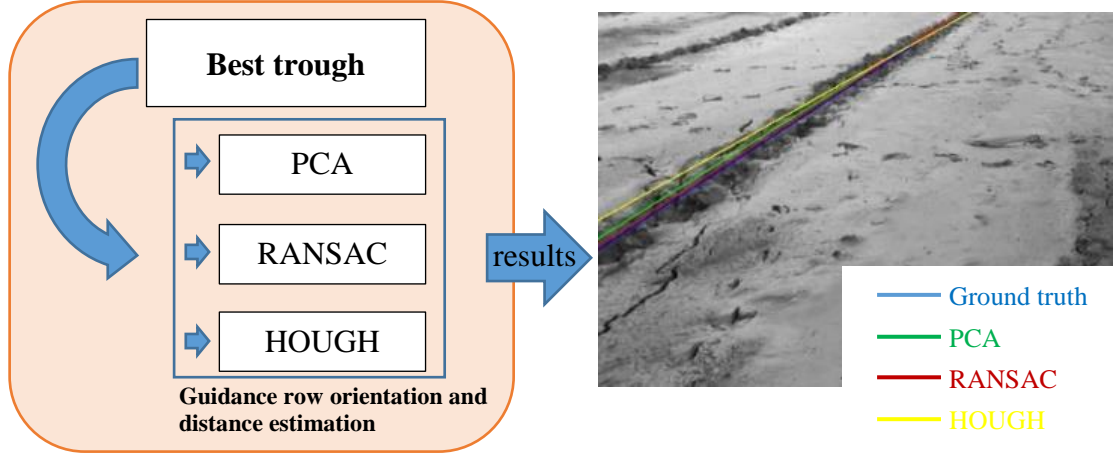
---

**3.3.1.2.5 Guidance Row Orientation and Distance Estimation.** The best trough connected component step returns a set of trough points. Then orientation ( $\phi$ ) and distance ( $d$ ) of the selected connected component estimate the parameters themselves as a guidance row for the tractor. Three methods are compared for straight-line estimation for predicting guidance row parameters as in Figure 3.10. The three methods are consist of PCA, RANSAC, and the Hough transform. In OpenCV C++, the PCA method is provided with the function of `pca_analysis`. For RANSAC and the

Hough transform. They can write in the C++ language. Table 3.3, Table 3.4, and Table 3.5 are the pseudocode of PCA, RANSAC and Hough transform respectively.

**Figure 3.10**

*Guidance Row Orientation and Distance Estimation*



**Table 3.3**

*Pseudocode for PCA Trough Estimation*

---

Input: Binary bird's eye view image  $\mathbf{I}$  containing selected best trough connected component, scaling  $s$  (200 pixel/meter), and pixel position  $\mathbf{P}_c$  of tractor coordinate system origin in  $\mathbf{I}$ .

Output: parameters  $\phi$  and  $d$  of trough line estimate.

---

$\mathbf{X} \leftarrow$   $2 \times N$  vector of  $(x, y)$  coordinates of pixels in trough

connected component  $\mathbf{C}$ , expressed as  $\begin{bmatrix} x_1 & \dots & x_N \\ y_1 & \dots & y_N \end{bmatrix}$

$$\bar{\mathbf{X}} \leftarrow \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$$

$\mathbf{I} \leftarrow$  identity matrix  $2 \times 2$

$$\mathbf{S} \leftarrow \mathbf{X} - [\mathbf{I} \times \bar{\mathbf{X}}] \times \mathbf{1}_{1 \times N}$$

$\mathbf{e} \leftarrow$  eigenvector of  $\mathbf{S}\mathbf{S}^T$  corresponding to largest eigenvalue  $\lambda$

$$\Delta \mathbf{L} = \mathbf{e}\lambda$$

$$\mathbf{L} = \bar{\mathbf{X}} + \Delta \mathbf{L}$$

$$\phi = \text{atan2}(-\Delta L_x, \Delta L_y)$$

$$\mathbf{P}_n = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{L} \\ \|\Delta \mathbf{L}\| \end{bmatrix}$$


---

---

```

Pr = Pc -  $\bar{\mathbf{X}}$ 
d = (Pr · Pn) / s
If Lx < Pcx
    d = -d
return φ, d

```

---

**Table 3.4**

*Pseudocode for RANSAC Trough Estimation*

---

Input: Binary bird's eye view image **I** containing selected best trough connected component, threshold  $\delta$  (14) scaling *s* (200 pixel/meter), and pixel position **P<sub>c</sub>** of tractor coordinate system origin in **I**.

Output: parameters *φ* and *d* of trough line estimate.

---

```

X ← 2xN vector of (x, y) coordinates of pixels in trough
connected component C, expressed as  $\begin{bmatrix} x_1 & \dots & x_N \\ y_1 & \dots & y_N \end{bmatrix}$ 
For 1000 iterations:
    Let xi, xj be two random sample points from X
    Let L be the line spanning xi, xj
    For each xk ∈ X
        Let dk be the perpendicular distance between xk and
L
    end for
    Let Nnew be the count of points xk for which dk <  $\delta$ 
    if Nnew > Nold
        Nold = Nnew
        Save L as Lbest
    end if
end for
(a, b, c) ← Lbest
 $\Delta\mathbf{L} = \mathbf{a} - \mathbf{b}$ 
 $\phi = \text{atan2}(-\Delta L_x, \Delta L_y)$ 
 $\mathbf{P}_n = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{L} \\ \|\Delta\mathbf{L}\| \end{bmatrix}$ 
Pr = Pc - a
d = (Pr · Pn) / s
If ax < Pcx
    d = -d
return φ, d

```

---

**Table 3.5***Pseudocode for Hough Transform Trough Estimation*


---

Input: Binary bird's eye view image  $\mathbf{I}$  containing selected best trough connected component, scaling  $s$  (200 pixel/meter), and pixel position  $\mathbf{P}_c$  of tractor coordinate system origin in  $\mathbf{I}$ .

Output: parameters  $\phi$  and  $d$  of trough line estimate.

---

$\mathbf{X} \leftarrow$   $2 \times N$  vector of  $(x, y)$  coordinates of pixels in trough

connected component  $\mathbf{C}$ , expressed as  $\begin{bmatrix} x_1 & \dots & x_N \\ y_1 & \dots & y_N \end{bmatrix}$

Let  $w$  be the width of the image and  $h$  be the height of image

$\mathbf{A} \leftarrow$  binned accumulator with  $\rho$  range from  $-w/2$  to  $w/2$  in 1-pixel increments,  $\Phi$  range from 0-180 in 1-degree increments

For each  $x \in \mathbf{X}$

Calculate  $\rho, \phi$  for  $x$

$\mathbf{A}[\rho, \phi] \leftarrow \mathbf{A}[\rho, \phi] + 1$

end for

select  $\rho, \phi$  from maximum accumulator of  $\mathbf{A}[\rho, \phi]$

if  $45 \leq \phi \leq 135$

$$\mathbf{L} = \begin{pmatrix} 0 & w \\ \left( \frac{(\rho - w/2) - (-w/2)\cos(\phi)}{\sin(\phi)} + \frac{h}{2} \right) & \left( \frac{(\rho - w/2) - (w - w/2)\cos(\phi)}{\sin(\phi)} + \frac{h}{2} \right) \end{pmatrix}$$

else

$$\mathbf{L} = \begin{pmatrix} \left( \frac{(\rho - w/2) - (-h/2)\sin(\phi)}{\cos(\phi)} + \frac{w}{2} \right) & \left( \frac{(\rho - w/2) - (w - h/2)\sin(\phi)}{\sin(\phi)} + \frac{w}{2} \right) \\ 0 & w \end{pmatrix}$$

$$\phi = \text{atan2}(-\Delta L_x, \Delta L_y)$$

$$\mathbf{P}_n = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{L} \\ \|\Delta \mathbf{L}\| \end{bmatrix}$$

$$\mathbf{P}_r = \mathbf{P}_c - L_1$$

$$d = (\mathbf{P}_r \cdot \mathbf{P}_n) / s$$

If  $L_x < P_{cx}$

$$d = -d$$

return  $\phi, d$

---

**3.3.1.2.6 Guidance Row Orientation and Distance Optimization.** After orientation and distance of guidance row are estimated from the previous step, The two parameters  $(\phi, d)$  feed as initial guesses to the optimizer which needs the parameters in tractor coordinate and the process of planar projection that generated the image. Maximum likelihood estimation (MLE) is used to optimize the parameters. The MLE helps to estimate an unknown parameter vector  $(\theta)$  by maximizing the joint probability of the observed data  $(x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n)$  under those parameters. The likelihood function is written as

$$\mathcal{L}(\theta; x_1, x_2, x_3, \dots, x_n) = \prod_{i=1}^n p(x_i | \theta). \quad (3)$$

We take the log of the likelihood to get

$$\ell(\theta; x_1, x_2, x_3, \dots, x_n) = \sum_{i=1}^n \log p(x_i | \theta) \quad (4)$$

Our goal is to find the optimal  $\theta^*$  maximizing this function:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} (\ell(\theta; x_1, x_2, x_3, \dots, x_n)). \quad (5)$$

In this research, two unknown parameters are the guidance row distance and orientation  $(d, \phi)$  that we want to estimate. For the observed data are the set of selected guidance row pixels that contain noise. We model the trough to be an inverted Gaussian function. The trough model is created as

$$Z_r = -he^{-\frac{(X_r \cos \phi + Y_r \sin \phi - d_i)^2}{2\sigma^2}}, \quad (6)$$

where  $X_r$ ,  $Y_r$ , and  $Z_r$  are a 3D point on the flat ground of the paddy field in the tractor coordinate system,  $d_i$  is the distance from the center of trough,  $\sigma$  is the standard deviation of trough's width which assumed to be 0.03 in this experiments,  $h$  is the deepest trough depth which assumed constant at 0.12 m in our experiments,  $\phi$  is the guidance row orientation, and  $d$  is the guidance row distance. The sign of  $d$  is negative when the guidance row is to the right of the tractor; otherwise, it is positive.

The camera model maps points between tractor and image coordinates by using backprojection of image point can be rewritten as

$$\begin{bmatrix} X_r \\ Y_r \\ Z_r \\ 1 \end{bmatrix} = \begin{bmatrix} T_{cx} \\ T_{cy} \\ T_{cz} \\ 1 \end{bmatrix} + \beta \begin{bmatrix} \mathbf{P}^+ \\ v \\ 1 \end{bmatrix}. \quad (7)$$

where

$$\mathbf{P}^+ = \begin{bmatrix} P_{11}^+ & P_{12}^+ & P_{13}^+ \\ P_{21}^+ & P_{22}^+ & P_{23}^+ \\ P_{31}^+ & P_{32}^+ & P_{33}^+ \\ P_{41}^+ & P_{42}^+ & P_{43}^+ \end{bmatrix},$$

$$\mathbf{P}_1^+ = [P_{11}^+ \ P_{12}^+ \ P_{13}^+],$$

$$\mathbf{P}_2^+ = [P_{21}^+ \ P_{22}^+ \ P_{23}^+],$$

$$\mathbf{P}_3^+ = [P_{31}^+ \ P_{32}^+ \ P_{33}^+],$$

$$\mathbf{P}_4^+ = [P_{41}^+ \ P_{42}^+ \ P_{43}^+].$$

The image point is  $[u \ v \ 1]^T$ ,  $\beta$  is a variable parameter used to generate all points on the back projection, The camera's position denote as  $[T_{cx} \ T_{cy} \ T_{cz} \ 1]^T$ , and  $\mathbf{P}^+$  is the pseudo-invert of homogeneous camera matrix  $\mathbf{P}$ . Thus, we can compute the depth of trough as

$$Z_r = \frac{T_{cz} + \beta([\mathbf{P}_3^+][\mathbf{I}])}{1 + \beta([\mathbf{P}_4^+][\mathbf{I}])} = -he \frac{\left( \frac{T_{cx} + \beta([\mathbf{P}_1^+][\mathbf{I}])}{1 + \beta([\mathbf{P}_4^+][\mathbf{I}])} \cos \phi + \frac{T_{cy} + \beta([\mathbf{P}_2^+][\mathbf{I}])}{1 + \beta([\mathbf{P}_4^+][\mathbf{I}])} \sin \phi - d \right)^2}{-2\sigma^2}, \quad (8)$$

where  $\mathbf{I}$  is an image point in a contour of best trough selection. Only the  $\beta$  is unknown, can be compute using the Newton-Raphson iterative root finding method. The function of  $\beta$  be define as

$$f(\beta) = \frac{T_{cz} + \beta([\mathbf{P}_3^+][\mathbf{I}])}{1 + \beta([\mathbf{P}_4^+][\mathbf{I}])} + he \frac{\left( \frac{T_{cx} + \beta([\mathbf{P}_1^+][\mathbf{I}])}{1 + \beta([\mathbf{P}_4^+][\mathbf{I}])} \cos \phi + \frac{T_{cy} + \beta([\mathbf{P}_2^+][\mathbf{I}])}{1 + \beta([\mathbf{P}_4^+][\mathbf{I}])} \sin \phi - d \right)^2}{-2\sigma^2} = 0, \quad (9)$$

and the deferential function of  $\beta$  is equation below

$$f'(\beta) = \frac{m_3}{1+\beta m_4} - \frac{m_4}{1+\beta m_4} Z_r + \left( \frac{(d - Y_r \sin(\theta) - X_r \cos(\theta))^2}{-2\sigma^2} \right) \times \dots$$

$$\left( \frac{(d - Y_r \sin(\theta) - X_r \cos(\theta)) \left( \frac{m_1}{1+\beta m_4} \cos(\theta) + \frac{m_2}{1+\beta m_4} \sin(\theta) - \frac{m_4}{1+\beta m_4} \cos(\theta) X_r - \frac{m_4}{1+\beta m_4} \sin(\theta) Y_r \right)}{\sigma^2} \right),$$

(10)

where

$$X_r = \frac{(t_{cx} + \beta[\mathbf{P}_1^+][\mathbf{I}])}{1 + \beta[\mathbf{P}_4^+][\mathbf{I}]}$$

$$Y_r = \frac{(t_{cy} + \beta[\mathbf{P}_2^+][\mathbf{I}])}{1 + \beta[\mathbf{P}_4^+][\mathbf{I}]}$$

$$Z_r = \frac{(t_{cz} + \beta[\mathbf{P}_3^+][\mathbf{I}])}{1 + \beta[\mathbf{P}_4^+][\mathbf{I}]}$$

$$m_1 = [\mathbf{P}_1^+][\mathbf{I}]$$

$$m_2 = [\mathbf{P}_2^+][\mathbf{I}]$$

$$m_3 = [\mathbf{P}_3^+][\mathbf{I}]$$

$$m_4 = [\mathbf{P}_4^+][\mathbf{I}]$$

We initialize with  $z_r=0$ , giving  $\beta_0 = \frac{-T_{cz}}{[\mathbf{P}_3^+][\mathbf{I}]}$  and the iterative update rule

$$\beta_{n+1} = \beta_n - \left[ \frac{f(\beta)}{f'(\beta)} \right]. \quad (11)$$

We allow a maximum number of iterations equal to 40. After  $\beta$  is estimated for every pixel in the region of interest for the best selected trough, we can calculate the average log-likelihood. For a particular  $\phi$  and  $d$ , we now have  $Z_r$  for every pixel in the region. In this experiment, the likelihood needs to set low probability to black pixels in the trough region and high probability to black pixels in the region because black pixels are assumed to represent the ground plane and white pixels are assumed to represent trough points that have some depth. The model for black and white pixels is a sigmoid in the height of the trough. We allow for scaling and phase shift of the sigmoid:

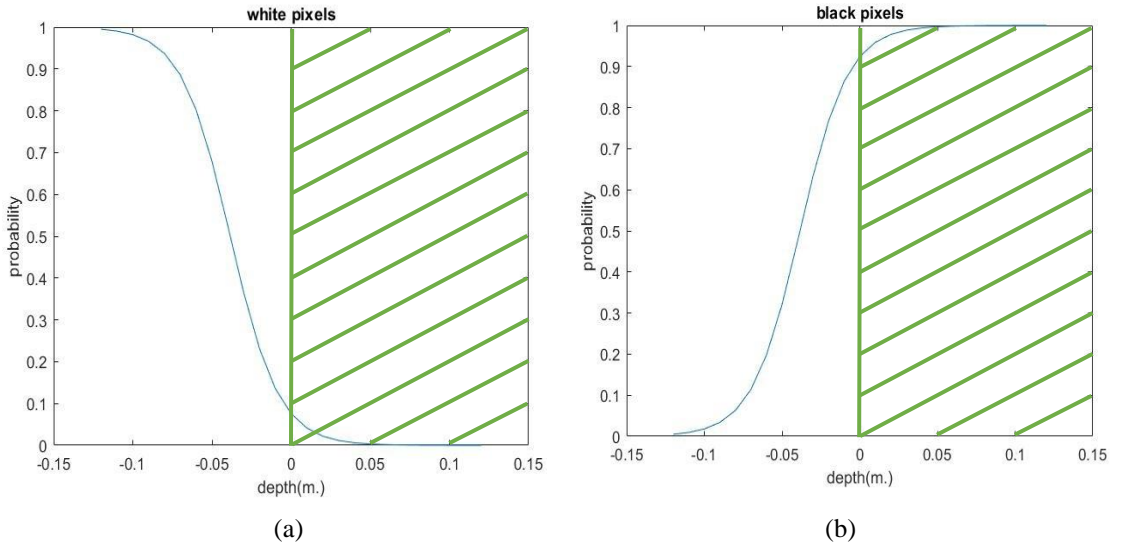


$$\begin{aligned}
p(B | u_b, v_b) &= \frac{1}{1 + e^{-(\alpha Z_r - \gamma)}} \\
p(W | u_b, v_b) &= 1 - \frac{1}{1 + e^{-(\alpha Z_r - \gamma)}}.
\end{aligned} \tag{12}$$

To miss the optimal condition, the optimization should move the trough which make most of white pixels are inside of it and mostly black pixels are outside of it. We use  $\alpha = 65$ , and  $\lambda = -2.5$ . Sample graphs of  $p(B | u_b, v_b)$  and  $p(W | u_b, v_b)$  as functions of trough depth. It shows in Figure 3.11.

**Figure 3.11**

*Relationship between Trough's Depth and Assumed Probability of Obtaining White and Black Pixels (a) Probability of White Pixels (b) Probability of Black Pixels.*



$$\ell(\phi, d; (u_b, v_b)) = \sum_{u_b, v_b \in B} \log(p(B | u_b, v_b)) + \sum_{u_b, v_b \in W} \log(p(W | u_b, v_b)) \tag{13}$$

We optimize  $\ell(\theta)$  using gradient ascent because of its flexibility. We begin with a guidance row distance and orientation estimate from one of three approaches (PCA, RANSAC, Hough transform), use Newton-Raphson to calculate the depth  $Z_r$  for each pixel, and then use finite differences to estimate

$$\nabla \ell_i = \begin{bmatrix} \frac{\partial \ell}{\partial \phi} \\ \frac{\partial \ell}{\partial d} \end{bmatrix}. \quad (14)$$

We update the guidance row distance and orientation using

$$\begin{bmatrix} \phi_{i+1} \\ d_{i+1} \end{bmatrix} = \begin{bmatrix} \phi_i \\ d_i \end{bmatrix} + \eta \nabla \ell_i. \quad (15)$$

Where  $\eta$  is a learning rate matrix. We use  $\begin{bmatrix} 0.05 & 0 \\ 0 & 0.1 \end{bmatrix}$ , and  $i$  is the iteration. We limit orientation updates to no more than 0.05 radians ( $|\phi_{i-1} - \phi_i| \leq 0.05$ ) and distance updates less than 0.1 ( $|d_{i-1} - d_i| \leq 0.1$ ).

We repeat until the log-likelihood is stable or until 25 iterations. The guidance row distance and orientation that maximize the log-likelihood are the optimal parameters. Pseudocode for Maximum likelihood guidance distance and orientation estimation as we use it is provided in Table 3.6

**Table 3.6**

*Pseudocode of Maximum Likelihood Guidance Distance and Orientation Estimation*

---

Input: black and white bird's eye view image  $\mathbf{I}$  of best trough, initial guess of distance  $d$ , initial guess of row orientation  $\phi$ . Initial guess may be from one of three methods (PCA, RANSAC, HOUGH).

Output: parameters of optimized trough distance and orientation estimates  $d^*, \phi^*$

---

$\mathbf{X} \leftarrow$  2xN vector of  $(x, y)$  coordinates of pixels in trough

connected component  $\mathbf{C}$ , expressed as  $\begin{bmatrix} x_1 & \dots & x_N \\ y_1 & \dots & y_N \end{bmatrix}$

$d, \phi \leftarrow$  initial parameter estimates

$B$ : set of black pixel positions in the trough region

$W$ : set of white pixel positions in the trough region

$d^* \leftarrow d$ ;  $\phi^* \leftarrow \phi$

$i \leftarrow 0$

while  $i < 25$

For each  $x_r \in \mathbf{X}$

Compute trough depth  $Z_r$  for every point in  $\mathbf{C}$  using

$d, \phi$ .

---

---

```

compute  $\begin{bmatrix} \phi_{i+1} \\ d_{i+1} \end{bmatrix} = \begin{bmatrix} \phi_i \\ d_i \end{bmatrix} + \eta \nabla \hat{\ell}_i$ .
If  $\hat{\ell}_i(\phi, d)$  is best so far
 $\phi^* \leftarrow \phi$  ;  $d^* \leftarrow d$ 
i  $\leftarrow$  i+1
return  $\phi^*, d^*$ 

```

---

### 3.3.2 Path Planning

Waypoint is designed to world coordinate in X,Y axis (East, North); the number of waypoints depend on the area of different fields. The seeding robot will start from the first row which starting point is at the center of the tractor and then go straight line until finish the design of the first row's distance. After that, make a U-turn to the second row which is parallel with the previous row as shown in Figure 3.12. The axis of path planning define with  $X', Y'$ . For straight line waypoints design as

$$\begin{bmatrix} X \\ Y \end{bmatrix}_{n,m} = \begin{bmatrix} X' \cos(\theta) + Y' \sin(\theta) \\ -X' \sin(\theta) + Y' \cos(\theta) \end{bmatrix}, \quad (16)$$

where  $\theta$  is the bearing of the tractor orientation concerning the North direction of the world coordinate,  $m$  is the number of points in the row, and  $n$  is the number of rows.

The waypoint at the curve line defines as

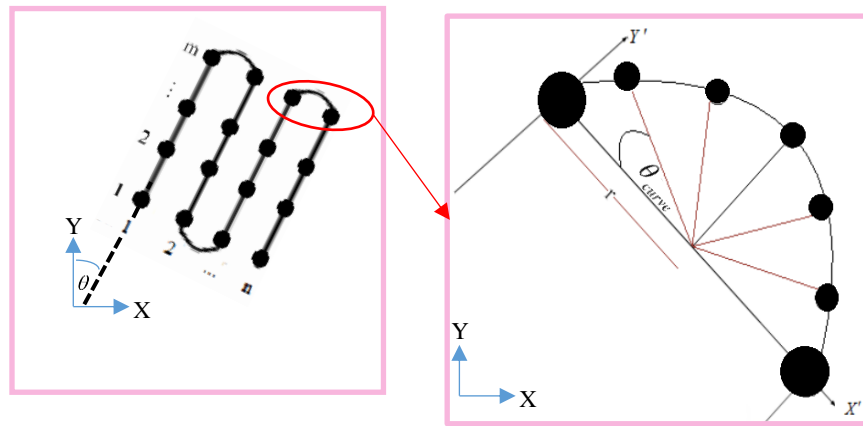
$$\begin{bmatrix} X'_{curve} \\ Y'_{curve} \end{bmatrix} = \begin{cases} \begin{bmatrix} r - r \cos(\theta_{curve}) \\ r \sin(\theta_{curve}) \end{bmatrix} & \text{if turn right} \\ \begin{bmatrix} r - r \cos(-\theta_{curve}) \\ r \sin(-\theta_{curve}) \end{bmatrix} & \text{if turn left} \end{cases}, \quad (17)$$

thus,

$$\begin{bmatrix} X \\ Y \end{bmatrix}_{n,m} = \begin{bmatrix} X'_{curve} \cos(\theta) + Y'_{curve} \sin(\theta) \\ -X'_{curve} \sin(\theta) + Y'_{curve} \cos(\theta) \end{bmatrix}, \quad (18)$$

**Figure 3.12**

*Path Planning*

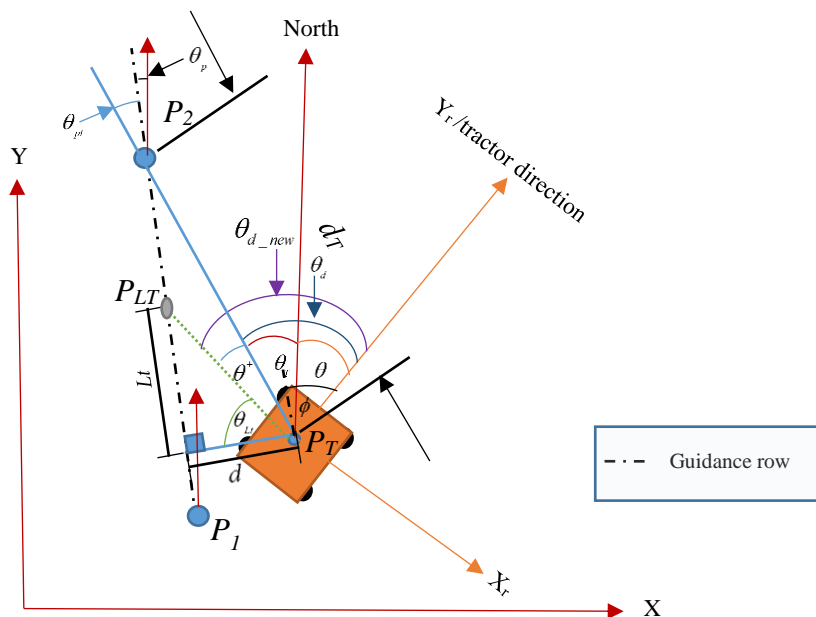


### 3.3.3 Path Tracking Control System

In this work, we are tracking the path by controlling two subsystems: the velocity and heading of tractor systems. The tractor's orientation and guidance row distance are fed back to the control loops. Figure 3.13 describes the definition of the desired orientation and row distance in planar.

**Figure 3.13**

*Parameters in the World Coordinate and the Tractor Coordinate*



In this part, Pure Pursuit and look-ahead distance are used to compute the orientation desired which is called desired look-ahead orientation ( $\theta_{d\_new}$ ). The purpose of look-ahead distance ( $L_t$ ) is to help the tractor get closer to the desired path as soon as possible. The look-ahead distance  $L_t$  is written as

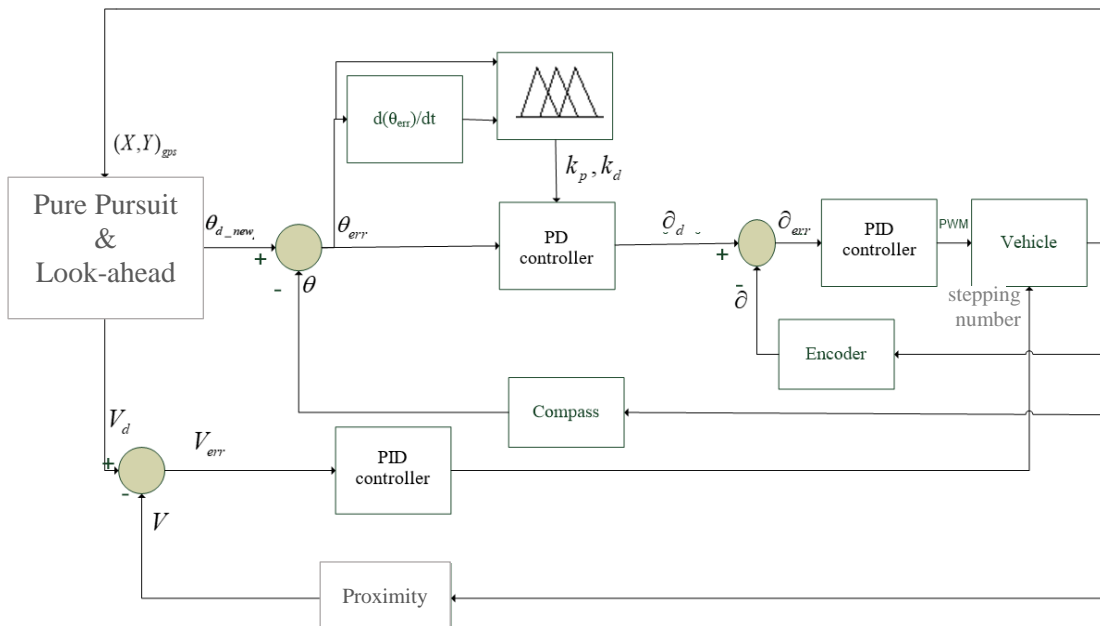
$$L_t = k\sqrt{d_T^2 - d^2}, \quad (19)$$

where,  $k$  is look-ahead gain,  $d_T$  is distance from tractor to forward way point and  $d$  is guidance row distance.  $\theta_{d\_new}$  is computed by wrapping between angular deviation of look-ahead direction with respect to the forward waypoint direction ( $\theta^+$ ) and the desired yaw orientation ( $\theta_d$ ) as

$$\theta_{d\_new} = \text{wrap}(\theta_d, \theta^+). \quad (20)$$

**Figure 3.14**

*Block Diagram of the Control System for Path Tracking*



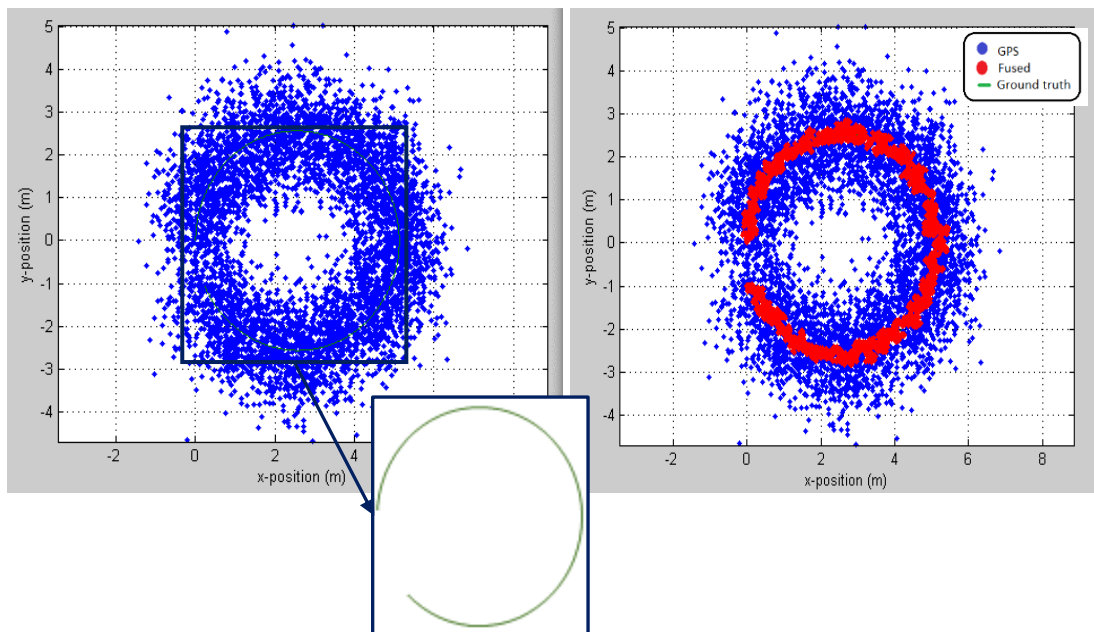
The block diagram of the control system for the path tracking is shown in Figure 3.14. There are two reference variables: the desired velocity of tractor  $V_d$  and desired look-ahead orientation, which are obtained from the Pure Pursuit and look-ahead. Thus, two control loops: velocity and orientation loop control are tuned and working independently. In the velocity loop, the PID controller computes the number of stepping commands based on the velocity error. For the orientation loop control, the outer loop-

based PD controller is used to calculate the desired steering wheel angle ( $\delta$ ) and the PID controller is implemented for the steering angle control. For the PD controller, the controller gains are adapted based on the amount of orientation error and rate of orientation error. These variables are used to find the suitable gain based on fuzzy logic.



**Figure 4.2**

*Result of Generate Circle Path and GPS Position*



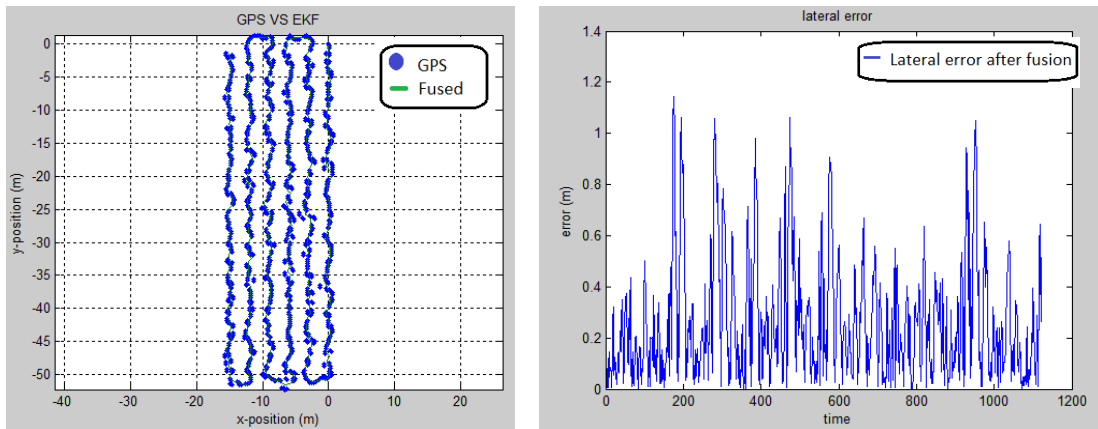
#### **4.1.2 Tracking Planning Path**

After the first simulation, we knew that EKF helps to reduce unwanted data. The purpose of the second simulation is to test the path tracking control system. In this simulation, the tracking control system uses Pure Pursuit and look-ahead to compute the desired orientation. At the outer loop, the PD controller is implemented to calculate the desired steering wheel angle, and the PID controller is used for the steering angle control. The velocity is still set as a constant speed at 0.3 m/s. The distance between each waypoint is set at 2.50 meters. The result is shown in Figure 4.3, The outputs are compared between GPS in blue dots and the fusion in the green line. As a result of adding EKF, the lateral errors are smaller than GPS. The maximum lateral error after fusion is 1.24 meters.



**Figure 4.3**

*Result on Simulink Model of Tracking GPS with EKF*



## **4.2 Experiment Result Tracking Based on GPS without EKF in Paddy Field**

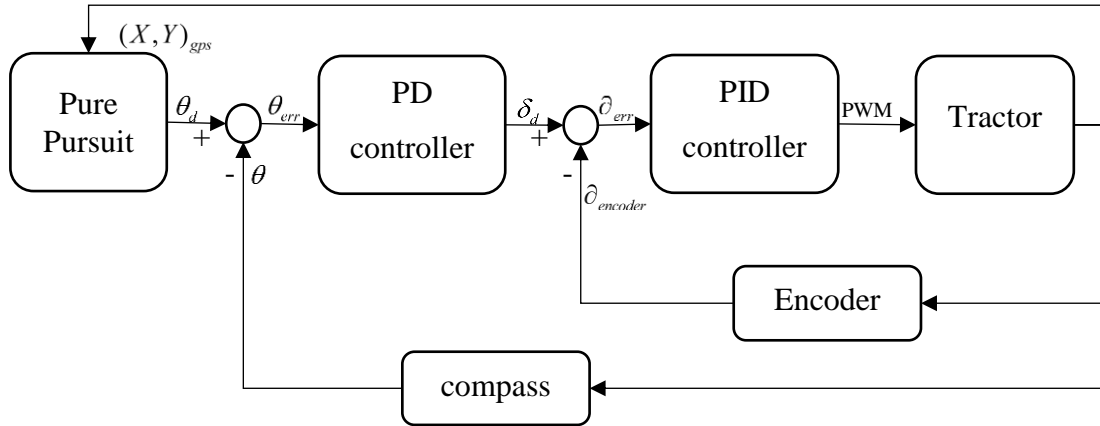
The experiments in the paddy field were conducted at the Asian Institute of Technology field. We tested in 2 main parts. The first is tracking based on GPS localization without EKF, and the second is localization based on a machine vision system.

### **4.2.1 Control Based on Pure Pursuit for Estimating Orientation**

In this experiment, we set the velocity of the tractor to be constant at 0.3 m/s. Pure Pursuit method estimate the orientation. The orientation loop control consists of the outer loop-based PD controller and the PID controller for controlling the steering angle. The diagram of tracking control using orientation is shown in Figure 4.4. The result of tracking straight line path is shown in Figure 4.5. The maximum error is about 0.52 m when straight line tracking.

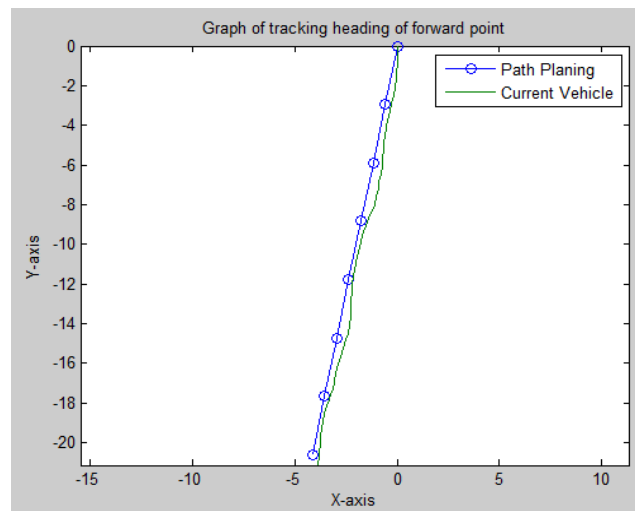
**Figure 4.4**

*Tracking Using GPS and Steering Controller Based Pure Pursuit for Estimating Orientation*



**Figure 4.5**

*The Straight Line Result of Tracking Based Pure Pursuit for Estimating Orientation*



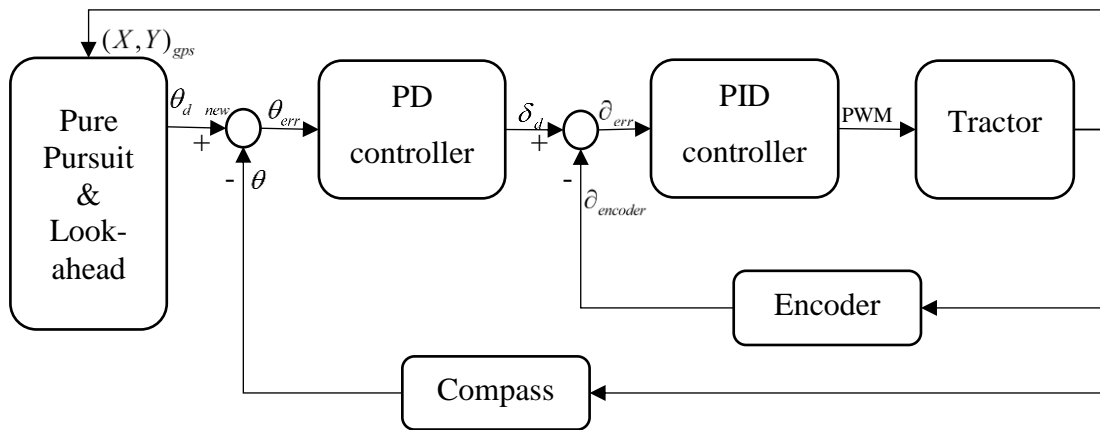
#### **4.2.2 Control Based on Pure Pursuit and Look-ahead for Estimating Orientation**

**4.2.2.1 PD Controller.** In this experiment, we set the velocity of the tractor to be constant speed at 0.3 m/s same as the previous method. The difference between this experiment and the previous experiment is the only orientation computed from both Pure Pursuit and the look-ahead method. The method diagram is shown in Figure 4.6. The result of controlling and tracking straight line path based Pure Pursuit and look-ahead to estimate orientation is shown in Figure 4.7. The maximum lateral error of

tracking is 0.18 m. The mean error of this experiment is 0.06 meter. If we compared this controller and the previous controller, the result shows that this controller has better tracking than the previous controller does.

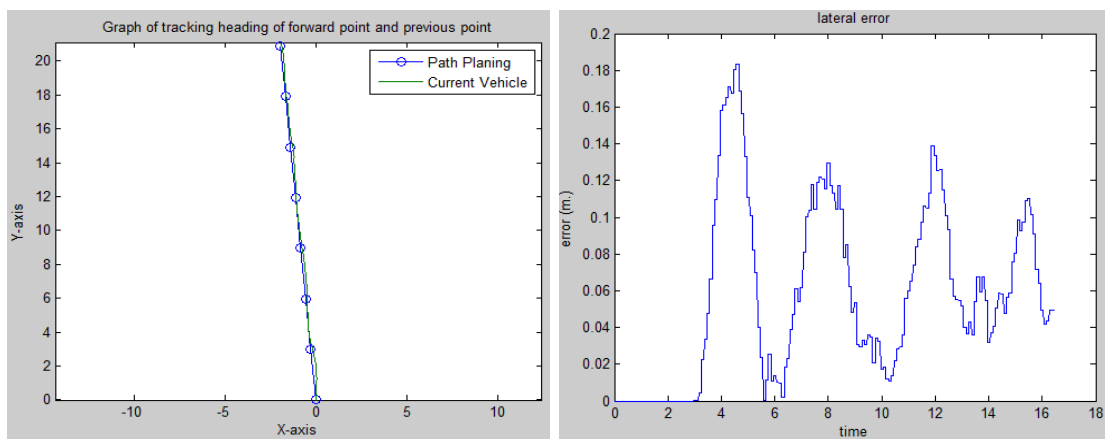
**Figure 4.6**

*Tracking using GPS and Steering Controller Based Pure Pursuit and Look-ahead for Estimating Orientation*



**Figure 4.7**

*Result of Tracking Using GPS and Steering Controller Based Pure Pursuit and Look-ahead for Estimating Orientation*

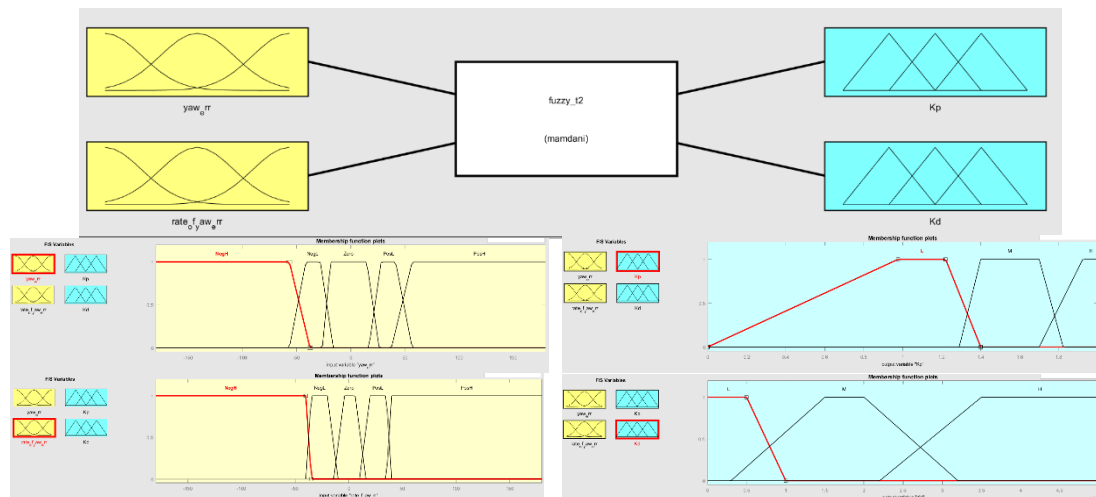


**4.2.2.2 Fuzzy Self-adjustment for the PD Controller.** The control system for the path tracking of this experiment is shown in Figure 3.14. There are two reference variables: the desired velocity of tractor and desired look-ahead orientation. For control

orientation loop control, it is the same method as using PD control based Pure Pursuit and look-ahead except for the gain parameters of the PD are coming from fuzzy design. The fuzzy rule of this experiment is shown in Figure 4.8. The PID controller is implemented as an inner loop for the steering angle control. For the velocity loop, the PID controller estimates the number of stepping commands to control steering. The result of U-turn path tracking in the paddy field is shown in Figure 4.9. The maximum lateral error from the desired path is 0.68 m.

**Figure 4.8**

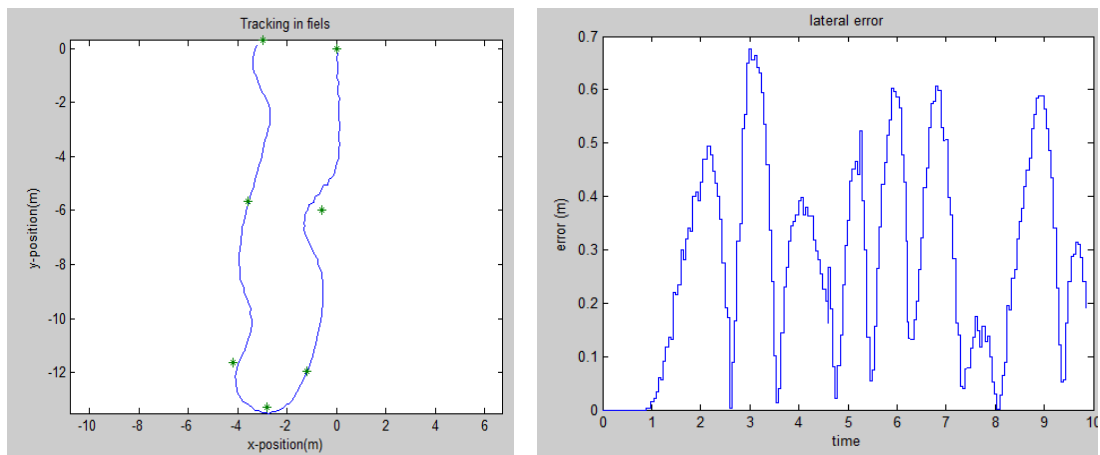
*Fuzzy Rule*



Inputs			Outputs			
$\theta_{err}$	operation	$\frac{\partial \theta_{err}}{\partial t}$			$K_p$	$K_d$
NH	and	NH	then		H	H
NH	and	NL	then		H	M
NH	and	Zero	then		H	M
NH	and	PL	then		H	M
NH	and	PH	then		H	H
NL	and	NH	then		M	H
NL	and	NL	then		M	M
NL	and	Zero	then		M	L
NL	and	PL	then		M	M
NL	and	PH	then		M	H
Zero	and	NH	then		L	M
Zero	and	NL	then		L	M
Zero	and	Zero	then		L	L
Zero	and	PL	then		L	M
Zero	and	PH	then		L	M
PL	and	NH	then		M	H
PL	and	NL	then		M	M
PL	and	Zero	then		M	L
PL	and	PL	then		M	M
PL	and	PH	then		M	H
PH	and	NH	then		H	H
PH	and	NL	then		H	M
PH	and	Zero	then		H	M
PH	and	PL	then		H	M
PH	and	PH	then		H	H

**Figure 4.9**

*Result of U-turn Path Tracking Based Fuzzy Self-adjustment for PD*



### 4.3 Localization Base on Machine Vision System

We have a test based on a machine vision system for five experiments and all the experiments were conducted at the Asian Institute of Technology over the period 2018-2020. We seeded wet fields at different times of the day. Thus, in the five experiments, we cannot control moisture, illumination, and shadow.

A summary of the results of trough detection over five experiments is shown in Table 4.1. Over the five experiments, the first three experiments have lower detection rates, with F1-score accuracy less than 0.8. In the experiments 2019/2 and 2020/1, we observe more accurate detection. These differences are come from water level and illuminate. Different times of day cause changes in the illumination, moisture of the soil, and shadow patterns. The effect of our detection method is its sensitivity to such imaging conditions. We plan to improve trough detection rates in future work using work deep learning.

Next, we consider the accuracy of guidance row localization in comparison to ground truth. Only in the experiments 2019/2, and 2020/1, ground truth markers were placed in the center of the trough, as shown in Figure 4.10 to measure the accuracy of the trough estimated by our methods. For all images, we manually drew a straight line through the markers and transformed the resulting line into the tractor coordinate system.

**Table 4.1**

*Trough Detection Results*

<b>Experiment</b>	<b>Number</b>	<b>Hits</b>	<b>False positives</b>	<b>Precision</b>
2018/1	30	21	9	0.70
2018/2	30	24	6	0.80
2019/1	39	27	12	0.69
2019/2	35	32	3	0.91
2020/1	13	13	0	1.00

**Figure 4.10**

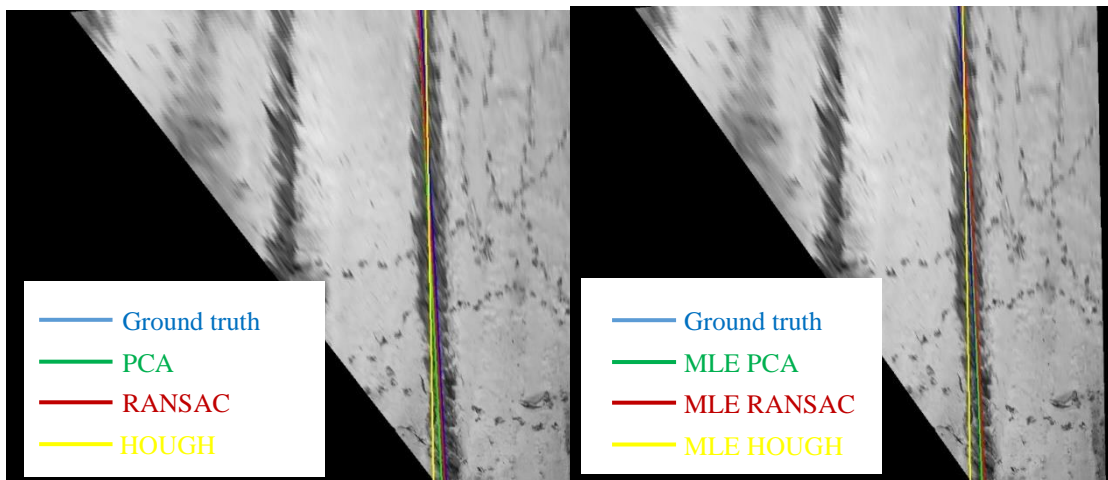
*The Ground Truth of Guidance Row Distance and Orientation Using Markers*



Example estimates are shown in Figure 4.11. The result is a straight line in the tractor ground plane, from which we can obtain guidance row distance and orientation. PCA result, RANSAC result, and HOUGH result are shown in Figure 4.11 (a). Figure 4.11 (b) shows optimized guidance row estimates using MLE based on the estimates of initial guesses. To measure the accuracy of the guidance row estimation method, we used the correctly detected guidance rows (hits) in the last two experiments. The guidance row error is the absolute mean error in perpendicular distance between the estimated lines and the ground truth. A detail of the results of guidance row orientation and distance estimation shows in Table 4.2. The results of guidance row estimation are shown in Figure 4.12.

**Figure 4.11**

*Example Estimates of Guidance Row Orientation and Lateral Distance*



(a) Estimates by PCA, RANSAC, and HOUGH (b) MLE Estimates Using Each Result from (a) for Initialization

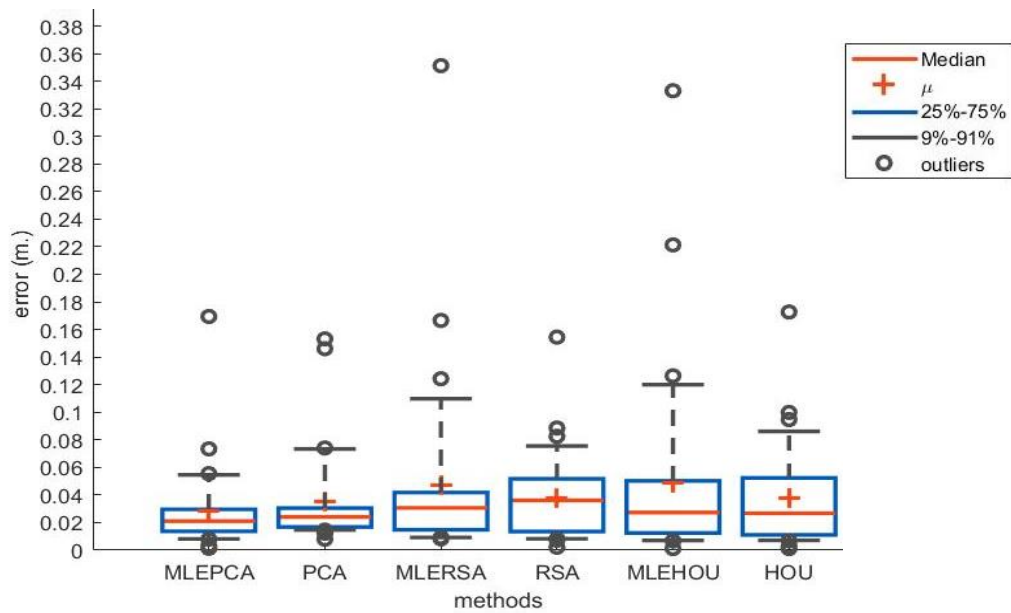
**Table 4.2**

*Summary of Guidance Row Estimation Results in Experiments 2019/2 and 2020/1*

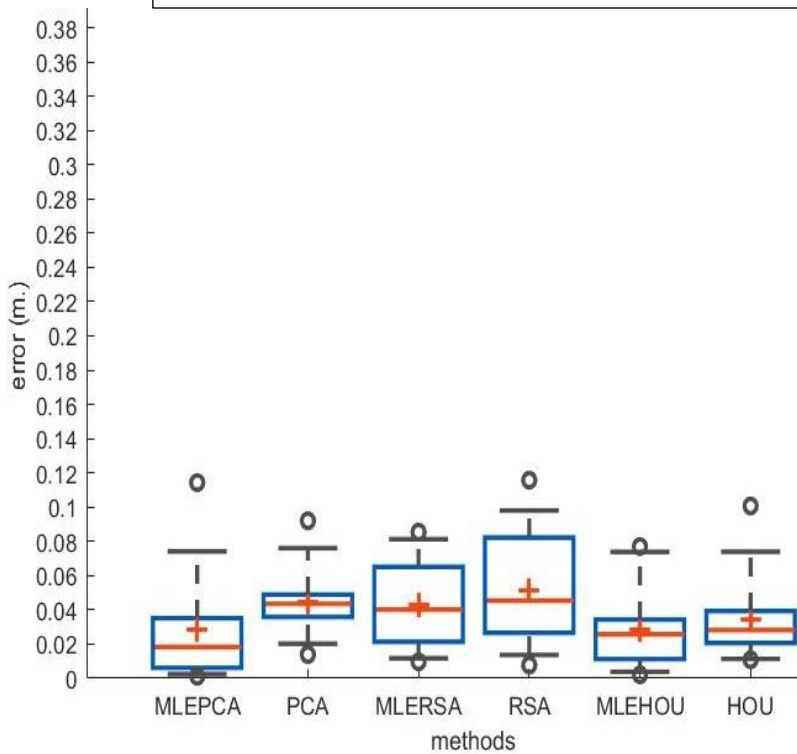
Approach	Experiment	Guidance row error (mm.)					
		Mean	SD	Variance	Median	Quartile 3	Maximum
<b>PCA</b>	2019/2	35.0	34.2	1.7	24.1	30.5	153.2
	2020/1	44.6	19.5	0.4	43.5	48.9	92.1
<b>MLE PCA</b>	2019/2	28.4	30.2	0.9	21.1	29.6	169.4
	2020/1	28.1	30.7	0.9	18.3	35.1	114.3
<b>RANSAC</b>	2019/2	37.8	31.3	1.0	36.1	51.8	154.4
	2020/1	51.0	33.1	1.1	45.4	82.2	115.7
<b>MLE RANSAC</b>	2019/2	46.8	65.2	4.3	30.7	41.8	351.3
	2020/1	42.8	25.7	0.7	40.2	65.1	85.4
<b>HOUGH</b>	2019/2	37.6	36.5	1.3	26.7	52.4	172.7
	2020/1	34.4	24.1	0.6	28.2	39.3	100.7
<b>MLE HOUGH</b>	2019/2	48.8	68.0	4.6	27.3	51.1	333.1
	2020/1	28.7	23.1	0.5	25.8	39.3	77.0

**Figure 4.12**

*Guidance Row Estimation Results of Experiments 2019/2 and 2020/1*



Lateral Distance Error in Experiment 2019/2



Lateral Distance Error in Experiment 2020/1



A summary of the result shows that MLE PCA has mean error levels for the optimized method are 28.4, and 28.1 mm which are optimized in both two-year experiments. If we focus only on quartile 3, all MLE methods show improvements in every three methods of both years. Some results show that mean optimized errors are higher than the mean unoptimized because the reason for the estimates is that the unoptimized estimates are not close enough to the ground truth to attract the optimizer; this reason makes outliers in the results of the optimized estimate.

When we combine the results of Experiments 2019/2 and 2020/1, we find that optimization makes RANSAC and HOUGH estimate slightly worse overall but improves in PCA. The differences using repeated measures two-tailed t-test, which  $H_0 : \mu = 0$ , were only significant for PCA. The summary of significance shows in Table 4.3.

**Table 4.3**

*Difference between Each Line Fitting Method and the Optimized Line Fitting Method, Over Experiments 2019/2 and 2020/1*

<b>method</b>	<b>MLE PCA- PCA</b>	<b>MLE RANSAC- RANSAC</b>	<b>MLE HOUGH- HOUGH</b>
<b>Average difference</b>	-0.00950704	0.00407204	0.00630311
<b>df</b>	44	44	44
<b><i>T stat</i></b>	-2.354218	0.566035	0.962463
<b><i>P-value</i></b>	0.023090	0.574245	0.341076
<b><i>sig</i></b>	Yes	No	No

In conclusion, PCA is the best performance if we consider the results with its compute time. Among the optimized methods, MLE PCA gives the best optimized lateral distance and orientation estimates than using the PCA method alone. The cons of optimization are time consuming. The optimized methods used an average of 46 seconds per frame while unoptimized methods needed the time in less than 1 second as shown in Table 4.4. Although the speed could be further improved by code-level

optimization of our implementation, the compute resource gap between PCA and MLE PCA is still wide.

**Table 4.4**

*Time consumption of Each Method over Experiments 2019/2 and 2020/1*

<b>Activity</b>	<b>Minimum (s/frame)</b>	<b>Maximum (s/frame)</b>	<b>Average (s/frame)</b>
<b>Guidance row segmentation and best guidance row selection</b>	0.10	0.22	0.14
<b>PCA</b>	0.00008	0.00053	0.00022
<b>MLE PCA</b>	0.10	92.20	46.24
<b>RANSAC</b>	0.19	1.35	0.68
<b>MLE RANSAC</b>	9.50	84.89	46.22
<b>HOUGH</b>	0.03	0.20	0.09
<b>MLE HOUGH</b>	15.14	84.17	46.17

## **CHAPTER 5**

### **CONCLUSION AND FUTURE WORK**

In this chapter, the conclusion and future work of this work are described below.

#### **5.1 Conclusion**

Currently, the tracking controller of the tractor uses GPS and compass as guidance to control steering. At the outer loop, we fed the orientation of the tractor concerning the North direction and compute the error between look-ahead orientation and tractor orientation. The PD based fuzzy logic is used for computing desired steering. In the inner loop, PID is used to estimate the PWM command to control the DC motor attached to the tractor's steering. Another independent control loop is the velocity control loop based PID controller. Both independent desired velocity and desired orientation are estimated from Pure Pursuit and the look-ahead method. The result shows the error is less than 1 m.

In this research, the methods for localization of a guidance row in a paddy field have been described. We explore three different methodologies and provide a method for further optimizing initial guesses using MLE. The MLE method provides more accurate results than the methods not using it. The best approach is MLE PCA. In two experiments, PCA has an average mean error of 39.8 mm, while MLE PCA has an average mean error of 28.25 mm, an improvement of 29%. The optimization method thus enables centimeter-level accuracy for guidance row mapping.

#### **5.2 Future Work**

For future work, we will test the method in real-time on the tractor while it is running. These improvements are needed. First, we will use an inertial measurement unit (IMU) to measure roll, pitch, and yaw, along with compensation for the camera motion measured by the IMU rather than using a gimbal to stabilize the camera. Second, a real-time operating system (RTOS) should be used to ensure the system does not miss real-time events. Another area for improvement is the trough detection rates. This process can be improved using deep learning for the detection of the trough pixels. Finally, with real-time responses and a high trough detection rate, autonomous seed planting will be

possible through the integration of steering and velocity control. Farmers will benefit from the low cost and high accuracy of the system.

## REFERECE

- Astrand, B., & Baerveldt, A. J. (2005). A vision based row-following system for agricultural field machinery. *Mechatronics*, 15(2), 251–269. <https://doi.org/10.1016/j.mechatronics.2004.05.005>
- Backman, J., Oksanen, T., & Visala, A. (2012). Navigation system for agricultural machines: Nonlinear model predictive path tracking. *Computers and Electronics in Agriculture*, 82, 32-43. <https://doi.org/10.1016/j.compag.2011.12.009>
- Bakker T, van Asselt K, Bontsema J, Müller J, van Straten G. (2011). Autonomous navigation using a robot platform in a sugar beet field. *Biosystems Engineering*, 109(4), 357–368.
- Carballido, J., Perez-Ruiz, M., Gliever, C., & Agüera, J. (2012). Design, development and lab evaluation of a weed control sprayer to be used in robotic systems. In *Proceedings of the First International Conference on Robotics and Associated High-technologies and Equipment for Agriculture. Applications of automated systems and robotics for crop protection in sustainable precision agriculture,(RHEA-2012) Pisa, Italy-September 19-21, 2012* (pp. 23-29). University of Pisa.
- Chaivivatrakul, S., & Dailey, M. N. (2014). Texture-based fruit detection. *Precision Agriculture*, 15(6), 662–683. <https://doi.org/10.1007/s11119-014-9361-x>
- 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(1-2), 31-38. <https://doi.org/10.1016/j.compag.2007.04.008>
- Han, X. Z., Kim, H. J., Kim, J. Y., Yi, S. Y., Moon, H. C., Kim, J. H., & Kim, Y. J. (2015). Path-tracking simulation and field tests for an auto-guidance tillage tractor for a paddy field. *Computers and Electronics in Agriculture*, 112, 161-171. <https://doi.org/10.1016/j.compag.2014.12.025>
- Jiang, G., Wang, Z., & Liu, H. (2015). Automatic detection of crop rows based on multi-ROIs. *Expert Systems with Applications*, 42(5), 2429–2441. <https://doi.org/10.1016/j.eswa.2014.10.033>
- Kanagasingham, S., Ekpanyapong, M., & Chaihan, R. (2020). Integrating machine vision-based row guidance with GPS and compass-based routing to achieve autonomous navigation for a rice field weeding robot. *Precision Agriculture*, 21(4), 831–855. <https://doi.org/10.1007/s11119-019-09697-z>
- Kayacan, E., Kayacan, E., Ramon, H., & Saeys, W. (2015). Towards agrobots: Identification of the yaw dynamics and trajectory tracking of an autonomous tractor. *Computers and Electronics in Agriculture*, 115, 78-87. <https://doi.org/10.1016/j.compag.2015.05.012>

- Kraus, T., Ferreau, H. J., Kayacan, E., Ramon, H., De Baerdemaeker, J., Diehl, M., & Saeys, W. (2013). Moving horizon estimation and nonlinear model predictive control for autonomous agricultural vehicles. *Computers and electronics in agriculture*, 98, 25-33. <https://doi.org/10.1016/j.compag.2013.06.009>
- Leemans, V., & Destain, M.-F. (2006). Application of the Hough Transform for Seed Row Localisation using Machine Vision. *Biosystems Engineering*, 94(3), 325–336. <https://doi.org/10.1016/j.biosystemseng.2006.03.014>
- Leemans, V., & Destain, M.-F. (2007). A computer-vision based precision seed drill guidance assistance. *Computers and Electronics in Agriculture*, 59(1–2), 1–12. <https://doi.org/10.1016/j.compag.2007.04.003>
- Li, H., Lee, W. S., & Wang, K. (2016). Immature green citrus fruit detection and counting based on fast normalized cross correlation (FNCC) using natural outdoor colour images. *Precision Agriculture*, 17(6), 678–697. <https://doi.org/10.1007/s11119-016-9443-z>
- Morimoto, E., Suguri, M., & Umeda, M. (2005). Vision-based navigation system for autonomous transportation vehicle. *Precision Agriculture*, 6(3), 239–254. <https://doi.org/10.1007/s11119-005-1384-x>
- Nagasaka, Y., Umeda, N., Kanetai, Y., Taniwaki, K., & Sasaki, Y. (2004). Autonomous guidance for rice transplanting using global positioning and gyroscopes. *Computers and electronics in agriculture*, 43(3), 223-234. <https://doi.org/10.1016/j.compag.2004.01.005>
- Nieuwenhuizen, A. T., Hofstee, J. W., & Van Henten, E. J. (2010). Performance evaluation of an automated detection and control system for volunteer potatoes in sugar beet fields. *Biosystems Engineering*, 107(1), 46-53. <https://doi.org/10.1016/j.biosystemseng.2010.06.011>
- Noguchi, N., & Barawid Jr, O. C. (2011). Robot farming system using multiple robot tractors in Japan agriculture. *IFAC Proceedings Volumes*, 44(1), 633-637.
- Noguchi, N., & Barawid Jr, O. C. (2011). Robot farming system using multiple robot tractors in Japan agriculture. *IFAC Proceedings Volumes*, 44(1), 633-637. <https://doi.org/10.3182/20110828-6-IT-1002.03838>
- Perez-Ruiz, M., Slaughter, D. C., Gliever, C. J., & Upadhyaya, S. K. (2012). Automatic GPS-based intra-row weed knife control system for transplanted row crops. *Computers and Electronics in Agriculture*, 80, 41-49. <https://doi.org/10.1016/j.compag.2011.10.006>
- Peruzzi, A., Frascioni, C., Martelloni, L., Fontanelli, M., & Raffaelli, M. (2012). Application of precision flaming to maize and garlic in the RHEA project. In *Proceedings of the First International Conference on Robotics and Associated High-technologies and Equipment for Agriculture. Applications of automated systems and robotics for crop protection in sustainable precision agriculture, (RHEA-2012) Pisa, Italy-September 19-21, 2012* (pp. 55-60). University of Pisa.

- Qureshi, W. S., Payne, A., Walsh, K. B., Linker, R., Cohen, O., & Dailey, M. N. (2017). Machine vision for counting fruit on mango tree canopies. *Precision Agriculture*, 18(2), 224–244. <https://doi.org/10.1007/s11119-016-9458-5>
- Reiser, D., Sehsah, E. S., Bumann, O., Morhard, J., & Griepentrog, H. W. (2019). Development of an autonomous electric robot implement for intra-row weeding in vineyards. *Agriculture*, 9(1), 18. <https://doi.org/10.3390/agriculture9010018>
- Stentz A, Dima C, Wellington C, Herman H, Stager D. A. (2002). System for semi-autonomous tractor operations. *Autonomous Robots*, 13(1), 87–104
- Vieri, M., Lisci, R., Rimediotti, M., & Sarri, D. (2012). The innovative RHEA airblast sprayer for tree crop treatment. In *Proceedings of the First International Conference on Robotics and Associated High-technologies and Equipment for Agriculture. Applications of automated systems and robotics for crop protection in sustainable precision agriculture,(RHEA-2012) Pisa, Italy-September 19-21, 2012* (pp. 93-98). University of Pisa.
- Xue, J., Zhang, L., & Grift, T. E. (2012). Variable field-of-view machine vision based row guidance of an agricultural robot. *Computers and Electronics in Agriculture*, 84, 85–91. <https://doi.org/10.1016/j.compag.2012.02.009>