



**HAL**  
open science

# Design, Development and Evaluation of a System for the Detection of Aerial Parts and Measurement of Growth Indices of Bell Pepper Plant Based on Stereo and Multispectral Imaging

Vahid Mohammadi

► **To cite this version:**

Vahid Mohammadi. Design, Development and Evaluation of a System for the Detection of Aerial Parts and Measurement of Growth Indices of Bell Pepper Plant Based on Stereo and Multispectral Imaging. Signal and Image Processing. Université Bourgogne Franche-Comté; Tarbiat Modares University (Téhéran, Iran), 2022. English. NNT : 2022UBFCK109 . tel-04087258

**HAL Id: tel-04087258**

**<https://theses.hal.science/tel-04087258>**

Submitted on 3 May 2023

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

**THÈSE DE DOCTORAT EN COTUTELLE**  
**DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ**  
**PRÉPARÉE À L'UNIVERSITÉ DE BOURGOGNE**  
**ET DE L'ÉTABLISSEMENT UNIVERSITÉ DE TARBIAT MODARES**

Ecole doctorale n°37  
Sciences Pour l'Ingénieur et Microtechniques  
(SPIM)

Doctorat de **Instrumentation et Informatique de l'Image**

Thèse présentée par  
**Vahid MOHAMMADI**

Titre

**Design, Development and Evaluation of a System for the Detection of Aerial  
Parts and Measurement of Growth Indices of Bell Pepper Plant Based on Stereo  
Imaging and Spectral Analysis**

Thèse présentée et soutenue à Tehran, le 21 / 09 / 2022

**Composition du Jury**

Mohammad Hadi Khoshtaghaza	Professor at Tarbiat Modares University	Président
Jon Yngve Hardeberg	Professor at NTNU	Rapporteur
Soleiman Hosseinpour	Associate professor at Tehran university	Rapporteur
Olivier Laligant	Professor at the University of Burgundy	Examineur
Ahmad Banakar	Associate professor at Tarbiat Modares University	Examineur
Pierre Gouton	Professor at University of Burgundy	Directeur de thèse
Saeid Minaei	Professor at Tarbiat Modares University	Codirecteur de thèse
Alireza Mahdavian	Assistant Professor at Tarbiat Modares University	Conseiller



*To all lovers of machine vision!*

# ACKNOWLEDGEMENTS

First, I would like to wholeheartedly appreciate my supervisors, Prof. Saeid Minaei and Prof. Pierre Gouton for all their support, guidance, advice, and efforts. This was very valuable to me that they did for me whatever was possible. Each day, each conversation I learned a lesson from them. No word can appreciate them enough. Second, I thank my advisors, Dr. Ali Reza Mahdavian and Prof. Mohammad Hadi Khoshtaghaza for their advice and encouragement.

Furthermore, I am very much grateful to the reviewers, Prof. Jon Yngve Hardeberg and Dr. Soleiman Hosseinpour, for going through the entire dissertation and giving me critical comments and suggestions, which made the dissertation more complete and clear. I am especially grateful to the examiners, Prof. Olivier Laligant, Dr. Ahmad Banakar, Prof. Gholamhassan Najafi, and Dr. Keivan Ansari for assisting to the defense and providing conducive comments.

I also need to appreciate the help of technicians with the development of the systems and instrumentation. This includes Mr. Akbar Kulivand, Mr. Mathieu Rossé and Mr. Davoudi-Nasab.

Next, I thank—from the bottom of my heart—my father Ali, my mother Mahin, my brothers Mohammad Hossein and Hassan and my dear fiancé Houda, for their unlimited kindness and support. God has truly blessed me to have them.

This dissertation was started in Tarbiat Modares University, Iran and then became a joint (cotutelle) program with collaboration of Université de Bourgogne, France. Here, I cordially appreciate the Cultural section of Embassy of France in Iran and the Iranian Ministry of Science, Research and Technology for supporting this dissertation and collaboration between two universities.

# ABSTRACT

Monitoring plants growth including the measurement of physical properties, counting plants leaves, detection of plants and their separation from weeds brings much benefits to the producers. Preferable techniques for this task are those that are non-destructive because plant is a very sensitive creature and any manipulation of which can affect its growth or lead to losing leaves or branches. Imaging techniques are of the best solutions for plant growth monitoring and geometric measurements. This project involves the use of stereo imaging and multispectral data analysis for examining, separation, and classification of crop plant/weeds. Active and passive stereo imaging were employed for the estimation of physical properties and counting leaves and multispectral data was utilized for the separation of crop and weed. Bell pepper plant was used for imaging measurements during a 30-day period and for crop/weed separation, the spectral responses of bell pepper and five weeds were measured. Nine physical properties of pepper leaves (i.e., main leaf diameters, leaf area, leaf perimeter etc.) were measured using a scanner and used as a database. These were also used for comparing the estimated values with the actual values. The stereo system consisted of two webcams and a video projector. First the stereo system was calibrated using sample images of a standard checkerboard in different positions and angles. The system was controlled using the computer for switching a light line on, recording videos of both cameras while light is being swept on the plant, and switching the light off. The processing algorithm filtered the images for removing noise and then thresholded the unwanted pixels of the background environment. Next, using the Center of Mass peak detection method the central part of the light line was extracted. The images were then rectified and the correspondent pixels were detected and used for the 3D model development. The obtained point cloud was transformed to a meshed surface and used for measurement of physical properties. Passive stereo imaging was utilized for leaf detection and counting. Six different matching algorithms and three cost functions were used for passive stereo matching. In order to obtain spectral responses of plants, they were moved to the laboratory, leaves were detached from the

plants and placed on a blur dark background. Spectral measurements were carried out in the 400 – 1000 nm range using a spectroradiometer. To reduce the dimensionality of the data, PCA and wavelet transform were used. Results of this study showed that stereo imaging can be used as an inexpensive and non-destructive tool for agriculture, specifically, for plant detection and classification. Use of Active Stereo Imaging (ASI) during the primary stage of growth provides acceptable results, however, it is unable to detect and reconstruct all leaves and plant parts during the later stages. It was observed that red color was the best choice for the light stripe color. The results showed that as a trade-off between field of view and angle view of the cameras, 90 degrees angle was the best. Using ASI, yielded  $R^2$  values of 0.978 and 0.967 for estimation leaf area and perimeter, respectively. The result of segmentation of the point clouds could count the leaves with the success rate of 96.053 %. However, the overall success rate of the passive stereo algorithm for leaf counting was 84.32 %. It was observed that generally BP algorithm provides better results for disparity map development. For cost functions, SAD provided better results compared to MSE and NCC. The results of separation of crop and weeds using spectral data were very promising and the classifier—which was based on deep learning—was capable of completely separating pepper plants from five weeds.

# TABLE OF CONTENTS

I CONTEXT AND STATE OF THE ART.....	1
1 INTRODUCTION.....	2
1.1 Context and problem.....	2
1.2 Objectives.....	5
1.3 Contributions and structure of the thesis.....	6
2 State Of The Art.....	8
2.1 Monitoring techniques.....	8
2.2 Stereo 3D reconstruction.....	11
2.2.1 Passive stereo imaging.....	12
2.2.2 Active stereo imaging.....	13
2.2.3 Comparison of active and passive systems.....	14
2.2.4 Pinhole Camera model.....	15
2.2.5 Triangulation and epipolar geometry.....	17
2.2.6 Rectification.....	18
2.2.7 Delaunay Triangulation.....	20
2.2.7.1. D-dimensional Delaunay.....	21
2.3 Light stripe peak detection.....	21
2.3.1 Thresholding detection.....	22
2.3.2 Gaussian detector.....	22
2.3.3 Linear Interpolation.....	23
2.3.4 Center-of-mass.....	23
2.3.5 Parabolic Estimator.....	23
2.4 Noise detection.....	24
2.5 Camera calibration.....	24
2.5.1 Calibration methods.....	24
2.5.2 Internal parameters.....	25
2.5.3 External parameters.....	26
2.5.4 Stereo camera calibrators.....	26



2.6 Point cloud segmentation .....	28
2.6.1 Distance clustering .....	29
2.6.2 Shape clustering .....	30
2.6.3 Edge based segmentation .....	31
2.6.4 Region based segmentation.....	31
2.6.5 Attributes based segmentation .....	32
2.6.6 Model based segmentation.....	32
2.6.7 Graph based segmentation .....	33
2.7 Spectral data for agriculture .....	33
2.8 Dimensionality reduction techniques.....	36
2.8.1 Partial least square regression .....	37
2.8.2 High Correlation Filter.....	37
2.8.3 Random Forests / Ensemble Trees .....	37
2.8.4 Principal Component Analysis (PCA) .....	37
2.8.5 Backward Feature Elimination.....	38
2.8.6 Forward Feature Construction.....	38
2.8.7 Wavelet Transforms .....	38
<b>II METHODOLOGY .....</b>	<b>40</b>
<b>3 ACTIVE STEREO IMAGING .....</b>	<b>41</b>
3.1 Plants .....	42
3.2 Leaf scanner .....	43
3.3 Database of leaf features .....	44
3.4 Active stereo system .....	45
3.4.1 Temperature and humidity sensor.....	45
3.4.2 Light sensor .....	46
3.4.3 Light Projector .....	47
3.4.4 Micro-servomotor .....	48
3.4.5 Camera .....	49
3.4.6 Electronic unit .....	50
3.4.7 Graphic interface.....	50

3.4.8	System .....	51
3.4.9	Positioning of the cameras .....	53
3.4.10	Calibration .....	55
3.4.11	Preparation of checkerboard pattern.....	55
3.4.12	Evaluation of calibration accuracy .....	56
3.4.13	Taking active stereo images .....	58
3.4.14	Light strip color .....	58
3.4.15	Noise removal.....	59
3.4.16	Light stripe peak .....	60
3.4.17	Peak detection and extraction.....	60
3.4.18	Camera angle .....	60
3.4.19	Processing.....	61
3.4.20	Stereo Image Rectification .....	62
3.4.21	Triangulation .....	69
4	PASSIVE STEREO IMAGING.....	73
4.1.1	Acquisition system .....	73
4.1.2	Stereo images .....	73
4.1.3	Calibration.....	74
4.1.4	Processing algorithm.....	75
4.1.5	Depth estimation and scaling .....	77
4.2	Stereo matching and cost functions.....	79
4.2.1	Dataset.....	80
4.2.1.1	Block Matching Algorithm .....	80
4.2.1.2	Block Matching with Dynamic Programming .....	81
4.2.1.3	Belief Propagation.....	81
4.2.1.4	Histogram of Oriented Gradient.....	81
4.2.2	Matching Cost .....	82
5	SPECTRAL ANALYSIS .....	84
5.1	Application of multispectral data .....	84
5.1.1	Plants.....	84

5.1.2	Spectral Data .....	84
5.1.3	Measurements .....	85
5.1.4	Continuous Wavelet Transform .....	87
5.1.5	Convolutional Neural Network .....	88
5.1.6	<i>k</i> -means clustering .....	89
III RESULTS, DISCUSSIONS AND CONCLUSIONS .....		92
6 RESULTS AND DISCUSSION .....		93
6.1	Physical Properties Database .....	93
6.1.1	Leaf perimeter .....	93
6.1.2	Leaf area .....	94
6.1.3	Compare leaf area with theoretical models .....	94
6.2	Active stereo imaging .....	96
6.2.1	Comparison of laser and projector .....	96
6.2.1.1	Selection of the light stripe color .....	97
6.2.2	Processing the images .....	99
6.2.3	Light peak detection .....	99
6.2.4	Light stripe .....	99
6.2.5	Camera calibration accuracy .....	101
6.2.6	Cameras angle analysis .....	101
6.2.7	Leaf counting .....	101
6.2.8	Leaf geometrical properties .....	110
6.3	Passive stereo image processing .....	117
6.3.1	Segmentation .....	113
6.3.2	Number of leaves .....	115
6.3.3	Physical properties .....	116
6.4	Stereo matching algorithm .....	117
6.4.1	SAD .....	117
6.4.2	MSE .....	119
6.4.3	NCC .....	121
6.5	Spectral data for separation of plants .....	123

6.5.1 Separation using deep learning .....	123
6.5.2 Separation using $k$ -means clustering .....	128
7 CONCLUSIONS .....	131
8 SUGGESTIONS .....	134
REFERENCES .....	137
Appendix A .....	154
Appendix B .....	155
Appendix C .....	161
Appendix D .....	163

## TABLE OF Figures

<b>Figure 1.1.</b> Examples of images with different level of texture features; left) low level of texture and right) high level of texture .....	4
<b>Figure 2.1.</b> A classification of 3D reconstruction techniques.....	10
<b>Figure 2.2.</b> Example of 3D imaging in agriculture (Rusinkiewicz et al., 2002).....	10
<b>Figure 2.3.</b> Left and right images of a stereo camera. ....	11
<b>Figure 2.4.</b> Schematic of a passive stereo imaging system.....	12
<b>Figure 2.5.</b> Schematic of an active stereo imaging system (Bianco et al. 2013).....	14
<b>Figure 2.6.</b> Multiple view geometry in computer vision (Werner and Zisserman 2002) .....	15
<b>Figure 2.7.</b> Epipolar geometry; a) epipolar line segment corresponding to one ray, b) corresponding set of epipolar lines and their epipolar plane (Szeliski, 2010).....	18
<b>Figure 2.8.</b> The rectification algorithm presented by Loop and Zhang (1999); a) original image pair, b) images transformed so that epipolar lines are parallel, c) rectified images and d) final rectification by minimizing horizontal distortions.....	19
<b>Figure 2.9.</b> Estimation of disparity map; a) left image, b) right image, and c) disparity map (Liu et al. 2020).....	20
<b>Figure 2.10.</b> An example of Delaunay triangulation. ....	21
<b>Figure 2.11.</b> Light stripe peak.....	22
<b>Figure 2.12.</b> Example of distorted and corrected grids.....	26
<b>Figure 2.13.</b> An example of a segmented point cloud of a road surrounding (Xiang et al., 2018). ....	28
<b>Figure 2.14.</b> A simple example of Euclidean distance clustering.....	30
<b>Figure 2.15.</b> Example of edge-based segmentation (Che et al., 2019). ....	31
<b>Figure 2.16.</b> Segmented plant leaves using model-based clustering (Gélard et al., 2017). .....	33
<b>Figure 2.17.</b> Behavior of surface against passage of electromagnetic waves.....	34
<b>Figure 3.1.</b> Top-view RGB images of different bell-pepper samples used for stereo segmentation. ....	43

<b>Figure 3.2.</b> Hand-held leaf scanner and a sample image of a leaf.....	43
<b>Figure 3.3.</b> Measurement of geometrical properties of leaves; a) the scanned leaf, b) the processed image.....	45
<b>Figure 3.4.</b> The temperature and moisture sensor. ....	46
<b>Figure 3.5.</b> The light sensor. ....	47
<b>Figure 3.6.</b> Light projector used for active stereo imaging. ....	48
<b>Figure 3.7.</b> Micro-servomotor used for displacement of the laser.....	49
<b>Figure 3.8.</b> The LogiTech webcam.....	49
<b>Figure 3.9.</b> The controlling circuit of sensors.....	50
<b>Figure 3.10.</b> Graphical interface developed for measurements. ....	51
<b>Figure 3.11.</b> Chassis developed for holding the imaging system. ....	51
<b>Figure 3.12.</b> Developed holder for the cameras and the projector.....	52
<b>Figure 3.13.</b> Telescopic hand for holding the cameras.....	52
<b>Figure 3.14.</b> The imaging and measurement set-up.....	53
<b>Figure 3.15.</b> Block diagram of the stereo system. ....	53
<b>Figure 3.16.</b> Field of view and common area of cameras.....	54
<b>Figure 3.17.</b> Calculation of camera distance from the plant.....	55
<b>Figure 3.18.</b> The standard checkerboard used for calibration. ....	56
<b>Figure 3.19.</b> Sample images used for calibration taken with the left camera.....	56
<b>Figure 3.20.</b> The error measurement of reprojection in calibration of stereo cameras..	57
<b>Figure 3.21.</b> Bar chart of accuracy measurement of calibrated stereo cameras. ....	57
<b>Figure 3.22.</b> Spatial position of stereo cameras based on calibration and position of checkerboard.....	58
<b>Figure 3.23.</b> Sample images of left and right cameras; Top) a test object and down) a plant. ....	58
<b>Figure 3.24.</b> Different colors of light stripe (From left: red, green, blue). ....	59
<b>Figure 3.25.</b> Noise in the form of light grains on the images because of scattering of light on the surrounding.....	60
<b>Figure 3.26.</b> Histogram of the region including the light stripe on plant leaf. ....	60
<b>Figure 3.27.</b> Positioning of the cameras in relation to each other. ....	61

<b>Figure 3.28.</b> The usage of a simple plane for evaluation of camera angle. ....	61
<b>Figure 3.29.</b> Removing the light stripe on the ground from the left image. ....	62
<b>Figure 3.30.</b> Stereo camera pinhole model. ....	63
<b>Figure 3.31.</b> The structure of the stereo vision system. ....	64
<b>Figure 3.32.</b> The virtual parallel system. ....	65
<b>Figure 3.33.</b> Transformation of coordinate system.....	66
<b>Figure 3.34.</b> Schematic of the coordinate systems transformation. ....	67
<b>Figure 3.35.</b> Schematic of the left camera's reprojection. ....	69
<b>Figure 3.36.</b> An example of k-d tree.....	71
<b>Figure 4.1.</b> Schematic diagram of the stereo imaging system. ....	73
<b>Figure 4.2.</b> Original images used in stereo imaging for the depth detection. ....	74
<b>Figure 4.3.</b> The sample images used for calibration. ....	75
<b>Figure 4.4.</b> The thresholded image and its result after filling the holes. ....	77
<b>Figure 4.5.</b> Watershed image. ....	77
<b>Figure 4.6.</b> The overlapped right and left images after rectification. ....	78
<b>Figure 4.7.</b> Disparity map obtained for the distance of leaves from the camera. ....	78
<b>Figure 4.8.</b> The real picture of leaves after thresholding of 3D points image. ....	79
<b>Figure 4.9.</b> Sample left and right images with the disparity map. ....	80
<b>Figure 4.10.</b> Total of left images used in this study provided by The Middlebury Stereo Dataset. ....	80
<b>Figure 4.11.</b> Diagram showing the course of the whole algorithms.....	83
<b>Figure 5.1.</b> Spectral distribution of light sources.....	85
<b>Figure 5.2.</b> The measurement system and illumination.....	86
<b>Figure 5.3.</b> The spectral response of a) Pepper, b) Bindweed, c) Nutsedge, d) Plantago lanceolata, e) Potentilla, and f) Sorrel.....	87
<b>Figure 5.4.</b> The scalogram of CWT colored as an RGB image.....	88
<b>Figure 5.5.</b> Layer graph of the trained network. ....	88
<b>Figure 5.6.</b> The architecture of the GoogleNet. ....	89
<b>Figure 5.7.</b> Block diagram of the proposed algorithm.....	89
<b>Figure 6.1.</b> Leaf perimeter during growth period. ....	94

<b>Figure 6.2.</b> Leaf area during growth period.....	94
<b>Figure 6.3.</b> The estimated leaf area based on three theoretical model.....	95
<b>Figure 6.4.</b> Light reflection around the light stripes taken by: Left) projector and Right) laser projection (the contrast has been ameliorated).....	96
<b>Figure 6.5.</b> Light stripe width. ....	97
<b>Figure 6.6.</b> Light stripes and their scattering in the edges. ....	98
<b>Figure 6.7.</b> Adjacent pixels to the red light stripe.....	98
<b>Figure 6.8.</b> Adjacent pixels to the green light stripe.....	98
<b>Figure 6.9.</b> Adjacent pixels to the blue light stripe.....	99
<b>Figure 6.10.</b> Effect of smoothing filter on the light stripe; Left) original image and Right) after filtering. ....	99
<b>Figure 6.11.</b> Light peak detection; a) Gaussian, b) Center of mass, Parabolic and Linear Interpolation.....	100
<b>Fig. 6.12.</b> The combined light stripes projected on the plant. ....	101
<b>Figure 6.13.</b> Detection of checkerboard and correspondence of the predicted and actual centers in a pair of images (75°).....	102
<b>Figure 6.14.</b> Average calibration error for all pairs of calibration images in 75°.....	102
<b>Figure 6.15.</b> Position of cameras and checkerboards in the global coordination system (75°).....	102
<b>Figure 6.16.</b> Detection of checkerboard and correspondence of the predicted and actual centers in a pair of images (90°).....	103
<b>Figure 6.17.</b> Average calibration error for all pairs of calibration images in 90°.....	103
<b>Figure 6.18.</b> Position of cameras and checkerboards in the global coordination system (90°).....	103
<b>Figure 6.19.</b> Detection of checkerboard and correspondence of the predicted and actual centers in a pair of images (105°).....	104
<b>Figure 6.20.</b> Average calibration error for all pairs of calibration images in 105°.....	104
<b>Figure 6.21.</b> Position of cameras and checkerboards in the global coordination system (105°).....	104



<b>Figure 6.22.</b> 3D development of simple plane in different camera angles; a) 105°, b) 90° and c) 75°.	106
<b>Figure 6.23.</b> Result of 3D reconstruction of top of a cylinder; Left) point cloud and Right) mesh.	107
<b>Figure 6.24.</b> 3D point cloud of a single leaf.	107
<b>Figure 6.25.</b> Segmented point cloud and discrimination of leaves.	108
<b>Figure 6.26.</b> Comparison four different active stereo imaging systems done by Gupta and Ibaraki (2014); Left) point cloud, middle) depth map, and right) segmented model.	108
<b>Figure 6.27.</b> An example of a meshed point cloud of a leaf into a 3D surface and the magnified triangulation connections.	109
<b>Figure 6.28.</b> The 3D model developed based on the segmented point cloud.	109
<b>Figure 6.29.</b> The predicted number of leaves compared to the actual number.	110
<b>Figure 6.30.</b> Leaf perimeter using Alphashape.	111
<b>Figure 6.31.</b> An example of the convex hull of a leaf from two different points of view.	111
<b>Figure 6.32.</b> Regression of the estimated and actual values for leaf length and width.	112
<b>Figure 6.33.</b> Regression of the estimated and actual values for leaf area and perimeter.	112
<b>Figure 6.34.</b> Regression of the estimated and actual values.	113
<b>Figure 6.35.</b> Image of the extracted leaves after removing the background and filling holes.	115
<b>Figure 6.36.</b> Pseudo-coloring of leaves after detection.	115
<b>Figure 6.37.</b> Estimated leaf area versus the actual area.	116
<b>Figure 6.38.</b> Estimated leaf perimeter versus the actual perimeter.	117
<b>Figure 6.39.</b> Left and right images and the disparity maps achieved for different matching algorithms based on SAD cost function.	118
<b>Figure 6.40.</b> Left and right images and the disparity maps achieved for different matching algorithms based on SAD cost function.	118

<b>Figure 6.41.</b> Left and right images and the disparity maps achieved for different matching algorithms based on SAD cost function.....	119
<b>Figure 6.42.</b> Left and right images and the disparity maps achieved for different matching algorithms based on MSE cost function. ....	120
<b>Figure 6.43.</b> Left and right images and the disparity maps achieved for different matching algorithms based on MSE cost function. ....	120
<b>Figure 6.44.</b> Left and right images and the disparity maps achieved for different matching algorithms based on MSE cost function. ....	120
<b>Figure 6.45.</b> Left and right images and the disparity maps achieved for different matching algorithms based on NCC cost function. ....	121
<b>Figure 6.46.</b> Left and right images and the disparity maps achieved for different matching algorithms based on NCC cost function. ....	121
<b>Figure 6.47.</b> Left and right images and the disparity maps achieved for different matching algorithms based on NCC cost function. ....	122
<b>Figure 6.48.</b> The whole comparison of all algorithms for all the images. ....	123
<b>Figure 6.49.</b> The scalogram of CWT colored as an RGB image; a) pepper and b) Bindweed. ....	124
<b>Figure 6.50.</b> Diagram showing the network training process and accuracy per iteration. ....	126
<b>Figure 6.51.</b> Diagram of loss function values per iteration during the network training process. ....	126
<b>Figure 6.52.</b> First convolutional layer weights; a) pepper and b) Bindweed.....	127
<b>Figure 6.53.</b> First convolutional layer weights; a) pepper and b) Bindweed.....	127
<b>Figure 6.54.</b> Confusion matrix of the network for validation samples; a) pepper and two weeds, b) pepper and three weeds, c) pepper and four weeds, and d) pepper and five weeds. ....	127
<b>Figure 6.55.</b> Scatter plots (PCA) of clustered spectral data: a) pepper and one weed, b) pepper and two weeds, c) pepper and three weeds, d) pepper and four weeds. ....	129
<b>Figure 6.56.</b> Scatter plot (PCA) of the clustered spectral data for pepper and five weeds. ....	129

**Figure 6.57.** Optimal number of clusters; a) Calinski-Harabasz, b) Davies-Bouldin, c) Silhouette, and d) Gap ..... 130

**Figure 6.58.** Scatter plot (PCA) of clustered data for 700-1000 nm..... 130

## List OF Tables

<b>Table 3-1.</b> Leaf physical properties used for morphological estimation of leaf shape. .....	44
<b>Table 3-2.</b> Technical specifications of DHT22 temperature and humidity sensor. ...	46
<b>Table 3-3.</b> Technical Specifications of GY302 Light Intensity Sensor. ....	47
<b>Table 3-4.</b> Technical specifications of the projector. ....	48
<b>Table 3-5.</b> Technical specifications of the cameras. ....	49
<b>Table 4-1.</b> The databases of stereo and ground truth images used in this study. ....	80
<b>Table 5-1.</b> Technical specifications of the spectroradiometer. ....	85
<b>Table 6-1.</b> Actual physical properties of leaves. ....	93
<b>Table 6-2.</b> Measurement of glossiness of bell pepper leaves. ....	97
<b>Table 6-3.</b> Measurement error of plate properties for different camera and plate angles.....	105
<b>Table 6-4.</b> Reconstruction in different camera angles. ....	106
<b>Table 6-5.</b> Estimated physical properties of leaves.....	111
<b>Table 6-6.</b> Minimum and maximum of average error for different algorithms. ....	122
<b>Table 6-7.</b> Number of samples used for training the network.....	123
<b>Table 6-8.</b> The training information of the network for pepper and three weeds...	125
<b>Table 6-9.</b> The training information of the network for pepper and four weeds. ...	125
<b>Table 6-10.</b> The training information of the network for pepper and five weeds...	125



I

# CONTEXT AND STATE OF THE ART

# INTRODUCTION

## 1.1 Context and problem

Increase in agricultural production and its optimization is necessary to supply the ever-increasing population of the world and to transcend phenomena such as famine, disease and environmental pollutions. Thus, due to lack of natural resources and limitations of energy supply, it is necessary to lower costs and produce food and agricultural products in shorter time periods. Comprehensive changes in the current agricultural infrastructure, wide cooperation of entities, proper management and exploitation of resources and utilities, wise scheduled organization, and scientific management of growth and harvest are strategies to handle the above-mentioned problems. Therefore, without using the up-to-date technologies, it is not possible to improve agricultural productions and use resources efficiently.

On the other side, plant growth and its fruit production is the outcome of a complicated interaction between genotypic properties and growing conditions. It is possible to evaluate plant growth destructively by routinely harvesting plants parts and nondestructively by taking numerous measurements while leaves still are connected to the plant. Both approaches have benefits and drawbacks: The growth of a single plant can be tracked over time by nondestructive instruments, and variations in growth between plants can be researched. On the other side, touching the plant repeatedly while measuring them could potentially impede their development. The optimum equation representing the growth of the plant is normally chosen once the measurements, either destructively or nondestructively, have been taken. These

equations are used for predictions of growth but not for real-time monitoring of the growth. Therefore, systems or models are needed that can predict the behavior of plants as a function of growing conditions so that the growth conditions can be adjusted based on that. These planting conditions include different production aspects such as light, temperature, moisture, CO<sub>2</sub>, and nutrition necessary for plants. Thus, production enhancement requires quantitative knowledge of plant structure, its behavior and growth conditions.

There are many non-destructive methods (i.e. thermal imaging, fluorescent imaging, visible camera systems, 3D imaging, synthetic aperture radar etc.) for the instant measurement of growth but the complicated appearance of plants and frequent changes of their structures during growth have limited the usage of such methods (Kacira and Ling, 2001; Chaerle et al., 2009; Bagavathiappan et al., 2013; Zhang et al. 2019). Mathematical plant growth models are of the tools for describing and understanding growth trends of plants which can be used for developing a production plan. These models are utilized for decision-making systems, greenhouse controlling systems and production prediction. However, these models are not able to detect the growth irregularities and unexpected production incidents. Hence, always there has been a need to novel and accurate systems.

Image processing has always been into attention of scientists as it is one of the fastest and most accurate non-destructive monitoring techniques. Indeed, image processing is used in the applications that like human vision, the physical properties and appearance of the problem are considered for the detection and prediction. Image processing can replace common models and detect growth trend and unexpected growth changes in a higher accuracy. This technology can adjust plant needs in each moment of its growth based on the conditions and in addition to preventing the waste of water and nutrients, provide the optimized and proper condition of growth. This leads to fully-automized production of agricultural plants. As the goal is to provide a non-contact and non-invasive monitoring system, a system is proposed that is based on imaging and image processing techniques.



One of image processing techniques that is used for the estimation of 3D properties based on 2D images is Stereo Image processing. Stereo Imaging provides the ability to determine depth, volume, surface properties and build 3D models. Therefore, by using stereo imaging a more realistic idea of the problem can be achieved which this helps to detect barriers and overlapped objects. Stereo imaging can be passive or active. Passive stereo imaging includes only two cameras. In this imaging method, the depth estimation is not performed well for the objects with no texture or environmental difference. However, active stereo imaging has the advantage of detecting the depth and 3D angles in high accuracy. Also, it is advantageous for the imaging at night that this is important as during the day light intensity changes and affects the image processing results. Another advantage is that the active stereo set-up can be proposed at a lower price as this factor is of high importance for agricultural producers. Active stereo is useful in regions where there is a lack of light and/or texture (Szeliski, 2010). The infrared projector or another light source will illuminate the scene with a light texture. But along with its positiveness, there are some disadvantages such as active stereo will lose its effectiveness in direct sunlight and in regions with a high interference of the same external light source technology used.



*Figure 1.1. Examples of images with different levels of texture features; left) low level of texture and right) high level of texture*

Then the following are some of the benefits of using an active stereo vision system:

- It works with scenes that lack adequate features, such as edges or corners, that are connected with intensity discontinuities;
- the correspondence problem is completely missing in the stereo matching process. The plane equation of the light stripe would have been computed

from the projector calibration procedure for each pixel in the image that is illuminated by a particular light stripe from the projector or laser unit, and simple triangulation is all that is required to compute the 3D coordinates of the pixel.

- It produces a highly detailed depth (or range) map.
- It may be used in a wide range of shape measurement and flaw detection tasks.

After the development of such a system, bell-pepper plant has been considered for its evaluation. Bell-pepper is Herbaceous plant and has straight and strong stems. The length of stems reaches 0.5-1.5 m and 3 to 4 fruits are produced at the end of each stem (Marcelis et al. 2006). The amount of bell-pepper production of the world in the year 2019 was 57 034 382 ton of which China ranked first having the production of 37 985 275 ton (FAO 2019). Bell pepper is a nutritious vegetable and effective for different diseases. Capsaicin, as the most active ingredient in pepper, lowers blood cholesterol and reduces the intensity of pain messages in the body. It also has antioxidant properties that protect the body's cells from damage by free radical molecules and also has antibacterial properties (Barzegar 2013).

Accordingly, a system is needed that non-destructively and with high accuracy can monitor the plant growth and determine the needs of plant including light, temperature and moisture based on the current situation of the plant. So, the goal of the system is to acquire plant images, detect and determine the number and size of leaves, estimate the growth stage, and monitor growth behavior.

## **1.2 Objectives**

Growth conditions of plants are different in various stages of growth and need to be adjusted frequently. These conditions include amount of irrigation, soil moisture, relative humidity, light, and nutrition. By adjusting the growth conditions and revising them when needed, higher production in a shorter period can be achieved which means optimal use of resources for maximum production. As adjusting growing conditions depends on the determination of plant lifetime and growth stage, development of non-

destructive accurate systems for the estimation of plant lifetime is necessary. The existing methods such as growth models, not only should be recalibrated in each area, but also cannot detect unexpected variations in growth rates. In addition, most other techniques and systems for monitoring plants are very expensive for agricultural producers. Production optimization, control of growth conditions, prevention of diseases, production increase, product quality improvement, mechanized production, lower production costs, and handling irregularities of growth are among the benefits of adjustment and management of growth conditions.

Hence, objectives of the current study are:

- Development of a stereo imaging system for the estimation of growth stages; This involves construction of a low-cost system, use of video projector instead of laser light, comparison of different light colors and a geometrical rectification.
- Determination of number, area, and physical properties of leaves based on image processing; This goal involves the use of image processing techniques for the extraction of the desired properties from the point cloud and 3D models.
- 3D modelling of aerial parts of bell-pepper plant and determination of growth stages; Which includes the development of 3D models of plants based on the obtained point clouds and construction of growth curves.
- Application of Multispectral imaging for the detection and separation of pepper plants from weeds; This is to study the separation of bell pepper plant from weeds based on spectral data as a preprocess for localizing each plant and monitoring its growth.

### **1.3 Contributions and structure of the thesis**

This thesis attempts to define the problem of growth monitoring of plants, describe fundamentals and theoretical basics of stereo image processing, consider the background of this research, and explain the methodology. In the first section, the problem and objectives are considered as well as the theoretical background of the research. The second section presents the tools, instrumentation and the developed

system as well as the processing methods used to achieve the objectives. In the third section, the results obtained from the experiments, data analysis, the application of artificial intelligence and comparison of results with other reported research is presented. Furthermore, conclusions and perspectives of this research are provided.

Monitoring the growth of plants at different stages of growth has always been the focus of researchers and for decades they have been looking for practical and effective solutions. Therefore, several mathematical models for different stages of growth and for different plants have been developed and different tools were researched and studied. One of the most important tools has been machine vision, which provides the closest tool to human understanding of plant growth and development. Various imaging methods and different processing techniques were examined. But one of the novel and accurate methods is stereo image processing, which is based on two images from different viewing angles. In addition to the previous methods and tools, this method can provide useful information about the three-dimensional condition of the plant.

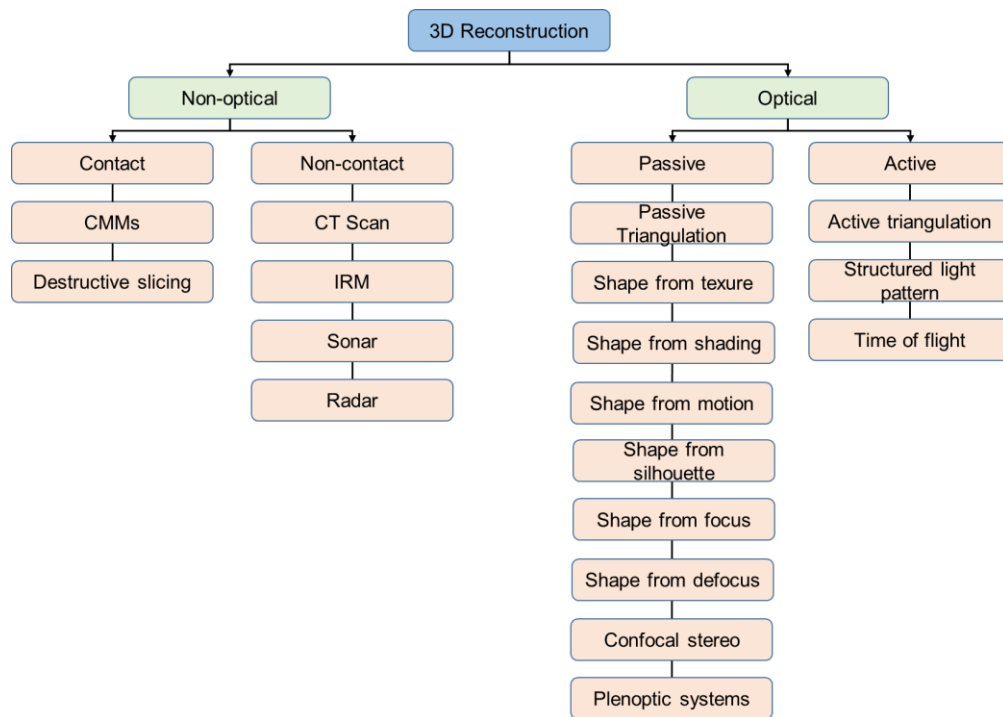
## 2.1 Monitoring techniques

There have been used different techniques for monitoring purposes in agriculture (Loch et al., 2005; Hashemi et al., 2010; Jin and Tang, 2009). These techniques are generally divided into destructive or non-destructive techniques. Imaging plants is not a simple work and is considered as a complicated imaging task. The complicated and special shape of leaves and how they cover each other pose difficulties on the imaging, monitoring and phenotyping of plants. Based on the application, different imaging techniques have been employed and studied. Imaging techniques include visible light imaging, thermal imaging, near infrared imaging, hyperspectral imaging, 3D imaging, laser imaging, MRI and CT. Where possible, these techniques have been used for agricultural applications as well. Visible imaging has normally been used for the measurement of leaf area, color, growth dynamics, perimeter, seed morphology, and

plant disease (L. Li et al. 2014; Chéné et al., 2012). For color imaging, no special camera is needed and almost all types of RGB cameras have been utilized in different projects (Khojastehnazhand et al. 2019). Near-Infrared (NI) imaging has been used along with visible imaging for monitoring and quality determination. Applications of NI in agriculture have been reported for forages, grains and grain products, oilseeds, coffee, tea, spices, fruits, vegetables, sugarcane, drinks, fats and oils, dairy products, eggs, meat, and other agricultural products (Xu et al. 2007; Wu et al. 2012; Huang et al. 2020).

Also, these techniques alone or in conjunction with other sensory technologies have been employed for 3D reconstruction of objects. The techniques for 3D reconstruction vary and highly depend on the application in use. Figure 2.1 presents a classification of 3D reconstruction techniques, however this is not a inclusive diagram and other techniques may be added to it (Billiot, 2013).

The goal of three-dimensional (3D) imaging is to capture the three-dimensional structure of sceneries and objects in our surroundings (Rusinkiewicz et al., 2002) (Figure 2.2). Color-texture information in the form of a registered 2D image, normally derived from ordinary digital image capture, is frequently included with the computed set of data points in 3D space. A 3D model, a 3D scan, or a 3D image are all terms used to describe 3D data that includes or excludes color/texture (Pears et al. 2012).



**Figure 2.1.** A classification of 3D reconstruction techniques.



**Figure 2.2.** Example of 3D imaging in agriculture (Rusinkiewicz et al., 2002).

A 3D imaging process' output can be analyzed and processed to extract data for a variety of applications, including object recognition, web shape search, face recognition for security and surveillance, robot navigation, mapping of the Earth's surface, forests, or urban areas, and clinical procedures in medicine.

Laser scanning (also known as LiDAR) is a type of 3D object scanning that combines controlled laser beam steering with a laser rangefinder. The scanner quickly captures the surface contour of objects, buildings, and landscapes by obtaining distance measurements in all directions. Combining various surface models derived

from different viewing angles, or admixing other known limitations, is required to create a comprehensive 3D model. In a technology similar to photogrammetry, small objects can be placed on a rotating pedestal.

## 2.2 Stereo 3D reconstruction

Stereo matching or stereo 3D reconstruction is the technique for producing the 3D model of a scene or an object by using 2D images (Dipanda et al., 2003; Jang et al., 2013). Stereo matching may use two or more images for 3D reconstruction (Besse et al., 2014; Schuldt et al., 2016). The concept of stereo matching emanated from human eyes and how human perceives depths. Hence, in stereo matching the differences of location of objects in two images of right and left cameras lead to achieving the depths information (Poggi et al., 2019). Figure 2.3 presents the left and right images and it can be seen that objects shift to left or right in comparison to another image. Appendix D presents several datasets for 3D models of different objects prepared in different research works.



*Figure 2.3. Left and right images of a stereo camera.*

As it will be discussed in the following sections, for simple configuration of cameras, the horizontal distance of objects in left and right images is inversely proportional to the distance of objects from the cameras (Heise et al., 2015). Based on this principle, the depth and then the 3D coordinations of each point in space will be calculated. While it looks simple, still the matching process of correspondent points of two images is a challenge which is called correspondence problem.

The application and use of stereo imaging techniques is growing fast. The interest in stereo imaging grown as it is a quick and accurate technique. Also, stereo imaging

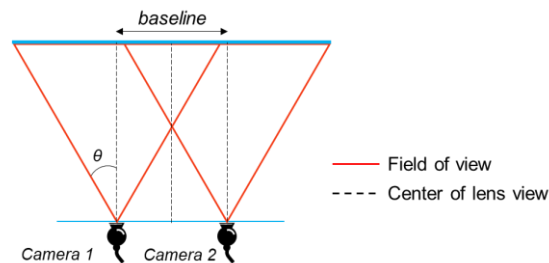


is a promising technique for the applications that require a non-contact method do to their sensitivity and manipulation considerations.

The use of this concept was started by photogrammetry for automatically constructing topographic maps using overlapping aerial images. However, this was progressed to build automatically images based on accurate matching of left and right images. For years, Stereo Imaging (SI) has been an active area of research for different fields of science and still it is of high importance for many applications. SI has been vastly used in different fields for the extraction of 3D information. In agricultural applications, SI has been employed for phenotyping, 3D reconstruction, and leaf counting (Guo and Xu 2017; Nguyen et al. 2016; T. T. Santos and De Oliveira 2012; T. SANTOS and Ueda 2013). However, in most cases, the expensive market scanners have been used. In this study, a cheap system based on webcams and normal video projector is presented.

### 2.2.1 Passive stereo imaging

Passive visual sensing is a type of visual sensing that doesn't require any other devices than cameras (Urquhart, 1997). In most cases, these algorithms were created during the early stages of computer vision research. The photos are the only input data, and no energy is emitted for sensing purposes. The sensing techniques were frequently intended to mimic the way human eyes work. When opposed to active procedures, which require additional devices, passive approaches have a competitive edge in terms of equipment cost.



*Figure 2.4. Schematic of a passive stereo imaging system.*

Stereo vision, trinocular vision, and a variety of monocular shape-from-X approaches, such as 3D shape from texture, motion parallax, focus, defocus, shadows, shading, specularities, occluding contours, and other surface discontinuities, are examples of passive techniques (Hartley and Zisserman, 2003). The issue is that retrieving three-dimensional information from a single two-dimensional image is an ill-posed problem. Stereo vision remains the only passive cue that provides sufficient precision.

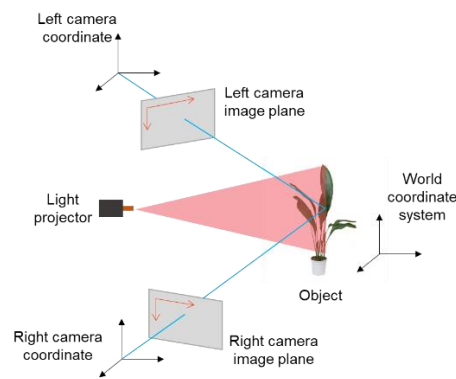
A significant disadvantage of the passive approach is that it necessitates the use of two cameras and that it is incompatible with untextured surfaces, which are typical in industrially created goods. In passive visual sensing, the demand for ambient light conditions is equally crucial. Stereo vision has the advantage of being fairly easy to implement and especially ideal for natural surroundings.

### **2.2.2 Active stereo imaging**

Active stereo imaging, as contrast to passive visual sensing, is a type of visual sensing approach. Only visible features with discernible texture gradients, such as intensity edges, are measured in the aforementioned situations of passive approaches (that employ ambient light) (Geng 2011). There is a related issue with the stereo configuration, for example, if the difference in location and orientation of the stereo images is minor, matching corresponding spots is simple; if the difference is considerable, it is challenging. When the difference in location and orientation of the stereo views is modest, however, the accuracy of the 3D reconstruction suffers (Pears et al. 2012).

Active sensing techniques have been developed in recent years to address the inadequacies of passive sensing (Latimer, 2005; Wang et al., 2012; Dong et al., 2018). These active systems are typically free of the correspondence issue and can measure with extreme precision (Dipanda and Woo, 2005). An external projecting device (such as a laser or LCD/DLP projector) is used in active sensing to actively emit light patterns that are reflected by the scene and detected by a camera (Cajal et al., 2015; Tu et al., 2019) (Figure 2.5). Lam (2006) used a patterned laser light as the light source

for the usage of active stereo imaging for medical diagnosis. That is to say, rather of relying on natural lighting, they probe the scene in some way. Active stereo techniques are generally more precise and reliable than passive imaging methods.



*Figure 2.5. Schematic of an active stereo imaging system (Bianco et al. 2013).*

Most ambiguities are resolved by active 3D vision sensors, which offer the geometry of an object or surroundings directly. To create the 3D coordinates, they only need a little help from the operator. Laser-based techniques, on the other hand, make the 3D information highly immune to background illumination and surface texture. As a result, active visual sensing is ideal for scenes with insufficient details. It is mainly ideal for interior locations because it requires lighting management, and both the camera and the projector must be pre-calibrated. The structured light system with coded patterns is based on active triangulation, which is one of several 3D range data collecting approaches in computer vision. Scanning a scene with a laser plane and detecting the location of the reflected stripe is a fairly basic way for obtaining depth information with structured light. The distortion along the detected profile can be used to compute depth information. Structured light systems and laser range finders map obtained data directly into a 3D volumetric representation, avoiding the correspondence problem that plagues passive sensing techniques. Scenes with no textural features, for example, can be readily modeled.

### **2.2.3 Comparison of active and passive systems**

What are the advantages and disadvantages of active and passive stereo imaging systems? In conclusion, because the computational overhead of passive

correspondences is reduced, active systems that can generate dense range images at high frame rates are often easier to develop (e.g. 30 fps for the Kinect) (Dias et al., 2006). Passive stereo might result in low density 3D reconstructions due to a lack of surface features or large-scale texture on the scene object, at least in the local regions where the surface texture or features (e.g. corners) are lacking (Sun et al., 2002).

For starters, it makes it impossible to establish the size and shape of the photographed object in its entirety. Second, when the imaged item is presented from many various views, it is challenging to produce accurate form visualizations. Active stereo systems, on the other hand, offer thorough form measurements and acceptable renderings for multi-viewpoint visualizations as long as the surface is not too dark (low reflectivity) or specular, and there aren't too many deep concavities ('missing portions'). When the density of features is low or the image sensing resolution is poor in comparison to the scale of the imaged texture, an active stereo system is the best option (Pears et al. 2012).

#### 2.2.4 Pinhole Camera model

The primary step in the development of an imaging system is to determine the camera model. In this study the pinhole camera model was utilized. The pinhole camera is a simplified camera model. Most previous research on active stereo imaging have been based on pinhole model (Lam, 2006; Coste, 2013; Kihlström, 2019; Weisenfeld, 2002). The light travels in a straight line. The light reflected by the object passes through the pinhole to form an inverted image on the imaging surface. The distance between the pinhole and the imaging surface is called the focal length. Generally speaking, the smaller the pinhole, the clearer the image, but too small a pinhole will cause diffraction, which will make the image blurry.

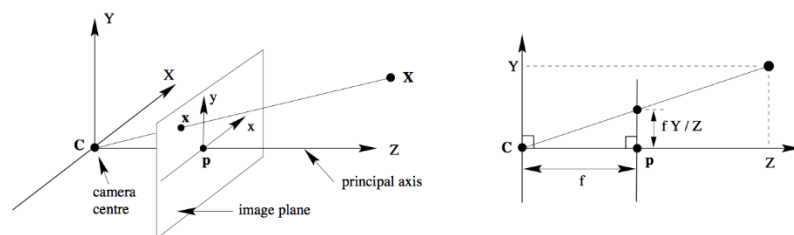


Figure 2.6. Multiple view geometry in computer vision (Werner and Zisserman 2002)

The light emitted from a point  $X$  in the external world passes through the small hole and is projected at the point  $x$  on the image plane. The quantitative relationship between a point  $X$  in 3D space and the corresponding point  $x$  coordinates on the imaging plane is:

$$(X, Y, Z) \rightarrow (x, y) \leftrightarrow (fX/Z, fY/Z)$$

We can write the transformation between 3D and 2D in the following form.

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$x = PX$$

When actually calculating, we first convert the 3D point into a 4-dimensional vector, and then multiply the transformation matrix to the left. The transformation matrix  $P$  known as the *Camera Projection Matrix* is completely determined by the camera parameters.

The above formula assumes that the principal point  $p$  is at the origin of the coordinate. The actual situation may not be the case, so the mapping becomes:

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} fX + Zp_x \\ fY + Zp_y \\ Z \end{bmatrix} = \begin{bmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, x = K[I|0]X$$

This matrix  $K$  is known as the *Camera Calibration Matrix*. In addition, it is also possible that our pixels are not square, so when we measure image coordinates in pixels, we need to introduce a non-equivalent scale factor  $m_x$ ,  $m_y$  in each direction. Specifically, the number of pixels per unit distance of the image coordinate in the  $x$  and  $y$  directions are  $m_x$ ,  $m_y$ , then our calibration matrix.

$$K = \begin{bmatrix} f_{m_x} & 0 & m_x p_x & 0 \\ 0 & f_{m_y} & m_y p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Finally, in order to increase generality, we also need to consider distortion parameters, although our current standard camera usually  $s=0$ ,

$$K = \begin{bmatrix} f_{mx} & s & m_x p_x & 0 \\ 0 & f_{my} & m_y p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

We can draw a mapping between the point in world coordinate system  $X$  and the point  $x$  in the image plane, represented by:

$$x = KR [I | -C]X$$

$$P = KR [I | -C]$$

Parameters in  $K$  are called the *camera intrinsics* and the remaining parameters  $R$  and  $C$  are called the *camera extrinsics*.

### 2.2.5 Triangulation and epipolar geometry

Given a pixel in one image, the correspondent pixel in the second image needs to be found and identified. This technique is called triangulation (Lam, 2006). A point is imaged and is seen in two different images. Then using calibration information, the 3D information of the point is obtained (Teutsch et al., 2005; Veitch-Michaelis, 2017). A variety of search techniques can be used to match pixels based on their local appearance as well as the motions of neighboring pixels. In stereo imaging, there are additional information available which are the positions and calibration data for the cameras. These information can be utilized to reduce the number of potential correspondences as well as reduce the computation cost.

Figure 2.7 shows how a point in the scene is projected on the images and the way these projected pixels are related. In this figure, the centers of the two cameras are  $C_0$  and  $C_1$ , and  $p$  is a three-dimensional space point, and its projection points on the imaging planes of the two cameras are  $x_0$  and  $x_1$ , respectively. Baseline is the connection between the optical centers of the two cameras  $C_0C_1$ . Epipolar plane is a plane containing the baseline. There is a set of epipolar planes (rotating with the

baseline as the axis). An example in this figure is  $C_0pC_1$ . And epipolar line is the line of intersection between the epipolar plane and the image plane. The example in the picture above is  $x_0e_0$  versus  $x_1e_1$ . These epipolar lines are the geometrical place of the point  $p$  on the images meaning that for finding the corresponding pixel of  $p$  on the second image, we need to just search on this epipolar line. This solution has been pretty common in previous research (Tošić and Frossard, 2010; Bianco et al. 2013; Lelas and Pribanić et al. 2016). Coste (2013) built disparity maps from rectification of images based on epipolar geometry.

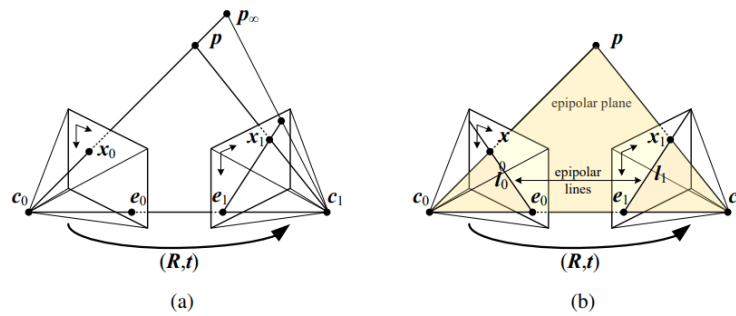
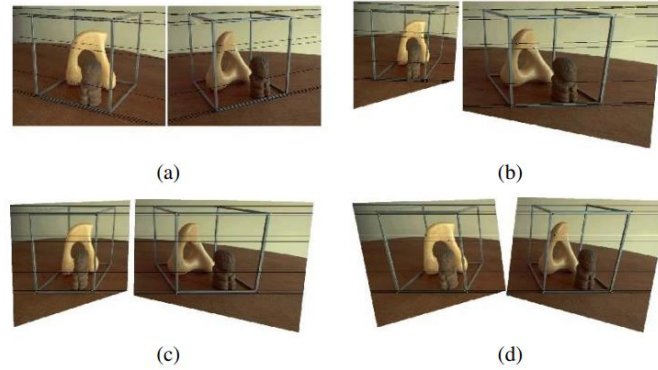


Figure 2.7. Epipolar geometry; a) epipolar line segment corresponding to one ray, b) corresponding set of epipolar lines and their epipolar plane (Szeliski, 2010).

## 2.2.6 Rectification

The epipolar geometry for a pair of cameras can be computed from the relative pose and calibration of the cameras. This can be done using seven or more point matches using the fundamental matrix. Once this geometry has been computed, it can be employed for the epipolar line corresponding to a pixel in one image to constrain the search for corresponding pixels in the other image. One of the techniques to do this is to use a general correspondence algorithm, such as optical flow. In this case, the pixels only on the epipolar line will be considered in the search for correspondence. A more efficient algorithm can be obtained by first rectifying (i.e, warping) the input images so that corresponding horizontal scanlines are epipolar lines. Then, it will be easier to consider horizontal lines for correspondence. Furthermore, there have been reported studies to ease and simplify correspondence problem. Chan (2002) used an active method for stereo matching which was the projection of specific patterns on the

object through structured lights. This technique could simplify matching problem and reduce computation cost.



**Figure 2.8.** The rectification algorithm presented by Loop and Zhang (1999); a) original image pair, b) images transformed so that epipolar lines are parallel, c) rectified images and d) final rectification by minimizing horizontal distortions.

A simple way for the rectification of the two images is to first rotate both cameras so that they are looking perpendicular to the line joining the camera centers  $c_0$  and  $c_1$ . Because of a degree of freedom in the tilt, it is better to use the smallest rotations achieving this angle. Afterwards, for determining the desired twist around the optical axes, make the up vector (the camera y axis) perpendicular to the camera center line. This helps to be sure that the corresponding epipolar lines would be horizontal and that the disparity for points at infinity will be zero. Finally, re-scaling of the images is done, if necessary, to account for different focal lengths, and to magnify the smaller image to the size of the bigger one for preventing aliasing. It should be noted that normally it is not possible to rectify an arbitrary collection of images simultaneously unless the optical centers of these images are collinear. However, the rotation of the cameras so that they all point in the same direction reduces the inter-camera pixel movements to scalings and translations. This rectified geometry is being used in a lot of stereo imaging set-ups and systems. This makes a straightforward way to build an inverse relationship between 3D depths  $Z$  and disparities  $d$ ,

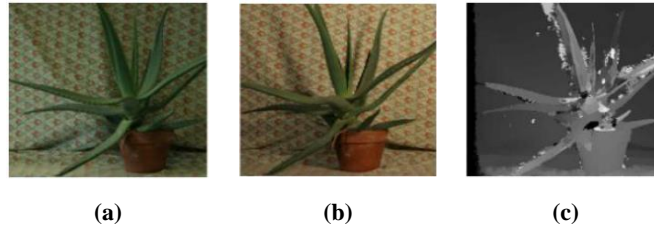
$$d = f \frac{B}{Z}$$

where  $f$  is the focal length (measured in pixels),  $B$  is the baseline, and



$$x' = x + d(x, y), \quad y' = y$$

describes the relationship between corresponding pixel coordinates in the left and right images. The task of extracting depth from a set of images then becomes the estimation of the disparity map  $d(x, y)$ . After rectification, we can easily compare the similarity of pixels at corresponding locations  $(x, y)$  and  $(x_0, y_0) = (x + d, y)$  and store them in a disparity space image (DSI)  $C(x, y, d)$  for further processing (Figure 2.9).



*Figure 2.9. Estimation of disparity map; a) left image, b) right image, and c) disparity map (Liu et al. 2020).*

### 2.2.7 Delaunay Triangulation

Having the rectified left and right images, it is time to calculate the 3D position of each point of the object. The known parameters are the 2D coordination of each point in the images (i.e. pixel address of the point in the image) and the information regarding the position of the cameras in space. Then, there is need to a technique to reach 3D coordiantes from 2D coordinates. Triangulation is the process of calculating the 3D position of points in images (Dorsch et al., 1994). In previous works, mostly researchers have used Delaunay Triangulation (Trujillo Romero, 2008). In mathematics and computational geometry, Delaunay's triangulation for a set of points called  $P$  on a plane is a triangulation called  $DT(P)$  so that none of the points  $P$  are in any of the perimeter circles of the triangle ( $DT$ ). Minimization of triangles maximizes the angles of triangles to prevent narrow triangles. This triangulation was invented by Boris Delaunay in 1934. For four or more points on the same circle (e.g. vertices of a rectangle) the Delaunay Triangle is not unique: both possible triangles that divide a quadrilateral into two triangles satisfy the Delaunay condition, for example, assuming that if all the circumferential circles of the triangles are empty. Considering enclosed spheres, the Delaunay trinity idea will be extended to three or more dimensions.

Generalization the metric system is preferred to the Euclidean system, but in these (Euclidean) cases the Delaunay trinity does not necessarily exist or is not unique.

Delaunay's trinity in the discrete space of the set  $p$  in the general position is taken from the double graphs of the Voronoi diagram of the set  $p$ . Special cases that include three points on a line and four points on a circle.

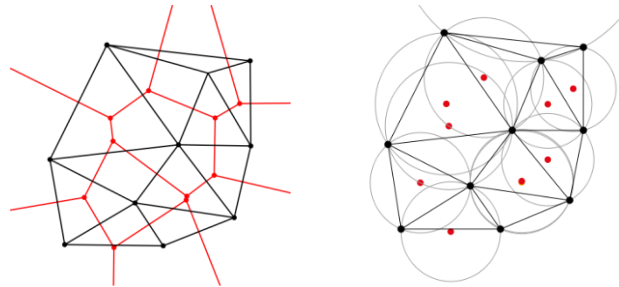


Figure 2.10. An example of Delaunay triangulation.

### 2.2.7.1. D-dimensional Delaunay

The dimensional  $d$  is the sum of the points  $p$  in Euclidean space.  $p$  is a group of points in the general position that results in no subsequent  $k$  subspace containing  $k + 2$  points or the next  $k$  sphere containing  $k + 3$  points for  $k \leq d - 1$  (for example for a set of points in  $\mathbb{R}^3$  no three dots are on a line, no four dots on a plane, no four dots on a circle, and no five dots on a sphere). Convex for a set of points in the  $d + 1$  space by giving extra coordinates  $2|p|$ , to each point  $p$  and taking the bottom corner of the convex cover and retransforming it to the next  $d$  space by removing the recent coordinates. It is the same assuming that all the sides of the cap are convex, the non-simple sides occur only when  $d + 2$  is one of the principal points on a next  $d$ -sphere as examples of points are not in the general position.

## 2.3 Light stripe peak detection

The peak point of a light stripe picture is located via light peak detection. The light projection angle, camera calibration parameters, and peak location in the images are used to retrieve the object's surface 3D information. As a result, the light stripe peak is one of the most crucial elements in the scanning process. [Schnee and Futterlieb \(2011\)](#) used dynamic models for robust and fast segmentation of a laser line. They

believed that dynamic models can better segment laser lines deformed because of shapes of objects. Zhang et al. (2013) presented a peak detection method based on continuous wavelet transform. It was observed that their proposed method led to good results comparable to the common techniques.

A typical profile of light stripe intensities is shown in Fig. 2.11. Despite the fact that the maximum pixel value is around 135, its precise location is unknown and must be calculated. As a result, we need to know which light peak detection approach in our 3D laser scanning system is the most accurate and exact.

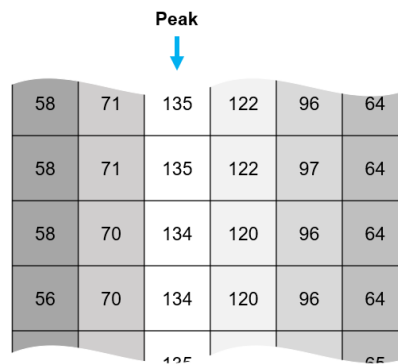


Figure 2.11. Light stripe peak.

### 2.3.1 Thresholding detection

A common way of detecting the light peak is thresholding. This technique removes the pixels smaller than a value that has been tested and validated for the detection of the brightest pixels. In this technique, it is needed to perform several tests to find the optimum amount of the threshold. Then, by thresholding the images, the pixels over the threshold are kept and used for 3D reconstruction.

### 2.3.2 Gaussian detector

The three highest, contiguous intensity values surrounding the recorded peak of the stripe are used in this technique, which assumes the observed peak shape meets a Gaussian profile. The laser light impinge on the scene is known to have a roughly Gaussian distribution. The peak's sub-pixel offset is determined by (Isa and Lazoglu, 2017):

$$\alpha = \frac{\ln(f(x-1)) - \ln(f(x+1))}{2(\ln(f(x-1)) - 2\ln(f(x)) + \ln(f(x+1)))}$$

### 2.3.3 Linear Interpolation

This method implies that the distribution of intensity values before and after the peak is defined by a simple, linear relationship. As a result, assuming the three highest intensities are determined as before:

if  $f(x+1) > f(x-1)$

$$\alpha = \frac{f(x+1) - f(x-1)}{2 * (f(x) - f(x-1))}$$

else

$$\alpha = \frac{f(x+1) - f(x-1)}{2 * (f(x) - f(x+1))}$$

### 2.3.4 Center-of-mass

The center-of-mass (CM) algorithm also assumes a Gaussian distribution for the spread of intensity values over the stripe. As a result, a simple weighted-average method can be used to determine the peak's location. The peak's sub-pixel position is indicated by:

$$\alpha = \frac{f(x+1) - f(x-1)}{f(x-1) + f(x) + f(x+1)}$$

### 2.3.5 Parabolic Estimator

The Taylor series expansion of the signal near the peak can be used to create a continuous version of the peak finder.  $\alpha$  is estimated if the peak is at  $f(x + \delta)$  while  $\delta$  is introduced and the received signal is  $f(x)$ :

$$\alpha = \frac{f(x-1) - f(x+1)}{2(f(x+1) - 2f(x) + f(x-1))}$$

## 2.4 Noise detection

The presence of noise in stereo images is very possible. This happens as a result of having bright light in the darkness of the environment. Reflections and refractions of the light on objects cause some level noise in the images. In case of using low-sensitivity cameras, this noise can appear more. [Amir and Thörnberg \(2017\)](#) for noise removal and spurious reflections in laser imaging used several methods including Spatial Compound Imaging, High Dynamic Range Extension, Gray Level Transformation, Center of Gravity and Most Similar Nearest Neighbor. The best precision achieved was  $\pm 4.2 \mu\text{m}$ . In a recent work, [Chiang and Lin \(2022\)](#) took benefit of multi-decoding pattern and epipolar line denoising methods. They studied a two-step denoising process for improved spatial resolution.

## 2.5 Camera calibration

The process of determining the internal camera geometric and optical characteristics (intrinsic parameters) and/or the 3D position and orientation of the camera frame relative to a certain world coordinate system (extrinsic parameters) in the context of three-dimensional data capture is known as camera calibration. The primary goal of calibration is to determine the projection model parameters that relate known 3D points to their projections, and then rebuild/identify the object using these values. The camera imaging geometry determines the relationship between the points' positions in object space and the corresponding points in the picture. The intrinsic/extrinsic settings of the camera define this geometry. Camera calibration is the process of determining these characteristics, as well as the camera's position and orientation in the world coordinate system.

### 2.5.1 Calibration methods

Calibrating stereo cameras is normally accomplished by calibrating each camera separately and then using geometric transformations to the external parameters to determine the stereo setup geometry. There are three types of camera calibration procedures currently available ([Li and Chen, 2003](#)):

- Closed form solution.
- Full nonlinear optimization.
- Two steps methods.

Lens distortion is not taken into account by the first-class methods, and comprehensive nonlinear optimization is difficult because there are at least 11 parameters to calibrate. Methods with two phases are more efficient and precise.

Even though it is possible to divide the parameters into two groups and perform nonlinear optimization only on a subset of the parameters, every calibration technique for radial distortion that is based on knowledge of the position of points in the world must search for the entire set of parameters at the same time.

The calibration procedure requires the user to:

- use a calibration grid.
- individuate the projections of calibration points in the image.

Another type of calibrating approach is one that does not require any prior knowledge of where points in the world are located. In this scenario, internal calibration may be distinguished from external calibration, and the optimization step can be carried out, for example, over the four internal parameters  $k_l$ ,  $O_x$ ,  $O_y$ , and  $a_x$ .

### **2.5.2 Internal parameters**

The internal parameters of a camera (in the pinhole camera concept) are  $f$ ,  $P_x$ ,  $P_y$ ,  $k_l$ ,  $O_x$ ,  $O_y$ ,  $a_x$ . If we know the values of  $f$ ,  $P_x$ , and  $P_y$  from the camera's technical specs, the calibration parameters are  $k_l$ ,  $O_x$ ,  $O_y$ , and  $a_x$ .

We define the relationship between the coordinates of the distorted and undistorted images in order to calibrate the internal parameters of each camera. It is feasible to undistort the image so that it can be treated as originating from a pinhole camera by knowing  $k_l$ ,  $O_x$ ,  $O_y$ ,  $a_x$ , ( $P_x$  and  $P_y$ ).

The distorted and undistorted images of the left and right cameras are exhibited in the following photographs.

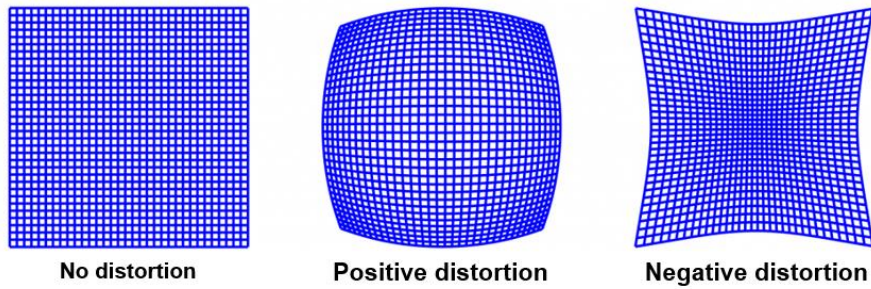


Figure 2.12. Example of distorted and corrected grids.

### 2.5.3 External parameters

Both the correspondence problem (identifying the epipolar lines for determining point correspondences) and triangulation require external parameters (for reconstruction).

In the following, we assume that we are dealing with a stereo camera in its typical configuration (i.e. parallel optical axes and coplanar picture planes), and that small angle approximation is valid (angles determining  $R$  are small).

The left camera is chosen as the global reference system, and the parameters to be discovered are the right camera's translation vector  $T$  and rotation matrix  $R$  in relation to the left camera.

It is necessary to locate the translation vector  $T$ .  $T_x$  is usually determined by measuring the distance between the centers of the cameras (that equals the baseline  $b$ ). We regard  $T_y$  and  $T_z$  to be 0 as well.

The three rotation angles around the right camera axis determine the rotation matrix  $R$  ( $\phi$ ,  $\theta$ , and  $\psi$ ). Rotation around the  $X$  axis ( $\phi$ ) and around the  $Y$  axis ( $\theta$ ) can be approximated as displacements across  $T_y$  ( $Y_{\text{off}}$ ) and  $T_x$  ( $X_{\text{off}}$ ) using small angle approximation.

### 2.5.4 Stereo camera calibrators

There are many applications for stereo camera calibration. Mostly these applications have been proposed based Matlab, OpenCV or Python languages ([Esteban et al., 2010](#)). There exists stand-alone applications for calibration as well.

The stereo calibrator of Matlab is called by the function `stereoCameraCalibrator` and provides a graphical interface for the user to perform the calibration. This calibrator is quite easy to use and functions pretty fast. The app requires the address of the folders of left and right calibration samples as well as the size of each square. The app provides the calibration error to redo the calibration or use it for further processing. The results of calibration—including the internal and external parameters—can be exported as a calibration file and be loaded anytime during the processing steps.

There are other toolboxes for Matlab that perform the calibration based on different criteria and techniques that can be found on the web. [Uematsu et al. \(2007\)](#) presented an application called 'D-Calib' for calibration multicamera systems. Their calibration approach included two steps: initializing camera parameters with a planar marker pattern and optimizing initial parameters by tracking two markers linked to a wand. Also, it was reported that the calibrator functions pretty fast and works well for large baseline systems.

Self-calibration calibration also has been studied for online stereo imaging. [Szczepanski 2019](#) proposed a self-calibration method that relies solely on data from the seen scene and excludes the usage of controlled models. First and foremost, they regard calibration to be a background system activity that must operate continuously in real time. Internal calibration was not the system's primary responsibility, but it was a procedure that high-level applications rely on. As a result, system constraints severely limited the algorithm's complexity, memory, and execution time. Because the suggested calibration approach used typical data from computer vision applications and required low resources, it could be buried within the application pipeline.

[Lam \(2006\)](#) used Ksai's algorithm for the calibration of the system. They used a light projection pattern and a single camera for active stereo imaging. Ksai's method is one of the simplest and most common methods for stereo calibration. The use of

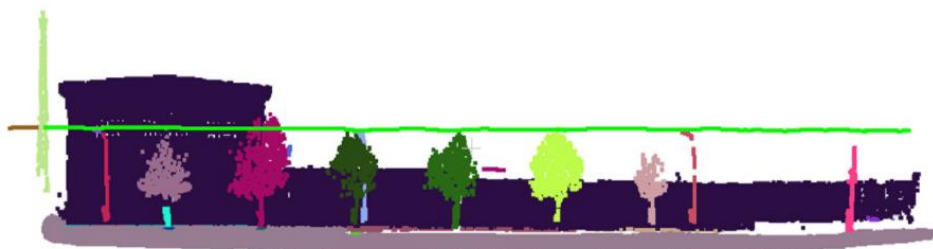


machine learning for stereo calibration or improvement of calibration has been studied in much research (Do 1999; Nichols 2001; Memon et Khan, 2001; Xing et al., 2007; Donné et al., 2016). Won et al. (2019) proposed an end-to-end deep neural network model for omnidirectional depth estimate using a multi-view stereo configuration with a large baseline.

## 2.6 Point cloud segmentation

The aim was to segment point clouds for counting leaves. In this regard, the use of clustering techniques for segmentation of the point clouds was necessary. Clustering methods allow data to be partitioned into subgroups, or clusters in an unsupervised manner. These segments, on the surface, appear to combine comparable observations together. As a result, clustering algorithms are very dependent on how this notion of similarity is defined, which is often field-specific. For exploratory data analysis, clustering techniques are frequently utilized. They also make up the majority of the procedures used in AI classification pipelines to produce well-labeled datasets in an unsupervised/self-learning manner.

Clustering techniques are particularly beneficial in situations where labeling data is costly. Take the task of annotating a big point cloud as an example. It can be a long and tiresome operation annotating each point with what it represents, to the point that persons performing it accidentally introduce inaccuracies due to inattention or exhaustion. Allowing a clustering algorithm to group comparable points together and then only involving a human operator when providing a label to the cluster is less expensive and possibly more efficient.



*Figure 2.13. An example of a segmented point cloud of a road surrounding (Xiang et al., 2018).*

Unsupervised algorithms have a more difficult time defining their purpose than supervised algorithms, which have a clear mission to complete (e.g. classification or regression). As a result, the model's success is more subjective. The fact that the goal is more difficult to define does not preclude a variety of performance indicators, which will be discussed below.

Clustering is the process of putting together the points that are the most similar or close together. Clustering is strongly reliant on the principles of distance and similarity.

These ideas will be quite helpful to formalize:

- (1) the distance between two observations;
- (2) the distance between an observation and a cluster;
- (3) the distance between two clusters.

### **2.6.1 Distance clustering**

The Euclidean and Manhattan distances are two of the most often used examples of distances. The "ordinary" straight-line distance between two places in Euclidean space is known as the Euclidean distance. The Manhattan distance is named after the two-dimensional distance traveled by a taxi on Manhattan's streets, which are all either parallel or perpendicular to one another.

As a result, a distance can be used to determine similarity: the greater the distance between two places, the less similar they are, and vice versa. We can convert a distance  $d$  between  $x$  and  $y$  into a similarity measure with just a smidgeon of arithmetic. Another frequent method for determining similarity is to utilize the Pearson correlation, which considers the form of the distribution rather than its amplitude, as the Euclidean distance does. It's crucial to pick the right distance measurement.

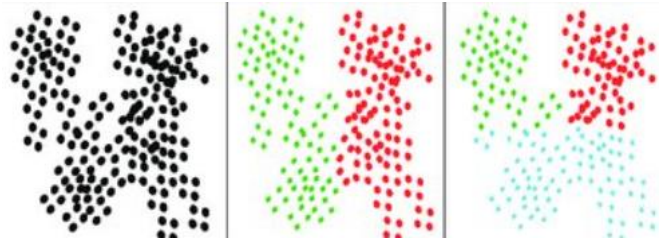


Figure 2.14. A simple example of Euclidean distance clustering.

## 2.6.2 Shape clustering

A cluster's form is a crucial feature that we first define as:

- (1) tightened inwards: two close points must be part of the same cluster.
- (2) separated by a large distance: two points separated by a large distance must belong to different clusters.

We frequently look for clusters that are tightening on themselves. Let's use the Euclidean distance as an example to translate these qualities. To begin, we may quickly compute the centroid of a cluster. The average of the distances between each of the locations in the cluster and the centroid can then be used to determine the cluster's homogeneity. A tightened cluster will have less heterogeneity than a cluster of distributed points in this way. The average of the homogeneity of each cluster may then be calculated to characterize not just one cluster, but all clusters in our dataset.

Second, the clusters should be separated from one another. The distance between the centroids of two clusters is commonly used to quantify this separation. We may calculate the average of these quantities on all of the obtained pairs of clusters once more.

Homogeneity and separation are the two criteria we must now optimize. We may simplify things by combining them into a single criterion, the Davies-Bouldin index. This measure compares intra-cluster distances (homogeneity) — which should be low — to inter-cluster distances (separation), which should be high. This index is all the weaker for a given cluster because all of the clusters are homogeneous and well separated.

### 2.6.3 Edge based segmentation

Edges describe the characteristics of an object's shape. To obtain segmented regions, edge-based methods detect the boundaries of several regions in point clouds. These approaches work on the premise of locating spots with a quick shift in intensity. Fig. 2.15 shows an example of edge-based segmentation of a point cloud.

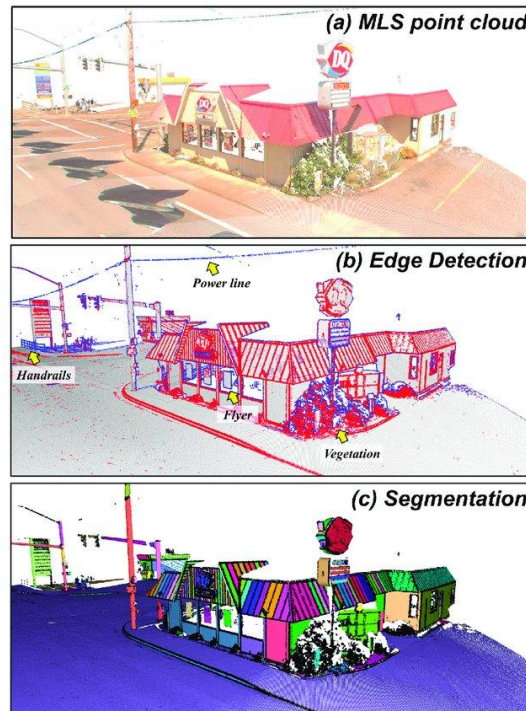


Figure 2.15. Example of edge-based segmentation (Che et al., 2019).

Although edge-based methods allow for quick segmentation, they have accuracy issues because they are all very susceptible to noise and unequal point cloud density, which are frequent in point cloud data.

### 2.6.4 Region based segmentation

Region-based approaches combine close points with similar qualities to create isolated areas and, as a result, find dissimilarity between the distinct regions using neighborhood knowledge. Methods that are based on regions are more accurate in terms of noise than methods that are based on edges. However, they have issues with excess or under segmentation, as well as accurately determining area borders.

### **2.6.5 Attributes based segmentation**

Attributes-based methods are robust methodologies based on point cloud data clustering attributes. There are two steps to these approaches. The first stage is to compute the attributes, and the second step is to cluster the point clouds based on the derived attributes. Clustering approaches allow for the incorporation of different cues into the segmentation process by accommodating spatial relationships and features. These methods have the drawback of being very reliant on the quality of derived qualities. To get the optimum separation between distinct classes, the properties of point cloud data should be computed properly.

The most reliable strategy for grouping points into homogeneous zones is to use attributes-based algorithms. Their outcomes are adaptable and precise. These methods, on the other hand, rely on the definition of a point's neighborhood and the point density of point cloud data. Another drawback of these methods is that they take a long time to process when dealing with multidimensional attributes for a large number of input points.

### **2.6.6 Model based segmentation**

Geometric primitive shapes (e.g., sphere, cone, plane, and cylinder) are used to group points in model-based techniques. Points with the same mathematical representation are placed together to form a segment. RANSAC (RANdom SAMple Consensus) is a well-known algorithm that is frequently utilized. RANSAC is a robust model for detecting mathematical properties such as straight lines, circles, and so on. For model fitting, this method is considered the gold standard. Many following works in 3D point cloud segmentation have adopted this first technique. [Gélard et al., 2017](#) used model-based approach for phenotyping of sunflower plants. Structure from Motion was used to create a 3D point clouds from RGB photos collected all around a plant. After that, the proposed method was used to segment and label plant leaves (Fig. 2.16).



*Figure 2.16. Segmented plant leaves using model-based clustering (G elard et al., 2017).*

Model-based approaches are based on pure mathematics. With outliers, they are quick and resilient. These approaches' fundamental flaw is their inaccuracy while dealing with various point cloud sources.

### **2.6.7 Graph based segmentation**

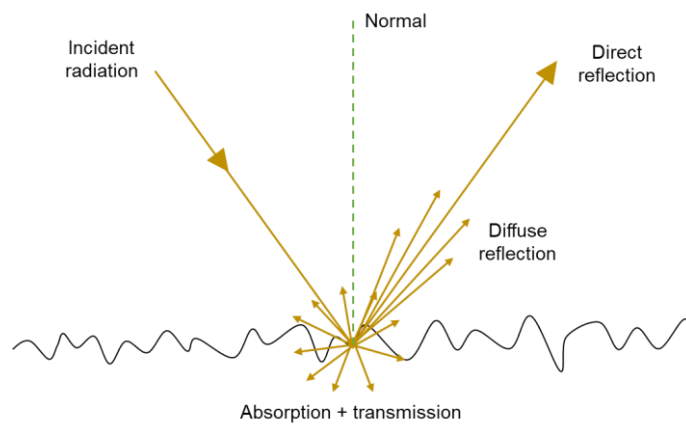
The point clouds are viewed as a graph in graph-based approaches. Each vertex represents to a point in the data, and the edges connect to specified pairings of surrounding points in a simple model. Because of their efficiency, graph-based approaches are becoming increasingly attractive for robotic applications. The FH algorithm is a well-known example of this strategy. This algorithm is simple and efficient, and it works similarly to Kruskal's technique for finding the shortest path in a graph.

In comparison to other methods, graph-based algorithms can separate complicated sceneries in point cloud data with superior results, especially when there is noise or uneven density. These methods, on the other hand, are rarely able to run in real time. Some of them may require further offline training or a co-registered sensor and camera setup.

## **2.7 Spectral data for agriculture**

Another contribution of this work was to study the ability of spectral data analysis for discrimination of bell pepper from weeds as the primary step for monitoring bell

pepper growth measurements. When optical radiation hits the surface of one of the many components of the environment, it can be reflected, transmitted, or absorbed, depending on the energy conservation principles (Fig. 2.17). The material and quality of the surface on which the radiation impinges determine the nature and intensity of this activity. The proportions, quantity, size, and form of the elements that make up the material will influence the parameters of the interaction, determining which features of the incident radiation will be affected and to what extent (Sandoval et al., 2010).



*Figure 2.17. Behavior of surface against passage of electromagnetic waves.*

The energy of an electromagnetic wave is proportional to its wavelength; the shorter the wavelength, the more energy is contained in the wave. As previously stated, when this energy comes to the surface of a body, it is either reflected from, collected by, or transferred by it. The wavelength, as well as the physical and chemical features of the body, influence the level and intensity of each process. Plants' spectrum reflectance is determined by the structure of their surface and the cells in their leaves, in addition to the absorption of their elements. The optical qualities of leaves are determined by their composition and structure, as well as their water content and biochemical concentration.

Spectral reflectance in the visible (VIS) and near-infrared (NIR) ranges, according to several research papers and scientists, is a potent and effective instrument for evaluating plant and agricultural features and conditions (Wang, 2018). As a result,

most agricultural research rely on measurements in the visible (400–700 nm) and near-infrared (700–2500 nm) wavelength ranges. Optical properties in these regions have been shown in numerous studies over the last few decades to be capable of detecting physiological and biological functions of plants and crops, potentially providing information for agricultural uses. Some researchers offered a set of wavelengths for which reflectance values can provide a wealth of information about the plant's state and performance. They specifically mentioned 550 nm (green reflectance peak), 650 nm (chlorophyll absorption band), 850 nm (infrared reflectance plateau), 1450 nm (water absorption band), 1650 nm (reflectance peak following 1450 nm), 1950 nm (water absorption band), and 2200 nm (water absorption band) (reflectance peak following water absorption band at 1950 nm) ([Luisa España-Boquera et al., 2006](#); [Carter and Knapp](#); [de Jong et al., 2012](#); [Xiaoping et al., 2017](#); [Zhao et al., 2019](#); [Lan et al., 2013](#); [Alchanatis et al., 2005](#); [Lin et al., 2017](#); [Zude-Sasse et al., 2002](#); [Feng et al., 2019](#); [Babellahi et al., 2020](#)). All of these studies suggest that if the organism in question is a plant, and the characteristics of one or more of these processes can be monitored, we can deduce important information about the plant's health and functionality from the data.

Among the methods most frequently employed for analyzing spectral data and classifications are K-nearest Neighborhood Classifier, Linear Discriminant Analysis (LDA), Principal Component Analysis (PCA), Normalized Difference Vegetation Indices, Fourier Transformation, Jeffries–Matusita Distance Measure, Support Vector Machines and Artificial Neural Networks. In previous works, these techniques have been utilized for the extraction of information from spectral data ([Zhang et al. 2002](#); [Bell et al. 2004](#); [Noble et al. 2009](#); [Longchamps et al. 2010](#); [Durgante et al. 2013](#); [Strothmann et al. 2017](#); [Zarco-Tejada et al. 2018](#); [Louargant et al. 2018](#)). [Symonds et al. \(2015\)](#) created a system for identifying plants in real time using discrete reflectance spectroscopy. In this study, three different laser diodes (i.e. 635 nm, 685 nm and 785 nm) were used. They reported that the system could make a practical discrimination for a vehicle speed of 3 km/h. In another work, [Nidamanuri \(2020\)](#) utilized machine learning to distinguish between kinds of tea plants. Canopy level hyperspectral



reflectance measurements were acquired for tea and natural plant species in the range of 350 nm to 2 500 nm. They could discriminate six out of nine tea plant varieties with accuracies between 75 % and 80 %. [Laba et al. \(2005\)](#), using derivative spectral analysis, examined how to distinguish between invasive wetland plant species. First, second and higher-order derivatives of the reflectance spectra of nine field plots were obtained. The results showed that dates in August were optimal from the point of view of species differentiability and could be selected for image acquisitions.

Convolutional Neural Networks (CNN) have recently been implemented and improved for classification purposes. The benefit of CNNs is that they develop their own feature sets through network training, which enables them to discriminate between samples that haven't been seen with a high performance rate ([Garibaldi-Márquez et al. 2022](#)). [Andrea et al. \(2017\)](#) discriminated maize and weed using CNNs. They verified LeNET, AlexNet, cNET and sent architectures and the cNET resulted to the best performance in terms of accuracy (95.05 %) and processing time (2.34 ms). [Xi et al. \(2020\)](#) proposed a network called MmNet consisting of the Local Response Normalization of AlexNet, GoogLeNet and VGG inception models. The proposed MmNet led to an accuracy of 94.50% and a time cost of 10.369 s. [Garibaldi-Márquez et al. \(2022\)](#) studied the use of shallow and deep learning techniques for the discrimination of plant and weed. VGG16, VGG19, and Xception models were trained and tested leading to the accuracies of 97.93 %, 97.44 % and 97.24 %, respectively.

## **2.8 Dimensionality reduction techniques**

When dealing with a high number of variables, the challenge of feature selection in classification typically emerges. In recent years, the requirement has grown as a result of the management of increasingly big databases, particularly in fields like genetics and image processing. As a result, the number of variables (features) used to classify an image or an object must be reduced ([Hamdi et al., 2014](#)).

As a result, the dimension reduction algorithms employed for this purpose strive to locate a projection of the data in a reduced-dimensional space while retaining the information contained within. It's possible that this projection will be linear or

nonlinear. Herein a short description of dimensionality reduction techniques is presented.

### **2.8.1 Partial least square regression**

Partial least squares (PLS) regression is a technique that reduces the predictors to a smaller set of uncorrelated components and performs least squares regression on these components, instead of on the original data. When your predictors are strongly correlated or when you have numerous predictors than observations and conventional least-squares regression either fails completely or yields coefficients with large standard errors, PLS regression is very helpful.

### **2.8.2 High Correlation Filter**

Data with comparable trends are more likely to contain similar data. Only one of these will be enough to input the machine learning model in this scenario. The Pearson's Product Moment Coefficient and the Pearson's chi square value are used to calculate the correlation coefficient between numerical and nominal data. The number of columns in a pair that have a correlation coefficient greater than a threshold is reduced to just one. A word of caution: correlation is scale sensitive, therefore for a meaningful correlation comparison, column normalization is essential.

### **2.8.3 Random Forests / Ensemble Trees**

In addition to being effective classifiers, Decision Tree Ensembles, often known as random forests, are beneficial for feature selection. One method for reducing dimensionality is to create a large, carefully designed set of trees against a target property, then utilize the usage statistics for each variable to discover the most informative subset of features. We can create a big number of extremely shallow trees (e.g. 2 levels) with each tree trained on a small percentage of the entire number of characteristics, resulting in a large number of trees. If an attribute is frequently chosen as the best split, it is most likely a useful trait to keep. The most predictive attributes are determined using a score based on attribute usage statistics in the random forest.

### **2.8.4 Principal Component Analysis (PCA)**

The statistical process Principle Component Analysis (PCA) orthogonally transforms the original  $m$  coordinates of a data set into a new set of  $n$  coordinates called principal components. The first principal component has the highest possible variance as a result of the transformation; each subsequent component has the maximum possible variance under the constraint of being orthogonal to the preceding components. By maintaining only the first  $n$  components, the data dimensionality is reduced while the majority of the data information, i.e. the variance in the data, is retained. It's worth noting that the PCA transformation is sensitive to the original variables' relative scale. Before using PCA, the data column ranges must be standardized. It's also worth noting that the new coordinates (PCs) are no longer true system-generated variables. When you use PCA to analyze your data, it loses its interpretability. PCA is not the transformation for your project if interpretability of the data is vital for your analysis.

### **2.8.5 Backward Feature Elimination**

The specified classification algorithm is trained on  $n$  input features in this technique at each iteration. Then, one by one, we eliminate input features and train the same model on  $n-1$  input features  $n$  times. We delete the input feature that caused the smallest increase in the error rate, leaving us with  $n-1$  input features. Following that, the classification is repeated using  $n-2$  features, and so on. Each iteration  $k$  yields a model with  $n-k$  features and an error rate of  $e$ . We specify the fewest number of features required to achieve that classification performance with the chosen machine learning method by selecting the maximum allowable error rate.

### **2.8.6 Forward Feature Construction**

Backward Feature Elimination is the inverse of this method. We begin with just one feature, gradually adding one feature at a time, i.e. the one that results in the greatest boost in performance. Backward Feature Elimination and Forward Feature Construction are both time-consuming and computationally intensive methods. They're practically only useful for data sets with a small number of input columns.

### **2.8.7 Wavelet Transforms**

A discrete wavelet transform (DWT) decomposes an input signal into a number of sets, each set consisting of a time series of coefficients that describes the signal's evolution in the corresponding frequency band. Wavelet transforms can be used to multidimensional data, including a data cube. This is done by first applying the transform to the first dimension, then to the second, and so on. Wavelet transforms have many real-world applications, including the compression of fingerprint images, computer vision, analysis of time-series data, and data cleaning.

## II

# METHODOLOGY

**3**

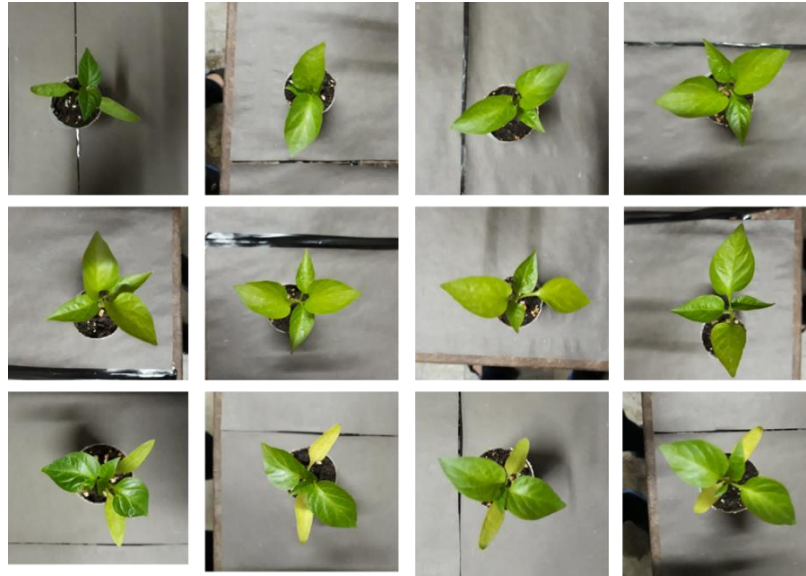
# **ACTIVE STEREO IMAGING**

To obtain the 3D model of the plants based on stereo images, a stereo system was developed. The stereo imaging system takes the image of the plant from two different positions and after processing, extracts the 3D point cloud which needs to be meshed for building the 3D model. In this chapter first the development of the imaging system is provided and then the calibration and processing.

### **3.1 Plants**

Bell-pepper seeds of the variety Cavendish F1 were procured and cultivated in a glass greenhouse that provides good exposure to the sun. The soil composition was 1:1:1 fertilized soil, coco peat and perlite. The plants were irrigated daily (i.e. around a centimeter) and temperature and moisture of the greenhouse were fully controlled. Stereo images were obtained every other day at early stages of growth. This included plants in 5th to 35th day of growth period. Figure 3.1 shows the RGB images of some of the samples from a top view.

The experiments of this research were performed in a greenhouse located in the Faculty of Agriculture of Tarbiat Modares University. One of the greenhouse rooms, which was equipped with a cooling system, was used for the experiments. Plastic pots were used for planting and growing bell peppers.



*Figure 3.1. Top-view RGB images of different bell-pepper samples used for stereo segmentation.*

### **3.2 Leaf scanner**

To build a database of actual physical properties of plants' leaves as well as measure these dimensions during growth period, a handheld scanner was prepared. The minimum and maximum dimensions of bell pepper plant leaves were extracted from the previous works (i.e. 6.5\*12.5 cm). Therefore, a scanning surface of 9\*14 cm was considered. To provide a light scanner, transparent plexiglass was used for the scanner. A webcam (A4Tech, China) with the resolution of 14 MP was used directly on the top of the scanner. This camera directly is connected to the laptop by USB cable.



*Figure 3.2. Hand-held leaf scanner and a sample image of a leaf.*



### 3.3 Database of leaf features

During growth, images of leaves were taken every two days and the physical properties of the leaves were extracted from these images. As growing plants is a slow process and to decrease the huge amount of image processing computations, two-day interval was chosen. Nine physical properties were utilized as indices representing leaf shape (Wu et al. 2007). The list of these properties and their definitions have been provided in Table 3-1. These properties were required to make a database of leaf properties and to train the neural networks based on actual leaf properties. As a whole, the physical properties of 311 leaves were obtained.

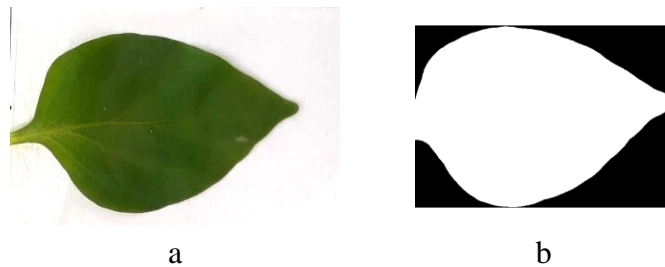
**Table 3-1.** Leaf physical properties used for morphological estimation of leaf shape.

Property	Description
Physiological length	The distance between the two terminals is defined as the physiological length. It is denoted as $L_p$
Physiological width	Drawing a line passing through the two terminals of the main vein, one can plot infinite lines orthogonal to that line. The number of intersection pairs between those lines and the leaf margin is also infinite. The longest distance between points of those intersection pairs is defined as the physiological width. It is denoted as $W_p$
Leaf area	The value of leaf area is easy to evaluate, just counting the number of pixels of binary value 1 on smoothed leaf image. It is denoted as $A$
Leaf perimeter	Denoted as $P$ , leaf perimeter is calculated by counting the number of pixels consisting leaf margin
Aspect Ratio	The aspect ratio is defined as the ratio of physiological length $L_p$ to physiological width $W_p$ , thus $L_p/W_p$
Form Factor	This feature is used to describe the difference between a leaf and a circle. It is defined as $4\pi A/P^2$ , where $A$ is the leaf area and $P$ is the perimeter of the leaf margin
Rectangularity	Rectangularity describes the similarity between a leaf and a rectangle. It is defined as $L_p W_p/A$ , where $L_p$ is the physiological length, $W_p$ is the physiological width and $A$ is the leaf area
Perimeter ratio of diameter	Ratio of perimeter to diameter, representing the ratio of leaf perimeter $P$ and leaf diameter $D$ , is calculated by $P/D$
Perimeter ratio of Lengths	This feature is defined as the ratio of leaf perimeter $P$ and the sum of physiological length $L_p$ and physiological width $W_p$ , thus $P/(L_p + W_p)$ .

The written algorithm cropped the image to the region of interest which is the leaf zone, extracted the green layer of the image. This helps in segmentation of the images

to foreground and background. Then a threshold was applied and produced a binary image containing the leaf shape. A filling function was used to fill the holes in the leaf image. These holes were appeared in some images because of dust, insects, or scanner uncleanness. Finally, a dilating filter was used to make the edge smoother as often by thresholding an uneven edge is produced which causes error in the measurements. The filter kernel used is as follows which is a small disk with the radius of one pixel:

$$k = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$



*Figure 3.3. Measurement of geometrical properties of leaves; a) the scanned leaf, b) the processed image.*

### **3.4 Active stereo system**

#### **3.4.1 Temperature and humidity sensor**

DHT22 sensor (Aosong Electronics Co., China) was used to measure temperature and humidity (Figure 3.4). DHT22 humidity sensor is a good quality sensor for measuring temperature and humidity. This sensor consists of two sensors, a capacitive humidity sensor and a heat resistor. There is also a very simple chip inside that converts analog to digital and produces digital output. The sensor specifications are presented in Table 3-2.



*Figure 3.4. The temperature and moisture sensor.*

**Table 3-2.** Technical specifications of DHT22 temperature and humidity sensor.

Required power	3.3-6 V, DC
Output signal	Digital signal with single port
Sensing element	Polymer capacitor
Operating range	humidity 0 – 100% RH, temperature -40 – 80 °C
Accuracy	Humidity $\pm 2\%$ RH, temperature $\pm 0.5$ °C
Resolution	Humidity $\pm 0.1\%$ RH, temperature $\pm 0.1$ °C
Sensing period	The average is 2 seconds
Dimensions	14 × 18 × 5.5 mm

### 3.4.2 Light sensor

The GY302 sensor (ROHM, China) was used to measure the intensity of light received daily during the growth stages. The sensor was placed inside a protective box to protect it from impact and dust (Figure 3.5). The GY-302 BH1750 luxmeter module is based on the BH1750 chip, which supports the I2C communication protocol, eliminating the need for an external converter such as an analog-to-digital converter. This sensor module can measure a wide range of light. Other technical specifications of the sensor are presented in Table 3-3.



*Figure 3.5. The light sensor.*

**Table 3-3.** Technical Specifications of GY302 Light Intensity Sensor.

Input voltage	3 – 5 V
Operating temperature range	-40 – 85 °C
Weight	2 g
Dimensions	2.3 × 14.4 × 19.5 mm
Working current	120 μA
communication protocol	I2C
Lighting range	0 – 65535 Lux

### 3.4.3 Light Projector

For light stripe, a video projector was used (NEC NP215 Projector, Sharp NEC Display Solutions, Ltd. Tokyo, Japan). The detailed information of the projector have been presented in Table 3-4. The use of laser causes problems that by video projector can be altered. The laser calibration is quite difficult and the laser line reflection on the plant leaves and environment also makes unwanted light points in the images. [Morris \(2006\)](#) used a laser for development of an active stereo 3D Scene reconstruction system. For reducing the laser intensity and power, they developed filters to be put in front of the laser projector. All modifications of light stripe are possible by video projectors including the quality of light, the speed of line travelling and thickness of the light stripe. The light stripe was projected on the field and travels until the end of the field. Then for the next measurement, the video again was played. [Weisenfeld \(2002\)](#) used laser for the projection of light stripe for active stereo measurement which

was equipped with a motor for motion control. [Teutsch et al. \(2005\)](#) proposed laser for the active vision for its intensity and reliability.



*Figure 3.6. Light projector used for active stereo imaging.*

**Table 3-4.** Technical specifications of the projector.

Brightness	2500 ANSI Lumens
Resolution	1024 × 768
Aspect Ratio	4:3 (XGA)
Contrast	2000:1
Display Type	1 × 1 cm DLP
Color Processing	8-bit
Video Modes	720p, 1080i, 576i, 576p, 480p, 480i
Lamp Type	180W UHP bulb
Lamp Life	3500 hours – 5000 hours
Lamp Model	60002853 / NP18LP
Included Lens	1.1x manual zoom , manual focus
Image Size	76 cm – 762 cm
Projector Size	9 cm × 31 cm × 25 cm
Weight	2.5 kg
Power	242 Watts, 100V – 240 V

#### 3.4.4 Micro-servomotor

In this work, the use of a light stripe for light projection on the plants was examined. For moving and swiping light on the plants with a good accuracy, a light micro-servomotor (MG90S, TIANKONGRC, China) was used. This micro-servomotor is a small servomotor with tiny gears having the torque of 2.2 K and input voltage of 4.8 V. The controlling signal is PWM. Also, it has the ability of circulating for 180°.



*Figure 3.7. Micro-servomotor used for displacement of the laser.*

### 3.4.5 Camera

Two webcams (LogiTech C525) made by LogiTech, China (Figure 3.8) were used. Cameras had a resolution of 8 megapixels. These cameras have a USB port and can be easily connected to a computer. The image size of this camera is  $1280 \times 720$  pixels. The technical specifications of the cameras are presented in Table 3-5. In previous research usually high sensitivity monochrome cameras have been used for active stereo imaging. Weisenfeld (2002) utilized two monochrome charge coupled devices for the development of an active stereo system for surface measurement.



*Figure 3.8. The LogiTech webcam.*

**Table 3-5.** Technical specifications of the cameras.

Camera model	HD Webcam C525
Manufacturer	LogiTech, China
Connection type	USB
USB Type	USB 2.0
Lens type	plastic sensor
Resolution	2 MP, real; 8 MP, software
Viewing angle	69°
Imaging (SD 4: 3)	240 × 320, 480 × 640, 2 MP, 8 MP

Imaging (W 16: 9)	360p, 480p, 720p
Video recording (SD 4: 3)	240 × 320, 480 × 640, MP 2
Video recording (W 16: 9)	360p, 480p, 720p
fps rate	30 Fps
Camera width	40.4 mm
Length	68.5 mm
Height	31.75 mm
Weight	88 g

### 3.4.6 Electronic unit

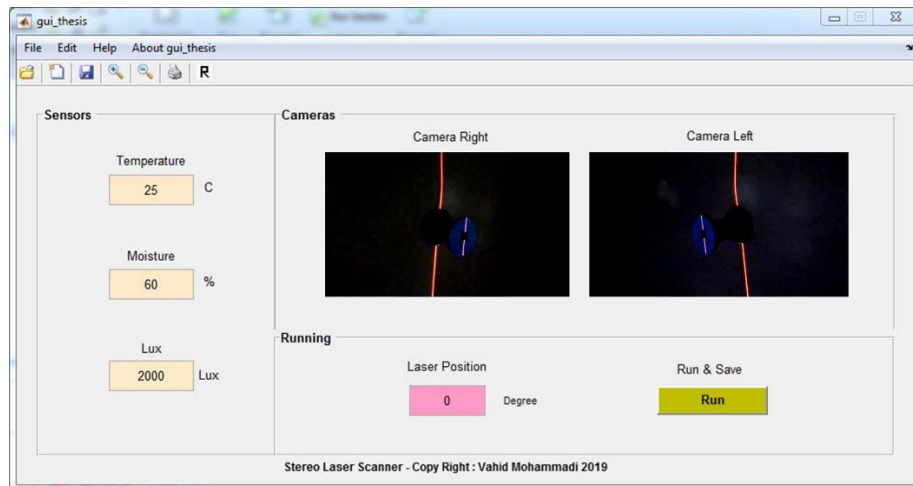
An electronic circuit was built to control the movement of the servo motor and the laser projection on the plant. The circuit takes command from a computer, turns on the laser, and shines it on the plant. After irradiation, the laser turns off and returns to its original state. The circuit also sends temperature, humidity and light intensity information to the computer for daily recording. This unit has parts such as ATmega8 microcontroller, ttl to USB cable, servo motor, temperature sensor (DHT22, China), humidity sensor (DHT22, China) and light intensity sensor (GY-302, China).



*Figure 3.9. The controlling circuit of sensors.*

### 3.4.7 Graphic interface

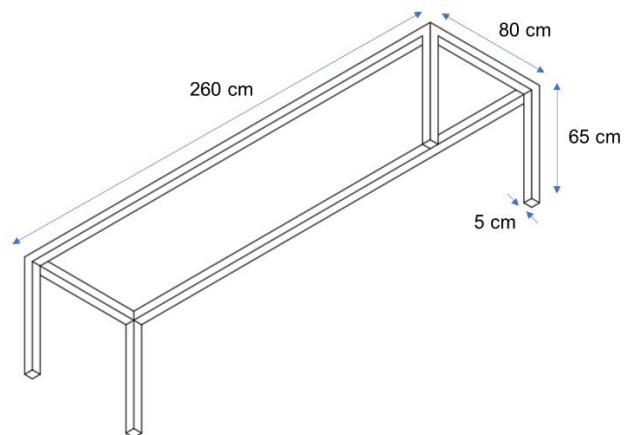
A graphical interface was created in MATLAB software to record videos and light projection on the plant. This interface helps the user to easily view and record information. Therefore, for each experiment, temperature, humidity and light intensity sensor information was recorded at the same time as the video. By pressing the Run key, the engine is started and the laser is projected on the plant. At the same time, the cameras start filming and at the end, the videos are saved.



*Figure 3.10. Graphical interface developed for measurements.*

### 3.4.8 System

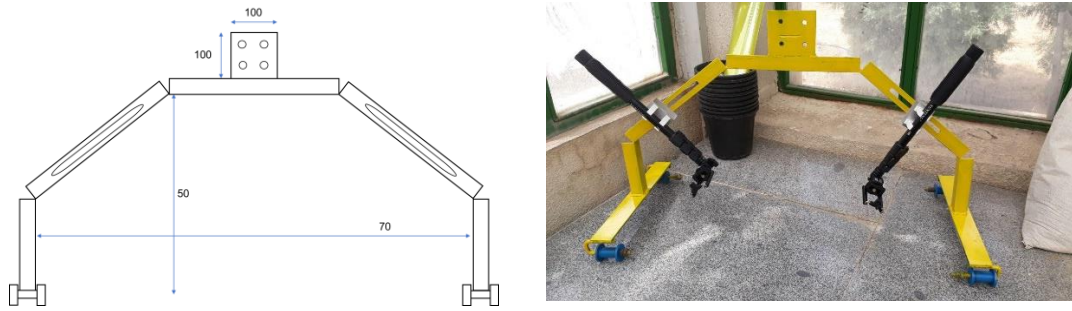
A chassis was built for putting the system on it and move all the system for different measurements. Figure 3.11 shows this chassis with the dimensions determined for imaging of pepper plants in early ages.



*Figure 3.11. Chassis developed for holding the imaging system.*

Next, a holder was designed and developed for holding the cameras and the projector. This holder needed to provide the ability to hold the cameras in different angles. In this case a small track was cut from the body of it that could provide the possibility to change the angle and baseline of left and right cameras.





*Figure 3.12. Developed holder for the cameras and the projector.*

To place the cameras on the system and easily adjust their positions, two telescopic hands were used. These arms could get longer three times more than the primary condition. There are fixtures on the arms that help to maintain the size for further measurements. Also, the top part of the hand is equipped with a camera holder that maintains the camera during the measurements.



*Figure 3.13. Telescopic hand for holding the cameras.*

The stereo imaging system consists of a projector and two cameras (Figure 3.14). The video projector was placed vertically on top of the plant. The videos including a light stripe were prepared and played during the measurements. Cameras and electronic units are connected to the computer via USB. The camera holder was made so that the angle of the cameras relative to each other and the distance of the cameras to the plant can be adjusted. The camera and video projector were placed on a rail to move at fixed and adjusted angles to produce images of other plants.

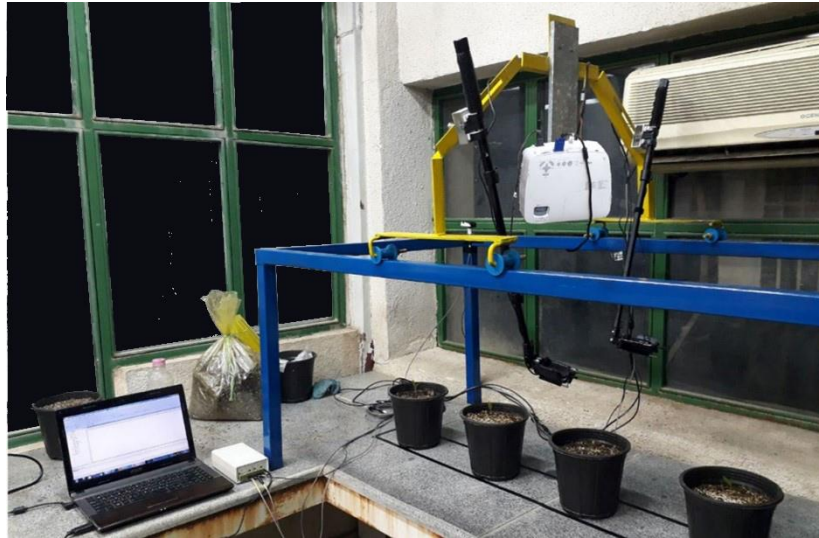


Figure 3.14. The imaging and measurement set-up.

Figure 3.15 provides the big picture of the system with different units and processing stages. The stereo system that takes two images at a time, needs to be calibrated and process the images by matching them and extraction of features. In the following sections, the processing stages are presented.

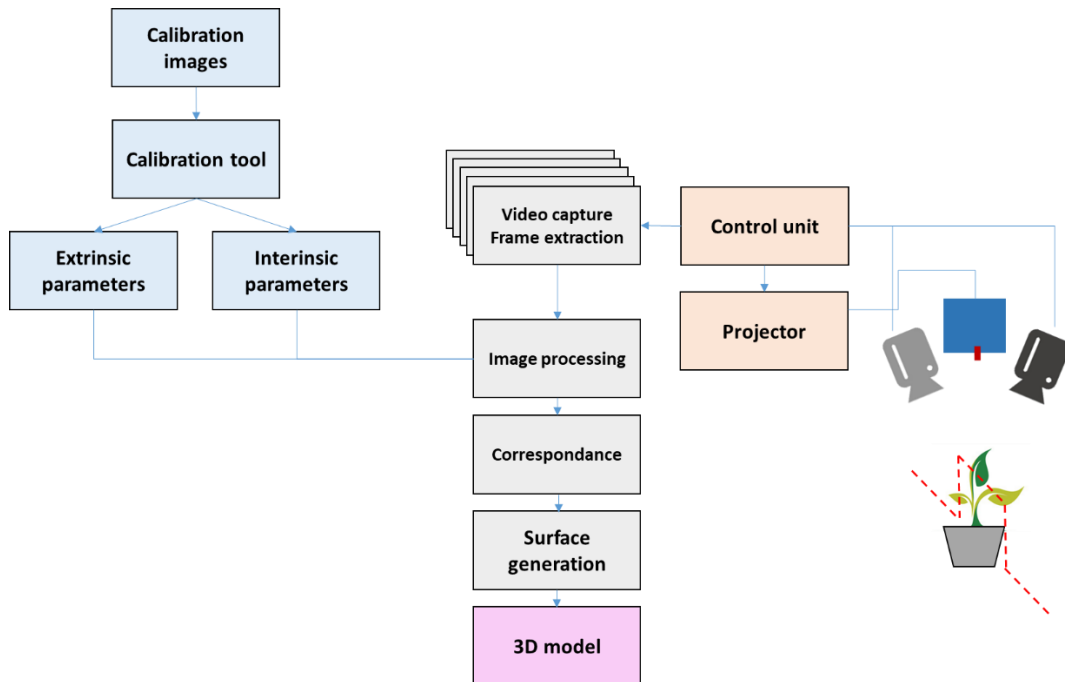


Figure 3.15. Block diagram of the stereo imaging system.

### 3.4.9 Positioning of the cameras

The cameras were symmetrically positioned on handles that provided the ability to adjust the angle between them. Figure below schematically illustrates how the cameras see a common area for taking images of the objects. The area that each camera sees is called “field of view.” Placement of the cameras on the holder should be in the way that they cover the region of interest for imaging including the object or plant.

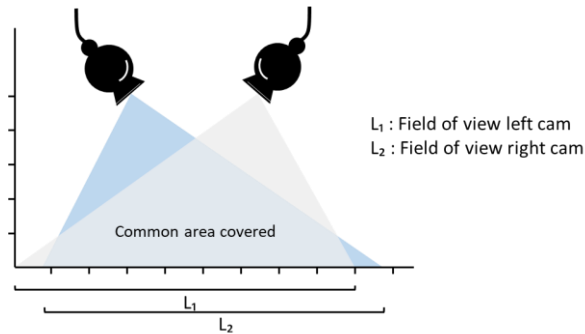


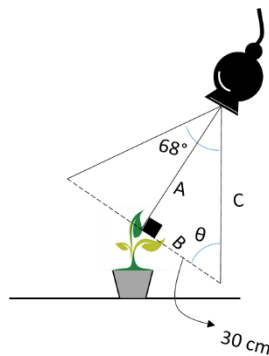
Figure 3.16. Field of view and common area of cameras.

As the FoV (Field of View) of the utilized cameras is 68 degrees, and an area of interest was about  $60 \times 60 \text{ cm}^2$ , based on the geometrical relation represented in figure below, a distance of about 44.477 cm was achieved. Considering the dimension of 60 cm,  $B$  is achieved as 30 cm.  $A$  is obtained from tangent of the angle  $\theta$  which equals to  $56^\circ$ . And  $C$  is calculated based on Pythagorean theorem which is equal to 53.65. Hence this amount was considered as vertical distance of the cameras with the plant.

$$\tan(\theta) = \frac{A}{B}$$

$$A = B * \tan(56) = 44.477 \text{ cm}$$

$$C = \sqrt{30^2 + 44.477^2} = 53.65 \text{ cm}$$



*Figure 3.17. Calculation of camera distance from the plant.*

It is also possible to take several images of the sample objects from different angles and points of view. This technique can help for the good positioning of the cameras. The cameras will be fixed and the object is put on a platform for being taken photos in different positions and distances (Trujillo Romero, 2008).

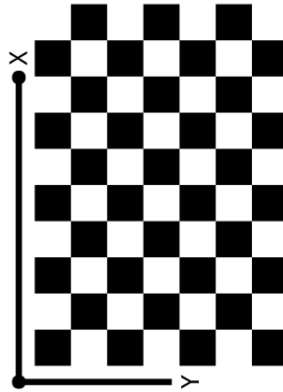
#### **3.4.10 Calibration**

For calibration of the stereo camera and extraction of internal and external parameters, the stereo calibrator toolbox of MATLAB (Mathworks Inc., the US, 2019b) was used. Calibration process of the stereo camera is done as follows:

- Preparation of images, cameras, and checkerboard pattern
- Uploading the images into the calibrator
- Calibration
- Measurement of accuracy of calibration
- Adjusting parameters for improvement of accuracy
- Exporting internal and external parameters to the code

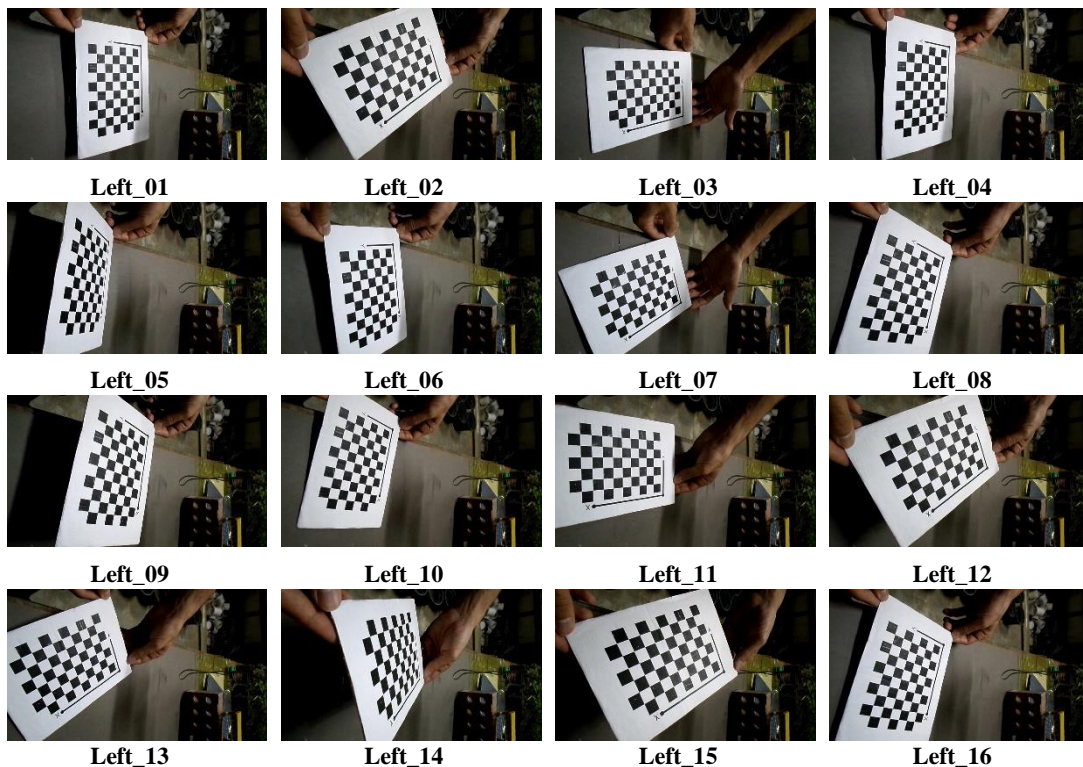
#### **3.4.11 Preparation of checkerboard pattern**

The standard checkerboard pattern provided by MATLAB was used in this study. The pattern must have black and white squares but the whole pattern must be rectangular. One side of the pattern must have an odd number of squares and another side an even number of those. This pattern was printed and glued on a cartoon plate (Fig. 3.18). No other signs and lines must be on the pattern.



*Figure 3.18. The standard checkerboard used for calibration.*

The squares on the printed pattern were measured that amounted to 16.5 mm. This size was used for the calibration in the calibrator application. By positioning the pattern in different locations, more than 50 pairs of images were obtained (Fig. 3.19). The images were taken in a way that the whole pattern was visible in the image. No processing was done on the images before presenting them to the calibrator.



*Figure 3.19. Sample images used for calibration taken with the left camera.*

### 3.4.12 Evaluation of calibration accuracy

First the calibrator application presents the reprojection errors. The reprojection errors are represented in distances (i.e. in pixels) between the detected points and the reprojected points. The calibrator application of MATLAB calculates the reprojection errors by projecting the checkerboard points from world coordinates, defined by the checkerboard, into image coordinates. This app compares the reprojected points with the corresponding points detected in the calibration images. Figure 3.20 shows how the reprojection error is presented.

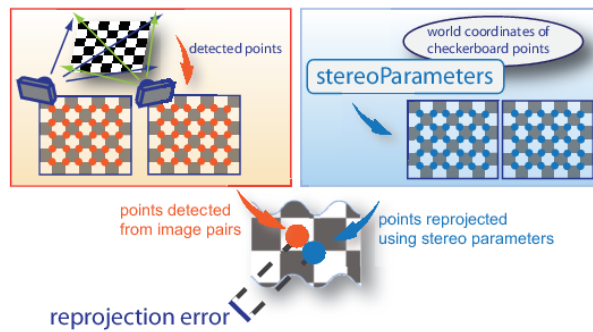


Figure 3.20. The error measurement of reprojection in calibration of stereo cameras.

Next, the mean reprojection error per image is provided in a bar graph, along with the overall mean error. This graph shows the proper pairs or the pairs that can be removed also.

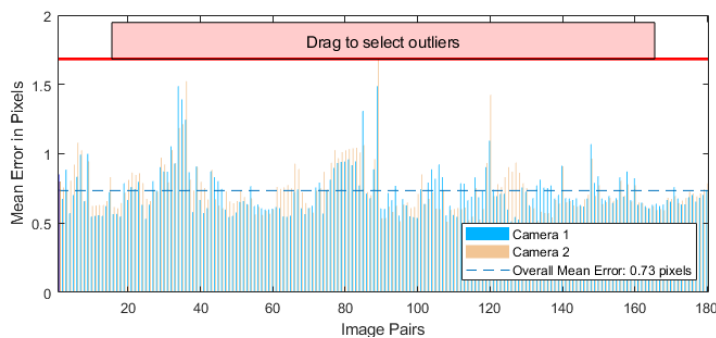


Figure 3.21. Bar chart of accuracy measurement of calibrated stereo cameras.

The calibrator app also provides a camera-centric view of the patterns and a pattern-centric view of the camera. These are presented as the 3-D extrinsic parameters of calibration. This graph is helpful to check the relative positions of cameras and patterns used for calibration. For example, a pattern that appears behind the camera indicates a calibration error.

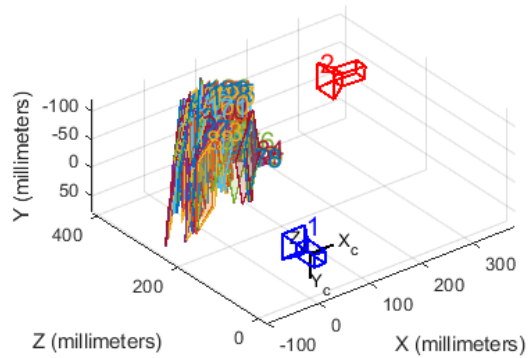


Figure 3.22. Spatial position of stereo cameras based on calibration and position of checkerboard.

### 3.4.13 Taking active stereo images

The light stripe was projected to the environment and simultaneously the right and left videos of light travelling were recorded and saved. All frames of the videos were extracted and used for processing. In this regards, first left frame corresponds to the first right frame meaning that they need to be treated at the same time. Figure 3.23 illustrates two sample pairs of left and right images for a test sample and a plant.

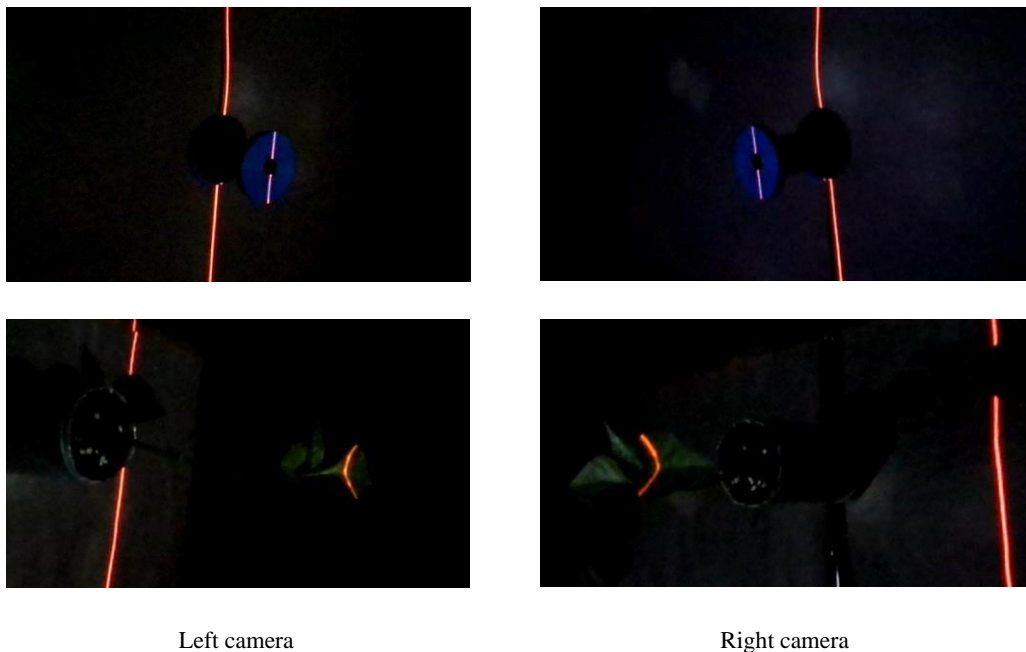
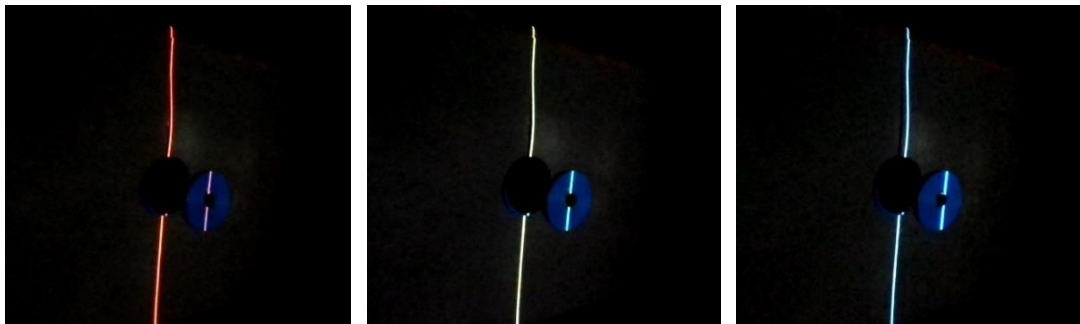


Figure 3.23. Sample images of the left and right cameras; Top) a test object and bottom) a plant.

### 3.4.14 Light strip color

In previous research, red light strip has been used (Beraldin et al., 2003; Morris, 2006; Tu et al., 2019). Weisenfeld (2002) used a red laser for intraoperative surface measurements using active stereo imaging. Of course, the main reason was the accessibility of red lasers. In the present study, due to the use of a video projector for light radiation, the ability to test other colors was also examined. For this purpose three colors literally red, green, and blue were evaluated. The light strips were made as video files to be executed by the computer and the desired light was projected on the environment by the projector.

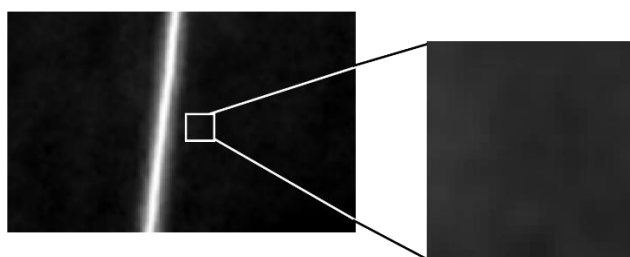


*Figure 3.24. Different colors of light stripe (From left: red, green, blue).*

#### **3.4.15 Noise removal**

Noise is a routine guest of digital images. Noise may occur because of different reasons including image sensor, light conditions or environmental causes. In active stereo imaging, normally a grain noise happens in the images as a result of scattering the light on the environment (Fig. 3.25). The presence of grainy noise specially in uniform parts of the image is more probable. This brings small white or light grey pixels in the image that may cause error in correspondence problem. This noise was removed by applying an averaging filter as the averaging filter weakens these pixels to lower values close to black ones. Also, a good way to minimize noise is to use Gaussian blur/smooth. However, it may cause the light stripe intensity to drop or the peak location to move (Isa and Lazoglu, 2017). Another way to reduce the amount of this noise is to adjust the intensity of light in the light stripe. In case of using lasers, it is needed to adjust the power supply and in case of using projectors, it is possible to adjust the light intensity in the videos or projector settings.

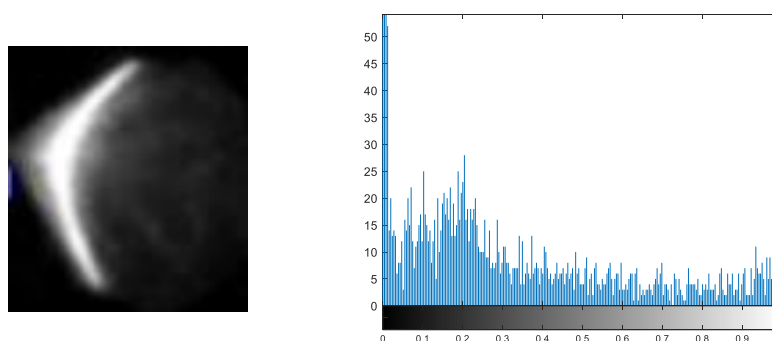




*Figure 3.25. Noise in the form of light grains on the images because of scattering of light on the surrounding.*

### **3.4.16 Light stripe peak**

It is noted that the light stripe on the leaves is not also a pure line. Fig. 3.26 shows the region of the leaf that light strip has passed on it and its histogram representing the pixels with different levels of intensity. This light scattering on the leaves is inevitable. In this regard, by thresholding the images, the bright part of the line is selected and used for further processing. Other pixels were turned to black. The application of four light peak detection techniques literally Gaussian, Center of Mass, Parabolic and Linear Interpolation was evaluated. These techniques help to remove the reflections of the light line and keep the major part of the light and lead to a narrow light line.



*Figure 3.26. Histogram of the region including the light stripe on plant leaf.*

### **3.4.17 Peak detection and extraction**

Several techniques were tested for light peak detection (i.e. Gaussian, Center of Mass, Linear Interpolation and Parabolic techniques). Each technique boosts the peak pixels values and degrades the others. However, the behavior of them brings different results on the images.

### **3.4.18 Camera angle**

The best angle between the right and left cameras is reported to be in right angle. This is due to the greater ease of calibration as well as providing a better field of view of the environment. However, to measure and test this, three angles of 75, 90 and 105 degrees were evaluated (Figure 3.27). The angle is considered as the angle between cameras toward the center of the object. The cameras were positioned and imaging was done for a simple plate with the predefined dimensions (i.e. 3 cm × 3 cm) (Figure 3.28). This plate was imaged in three different angles of 30, 60 and 90 degrees.

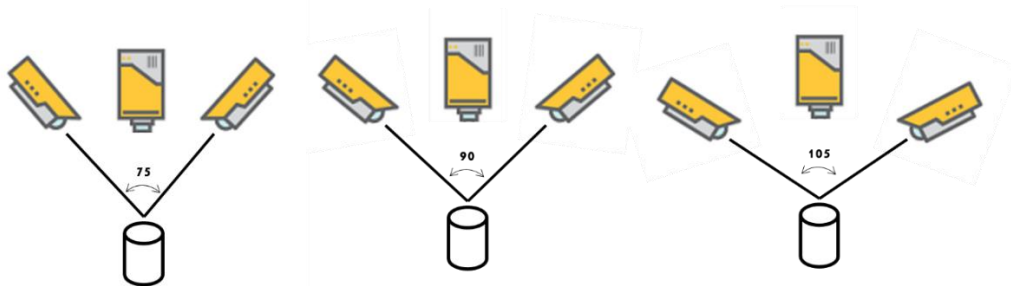


Figure 3.27. Positioning of the cameras in relation to each other.

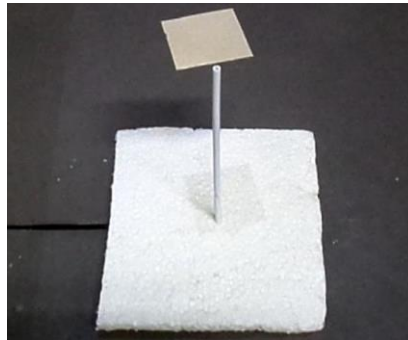


Figure 3.28. The usage of a simple plane for evaluation of camera angle.

### 3.4.19 Processing

Processing could be divided into five main steps. The first task was to remove the light stripes and spots everywhere except on the plant. To do so, the program was written to read the images from the first pixel and detect the non-zero pixels. For the left image, always the light stripe on the ground will be before the plant (Fig. 3.29). This idea was used to detect these lines and the plant and then keep the spots that are on the plants. Fig. 3.29 represents the area including the light stripe that is going to be removed. For the right image, the process is the same except that the light stripe on the

ground will be on the right side. Accordingly the image is read from the right side and detects the light stripe and removes it. Then the spots of light on the image will be the light spots on the plant leaves.

It is noted that these parts of the images should not be cut and the pixels should be replaced with black pixels. This is due to the calibration information that is connected to the size of images and changing the images' sizes will lead to mistakes in calculations and 3D developments.

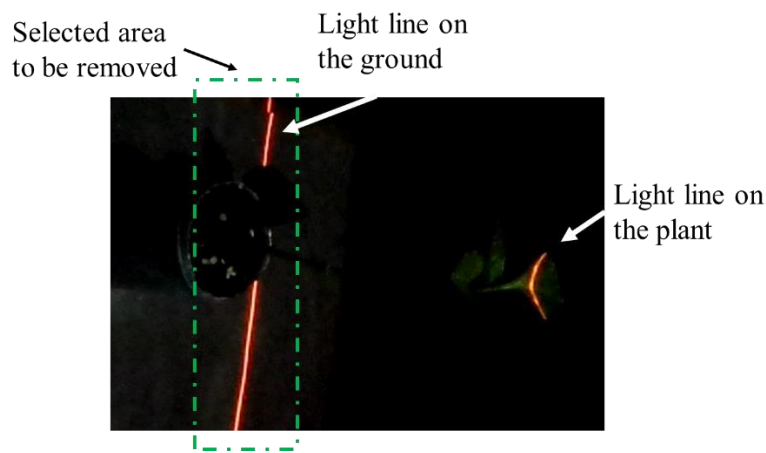


Figure 3.29. Removing the light stripe on the ground from the left image.

### 3.4.20 Stereo Image Rectification

For rectification, the traditional methods were tested. In all cases, the rectification was done very slowly and with huge computational costs including several hours of calculations. In some cases, the rectification could not be done as the matrices grow so big and out of memory of the PCs. In this regard, the rectification method presented by [Su and He \(2011\)](#) based on a geometric technique was employed.

#### *Camera model*

This method also uses the pinhole camera model describing the image position of a 3D point in the world by a pinhole view of the camera (Fig. 30). Let the world point be  $P_w(x, y, z)$  and image point be  $p_p(u, v)$ , the image point  $p_p$  in the image coordinate

system  $p_l(x_l, y_l)$  and in the camera coordinate system  $p_c(x_c, y_c, z_c)$  is obtained according to geometry.

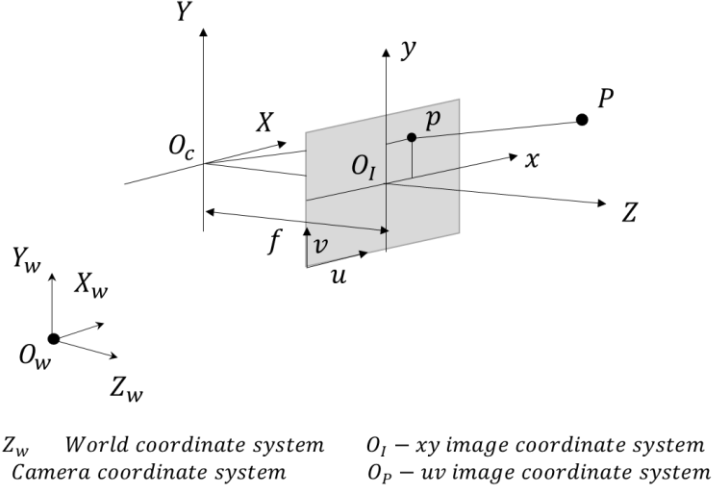


Figure 3.30. Stereo camera pinhole model.

$$\begin{cases} x_1 = \frac{u - u_0}{k_u} \\ y_1 = \frac{v - v_0}{k_v} \end{cases} \quad (1)$$

$$\begin{cases} x_c = f \cdot \frac{u - u_0}{f_u} \\ y_c = f \cdot \frac{v - v_0}{f_v} \\ z_c = f \end{cases} \quad (2)$$

where  $f$  is physical focal length of camera,  $k_u$  and  $k_v$  are the scale factors of the  $u$  and  $v$  axes,  $f_u$  and  $f_v$  are the product values of  $f$  and  $k_u, k_v$ ,  $(u_0, v_0)$  is the coordinates of the principal point.

Next, the cameras have been calibrated, then the intrinsic parameters and poses of the cameras are known. The pose of the right camera in relation to the left one and the intrinsic parameters are used as follows (Su and He 2011):

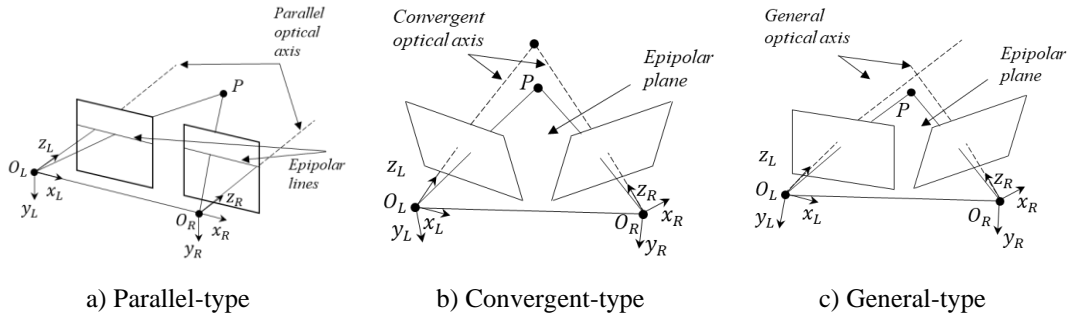
$${}^L F_L = [{}^L F_R \quad {}^L F_R] \quad (3)$$

$$K_L = \begin{bmatrix} f_{ul} & 0 & u_{0l} \\ 0 & f_{vl} & v_{0l} \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

$$K_R = \begin{bmatrix} f_{ur} & 0 & u_{0r} \\ 0 & f_{vr} & v_{0r} \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

*The stereo system*

As the figure below illustrates, the configuration of the two cameras can be divided into three categories literally parallel-type, convergent-type and general-type. In the parallel type the intrinsic parameters are the same and the cameras are on a line. However, there is the difficulty to make the system accurately. Convergent-type requires that the optical axes of two cameras should intersect in finite distance. The general type of configuration is totally unconstrained and in practice this is the common form.



*Figure 3.31. The structure of the stereo vision system.*

In Fig. 3.32, the optical centers of two cameras ( $O_L$ ,  $O_R$ ) and the world point  $P$  constitute the epipolar plane. The intersection of the epipolar plane and the camera's image plane is called epipolar line. Baseline is the line  $O_L O_R$  between the cameras' optical centers, and its intersection with the image plane is defined as epipole. When the projective point in one image plane of a world point is given, its corresponding point on the other image plane is found along the corresponding epipolar line, which called epipolar geometry constraint. Since the epipolar lines are not collinear in the general-type and convergent-type of stereo vision system (Fig. 3.31b and Fig. 3.31c), it needs substantial computation while using epipolar lines to search corresponding points.

To ensure the matching efficiency in convergent-type and general-type consistent with the parallel-type, we need to transform these two types into virtual parallel-type (Fig. 3.32), which is called stereo rectification.

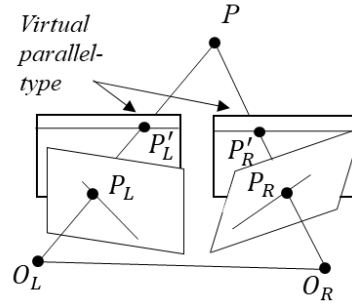


Figure 3.32. The virtual parallel system.

#### Rectification of stereo image pair

First, the original image points are converted to the coordinate system of the virtual parallel-type and then the image points which have been transformed to new coordinate system are mapped to the image plane of the virtual parallel-type by re-projecting. As we know the  $x$  axes of the two cameras coordinate systems coincide with baseline and toward to the same direction in parallel-type (Fig. 3.31a), therefore the first step of stereo rectification is to compute the transformation matrix which transforms the two cameras coordinate systems to parallel-type.

Assume left camera coordinate system is  $L(O_L-x_Ly_Lz_L)$ , the baseline is  $O_LO_R$ . And the objective is to obtain the transformation matrix of transforming the coordinate system  $L(O_L-x_Ly_Lz_L)$  to the new coordinate system  $L'(O_L-x'_Ly'_Lz'_L)$  whose  $x$  axis coincides with the baseline  $O_LO_R$ . First let the world coordinate system coincide with the coordinate system  $L$  and the coordinate of point  $O_R$  in the world coordinate system is  $O_R(t_1, t_2, t_3)$  which have been gained from  $LTR$ . And then according to geometry, the conversion from the left camera coordinate system  $L$  to new coordinate system  $L'$  is realized through rotating  $\alpha$  around  $y$  axis of the world coordinate system and then rotating  $\beta$  around  $z$  axis of the world coordinate system (Fig. 3.33).

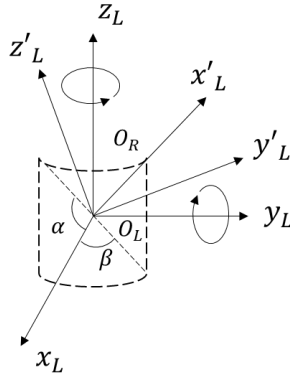


Figure 3.33. Transformation of coordinate system.

Then according to Niku (2001), the transformation matrix of the coordinate system  $L'$  relative to the coordinate system  $L$  is obtained:

$${}^L F_{L'} = R(z, \beta) \cdot R(y, \alpha) \quad (6)$$

where

$$R(y, \alpha) = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix} \quad (7)$$

$$R(z, \beta) = \begin{bmatrix} \cos\beta & -\sin\beta & 0 \\ \sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (8)$$

$\alpha$  and  $\beta$  is estimated by the coordinate of point  $O_R$ :

$$\sin\alpha = \frac{t_3}{\sqrt{t_1^2 + t_2^2 + t_3^2}} \quad (9)$$

$$\sin\beta = \frac{t_2}{\sqrt{t_1^2 + t_2^2}} \quad (10)$$

In practice, the destination is to get the coordinate of a point after being transformed from the coordinate system  $L$  to the coordinate system  $L'$  (Fig. 3.33), so the required transformation matrix is about the coordinate system  $L$  relative to the coordinate system  $L'$ . According to (6), the transformation matrix is obtained:

$${}^L F_L = {}^L F_L^{-1} = R^{-1}(y \ \alpha), R^{-1}(z \ \beta) = \begin{bmatrix} \cos\alpha \sin\beta & \cos\alpha \sin\beta & -\sin\alpha \\ -\sin\beta & \cos\beta & 0 \\ \sin\alpha \cos\beta & \sin\alpha \sin\beta & \cos\alpha \end{bmatrix} \quad (11)$$

Since cameras have been calibrated, the rotation transformation of right camera relative to left camera  ${}^L R_R$  is already known. Then the transformation matrix of right camera coordinate system is calculated with (11).

$${}^R F_R = {}^L F_L {}^L R_R \quad (12)$$

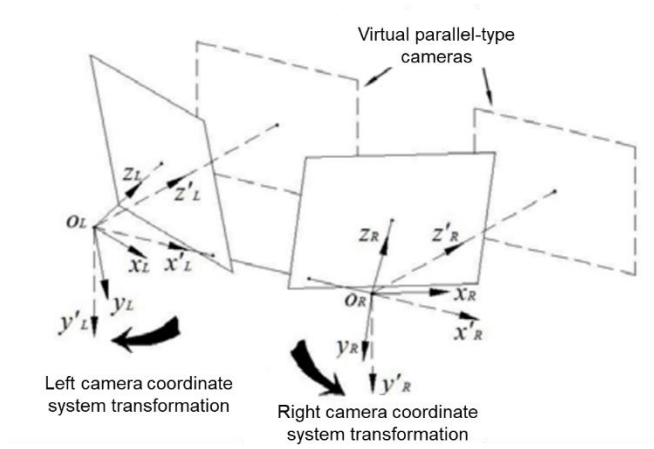


Figure 3.34. Schematic of the coordinate systems transformation.

Then the reprojection of points are done. To do so, first the coordinates of those points in each old camera coordinate system are estimated. Assume there is a left image point  $P_{pl}(u_l, v_l)$ , and then the coordinate of it in old left camera coordinate system  $P_{cl}(x_{cl}, y_{cl}, z_{cl})$  is obtained with (2).

$$\begin{cases} x_{cl} = f_l(u_l - u_{0l})/f_{ul} \\ y_{cl} = f_l(v_l - v_{0l})/f_{vl} \\ z_{cl} = f_l \end{cases} \quad (13)$$

Now the coordinate of the image point in new left camera coordinate system  $P'_{cl}(x'_{cl}, y'_{cl}, z'_{cl})$  is obtained through the transformation matrix which has been obtained.

$$P'_{cl} = {}^L F_L P_{cl} \quad (14)$$



As the same, a right image point in old right camera coordinate system is  $P_{cR}(x_{cr}, y_{cr}, z_{cr})$ , and its coordinate in new right camera coordinate system  $P'_{cR}(x'_{cR}, y'_{cR}, z'_{cR})$  is obtained.

$$\begin{cases} x_{cr} = f_r(u_r - u_{0r})/f_{ur} \\ y_{cr} = f_r(v_r - v_{0r})/f_{vr} \\ z_{cr} = f_r \end{cases} \quad (15)$$

$$P'_{cR} = {}^R F_R P_{cR} \quad (16)$$

After getting the coordinates of the image points in new coordinate system, we need to re-project them to the image plane of virtual camera to complete the rectification. As it shows in Fig. 3.35,  $P_{pl}(u_l, v_l)$  is an image point of the original left camera, and the coordinate of it in the new left camera coordinate system  $P'_{cL}(x'_{cl}, y'_{cl}, z'_{cl})$  is obtained after the above-mentioned method. Let the focal length of cameras be  $f_n$  and the principal point be  $(u'_0, v'_0)$  in parallel-type system, then the projection point of  $P_{pl}$  in the image plane of left virtual camera  $P'_{pl}$  is the intersection of line  $O_L P'_{cL}$  and plane  $z=f_n$ . According to mathematics, the equation of the line  $O_L P'_{cL}$  is as follow.

$$\frac{x}{x'_{cl}} = \frac{y}{y'_{cl}} = \frac{z}{z'_{cl}} \quad (17)$$

And the intersection point  $P'_{nL}(x'_{nl}, y'_{nl}, z'_{nl})$  is

$$\begin{cases} x'_{nl} = f_n \cdot x'_{cl}/z'_{cl} \\ y'_{nl} = f_n \cdot y'_{cl}/z'_{cl} \\ z_{nr} = f_n \end{cases} \quad (18)$$

Projecting the point  $P'_{nL}(x'_{nl}, y'_{nl}, z'_{nl})$  to the image plane of left virtual camera is executed with (1), and the coordinate of it in pixel coordinate system  $P'_{pl}(u'_l, v'_l)$  is as follow.

$$\begin{cases} u'_l = k_{nu} \cdot x'_{nl} + u'_{0l} = f_{nu} \cdot \frac{x'_{cl}}{z'_{cl}} + u'_{0l} \\ v'_l = k_{nv} \cdot y'_{nl} + v'_{0l} = f_{nv} \cdot \frac{y'_{cl}}{z'_{cl}} + v'_{0l} \end{cases} \quad (19)$$

In right original image, the corresponding image point of  $P_{pl}(u_l, v_l)$  is  $P_{pr}(u_r, v_r)$ , and its coordinate in right camera coordinate system is  $P'_{cR}(x'_{cr}, y'_{cr}, z'_{cr})$  which have

been obtained in (16). The same with left re-projection, the projection point of  $P_{pr}$  in the image plane of right virtual camera  $P'_{pr}$  is the intersection of line  $O_R P'_{cr}$  and plane  $z=f_n$ . the equation of the line  $O_R P'_{cr}$  is as follow.

$$\frac{x}{x'_{cr}} = \frac{y}{y'_{cr}} = \frac{z}{z'_{cr}} \quad (20)$$

And the intersection point  $P'_{nr}(x'_{nr}, y'_{nr}, z'_{nr})$  is

$$\begin{cases} x'_{nr} = f_n \cdot x'_{cr} / z'_{cr} \\ y'_{nr} = f_n \cdot y'_{cr} / z'_{cr} \\ z'_{nr} = f_n \end{cases} \quad (21)$$

Then the coordinate of  $P_{pr}(u_r, v_r)$  is obtained from (1), the result is as follows:

$$\begin{cases} u'_r = f_{nu} \cdot \frac{x'_{cr}}{z'_{cr}} + u'_0 \\ v'_l = f_{nv} \cdot \frac{y'_{cr}}{z'_{cr}} + v'_0 \end{cases} \quad (22)$$

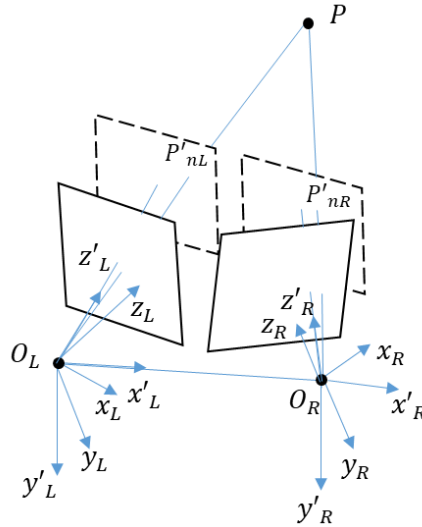


Figure 3.35. Schematic of the left camera's re-projection.

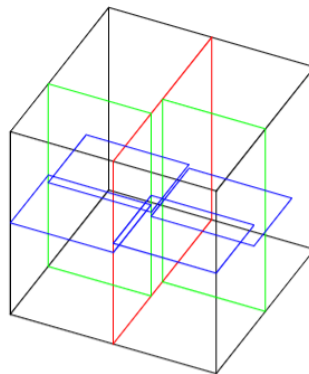
After re-projection, the original image information is transferred to the virtual parallel-type cameras' image plane, and then stereo matching is implemented rapidly as the rectification is finished.

### 3.4.21 Triangulation

The images are read line by line and the correspondent pixels are extracted. These pixels are saved in variables for finding the 3D coordinates. The triangulation technique and calibration information were used for 3D coordinate calculation. The 3D coordinates are then converted to a point cloud format (i.e. \*.ply format). This point cloud is the scattered points in the space. A list of datasets of 3D point clouds has been provided in Appendix D.

There are several methods for point cloud segmentation including edge-based methods, region-based methods, attribute-based methods, model-based methods, and graph-based methods. However, the choice of method depends on the application and complexity of shapes in the point cloud. In this work, the segmentation of the point cloud was done using the clustering of close points based on the Euclidean distance between them.

In the context of point clouds, Euclidean clustering processes points by searching for the closest points to a candidate point based on a distance threshold, then clustering points together as long as they fall within the threshold's range. The distance threshold is just the Euclidean distance  $d = \sqrt{x^2 + y^2 + z^2}$  between neighboring points, as the name implies. Constructing a  $k$ D-tree is a frequent approach to find a point's neighbors. A  $k$ D-tree is a data structure that divides the dimension space into  $k$  sub-dimensions and aids in the organization and localization of the closest points to a given position (Fig. 3.36). A minimum distance is determined and the points closer to each other less than this distance are considered as the same object (i.e. a leaf in this project).



*Figure 3.36. An example of k-d tree.*

The next step is to generate the mesh from the point cloud. Meshing the point cloud allows the production of surface and, consequently, the production of a three-dimensional model. The segmented point cloud was converted to a 3D surface using Delaunay's triangulation technique. This surface that consists of several objects is used for counting the leaves meaning that every object is considered as a leaf. The performance of the algorithm is presented below. [Sun et al. \(2002\)](#) used the combination of the space carving with Hilton's implicit surface-based method for 3D reconstruction.

The required specifications will then be extracted from the 3D model. To do this, first an  $\alpha$ -object ( $\alpha$ -shape) is generated that includes all points and then it is done using the triangulation method (i.e. Delaunay's triangulation technique). Suppose an  $S$  set of points is provided in 2D or 3D and the goal is to obtain something similar to the "shape produced by these points". This is a rather vague concept, and so there are many interpretations, one of which is the  $\alpha$ -object, a dense and unorganized dataset can be used to reconstruct the shape. In fact,  $\alpha$ -shape is defined by a boundary, which is a linear approximation of the original shape. Therefore, to produce the  $\alpha$ -object, circles are used so that the circumference of the circle coincides with the points and no points are inside the circles. Here  $\alpha$  is the radius of the circles.

- 1: Read the calibration images
- 2: Do calibration and extract calibration information
  - for i = 1 : number of images
    - 3: Read the first pair of images
    - 4: Apply smoothing filter and remove the holes
    - 5: Extract red layer of images
    - 6: Apply thresholding
    - 7: Keep light stripe peaks
    - 8: Remove the light stripes on the ground
    - 9: Rectify the images
      - for j = 1 : number of non-zero pixels
        - 10: Perform point matching
        - 11: Calculate the 3D coordinates
  - end
  - 12: Save the coordinates in a variable
  - end
- 13: Build the point cloud
- 14: Segment the point cloud into separated regions
- 15: Mesh the point cloud using triangulation
- 16: Develop the 3D model
- 17: Count plant leaves

## PASSIVE STEREO IMAGING

### 4.1.1 Acquisition system

A parallel stereo imaging system was developed to provide images right on the top of plants canopy. An 18 MP Canon cameras (EOS Kiss X5, Taiwan) was used for image acquisition. The camera for the second picture was moved to the right side of its rail. The sensor type of this camera is CMOS and the maximum resolution is 5184×3456. However, for this study the resolution of 3456×2304 was set. The central distance between the lenses was 8.7 cm in the right and left images. The camera was placed on a holder which was connected to a case. The case was mounted on a rail whereby the whole system could be moved for acquisition of other plants images. The distance between the lens face and the ground was 80 cm. All the image acquisition was taken in the afternoon and under daylight. This was due to two reasons; 1) to build a robust algorithm for real greenhouse situations and 2) this study did not aim at any color comparisons.

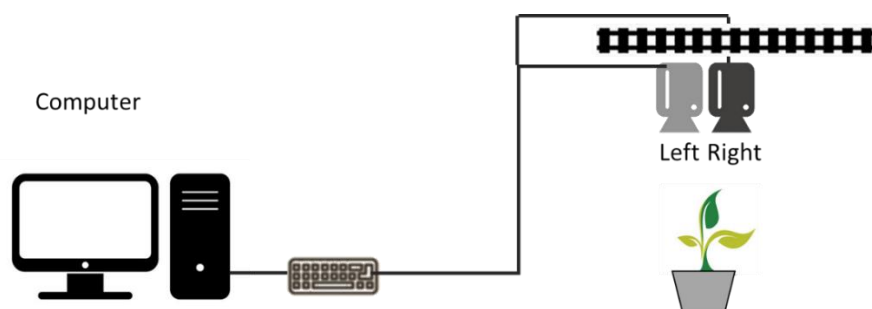
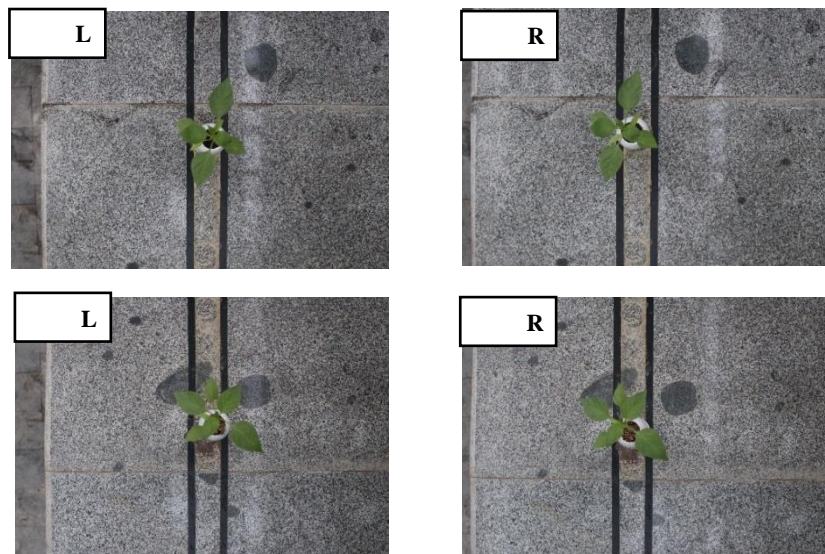


Figure 4.1. Schematic diagram of the stereo imaging system.

### 4.1.2 Stereo images

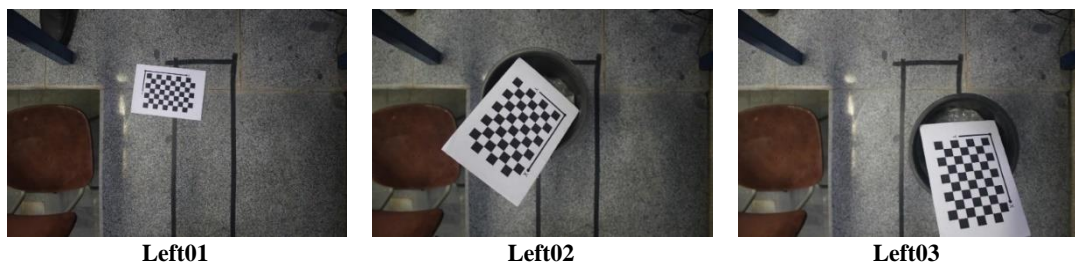
First the left image was taken and then the camera was moved to the right and the right image was taken. Figure 4.2 represents examples of left and right images. As it is seen, on the right and left borders of the images, the differences are clear that the cameras have seen the area with a small difference in point of view.

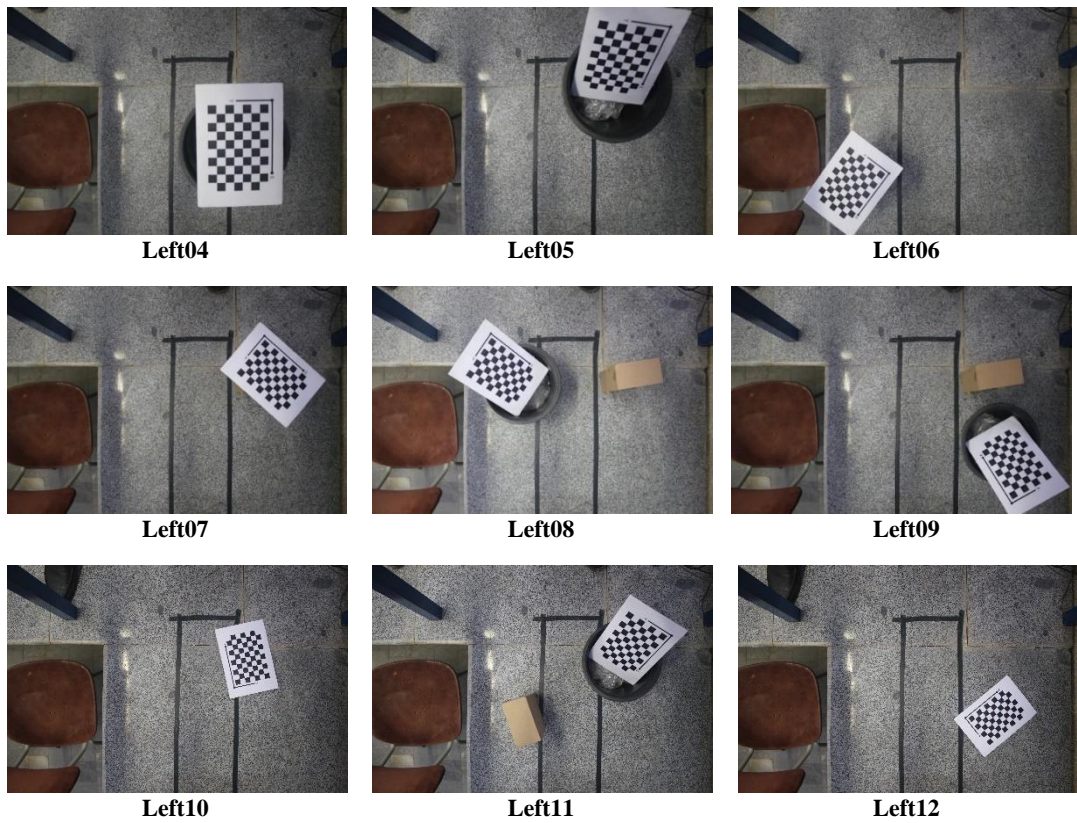


*Figure 4.2. Original images used in stereo imaging for the depth detection.*

### **4.1.3 Calibration**

The calibration was done based on the regulations described in the section 3.4.10 for active stereo calibration. 20 pairs of images were obtained from different points of view and positions in the fields of view of the cameras. The stereo calibration toolbox of MATLAB was used for the calibration. Figure 4.3 shows some of the calibration images used for passive stereo calibration.





*Figure 4.3. The sample images used for calibration.*

#### **4.1.4 Processing algorithm**

The developed algorithm included four stages: 1) calibration, 2) preprocessing, 3) segmentation and leaf detection, 4) calculations of physical properties, and 5) applying the distance scale. The functions performed by the algorithm are as follows:



1. Read a pair of images
2. Crop the images to the region of interest
3. Convert to HSV space
4. Extract the plant by hue thresholding
5. Filling the holes
6. Smoothing the broken edges
7. Apply entropy filter
8. Obtain gradient image
9. Calculate the watershed image
10. Obtain the boxes including each leaf or connected leaves
11. Apply the gradient information
  - if there is one leaf
  - 12. measure the properties of the leaf
  - elseif there are more than one leaf
  - 13. select the leaves based on watershed information
  - 14. measure the properties for each leaf
15. Calculate other physical properties related to the obtained information
16. Build the disparity map
17. Find the distances on the disparity map
18. Apply the scaling and estimate the real sizes

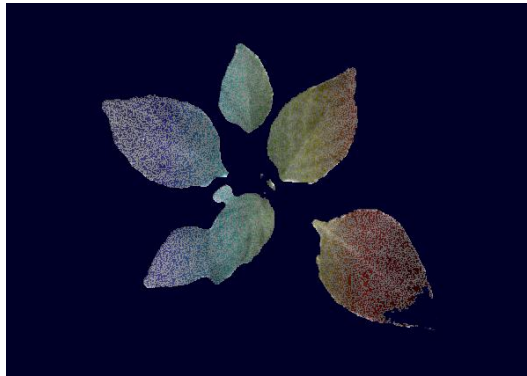
After cropping the region containing the plant, the image was converted to HSV color space. This conversion was very helpful for the thresholding of leaves and removal of the background and stems. Using trial and error, the values between 0.22 and 0.29 were chosen for thresholding. To prevent any small holes in binary images, a filling operation was performed (Fig. 4.4). These small holes that sometimes may not be clear in the image, can cause errors in the estimations. As thresholding may make edges uneven, using the opening function of MATLAB, edges were made smoother.



*Figure 4.4. The thresholded image and its result after filling the holes.*

In the next stage, entropy filter was applied and then the gradient magnitude was computed. The gradient is high at the borders and low inside the object. Next, the watershed image was obtained from the gradient image (Amandep 2014). This image was used for the detection of leaves and computing the properties (Figure 4.5).

The next step was to estimate the physical properties of the leaves. These are the properties mentioned in section 2.1. This estimation was done by extracting the shape of each leaf and using pixel-wise calculations. Afterwards, using the obtained disparity map, the distance of the leaves from the camera were estimated and used for the conversion of pixel values to centimeters.

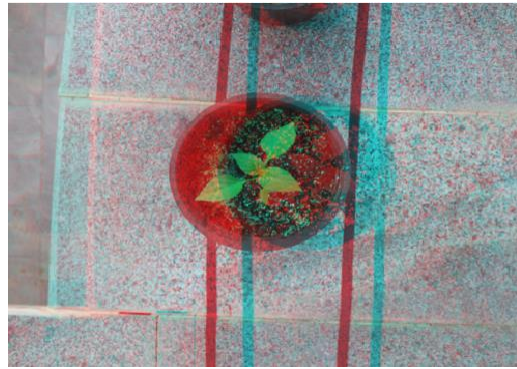


*Figure 4.5. Watershed image.*

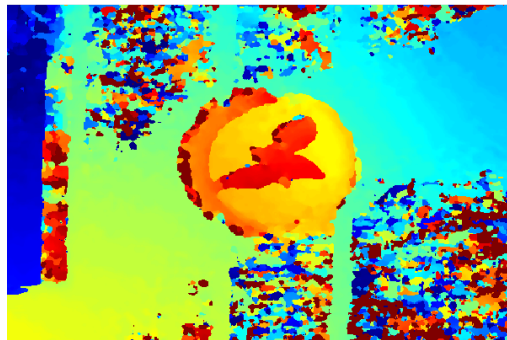
#### **4.1.5 Depth estimation and scaling**

The images were rectified based on the calibration data. This provided the undistorted and rectified left and right images. Fig. 4.6 illustrates the overlapped left and right images after rectification. Next, the disparity map was obtained by semi-global matching method with the block size of 7 pixels—the block size significantly

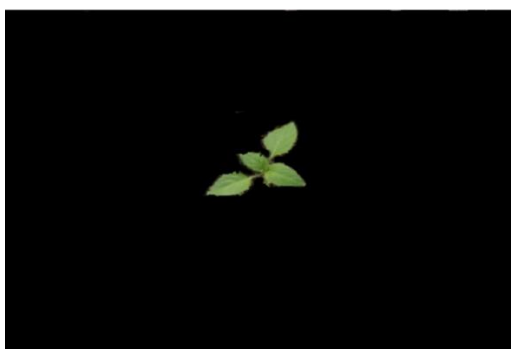
affects the results and it needs to be determined by trial and error. The algorithm calculated a measure of contrast using the Sobel filter and then computed the disparity for each pixel in the left image as the base. This was used for detecting the distance of the leaves and the estimation of the real size. The amount of disparity determines the amount of distance from the camera (Fig. 4.7). The larger the disparity, the closer the point to the camera. As Fig. 4.7 shows, the farthest points from the camera are in dark blue having distances of about 60 cm. Finally, the image of leaves was extracted based on the distance. Fig. 4.8 indicates the result of thresholding of the 3D image based on distance which was extracted from Fig. 4.8.



*Figure 4.6. The overlapped right and left images after rectification.*



*Figure 4.7. Disparity map obtained for the distance of leaves from the camera.*



*Figure 4.8. The real picture of leaves after thresholding of 3D points image.*

## 4.2 Stereo matching and cost functions

The use of matching techniques and cost functions is crucial in the development of the disparity map. Cost functions which also are called similarity measures define the “core” of many stereo matching algorithms. The similarity of image locations is measured in all stereo correspondence techniques. For all disparities under consideration, a matching cost is typically estimated at each pixel. Simple matching costs assume constant intensities at matching image sites, whereas more robust costs model certain radiometric shifts and/or noise (explicitly or implicitly). On gray and color images, common pixel-based matching costs include absolute differences, squared differences, sampling-insensitive absolute differences, or reduced versions. The sum of absolute or squared differences (SAD / SSD), normalized cross-correlation (NCC), and rank and census transforms are all common window-based matching costs ([Hirschmuller and Scharstein, 2007](#)).

In previous research, many different techniques and algorithms have been proposed for correspondence problem and stereo matching ([Heise et al. 2015](#)). In this work, we compared six different algorithms including Block Matching (BM), Block Matching with Dynamic Programming (BMDP), Belief Propagation (BP), Gradient Feature Matching (GF), Histogram of Oriented Gradient (HOG), and the proposed method. Also four cost functions namely Mean Squared Error (MSE), Sum of Absolute Differences (SAD), variant Sum of Differences (SD), Normalized Cross-Correlation (NCC) were used and compared. In addition to these methods, the use of deep learning was considered for stereo matching applications.

### 4.2.1 Dataset

Two stereo datasets were used to test our method (i.e. KITTI 2015 and middlebury). These datasets were used as they provide a large number of stereo pairs along with ground-truth images (Fig. 4.9). Table 4-1 represents the details of these datasets. These datasets were used in this study as they provide a variety of stereo pairs along with ground truth image which is necessary for training the network. Figure 4.10 represents the left images of all image pairs considered for the development of disparity map.



Figure 4.9. Sample left and right images with the disparity map.

Table 4-1. The databases of stereo and ground truth images used in this study.

No.	Source	Link to the database	Number of image pairs	Reference
1	Middlebury	<a href="https://vision.middlebury.edu/stereo/data/scenes2014/">https://vision.middlebury.edu/stereo/data/scenes2014/</a>	33	<i>Scharstein et al. 2014</i>
2	KITTI	<a href="https://www.cvlibs.net/datasets/kitti/eval_stereo.php">https://www.cvlibs.net/datasets/kitti/eval_stereo.php</a>	24	<i>Menze and Geiger 2015</i>



Figure 4.10. Total of left images used in this study provided by The Middlebury Stereo Dataset.

#### 4.2.1.1. Block Matching Algorithm

BM consists of sliding a window along the epipolar line and compares contents of that window with the reference window in the left image (Chen et al., 2001). Then a cost function will compute the difference between those two blocks, and finally the

block with most similarities with the reference block will be matched. Following this idea, the most crucial point in this algorithm is the choice of the block size, regardless to the cost function. In fact, if a small window size is chosen, there will be more details but also more of noise. On the other hand, a larger window size implies a smoother disparity map with less details and fails near boundaries (Gyaourova et al., 2003).

#### **4.2.1.2. Block Matching with Dynamic Programming**

This algorithm is used in order to avoid situations where BM fails to give a good result, especially when image pairs are textureless regions, repeated patterns and specularities (Nguyen et al., 2013; Aboali et al., 2017). The dynamic programming method was chosen as the global optimization technique for the disparity optimization phase since this algorithm optimizes the energy function to be NP-hard (non-deterministic polynomial-time hardness) for the purpose of smoothness and enhancement.

#### **4.2.1.3. Belief Propagation**

Belief propagation is a general method for solving optimization problems (Yedidia et al., 2003). This algorithm is normally specified in terms of probability distributions, but it can also be computed with negative log probabilities (Yedidia et al., 2000). The choice of this formulation is due to the low numerical errors and it directly employs the energy function concept. BP uses iterative message passing to get the overall best answer. Due to the simultaneous processing of these messages for all pixels in a single iteration step, a significant amount of computation and memory is needed. However, there are a number of fast stereo BP implementations available.

#### **4.2.1.4. Histogram of Oriented Gradient**

The idea behind HOG descriptor is that edge direction and intensity gradients can be used to characterize an object's form and appearance in an image (Dalal and Triggs, 2005). However, before we can calculate the histogram, we must first partition the image into smaller connected sections of a defined size. Only then a histogram of gradient orientations can be computed, but not for the full image, rather for each

individual cell, resulting in several histograms equal to the number of accessible cells, which add up to the required descriptor. It's also worth noting that orientation can be represented as angles between  $[0, 180]$  unsigned and  $[0, 360]$ , with the choice of (un-)signed depending on the project and the required degree of gradient accuracy.

Normalizing-by-contrast is a common optimization for intensity-based descriptors in which we compute the intensity over a bigger region termed block and then utilize this newly-found intensity information to normalize all the cells within the block (Anton, 2014).

#### 4.2.2 Matching Cost

Our initial cost function is the widely used Sum of Absolute Differences (SAD), which assumes brightness consistency for corresponding pixels and acts as our evaluation's baseline performance metric. Local stereo methods behave over a window, whereas global stereo methods use pixel-by-pixel differences (Equation 23) (Hamzah et al., 2010).

$$SAD = \sum_{(i,j) \in W} |I_1(i, j) - I_2(x + i, y + j)| \quad (23)$$

where  $I_1$  refers to the reference image, while  $I_2$  indicates the target image, and  $W$  indicates the square window for aggregation.

Another cost function that was used is the Mean Squared Error (MSE) (Equation 24). For images, the 'error' in MSE is a synonym of 'difference'. Then a difference between two image pairs is required to be obtained, but also the 'mean' is a synonym of 'average'. The sum of differences is needed to be divided by the number of the pixel. So, it can be called Average Squared Difference (Di Stefano et al., 2004).

$$MSE = \frac{1}{MN} \sum_{n=1}^M \sum_{m=1}^N [\hat{g}(n, m) - g(n, m)]^2 \quad (24)$$

Where  $g$  refers to the reference image, while  $\hat{g}$  indicates the target image.

Next, Normalized Cross-Correlation (NCC) (Equation 25) was the other cost function used for error measurement. NCC is a method for matching two windows around a pixel of interest that is widely used. The window's normalization accounts for variances in gain and bias (Heo et al., 2010). The statistically best method for

correcting Gaussian noise is NCC. However, because outliers create substantial errors in the NCC computation, NCC tends to obscure depth discontinuities more than many other matching costs (Zhang et al., 2009). Where  $I_1$  refers to the reference image, while  $I_2$  indicates to the target image,  $W$  indicates the square window for aggregation.

$$NCC = \frac{\sum_{(i,j)} I_1(i,j) \cdot I_2(x+i, y+j)}{\sqrt{\sum_{(i,j)} I_1^2(i,j) \cdot \sum_{(i,j)} I_2^2(x+i, y+j)}} \quad (25)$$

The flow chart below shows how the complete algorithm works. The left and right images and the ground truth are read and sent to preprocessing. During this step, the RGB images were transformed into gray images and then resized. The resizing was done to reduce the processing time. The format of the images from integer-eight was changed to double. Next, the images were presented to the matching algorithm.

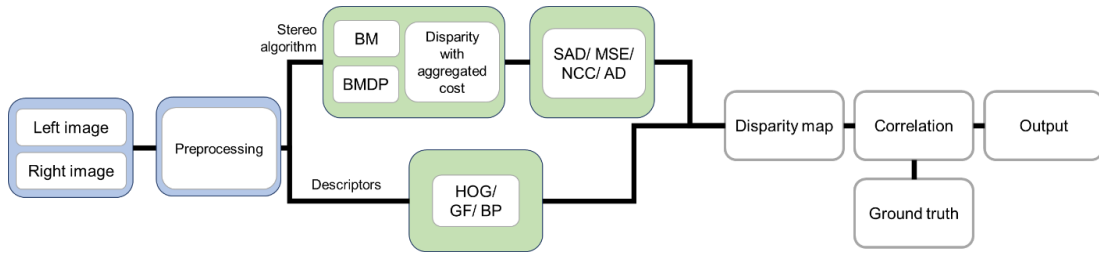


Figure 4.11. Diagram showing the course of the whole algorithm



## SPECTRAL ANALYSIS

### 5.1 Application of multispectral data

First step to use multispectral images for monitoring of plant growth and tracking is the possibility of the detection of plants from weeds. In this regard, the ability of multispectral data in separation of pepper and weeds was evaluated. This includes the study of possibility of separation and the selection of bands proper for separation and imaging applications.

#### 5.1.1 Plants

Bell pepper plants (variety of Cavendish F1) and five weeds (i.e. Bindweed, Nutsedge, Plantago lanceolata, Potentilla, and Sorrel) were used for this study. The samples were taken from young plants, of different leaf sizes and parts of the plants. Samples of bell pepper plant were taken from plants in vegetative and flowering stages of growth. Average height of plants was about 45 cm. For each plant about 40–50 samples were taken making up a total of 291 samples. These samples include leaves of various plants and of different sizes.

#### 5.1.2 Spectral Data

For illumination, one lamp of type A and one Halogen lamp were used. These lights were chosen as they provide enough energy in visible and NIR ranges. The spectral distribution of light intensity of A and Halogen sources have been shown in Fig. 5.1. Radiance of the A and Halogen were 1.373 and 4.326 W/(sr ×m<sup>2</sup>), respectively. These

light sources were measured by the spectroradiometer Specbos 1211 (JETI Technische Instrumente GmbH, Jena, Germany) (Table 5-1).

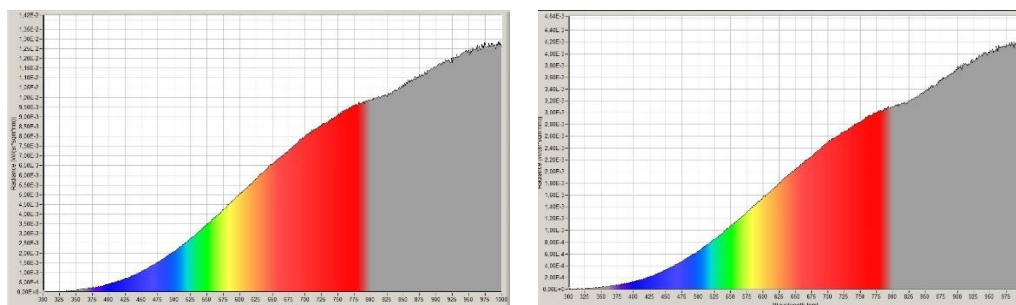


Figure 5.1. Spectral distribution of light sources.

The spectral responses were obtained in the range 400-1000nm using the spectroradiometer Specbos 1211 described before. This was done by dividing the spectral radiances on the source spectral distribution. The spectroradiometer was set at an angle of 90° in relation to the leaves and the standard observer of 2° was used for the measurements.

**Table 5-1.** Technical specifications of the spectroradiometer.

Spectral range	350 - 1000 nm
Optical bandwidth	4.5 nm
Measuring range	Luminance 0.2 - 100 000 cd/m <sup>2</sup> Illuminance 1 - 1 500 000 lx
Digital electronic resolution	15 bit ADC
Viewing angle	1.8° (Luminance mode)
Luminance uncertainty	± 4.4 % (Illuminant A @ 100 cd/m <sup>2</sup> , k=2)
Luminance repeatability	± 1 %
Chromaticity uncertainty	± 0.002 x, y (Illuminant A, k=2)
Color repeatability	± 0.0005 x, y (Illuminant A)
Max. wavelength error	± 0.5 nm (HgAr line source)
Measuring quantities	Illuminance, Irradiance xy and u' v' coordinates Dominant wavelength, Color purity Correlated Color Temperature (CCT) CRI, CQS, TM-30, TLCI, RGB Circadian metrics, PAR

### 5.1.3 Measurements

Lights were placed with 45 degrees related to the samples and the spectroradiometer was placed vertically toward the samples (i.e. zero degree for measurement) (Fig. 5.2). The plants were displaced with soil and roots from the farm and quickly transferred to the laboratory. Leaves were placed on a vague black background to reduce the light reflection. All the measurements were done under the same measuring conditions. Figure 5.3 represents the spectral radiances of samples for different plants.

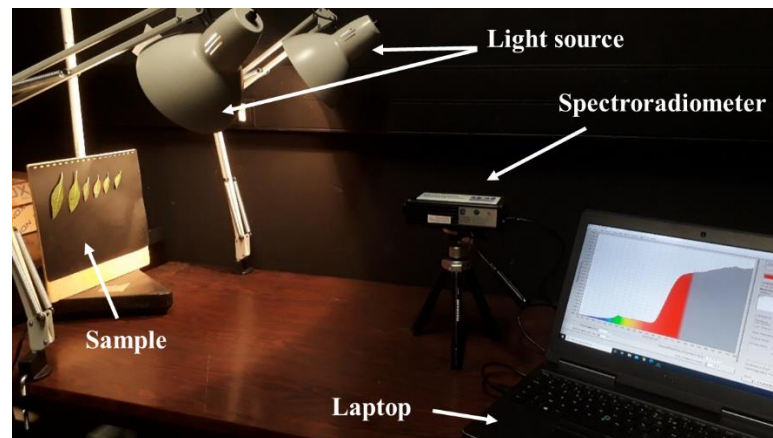
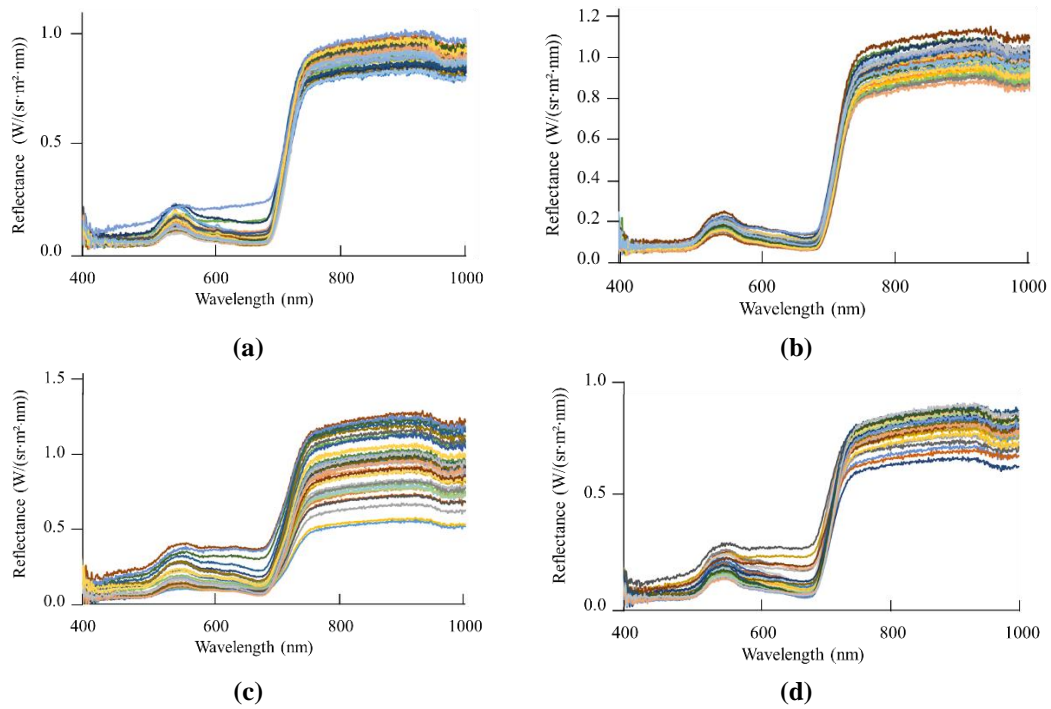
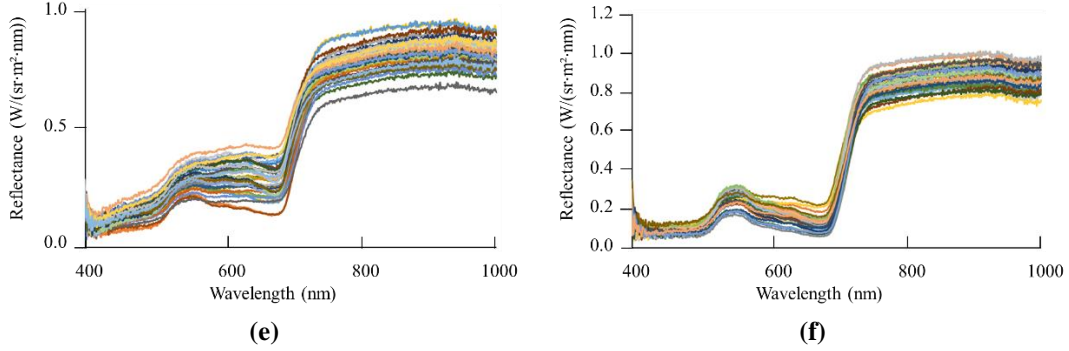


Figure 5.2. The measurement system and illumination.





**Figure 5.3.** The spectral response of a) Pepper, b) Bindweed, c) Nutsedge, d) *Plantago lanceolata*, e) *Potentilla*, and f) Sorrel.

#### 5.1.4 Continuous Wavelet Transform

Continuous Wavelet Transform (CWT) is used for the decomposition of a signal into wavelets. CWT is a perfect tool for mapping the changing properties of non-stationary signals. The basic functions of CWT are the scaled and shifted versions of the mother wavelet. The formula used for this transformation is as follows:

$$C(a, \tau) = \int \frac{1}{a} \psi\left(\frac{t - \tau}{a}\right) x(t) dt \quad (23)$$

Based on the formulae 23, the wavelet  $\psi(t)$  is shifted by  $\tau$  and scaled by factor  $a$ . In this study, Morse wavelet having the following formula was used:

$$\psi(\omega) = U(\omega) a_{p,\gamma} \omega^{\frac{p^2}{\gamma}} e^{-\omega^\gamma} \quad (24)$$

where  $U(\omega)$  represents the unit step and  $a$  is a normalizing constant.  $\gamma$  that controls the symmetry of the wavelet was set to 3 and  $p$  is the square root of the time-bandwidth product being in proportion to the wavelet duration was selected as  $\sqrt{60}$ . Hence, CWT was applied on all spectral responses and a database of scalograms was constructed. The scalogram is the absolute value of the continuous wavelet transform (CWT) of a signal or spectrum which is plotted as a function of time and frequency. These scalograms of the form 2D images were used for training the network and classification (Fig. 5.4).

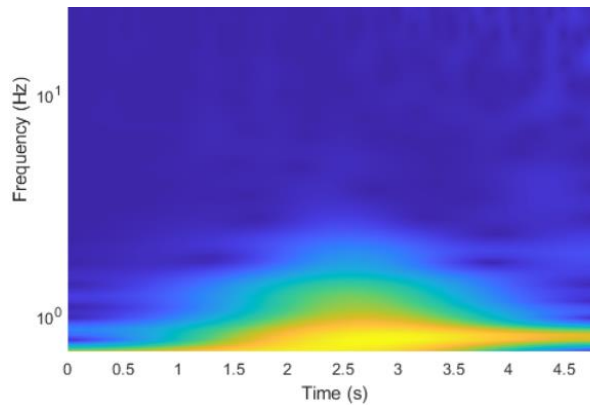


Figure 5.4. The scalogram of CWT colored as an RGB image.

### 5.1.5 Convolutional Neural Network

The pretrained neural network namely GoogLeNet was utilized in this study. This network was used for several reasons; first this is a quite strong network trained with a large database consisting of 1000 different categories; second, this saves time of trial and error for building new networks. In addition, a pretrained network can be easily used by other researchers working in the same field. The layer graph (which includes 144 layers) of the GoogLeNet has been shown in Fig. 5.5 and the architecture of it illustrated in Fig. 5.6.

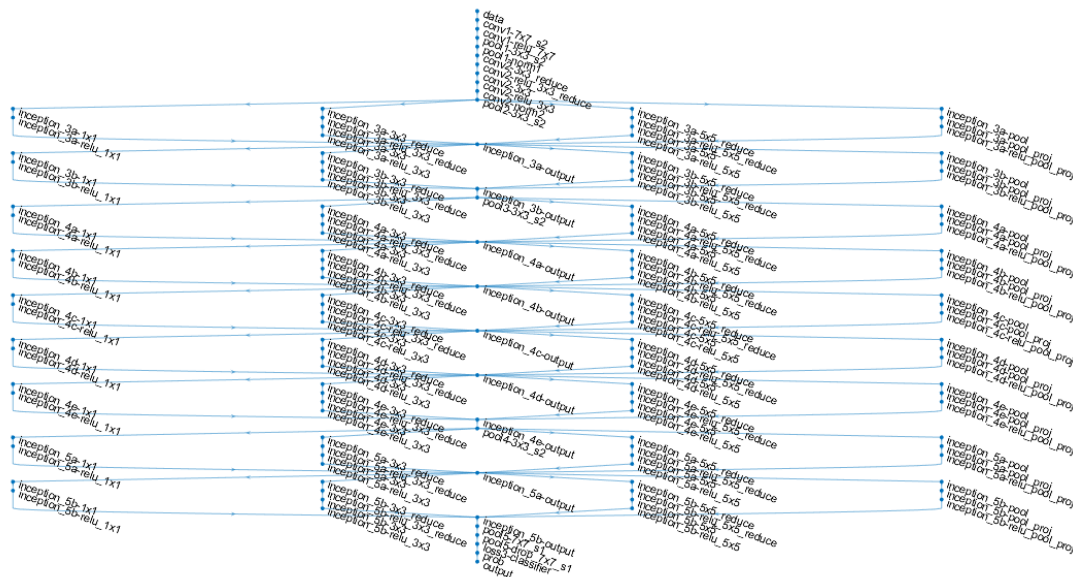


Figure 5.5. Layer graph of the trained network.

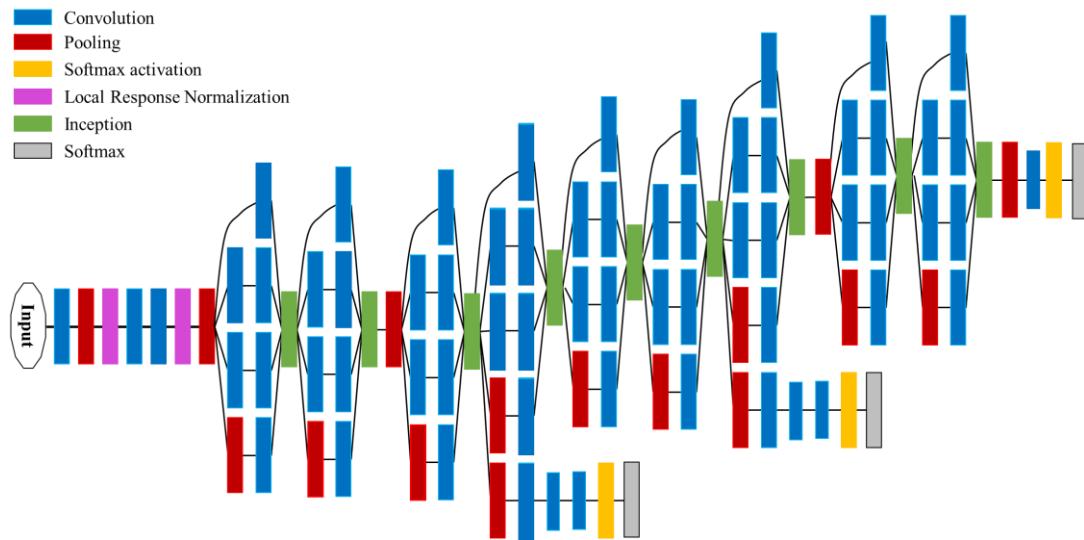


Figure 5.6. The architecture of the GoogleNet.

Initial learning rate of  $1 \times 10^{-4}$ , gradient threshold method of l2norm, and maximum epochs of 20 were used for building the network. The data were randomly divided into three groups of training, validation, and test. So, 70% was used for training, 15% was for validation and 15% was put aside for testing the network. To test the network, the spectral responses were given to the network to predict the type of plant. Then the accuracy of estimation was calculated as follows:

$$\text{Accuracy} = \text{Sum of true estimations} / \text{Total number of estimations}$$

To avoid overfitting, a dropout layer was employed that randomly sets input elements to zero at a level of probability. The flow chart of the proposed method is as shown in figure 5.7.

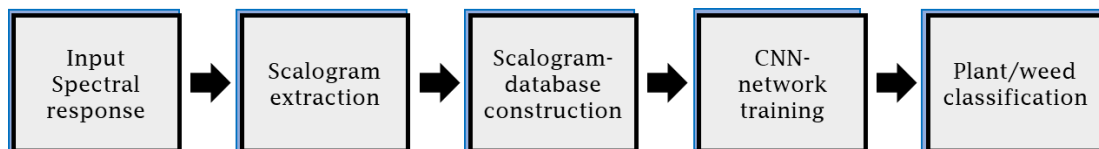


Figure 5.7. Block diagram of the proposed algorithm.

### 5.1.6 *k*-means clustering

The *k*-means algorithm is an iterative technique that attempts to split a dataset into *k* separate non-overlapping subgroups (clusters), each of which contains only one data

point. It attempts to make intra-cluster data points as comparable as possible while maintaining clusters as distinct as possible. It distributes data points to clusters in such a way that the sum of the squared distances between them and the cluster's centroid (arithmetic mean of all the data points in that cluster) is as small as possible. Within clusters, the less variance there is, the more homogenous (similar) the data points are.

The following is how the  $k$ -means algorithm works:

- $k$  is the number of clusters to specify.
- Initialize the centroids by shuffling the dataset and then picking  $K$  data points at random for the centroids without replacing them.
- Continue iterating until the centroids do not change. i.e. the clustering of data points does not change.
- Calculate the total of all data points' squared distances from all centroids.
- Assign each data point to the cluster that is closest to it (centroid).
- Calculate the cluster centroids by averaging all of the data points that correspond to each cluster.

To apply  $k$ -means clustering MATLAB software was used.  $K$ -means of MATLAB uses the squared Euclidean distance metric and the  $k$ -means++ algorithm for cluster center initialization. The  $k$ -means++ algorithm uses a heuristic to find centroid seeds for  $k$ -means clustering.

A sample code of  $k$ -means clustering for Silhouette technique has been presented in Appendix B. Four evaluation criteria, Calinski-Harabasz, Davies-Bouldin, Silhouette, and Gap methods were utilized. The experiments were done for  $k = 1:6$  groups. For more than two groups, the number of optimal clusters were tested. To represent the results of clustering, Principal Component Analysis (PCA) was applied and used. For the calculation of distances of points, the Euclidean distance was applied.

The spectral data of plants were imported to MATLAB and were used for  $k$ -means clustering. The spectral data were tested from 400 – 1000 nm with steps of one nanometer. Also, the data with steps of 5 and 10 nm were tested for clustering task.





# III

## RESULTS, DISCUSSIONS AND CONCLUSIONS

## RESULTS AND DISCUSSION

### 6.1 Physical Properties Database

The physical characteristics of the plants were measured and recorded using imaging and dimension extraction from leaf images during the growth stage. Nine physical properties were measured to monitor and estimate plant growth. This database was used to compare the estimated data with the real values and validation of growth estimates as well as artificial neural network training. In total, the physical properties of 311 leaves in different dimensions and ages were recorded. Table 6-1 shows the mean, minimum, and maximum values for these properties.

**Table 6-1.** Actual physical properties of leaves.

Property	Average $\pm$ SD	Min	Max
Physiological length (cm)	3.619 $\pm$ 1.757	0.862	10.393
Physiological width (cm)	1.987 $\pm$ 1.217	0.423	6.578
Leaf area (cm <sup>2</sup> )	6.719 $\pm$ 7.417	0.313	48.893
Leaf perimeter (cm)	9.528 $\pm$ 4.910	2.258	28.105
Aspect Ratio ( $\emptyset$ )	2.049 $\pm$ 0.534	1.178	3.383
Form Factor ( $\emptyset$ )	0.729 $\pm$ 0.082	0.433	0.912
Rectangularity ( $\emptyset$ )	1.303 $\pm$ 0.109	1.004	1.526
Perimeter ratio of diameter ( $\emptyset$ )	2.602 $\pm$ 0.192	2.209	3.542
Perimeter ratio of Lengths ( $\emptyset$ )	2.047 $\pm$ 1.046	1.556	6.913

#### 6.1.1 Leaf perimeter

The following diagram shows the trend of changes in the leaf perimeter of the three plants during the growing season. The values displayed for the perimeter per day are

the sum of the perimeter of all the plant leaves on that day. Actual leaf perimeter was obtained based on the images taken from the leaves.

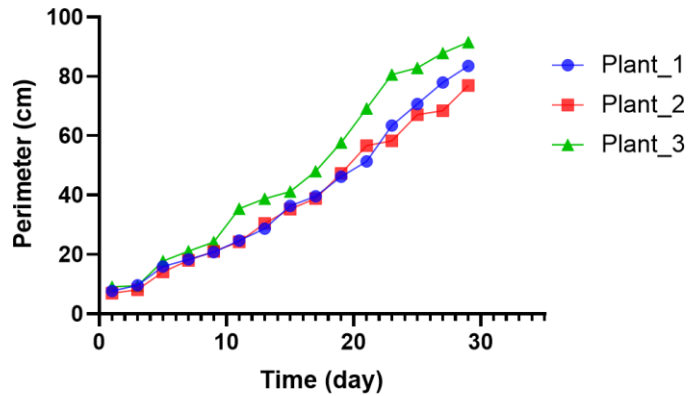


Figure 6.1. Leaf perimeter during growth period.

### 6.1.2 Leaf area

Diagram 6.2 shows the development and increase of leaf area for three plants during the growing season. Leaf area was measured every other day. The displayed areas are the sum of the plant leaf areas per day. Areas were obtained based on the images taken from the sheets on a daily basis.

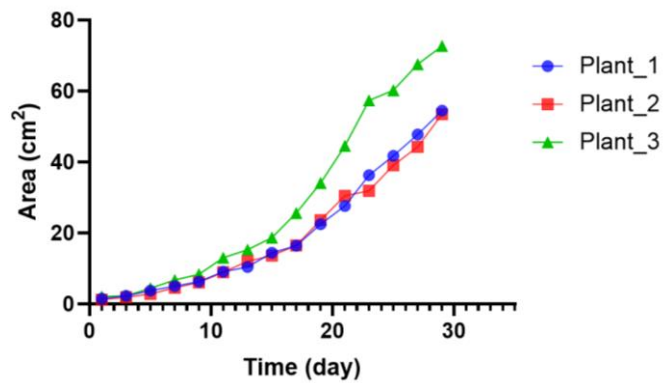


Figure 6.2. Leaf area during growth period.

### 6.1.3 Compare leaf area with theoretical models

Models which are based on the physical properties of the leaves, were plotted with the actual values of the leaf surface in Figure 6.3. The adjustment parameters of the

models were determined using MATLAB software. Model number 1 is the following linear model:

$$LA = K.LL.M$$

By fitting on the actual data,  $K$  was obtained equal to 0.25. Model number 2 is the following relation:

$$LA = \alpha(LL)^\beta$$

$\alpha$  was equal to 0.45 and  $\beta$  was equal to 1.2. Model number 3 is the exponential model as follows:

$$LA = \alpha \exp(\text{sqrt}(LL))$$

where  $\alpha$  was obtained as 0.2.

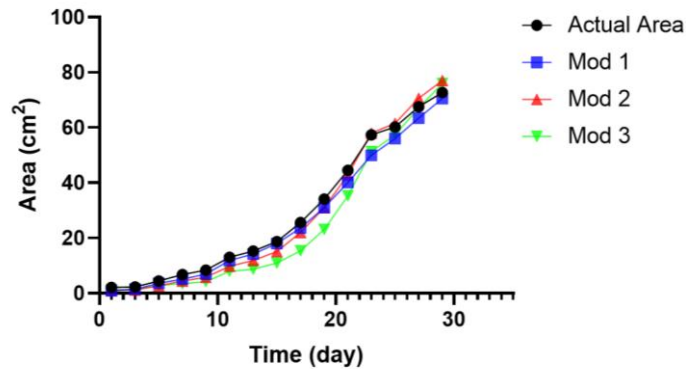


Figure 6.3. The estimated leaf area based on three theoretical model.

The accuracy of the models in estimating the leaf area was determined based on the main dimensions of the leaves using the RMSE value. Therefore, the RMSE value was 3.026 for the first model, 2.68 for the second model and 5.935 for the third model. Therefore, Model 2 most accurately estimated the leaf area using the original leaf length. Demirel et al. (2012) estimated growth curve parameters of pepper during first twelve weeks of growth. They measured plant height, plant x-x and y-y diameter and chlorophyll readings for growth estimation. It was observed that Linear, Gompertz and Logistic models performed best for pepper growth tracking. Sarıbaşı et al. (2018) studied the growth of pepper mathematically to propose a model. Inputs included in

transplant height, stem diameter, leaf area, total dry weight, leaf stomatal conductivity and leaf chlorophyll contents.

## 6.2 Active stereo imaging

### 6.2.1 Comparison of laser and projector

The use of laser for the projection of light was studied. Calibration of laser poses much difficulty on stereo imaging. One of the problems is to control the laser power. If the power is not enough, the laser is not detected in the images and if the power is strong the scattering of the laser on the environment and the leaves causes problems in image processing. Fig. 6.4 represents the light which is around the laser light in comparison to that of the projector. [Kashani et al. \(2015\)](#) divided the effective factors of light scattering in LiDAR and laser scanning into four categories; (1) target surface characteristics; (2) data acquisition geometry; (3) instrumental effects; and (4) environmental effects. [Bolkas \(2019\)](#) has reported that target surface characteristics including roughness and reflectance have large impacts on the intensity values of the reflected laser line on the surfaces. It has also been advised to reduce the laser intensity for shorter distances to reduce the amount of light scattering ([Kaasalainen et al. 2018](#)).



*Figure 6.4. Light reflection around the light stripes taken by: Left) projector and Right) laser projection (the contrast has been ameliorated).*

In this regard, to verify the problem of light scattering, the gloss (sheen) of some leaves were measured using a glossmeter (BYK, USA). Table 6-2 represents the average amount of gloss for 20 samples (front and back measurement). As it is seen the leaves are quite mate and referring to  $85^\circ$  that is the best measurement angle for super mate surfaces, the leaves are so mate and scatter almost all the received light.

Also, the effect of sub-surface scattering should be considered for leaves as live organs (Gigilashvili et al. 2021a; Gigilashvili et al. 2021b).

**Table 6-2.** Measurement of glossiness of bell pepper leaves.

Measurement degree	Leaf front			Leaf back		
	20°	60°	85°	20°	60°	85°
Gloss (sheen)	0.205±0.02	1.505±0.29	0.9±0.61	0.319±0.04	2.0±0.19	0.695±0.59

### 6.2.1.1. Selection of the light stripe color

Red light stripe has always been used in past research. One of the problems has been the scattering of laser on the objects as the light of lasers are quite strong and mostly it is hard to reduce the intensity (Morris, 2006). In the present study, due to the use of video projector and the possibility of using other colors, three different lights were evaluated. The use of projector provides ability to adjust the light much easier in the sense of light stripe width or light intensity. The width of the red light stripe was better and it had less light scattering. Examining the pixels of the stripes (average of ten images), it was observed that the width of the red stripe was 4.17 pixels, green was 5.06 and blue was 6.39 pixels. Weisenfeld (2002) used a red laser for the development of an active stereo system intraoperative surface measurements.



*Figure 6.5. Light stripe width.*

The scattering of the stripes, which affects how the optical stripe is extracted in subsequent processing, was also investigated. The light scatter at the edge of the blue stripe was higher than green and the green stripe was higher than red (Fig. 6.6). Figures 6.7-9 show the pixels adjacent to the light stripe and the brightness values. It can be seen that for the red stripe, the values decrease rapidly and at a distance of two pixels from the stripe, the brightness has reached less than 40, but for the green and blue

stripes, the changes in light intensity are milder and up to several pixels, still the values are large.



*Figure 6.6. Light stripes and their scattering in the edges.*

40	46	36	53	100	161	226	231	179	88
42	54	48	59	109	171	227	234	181	83
42	52	44	57	110	174	228	233	180	81
43	50	41	57	114	178	229	231	177	78
44	50	42	62	122	183	230	230	173	74
44	51	45	69	131	189	229	228	168	70
44	50	45	72	137	193	229	227	164	65
42	46	40	70	139	195	229	226	161	62
42	42	35	67	139	196	229	224	160	61

*Figure 6.7. Adjacent pixels to the red light stripe.*

64	61	60	78	181	250	251	248	252
64	61	63	80	180	249	251	249	253
64	61	68	83	179	247	251	250	253
64	61	72	88	180	245	250	251	253
63	62	76	93	185	245	250	252	252
63	64	77	99	191	247	250	252	252
62	65	77	104	198	249	250	252	252
62	66	76	106	202	250	250	252	251
62	69	78	120	206	251	248	248	253

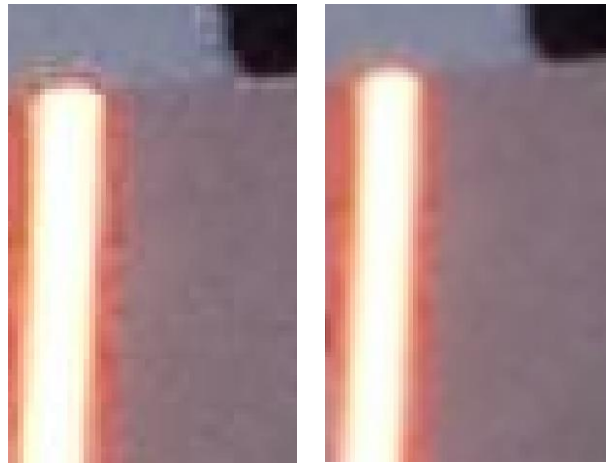
*Figure 6.8. Adjacent pixels to the green light stripe.*

62	67	53	61	150	234	245	246	243	239
53	64	57	72	160	239	245	246	244	240
48	60	62	86	174	242	245	245	244	239
53	58	68	99	186	244	244	244	245	230
59	56	71	107	194	245	244	244	245	219
54	55	74	115	209	248	242	246	245	213
56	51	70	119	213	248	245	246	244	202
58	48	67	126	219	248	245	246	243	191
59	52	67	136	223	248	246	246	244	185

*Figure 6.9. Adjacent pixels to the blue light stripe.*

### 6.2.2 Processing the images

First the images were read and a smoothing filter was used for making the light stripe smoother and help the thresholding of easier and cleaner. An averaging filter of size 3 x 3 was utilized. Figure 6.10 represents the effect of this filter on the images. As the figure shows the pixels around the light stripe are softer and the change of color is gradual. After filtering both images, the red layer of them was extracted. Then the resulting gray images were thresholded.



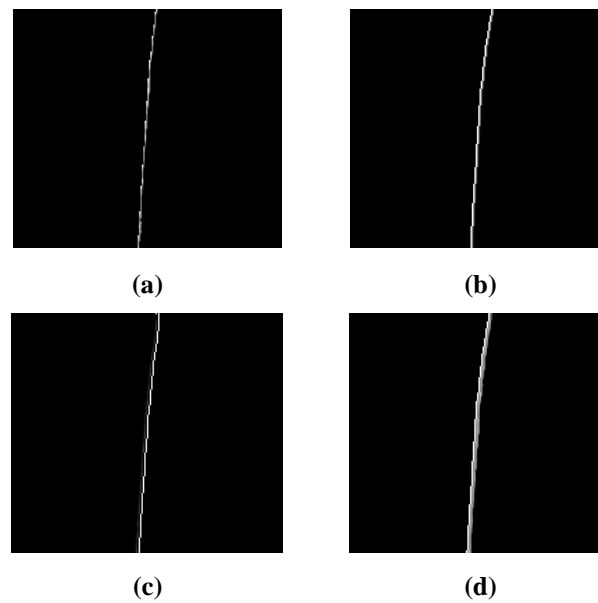
*Figure 6.10. Effect of smoothing filter on the light stripe; Left) original image and Right) after filtering.*

### 6.2.3 Light peak detection

Four different peak detection techniques including Gaussian, Center of Mass (CM), Parabolic and Linear Interpolation were tested. All techniques provide interesting



results for light peak detection. However, the best result was observed to be achieved by the CM for this project (Fig. 6.11). The Gaussian method narrowed the light line more than needed and in some parts resulted in disconnected lines. The Parabolic method was acceptable however, the resulting line became so thin and partly disjointed. The Linear Interpolation did not provide proper result and the resulting line got thicker and uneven in some parts.



*Figure 6.11. Light peak detection; a) Gaussian, b) Center of mass, Parabolic and Linear Interpolation.*

#### **6.2.4 Light stripe**

In each image—right or left—there is a spot or a line that is the projected light stripe on the plant. These spots or lines need to be extracted in each pair of images. Fig 6.12 represents the combined light stripes for all left and right frames. These images show that the image processing has been done very well and unwanted spots, dots and lines have been removed and only the light passed on the plants have been remained. This process is very critical as any unwanted spot in the environment causes mismatches in the correspondence process leading to inaccurate or wrong 3D shapes.



**Left camera**

**Right camera**

*Fig. 6.12. The combined light stripes projected on the plant.*

### **6.2.1 Camera calibration accuracy**

Calibration accuracy has a direct effect on stereo image processing. The calibration accuracy of stereo cameras was evaluated for three angles of 75, 90 and 105 degrees. Figure 6.13 shows how to identify the checkerboard, determine its position and how the predicted centers match the real centers. Figure 6.14 shows the calibration error for the angle of 75 degrees. It is observed that the calibration error is less than 1 pixel, which is an excellent value for the resolution of the cameras used. Compared to previous research works, this represents satisfactory precision for the proposed low-cost system which is one of the objectives of this study (Jang et al., 2013). Bi et al. (2021) developed an optical tracking system based on stereo vision. In this study also the calibration accuracy of pixel level was achieved. Wang and Shih (2021) proposed a novel method for measuring depth resolution based on active stereo imaging. In this study the calibration accuracy that was done using MATLAB calibrator was achieved equal to 0.1 pixels. Peiravi and Taabbodi (2010) proposed a photogrammetry-based calibration technique to find calibration parameters in an easier way. Figure 6.15 shows the location of the cameras in the calculated global coordinates.

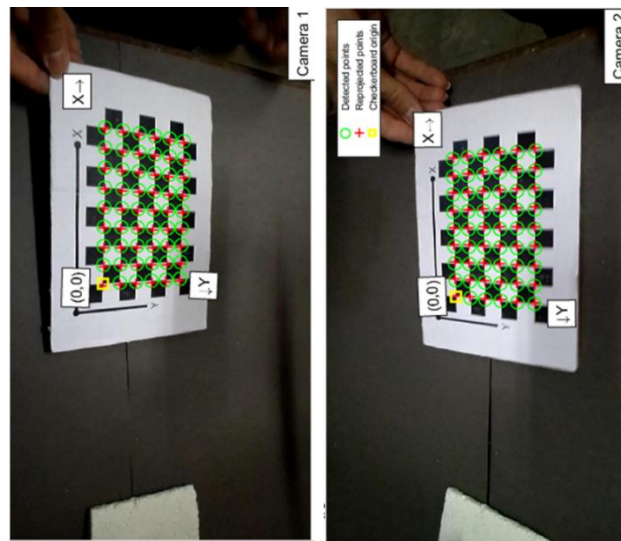


Figure 6.13. Detection of checkerboard and correspondence of the predicted and actual centers in a pair of images (75°).

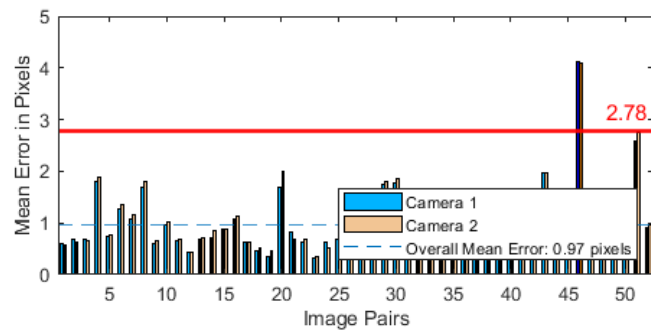


Figure 6.14. Average calibration error for all pairs of calibration images in 75°.

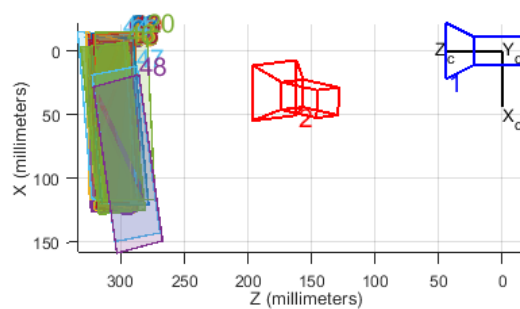


Figure 6.15. Position of cameras and checkerboards in the global coordination system (75°).

The correspondent points have been detected correctly and matched (Fig. 6.16). Calibration error was 1.14 pixel for the angle of 90 degrees between the cameras (Fig. 6.17). Fig. 6.18 represents the position of cameras in 3D space and the checkerboard images proportional to the cameras.

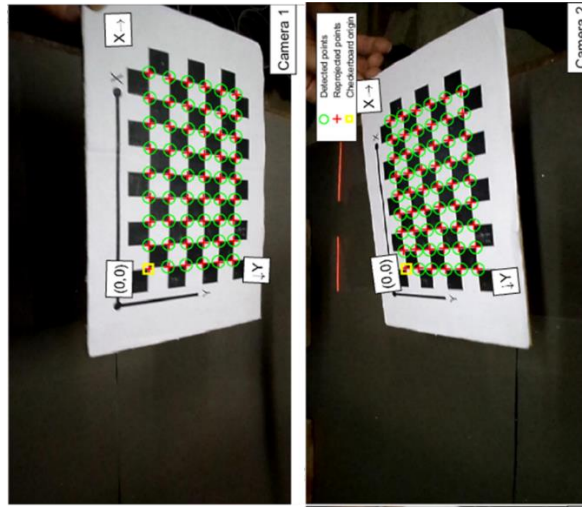


Figure 6.16. Detection of checkerboard and correspondence of the predicted and actual centers in a pair of images (90°).

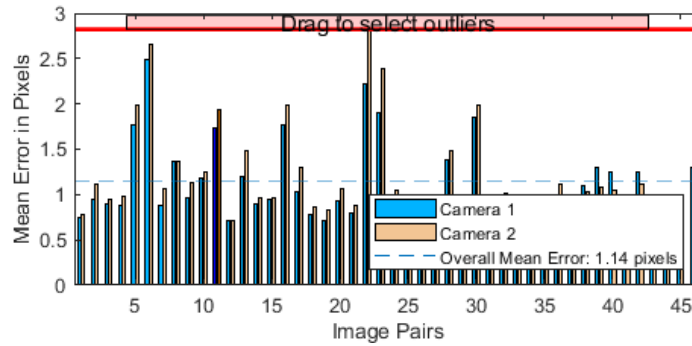


Figure 6.17. Average calibration error for all pairs of calibration images in 90°.

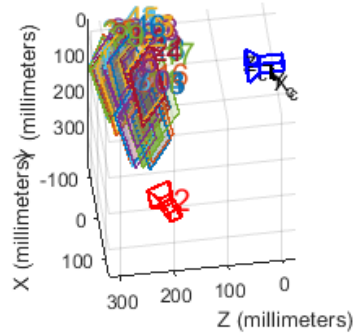


Figure 6.18. Position of cameras and checkerboards in the global coordination system (90°).

The overall calibration error in the angle of 105 degree amounted to 1.09 pixel (6.20). It is observed that the calibration error is almost equal in all degrees and the little difference is due to the number of images provided and a few improper pairs.

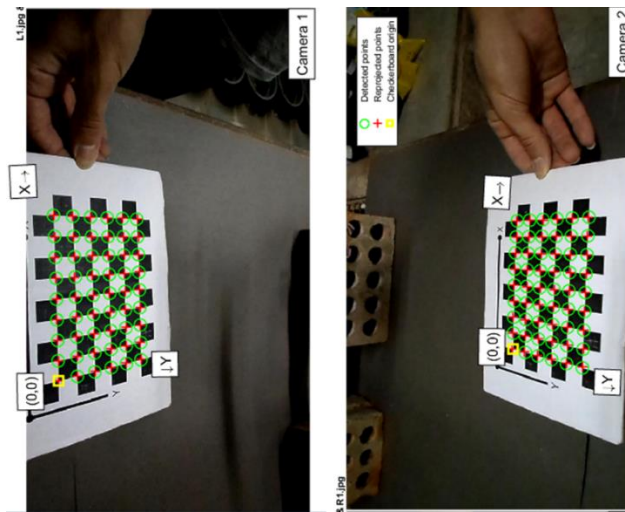


Figure 6.19. Detection of checkerboard and correspondence of the predicted and actual centers in a pair of images (105°).

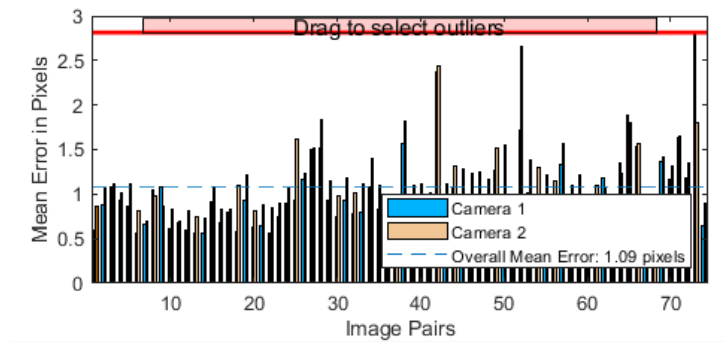


Figure 6.20. Average calibration error for all pairs of calibration images in 105°.

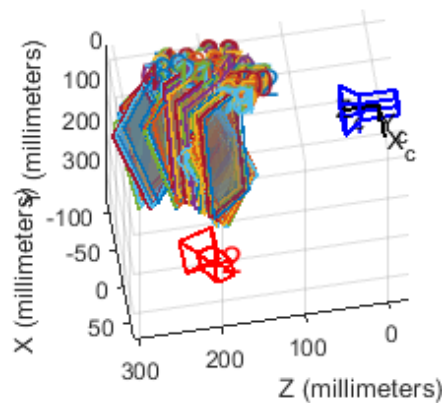


Figure 6.21. Position of cameras and checkerboards in the global coordination system (105°).

By the calibration, the intrinsic and extrinsic parameters were obtained. These parameters were saved in *.mat* file and used for further processing. Herein there is an

example of intrinsic matrix and rotation and translation matrices that build up the extrinsic matrix.

$$M_{int} = \begin{bmatrix} 6588.904939 & 0 & 0 \\ 0 & 6534.691113 & 0 \\ 3823.349831 & -978.496054 & 1 \end{bmatrix}$$

$$M_{Rot} = \begin{bmatrix} 0.4323 & 0.2872 & 0.8547 \\ 0.0266 & 0.9434 & -0.3304 \\ -0.9013 & 0.1656 & 0.40031 \end{bmatrix}$$

$$M_{Tran} = [-782.634656 \quad 343.181473 \quad 1707.997652]$$

### 6.2.2 Cameras angle analysis

The angle between the left and right cameras needs to be determined based on the applications. Three different angles were chosen and tested. As explained in chapter 3, a simple plate with predefined dimensions was used for this experiment. The plate was 3 cm × 3 cm. The developed 3D model of this plate was compared to the actual dimensions. Hence, width, length, and area were calculated for the measurement of the accuracy of modelling. Table 6-3 represents the percent error for different camera and plate angles. As the table presents, the best accuracy achieved when the angle between the cameras was 90°. Also, it is seen that the accuracy of measurements gets less for the smaller angles of plate.

**Table 6-3.** Measurement error of plate properties for different camera and plate angles.

Property	Plate angle	Camera angle (Percent error)		
		75°	90°	105°
Length (cm)	30°	12,333	10,000	14,333
	60°	9,333	4,333	10,667
	90°	2,333	1,333	2,333
Width (cm)	30°	29,333	6,667	8,333
	60°	10,667	6,333	8,667
	90°	3,667	1,667	2,000
Area (cm <sup>2</sup> )	30°	13,382	4,000	5,806
	60°	19,004	1,726	2,924
	90°	5,914	0,356	4,287

Figure 6.22 shows the result of reconstruction of the plate for each camera angle. Table 6-4 represents the quality of 3D reconstruction for different stereo camera angles. It was observed that in 90° the field of view is pretty good. In 105° the field of view of the cameras is quite big but it also covers unnecessary areas. For angle view

of the plate, the best view was  $75^\circ$  as both cameras can see the plate in most angles. But with smaller angles of the plate, it cannot be captured in one of the cameras and literally one of the cameras goes blind.

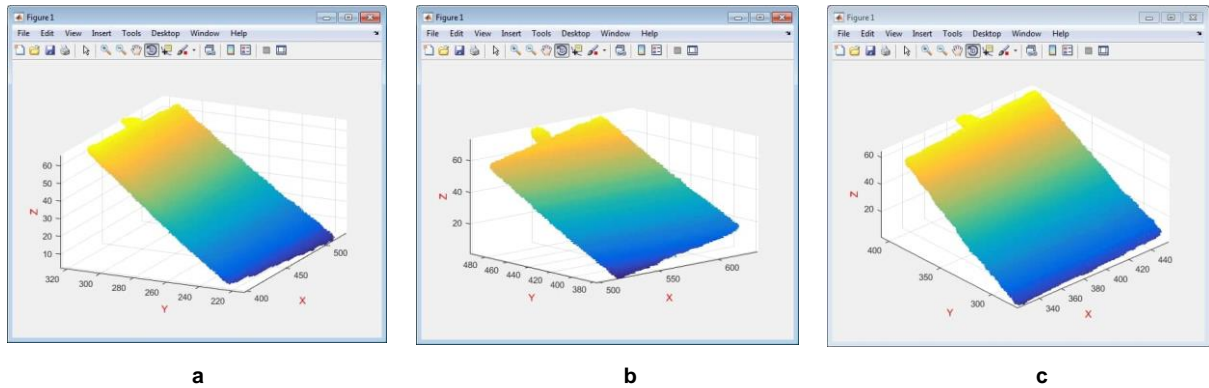


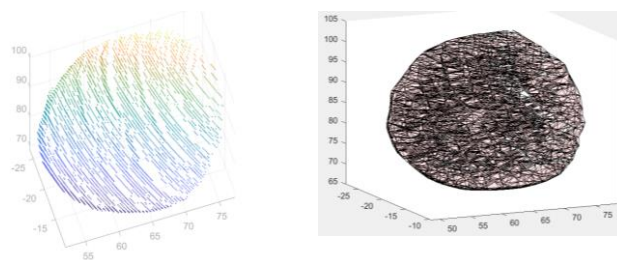
Figure 6.22. 3D development of simple plane in different camera angles; a)  $105^\circ$ , b)  $90^\circ$  and c)  $75^\circ$ .

Table 6-4. Reconstruction in different camera angles.

Property	Camera angle (Percent error)		
	$75^\circ$	$90^\circ$	$105^\circ$
Field of view	Good	Best	Good
Calibration accuracy	Best	Good	Good
Angle view	Best	Good	Poor
3D reconstruction ability	Good	Good	Poor

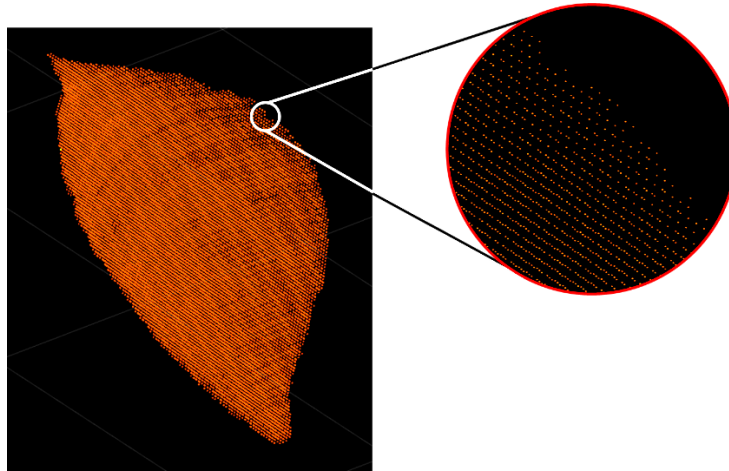
### 6.2.3 Leaf counting

The obtained 3D coordinates were converted to point clouds. Figure 6.23 illustrates an example of the point cloud extraction and surface reconstruction of a cylinder top part.



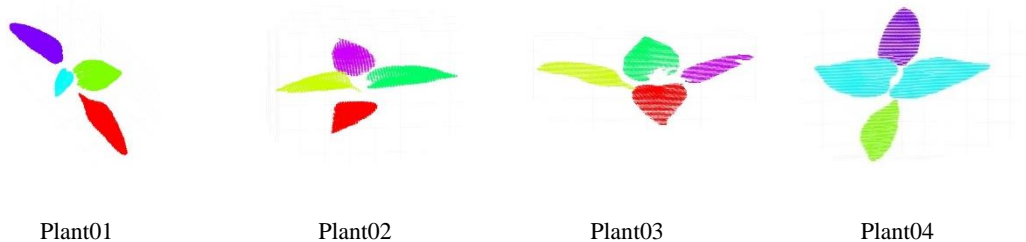
*Figure 6.23. Result of 3D reconstruction of top of a cylinder; Left) point cloud and Right) mesh.*

Figure 6.24 shows the point cloud of a single leaf extracted from the whole point cloud of a plant.



*Figure 6.24. 3D point cloud of a single leaf.*

Point clouds were segmented and used for leaf counting. The points close to each other were segmented as the same leaf. [Gupta and Ibaraki \(2014\)](#) compared four different active stereo technologies for plant 3D modelling and leaf segmentation. Fig. 6.25 represents their comparison and the point clouds that mostly are similar. [Jang et al. \(2013\)](#) took benefit of a two-step technique for the improvement of the accuracy. They built reliable correspondences based on a decoding technique of the projected pattern and then applied stereo matching technique. [Eigen et al. \(2014\)](#) proposed the extraction of the depth map based on a single image using Multi-Scale Deep Network. This solution is good in case of recovery of the occlusions and overlaid leaves.





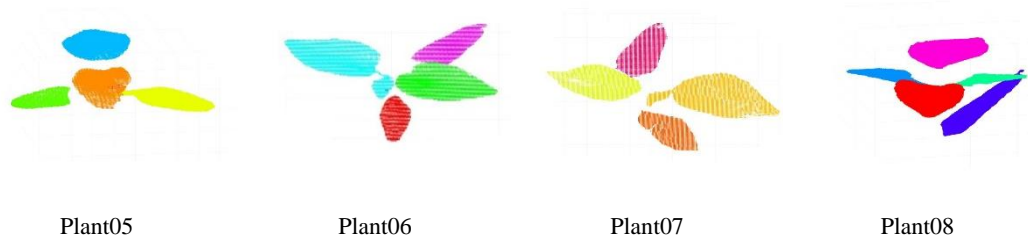


Figure 6.25. Segmented point cloud and discrimination of leaves.

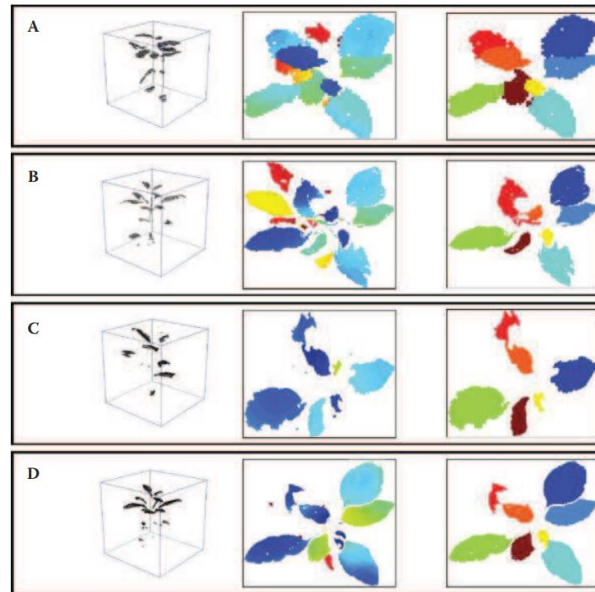
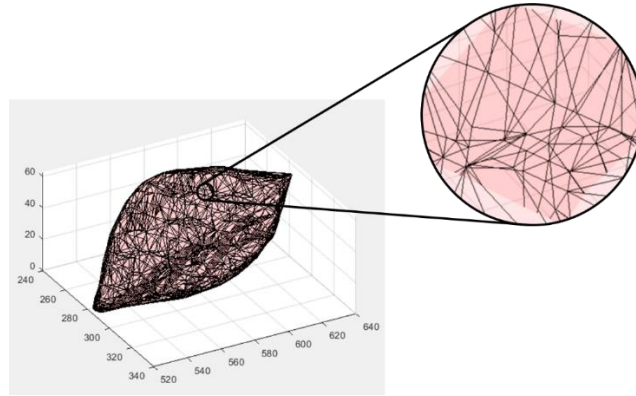


Figure 6.26. Comparison four different active stereo imaging systems done by Gupta and Ibaraki (2014); Left) point cloud, middle) depth map, and right) segmented model.

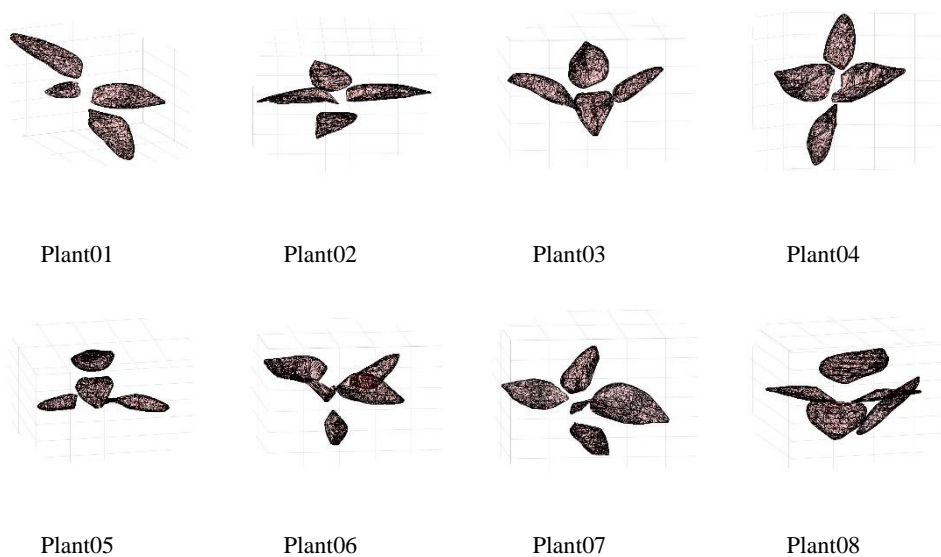
Point clouds were meshed into surfaces to measure physical properties of leaves. The surfaces are continuous models that provide the possibility of measuring the area and perimeter or other features (Serati et al. 2022; Massimiliano et al. 2022; Hossain and Lin, 2022). In the developed 3D models, all objects were counted, therefore any object was considered as a leaf. In point clouds or 3D models there is the possibility to remove a cluster of point clouds or a surface that cannot be a leaf. This is also a simple but important ability in 3D image processing. This can be used to remove the points or small objects that cannot be seen on the images but put error in counting objects. Rusinkiewicz et al. (2002) built the 3D model of objects by taking images of the objects in different angles and then combined the data together to develop a high quality 3D model of the objects. Figure 6.27 represents the result of meshing for a

sample leaf and shows the performance of triangulation in building the 3D shapes of leaves.



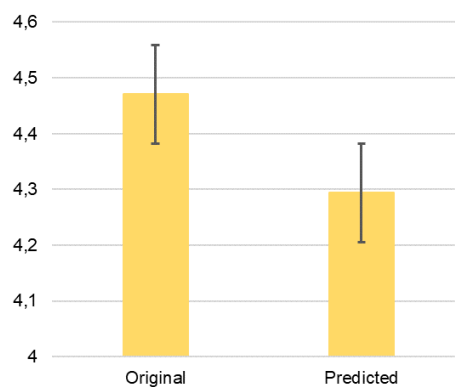
*Figure 6.27. An example of a meshed point cloud of a leaf into a 3D surface and the magnified triangulation connections.*

Fig. 6.28 represents the results of meshing for the samples shown in Fig. 6.25. The figure shows that Delaunay's triangulation has performed very well in the development of 3D surfaces. The results showed that the proposed method could count the leaves with the success rate of 96.053 %. [Dias \(2006\)](#) applied Delaunay triangulation for meshing point clouds. They reported redundant information of this technique specially if all data points were used.



*Figure 6.28. The 3D model developed based on the segmented point cloud.*

Figure 6.29 shows the average predicted number of leaves in comparison to the actual number. It is seen that the average of predicted leaves is lower than the actual number. This is due to the fact that in some cases the algorithm could not segment all leaves and some leaves were detected as one because they have been highly overlaid. One of the good ways to reduce the amount of occlusion is simulation. [Cajal et al. \(2015\)](#) simulated laser triangulation sensors scanning systems for design and development purposes. [Peiravi and Taabbodi \(2010\)](#) used two laser lights with two different colors along with a CCD camera for the minimization of occlusion.



*Figure 6.29. The predicted number of leaves compared to the actual number.*

#### 6.2.4 Leaf geometrical properties

For perimeter, the 3D leaves were projected to the XY plane and the perimeter of the 2D shapes were measured. Then the perimeter was scaled based on the angle of the leaf in proportion to the XY plane.

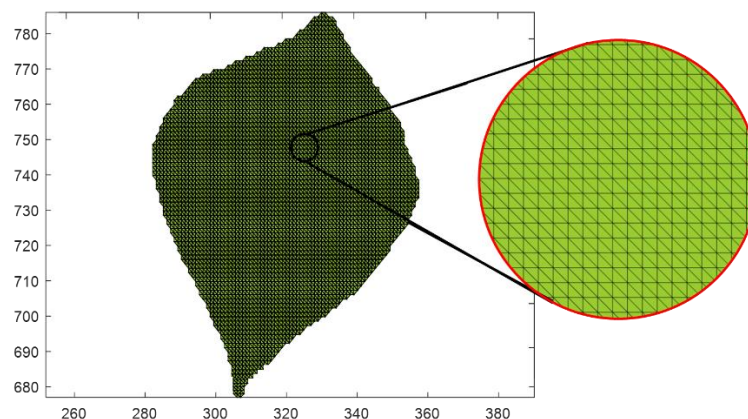


Figure 6.30. Leaf perimeter using Alphashape.

Leaf volume was measured using the convex hull volume. The convex hull volume is the volume that covers all the points. Fig. 6.31 represents the convex hull obtained for a sample leaf from two different points of view.

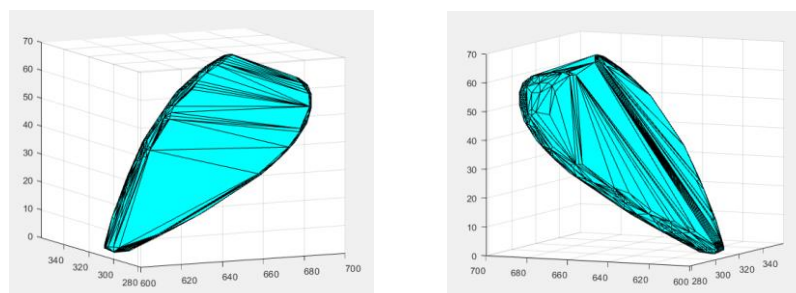


Figure 6.31. An example of the convex hull of a leaf from two different points of view.

Table 6-5 represents the average values of the estimated parameters based on the 3D models. The values provided in this table are the averages of all the leaves and the maximum and minimum values.

**Table 6-5.** Estimated physical properties of leaves.

Property	Average $\pm$ SD	Min	Max
Physiological length (cm)	2.588 $\pm$ 0.629	1.006	4.176
Physiological width (cm)	1.431 $\pm$ 0.541	0.446	2.815
Leaf area (cm <sup>2</sup> )	2.724 $\pm$ 1.545	0.356	7.676
Leaf perimeter (cm)	6.475 $\pm$ 1.732	2.407	11.267
Aspect Ratio ( $\emptyset$ )	1.957 $\pm$ 0.575	1.224	3.739
Form Factor ( $\emptyset$ )	0.755 $\pm$ 0.128	0.26	1.441
Rectangularity ( $\emptyset$ )	1.498 $\pm$ 0.523	0.652	3.912
Perimeter ratio of diameter ( $\emptyset$ )	2.505 $\pm$ 0.294	1.087	3.52
Perimeter ratio of Lengths ( $\emptyset$ )	1.618 $\pm$ 0.162	0.807	2.178

Figure 6.32 shows the regression of the actual and estimated values for leaf length and width. As the figure shows, the  $R^2$  values of 0.962 and 0.961 were achieved for the actual and estimated values of leaves' lengths and width, respectively.

The regression of the actual and estimated values for leaf area and perimeter are shown in Fig. 6.33. An  $R^2$  value of 0.978 was obtained for the regression of actual and

estimated values for leaf area that was much better than the  $R^2$  value obtained for that of perimeter (i.e. 0.967). [Paturkar et al. \(2020\)](#) used a 3D technique for the measurement of physical features of plants. They observed that the correlation between the measured values and ground truth values of leaf width had a correlation coefficient of 0.96 with RMSE of 0.11 cm.

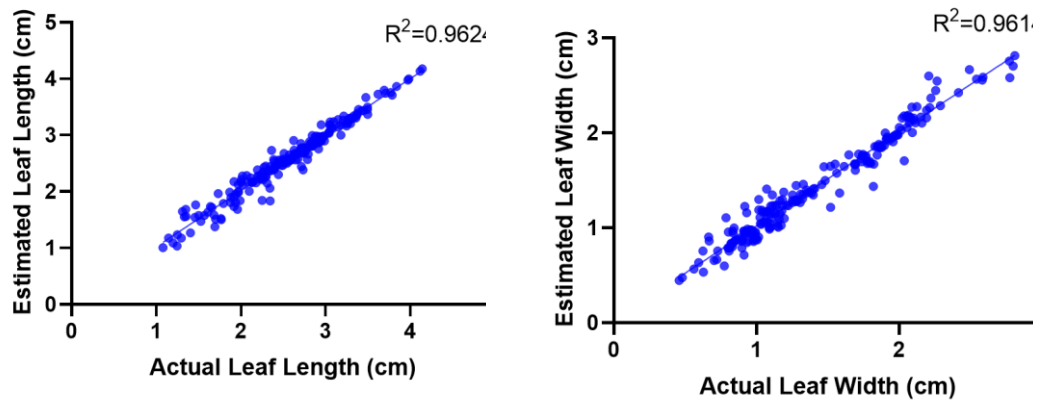


Figure 6.32. Regression of the estimated and actual values for leaf length and width.

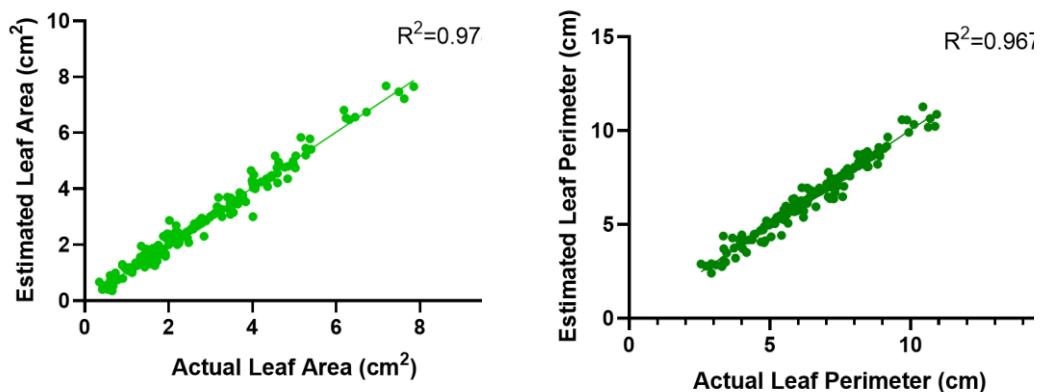


Figure 6.33. Regression of the estimated and actual values for leaf area and perimeter.

Fig. 6.34 represents the regression plots of the estimated and actual values of five other properties. It is observed that the highest  $R^2$  value obtained for Aspect Ratio amounting to 0.927 and the smallest  $R^2$  was obtained for Form Factor equal to 0.532. It is seen that as these properties are functions of other parameters, the error of the coefficient of determination is larger.

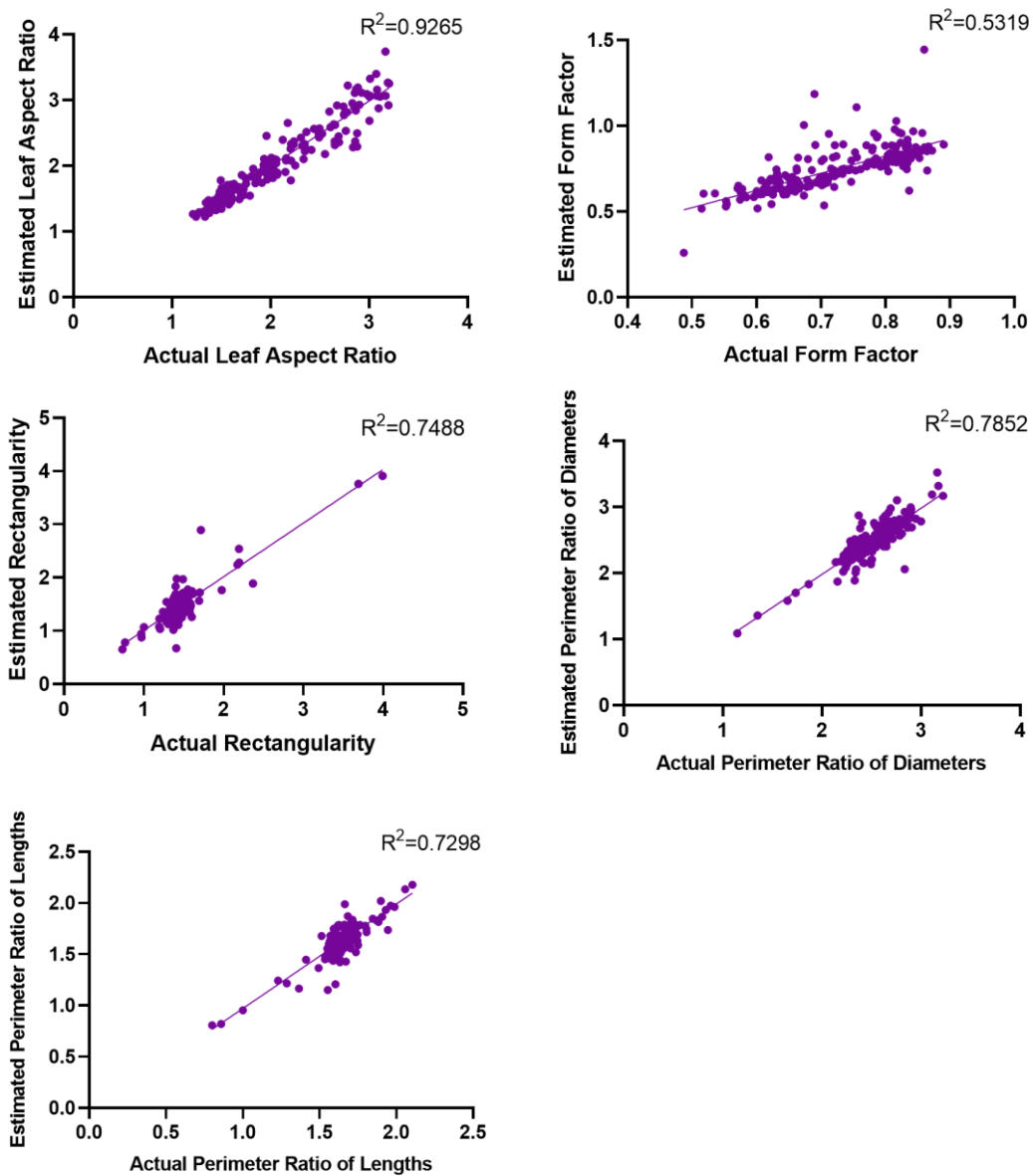


Figure 6.34. Regression of the estimated and actual values.

## 6.3 Passive stereo image processing

### 6.3.1 Segmentation

Figure 6.35 shows the results of pre-processing and segmentation steps for one plant from day 5 to day 27. The leaves have been properly extracted and the stems have been removed. [Shrestha et al. \(2004\)](#) used Otsu thresholding method for maize plant detection and reported positive results. [X. Li et al. \(2016\)](#), for monitoring lettuce

growth converted the images to HIS color space and applied thresholding. After filling the holes and denoising, *k*-means clustering algorithm was used. As it can be seen in Fig. 6.35, the detection and separation of leaves during the first days of growth is much easier and for the older plants with overlaid leaves this segmentation becomes much harder as the overlaying leaves make different and unpredictable shapes.

Using the obtained watershed image based on gradient magnitude, the leaves were colored. The result of pseudo-coloring of watershed images is given in Fig. 6.36 for the plants in the previous figure. In the watershed image, all the pixels for each leaf have the same index which can be used for other processing. In most cases the leaves have been identified and colored correctly. The algorithm proposed by [De Vylder et al. \(2011\)](#) also had proper results for the segmentation of leaves. [Lee \(2008\)](#) reported that the segmentation method worked properly for the images taken under sunlight conditions. Hence, a fresh weight model was developed and measurement error of the prediction equation was 2.49 g ([Lee 2008](#)). [Billiot et al. \(2013a\)](#) proposed shape from focus for the 3D reconstruction of plants. They used a single camera and developed the disparity map using shape from focus technique ([Billiot et al. 2013b](#)). [Denker \(2014\)](#) used multi-view stereo imaging based on four different camera views. They employed a GPU based stereo matching method resulted in a high quality reconstruction. [Chéné et al. \(2012\)](#) segmented plants for phenotyping applications by using Microsoft Kinect which consists of two CMOS sensors and an infrared camera. The detection was acceptable and bunches and leaves were segmented.

One of the solutions for overcoming the problem of overlaying leaves and true estimation of growth rate is to consider the whole plant for image processing. [Patil et al. \(2013\)](#), used a bunch of leaves of different plants. The proposed system included four steps of preprocessing, segmentation, feature extraction and classification which was performed using SVM. The classification accuracy of 94.73% was achieved in that study.

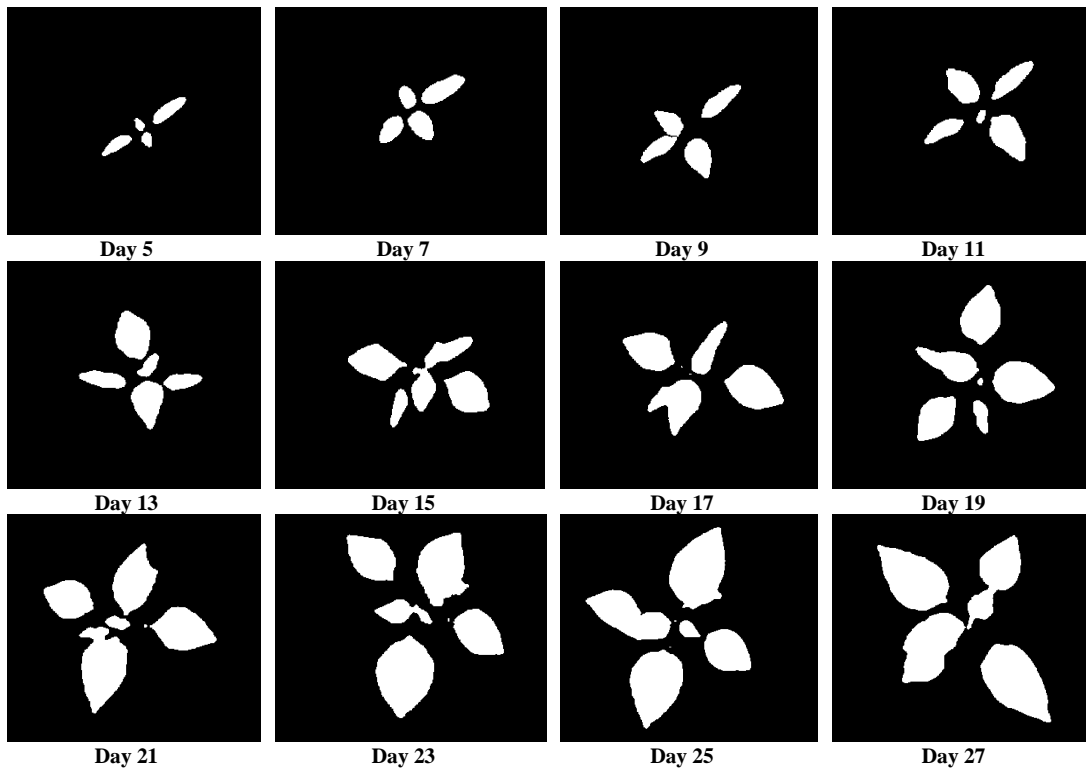


Figure 6.35. Image of the extracted leaves after removing the background and filling holes.

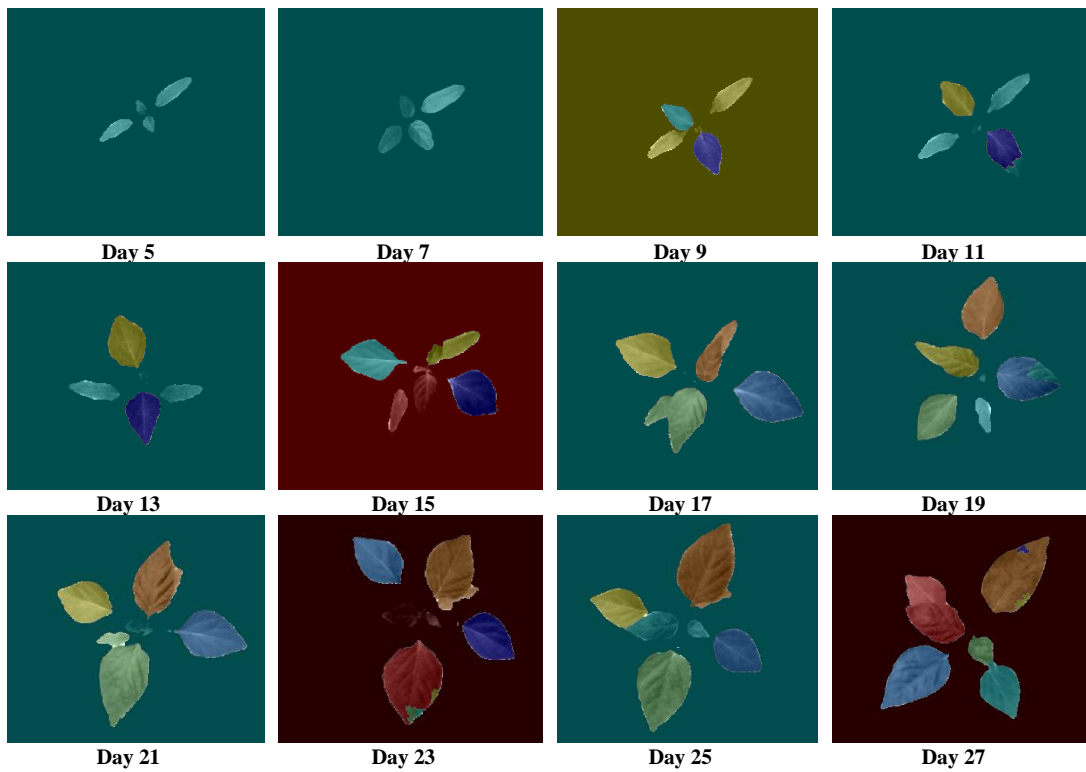


Figure 6.36. Pseudo-coloring of leaves after detection.

### 6.3.2 Number of leaves



One of the purposes of the algorithm was to count the leaves as every few days new leaves appear while others are lost. Thus, after the detection of leaves and before the calculation of leaf properties, the number of leaves is recorded. The overall success rate of the algorithm for leaf count was 84.32%. The undetected leaves were those that were fully covered or partially covered in such a way that could not be distinguished from the leaf directly above it.

### 6.3.3 Physical properties

Leaf physical properties were measured from the segmented images. Each detected leaf was associated with a label. Using the label the leaf could be extracted or processed. Figure 6.37 shows the regression between the estimated and actual leaf area values. The  $R^2$  of 0.944 was obtained for leaf area estimation. As Fig. 6.38 represents the  $R^2$  value of 0.931 was obtained for the regression between the estimated and actual leaf perimeters.

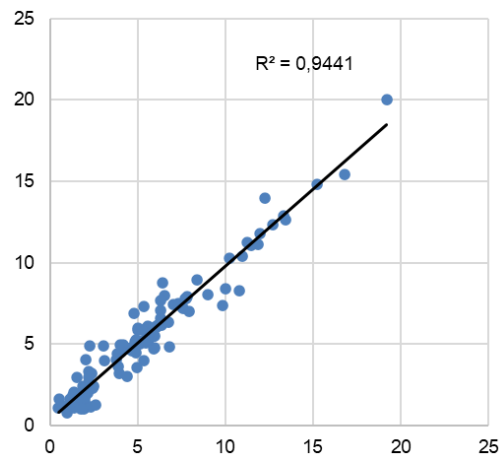


Figure 6.37. Estimated leaf area versus the actual area.

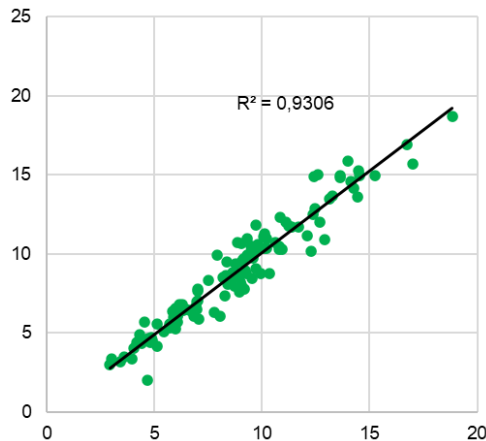


Figure 6.38. Estimated leaf perimeter versus the actual perimeter.

## 6.4 Stereo matching algorithm

Disparity maps were built for different matching algorithms and cost functions. Then the obtained results are compared with the ground truth. The mean error percentage was calculated as the average error for all the pixels of the image.

### 6.4.1 SAD

The results of matching techniques with the SAD cost function were pretty different. As Figure 6.39 shows, the BP provides the best result for the matching. However, it is observed that still the result depends on the image and the details inside. For example in the image of the motorcycle (Fig. 6.40), the DWAC had the best matching. Also, it is seen that the HOG technique is so sensitive to the details and in Fig. 6.40 the details have been detected much more neatly.

Besse et al. (2013) proposed Patch-Match BP to increase the precision of stereo matching and is an improvement of BP in sub-pixel level accuracy. Dipanda et al. (2003) used a GA-based algorithm for active stereo matching and reported high-accuracy matchings even for complicated objects and situations. Kihlström (2019) used deep learning for active stereo matching which received the left and right active stereo images and gave the disparity map as the result. They reported proper results of the proposed network compared to the previous techniques.

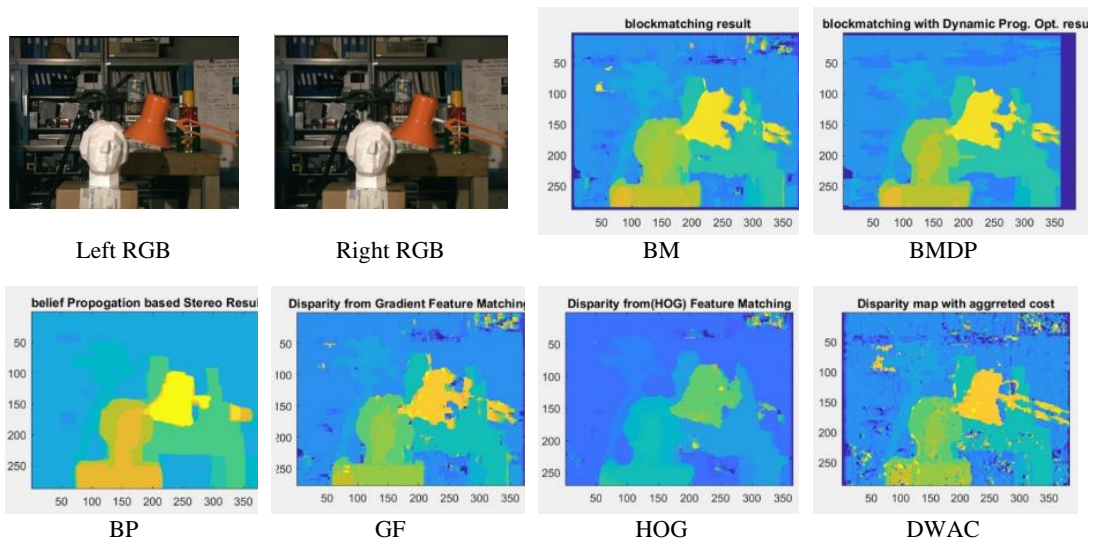


Figure 6.39. Left and right images and the disparity maps achieved for different matching algorithms based on SAD cost function.

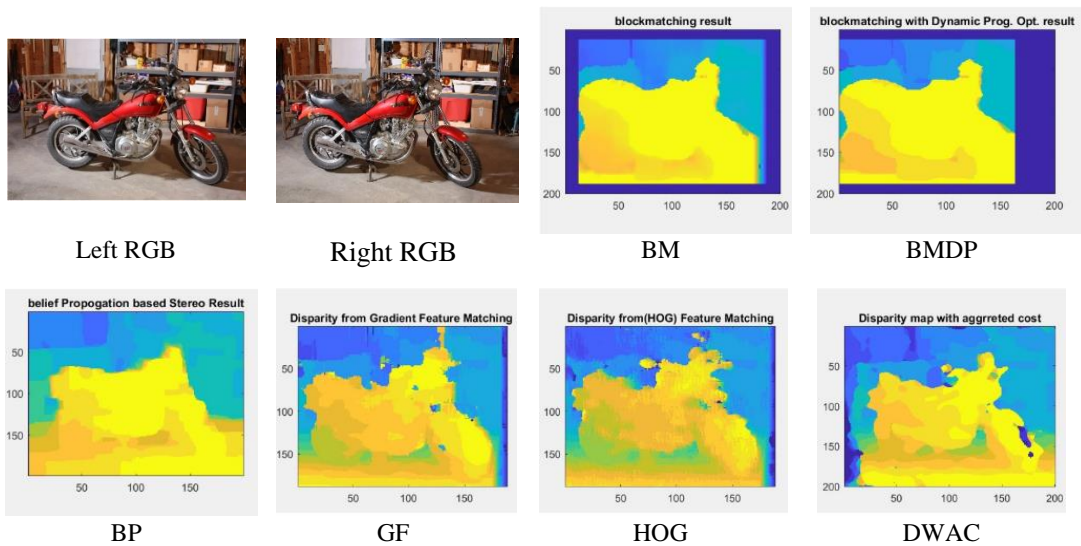


Figure 6.40. Left and right images and the disparity maps achieved for different matching algorithms based on SAD cost function.

Figure 6.41 presents the results for another example that has been a more difficult work to do as it is a less detailed image. It is seen that for this type of images, BM provides the least proper results while BP and HOG lead in better results. [Pinggera et al. \(2012\)](#) compared several matching techniques. They reported that HOG showed favorable results and better compared to Mutual Information or Local Self-Similarity descriptors.

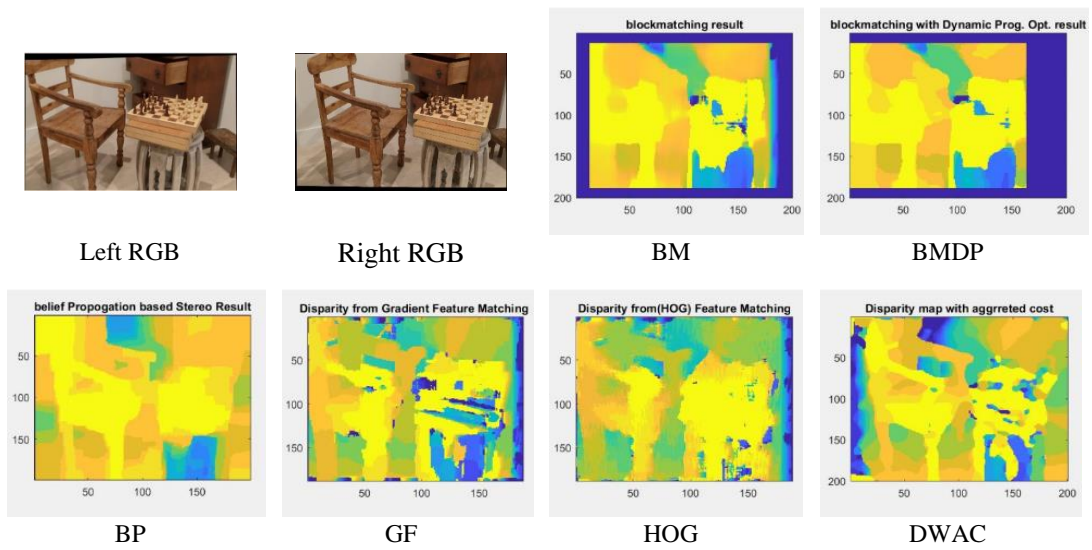
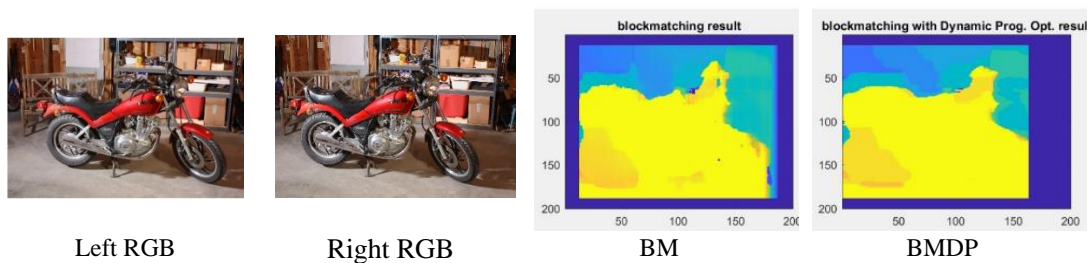


Figure 6.41. Left and right images and the disparity maps achieved for different matching algorithms based on SAD cost function.

## 6.4.2 MSE

Figure 6.42 provides the disparity maps that constructed using MSE cost function for different matching algorithms. The results are quite similar to the SAD algorithm. However, the disparity maps with MSE also are proper for BP, GF and HOG and for the rest the results are not very acceptable. Figure 6.43 provides an example of a difficult task of matching as the images contain less details and textures. [Aboali et al. \(2018\)](#) studied a stereo matching algorithm called Multistage Hybrid Median Filter. They used three cost functions including MSE, PSNR, and SSIM. In all cases of cost functions, the matching algorithm of Multistage Hybrid Median Filter had the best performance.



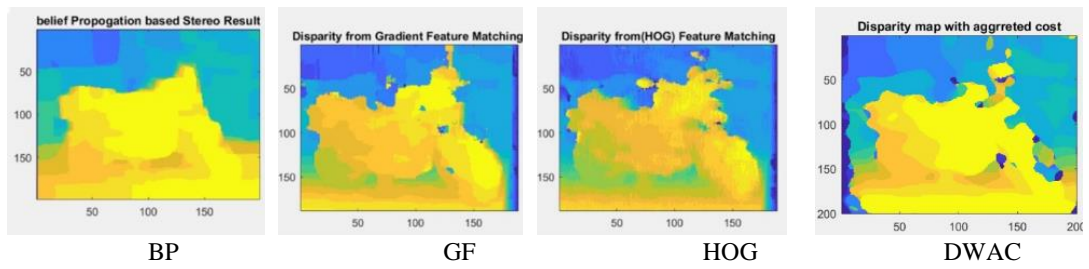


Figure 6.42. Left and right images and the disparity maps achieved for different matching algorithms based on MSE cost function.

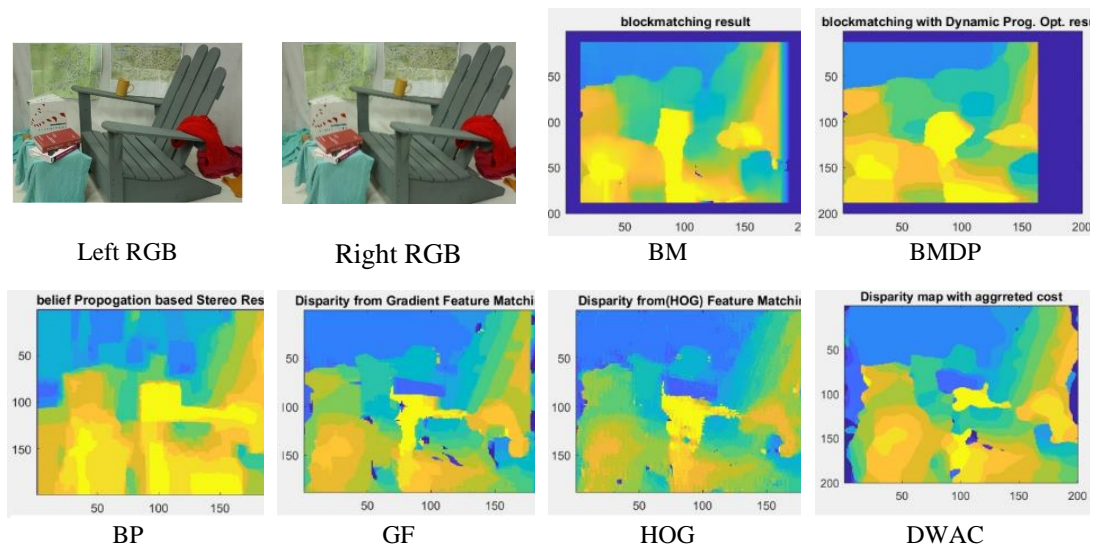


Figure 6.43. Left and right images and the disparity maps achieved for different matching algorithms based on MSE cost function.

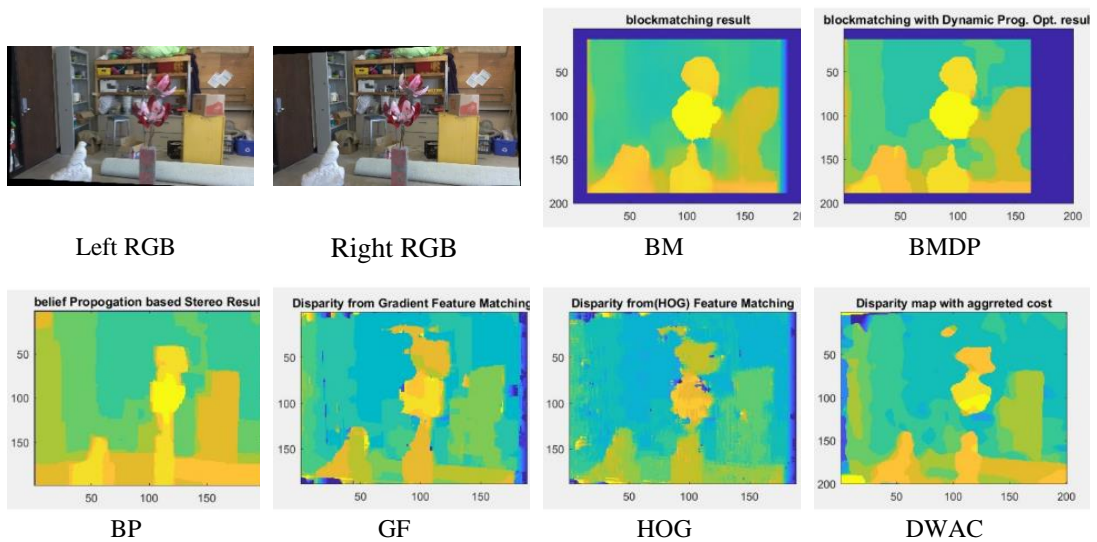


Figure 6.44. Left and right images and the disparity maps achieved for different matching algorithms based on MSE cost function.

### 6.4.3 NCC

Hereafter, the results obtained by NCC are presented. As figures 6.45 – 47 show, the performance of NCC was not as good as MSE and SAD for different matching algorithms. Here also the performance of HOG has been more precise and clear. [Hirschmuller and Scharstein \(2007\)](#) reported the same results. They found NCC the least interesting results while other techniques were much better.

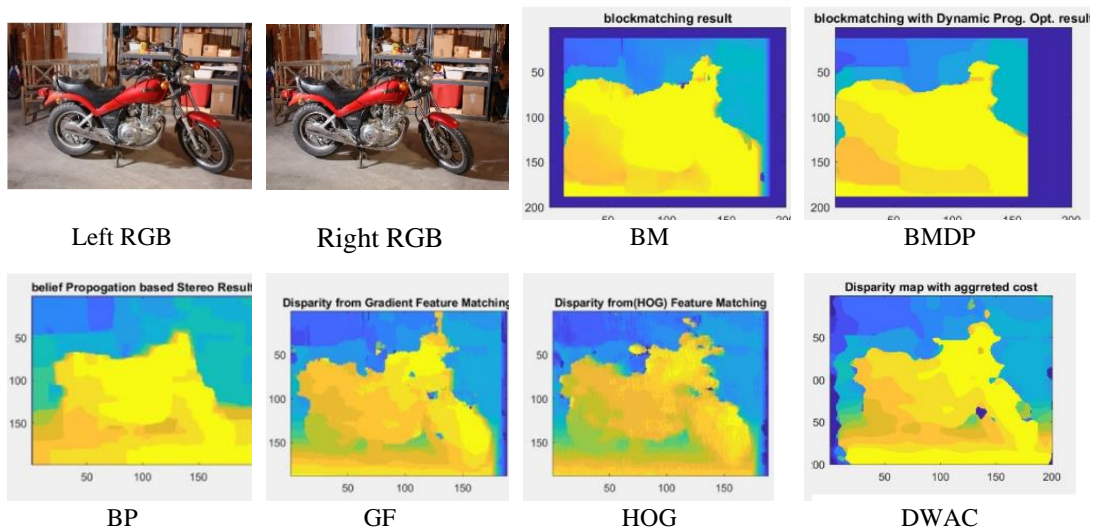


Figure 6.45. Left and right images and the disparity maps achieved for different matching algorithms based on NCC cost function.

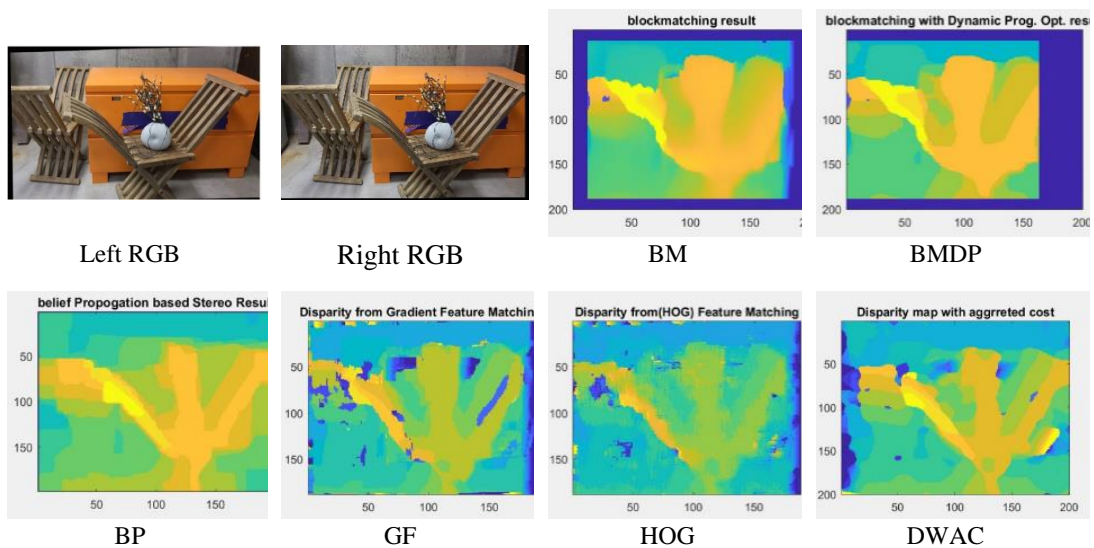


Figure 6.46. Left and right images and the disparity maps achieved for different matching algorithms based on NCC cost function.

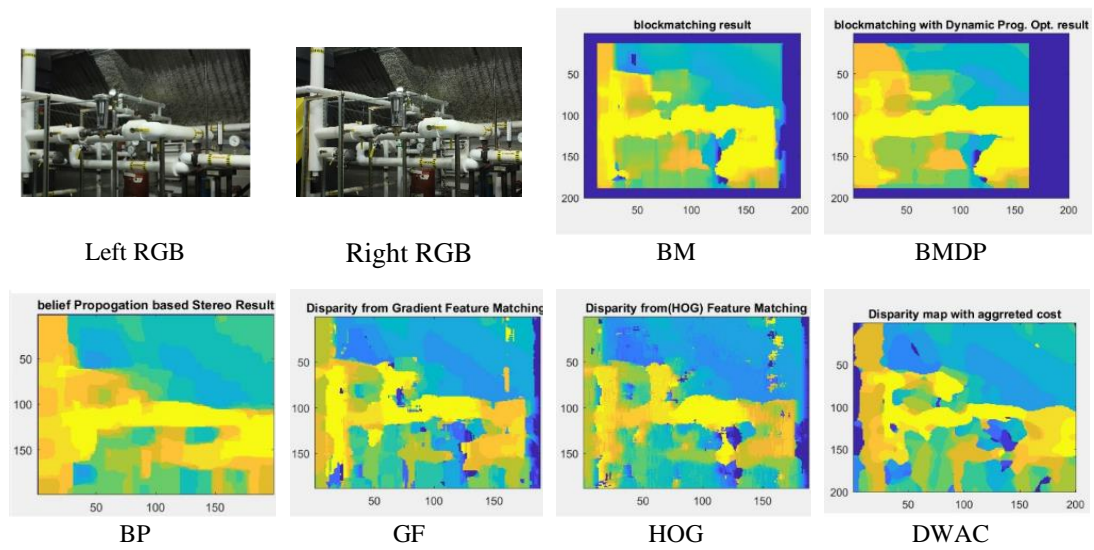


Figure 6.47. Left and right images and the disparity maps achieved for different matching algorithms based on NCC cost function.

Table 6-6 summarizes the minimum and maximum of average errors for all images used for disparity map development using different matching and cost algorithms. As the table indicates, the error strongly depends on the type of application, details, and textures inside the image. The best result was obtained by the use of BP and SAD.

**Table 6-6.** Minimum and maximum of average error for different algorithms.

Method	BM		BMDP		BP		GF		HOG		DWAC	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
MSE	0.113	0.853	0.106	0.903	0.272	0.933	0.099	0.854	0.128	0.905	0.043	0.841
NCC	0.236	0.881	0.156	0.925	0.315	0.941	0.153	0.904	0.246	0.91	0.121	0.892
SAD	0.291	0.919	0.253	0.94	0.347	0.964	0.21	0.919	0.295	0.94	0.102	0.904

Figure 6.48 shows the stereo matching accuracy for all image pairs used in this study. It is observed that BP had mostly the best accuracies, however for all algorithms the accuracy of matching has not been fixed and has experienced fluctuations for different types of images.

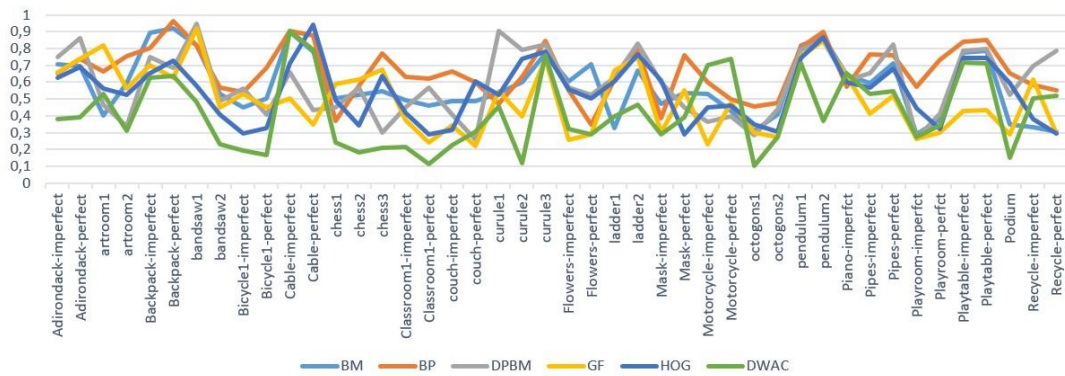


Figure 6.48. The whole comparison of all algorithms for all the images.

## 6.5 Spectral data for separation of plants

### 6.5.1 Separation using deep learning

For the separation of pepper from weeds, deep learning was utilized. Four networks were trained for considering the performance of the classification. Table 6-7 presents the plants and the number of samples used for building each network.

Table 6-7. Number of samples used for training the network.

Net	Plants	No of samples
1	Pepper, Bindweed, Nutsedge, Plantago	192
2	Pepper, Bindweed, Nutsedge, Plantago, Potentilla	241
3	Pepper, Bindweed, Nutsedge, Plantago, Potentilla, Sorrel	291

The deep network was a classifier based on RGB images. The spectral responses were transformed to scalograms. These scalograms of the form 2D images were used for training the network and classification. Examples of these scalograms have been given in Figure 6.49.



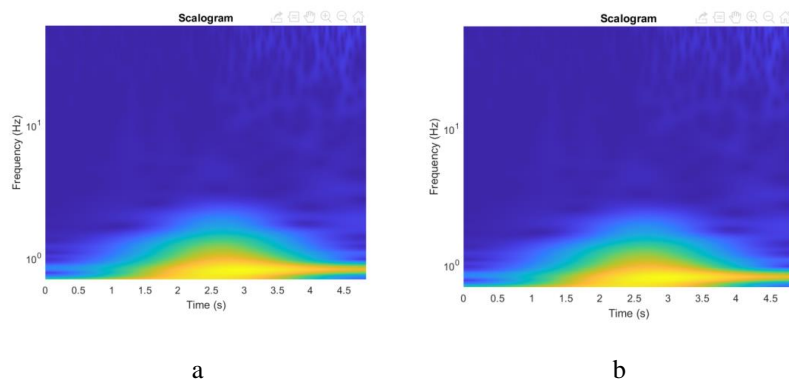


Figure 6.49. The scalogram of CWT colored as an RGB image; a) pepper and b) Bindweed.

Tables 6-8 to 6-10 provide the details of training of the networks. As the tables indicate, in all cases, the accuracy of networks reached 100%. Moshou et al. (2001) reported a neural network-based classifier for plants and weeds. Several networks were used and acceptable accuracies achieved. Wang et al. (2008) studied the spectral prediction of *Phytophthora infestans* infection on tomatoes using artificial neural networks. Benefitting from back-propagation neural networks, the correlation coefficients of 0.99 and 0.82 for field data and remote sensing image data were obtained. Li and He (2008) used back propagation networks for discrimination of varieties of tea plant based on spectral data where accuracy of 77.3 % was obtained. Nidamanuri (2020) also used artificial neural networks for the discrimination of tea plant varieties. It was reported that six out of nine varieties could be discriminated with accuracies ranging between 75 % and 80%.

Figure 6.50 illustrates the training process of the network for the last network (i.e. pepper and five weeds). It is observed that the training has been performed very well. In this figure, the most important element is the validation curve which has been improving and following the training data. It can be seen that in iteration 130, the network could completely discriminate the plants and weeds (i.e. 100% accuracy). Figure 6.51 presents the amount of loss function for each iteration. Minimizing loss function is based on the gradient descent algorithm. In every iteration, the gradient of the loss function is obtained and evaluated and then the weights for the descent algorithm are updated. In the figure, it can be seen that the training has been going

very well and the loss value for validation data has been gradually decreasing while following the training data.

**Table 6-8.** The training information of the network for pepper and three weeds.

Epoch	Iteration	Mini-batch Accuracy (%)	Validation Accuracy (%)	Mini- batch Loss	Validation Loss
1	1	20.00	46.87	2.0109	0.7489
3	10	80.00	75.00	0.4208	0.6102
5	40	73.33	75.00	52.76	0.4839
8	60	100.00	81.25	0.4354	0.4007
10	80	93.33	87.50	0.3902	0.2983
13	100	93.33	84.38	0.2370	0.2090
15	120	100.00	96.88	0.0596	0.1397
18	140	100.00	100.00	0.0308	0.1002
20	160	100.00	93.75	0.0240	0.1013
23	180	100.00	100.00	0.0410	0.0661
25	200	100.00	100.00	0.0243	0.0822

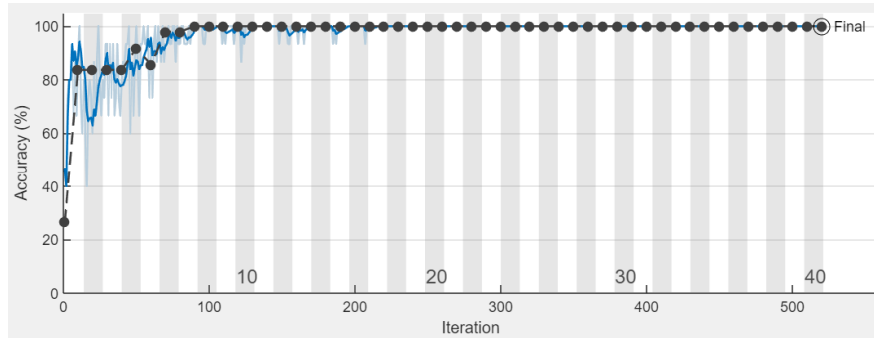
**Table 6-9.** The training information of the network for pepper and four weeds.

Epoch	Iteration	Mini-batch Accuracy (%)	Validation Accuracy (%)	Mini- batch Loss	Validation Loss
1	1	20.00	36.59	3.2537	0.8513
2	20	80.00	87.80	0.6030	0.4785
4	40	86.67	92.68	0.4330	0.2485
6	60	93.33	95.12	0.2118	0.1289
8	80	93.33	100.00	0.1713	0.0644
10	100	100.00	100.00	0.0755	0.0363
12	120	93.33	100.00	0.0750	0.0284
14	140	93.33	100.00	0.1924	0.0138
16	160	100.00	100.00	0.0521	0.0170
18	180	100.00	100.00	0.0197	0.0128
20	200	100.00	100.00	0.0070	0.0067

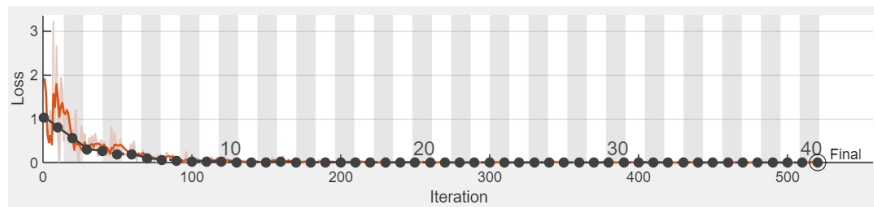
**Table 6-10.** The training information of the network for pepper and five weeds.

Epoch	Iteration	Mini-batch Accuracy (%)	Validation Accuracy (%)	Mini- batch Loss	Validation Loss
1	1	46.67	26.53	1.9153	1.0377
2	20	60.00	83.67	0.7427	0.5601
4	40	80.00	83.67	0.2899	0.2647
5	60	93.33	85.71	0.1805	0.1954
7	80	93.33	97.96	0.1487	0.0728

8	100	100.00	100.00	0.0674	0.0345
10	120	100.00	100.00	0.0217	0.0229
11	140	100.00	100.00	0.0056	0.0183
13	160	100.00	100.00	0.0058	0.0206
14	180	100.00	100.00	0.0159	0.0062
16	200	100.00	100.00	0.0109	0.0045

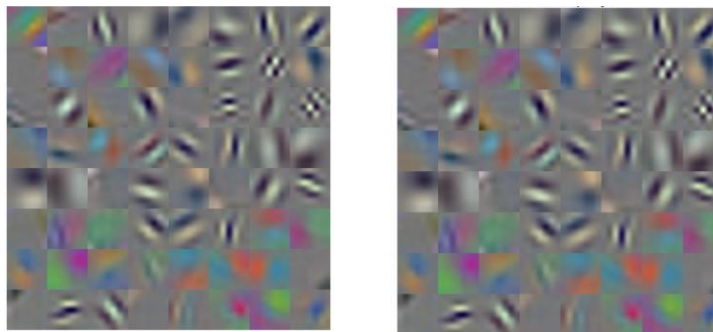


**Figure 6.50.** Diagram showing the network training process and accuracy per iteration.

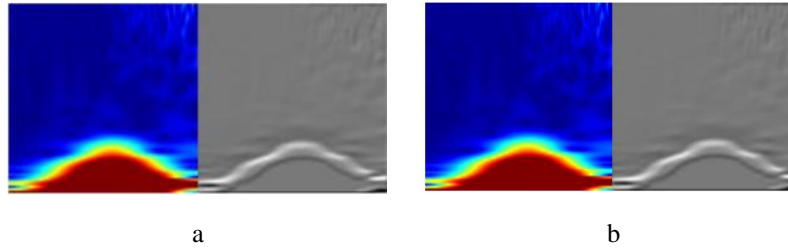


**Figure 6.51.** Diagram of loss function values per iteration during the network training process.

In CNNs the primary layers provide the most important information for image processing applications. These layers hold the information about edges, shapes and curves. Figure 6.52 represents the weights for the first convolutional layer. Each layer of the network consists of many channels. Figure 6.53 illustrates an original scalogram as the input of the network, and the strongest convolutional channel corresponding to it.

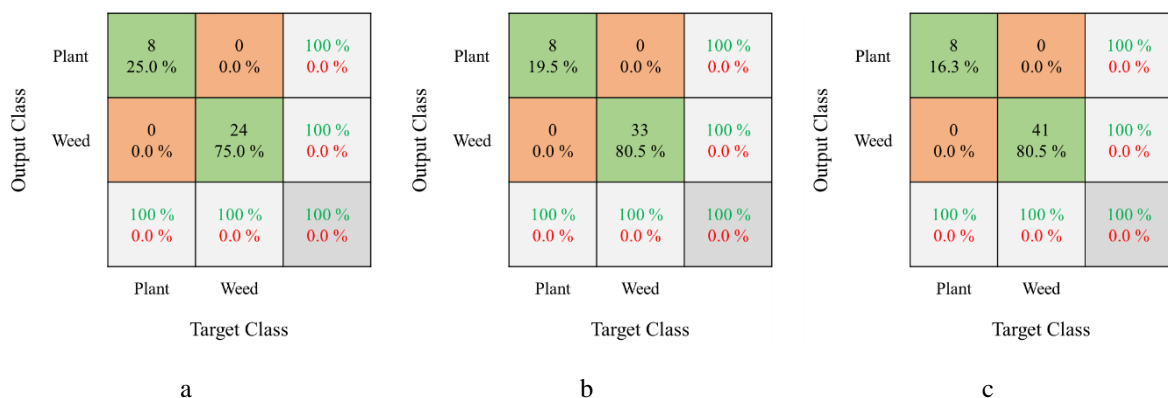


a b  
**Figure 6.52.** First convolutional layer weights; a) pepper and b) Bindweed.



a b  
**Figure 6.53.** First convolutional layer weights; a) pepper and b) Bindweed.

Figure 6.54 represents the confusion matrices for describing the networks performance. In these matrices, output class is the predicted classification and the target class refers to the actual classes. It is seen that for example for the first network, the algorithm has randomly chosen 8 plant samples and 16 weed samples that have been classified completely and correctly and no plant or weed has been classified in the other group. In a recent work, [Shirzadifar et al. \(2018\)](#) used soft independent modelling of the class analogy method for discrimination of three weeds based on spectral data. They reported NIR area as the best area for the discrimination. The proposed method could discriminate three weed species with 100% accuracy for 63 samples. [de Souza et al. \(2020\)](#) studied the differentiation of sugarcane from weeds based on spectral data and using soft independent modelling by class analogy, obtained an accuracy of 97.4.



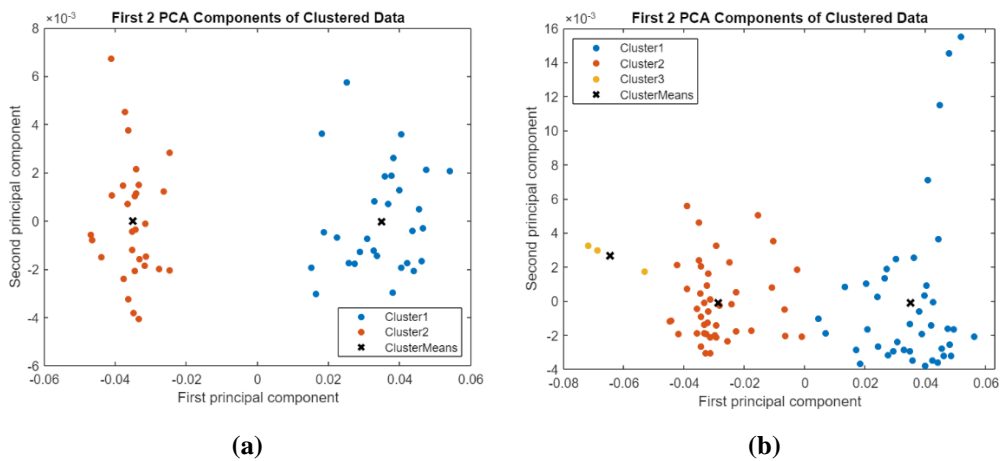
**Figure 6.54.** Confusion matrix of the network for validation samples; a) pepper and two weeds, b) pepper and three weeds, c) pepper and four weeds, and d) pepper and five weeds.

For the unseen data, still the performance of the network was significant. In case of training the network with four plants (i.e. bell pepper and three weeds), the accuracy was 0.8631. It seems that this has been because of the low number of samples. In case of training the network with five plants, the total of 241 samples, the accuracy of 1.00 was achieved. And in the last case, the network based on six different plants achieved an accuracy of equal to 0.9655 that still is a remarkable performance.

### 6.5.2 Separation using $k$ -means clustering

$k$ -means clustering was used for the classification of spectral data into different groups. Figure 6.55 represents the clustering results for pepper and different weeds (i.e. the weeds mentioned before). It is observed that the separation of pepper from one weed is perfectly done but for more weeds it was a bit hard to separate as still it was possible.

Figure 6.56 illustrates the result of  $k$ -means clustering of spectral data for pepper and five different weeds. The figure shows that the separation is pretty difficult based on this PCA result.



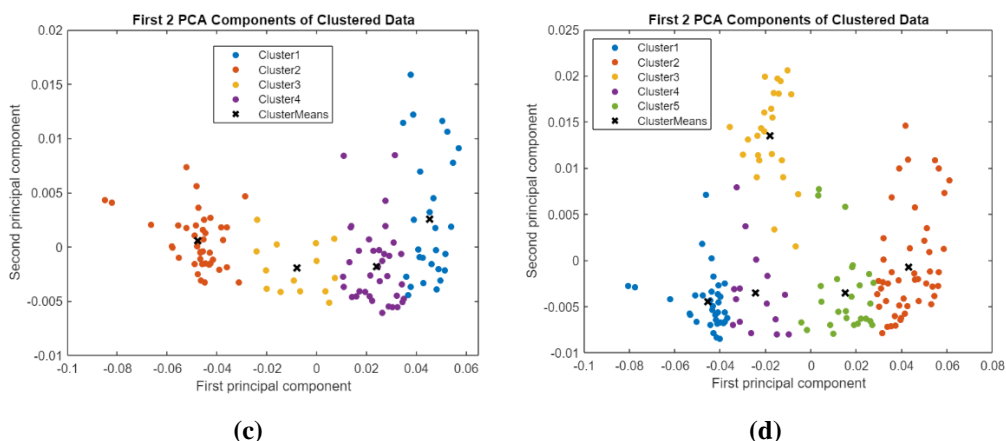


Figure 6.55. Scatter plots (PCA) of clustered spectral data: a) pepper and one weed, b) pepper and two weeds, c) pepper and three weeds, d) pepper and four weeds.

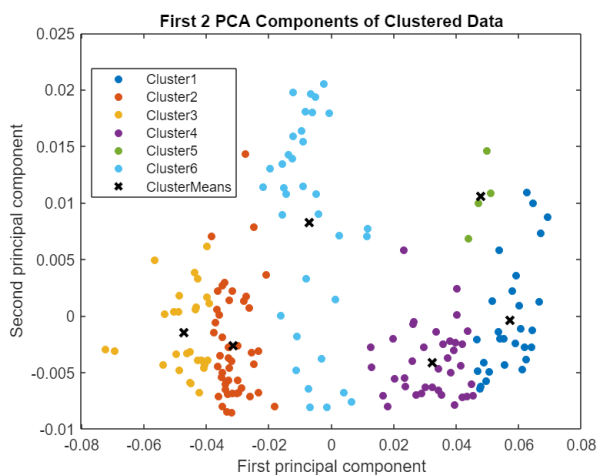


Figure 6.56. Scatter plot (PCA) of the clustered spectral data for pepper and five weeds.

Optimal number of clusters was searched using different methods (Fig. 6.57). It is seen that the data are best classifiable into two groups or into six groups. Therefore, based on Calinski-Harabasz method, the data can be classified into 6 groups. However, to improve the classification, the spectral data can be optimized as well. In this case, the clustering was done for the NIR part (i.e. 700 – 1000 nm) and the VIS part of data was removed. The result is remarkable and all plants are separated (Fig. 6.58).

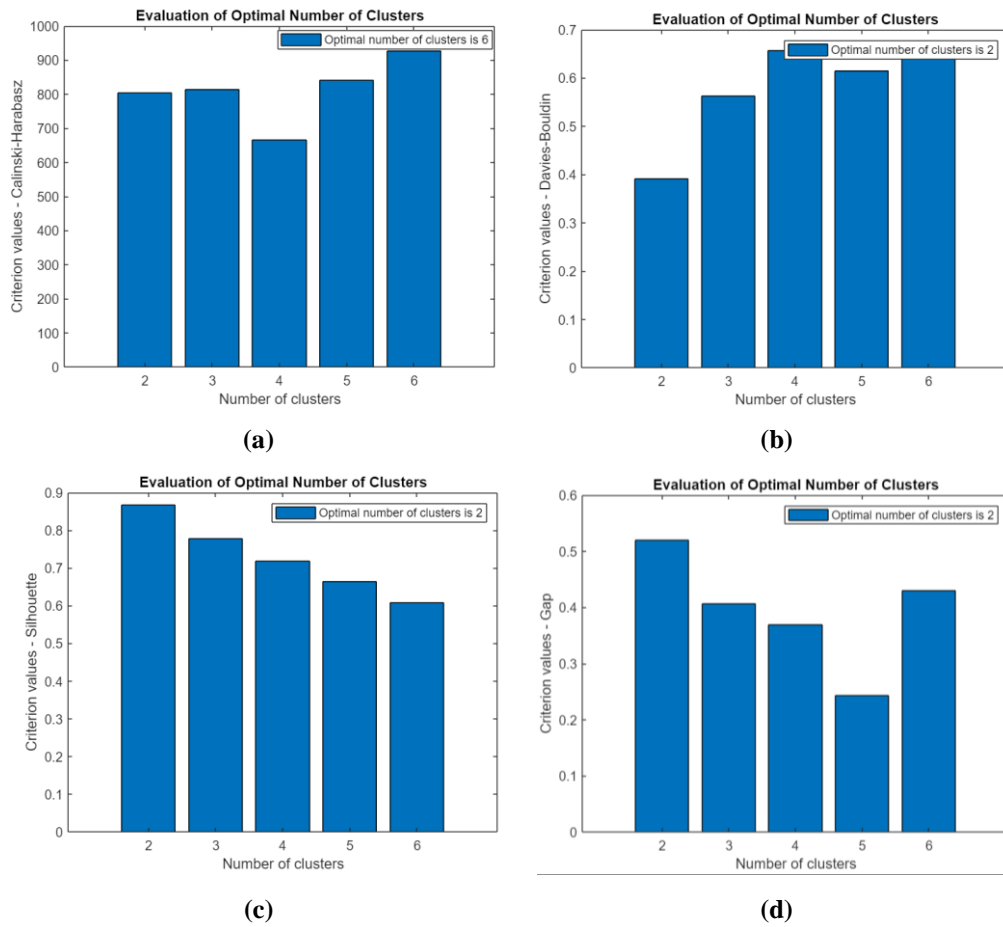


Figure 6.57. Optimal number of clusters; a) Calinski-Harabasz, b) Davies-Bouldin, c) Silhouette, and d) Gap

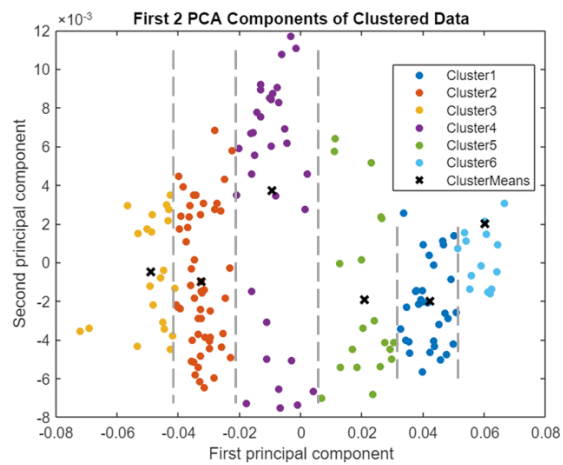


Figure 6.58. Scatter plot (PCA) of clustered data for 700-1000 nm.

## CONCLUSIONS

In this work, the use of stereo and multispectral imaging for growth monitoring and geometrical measurements of bell pepper plant was studied. Active and passive stereo imaging were employed for imaging bell pepper plant during a period of 35 days.

It was observed that the best color for the projected light in active stereo imaging will be the red color. The scattering of red light was less and the extraction of the light stripe was easier. For the prepared videos of light stripes with the same light thickness, the green and blue lights had bigger lines on the environment and were scattered more. However, the choice of color depends also on the camera that is used for imaging and the camera's characteristics should be considered in other works.

It was observed that the more the calibration images the better the calibration accuracy. As a rule of thumb 20 pairs is enough but it is suggested to provide more than 40 pairs of images to achieve a good calibration accuracy. As long as the calibration image pairs are enough and proper, the calibration is done with good precision for different stereo camera angles. The calibration accuracy was about one pixel for  $75^\circ$ ,  $90^\circ$  and  $105^\circ$  angles between the cameras that was acceptable for all cases. However, as a trade-off between field of view and angle view of the cameras, 90 degrees angle was the best. During calibration, it is important that the software identifies the correspondent images correctly.

For the angle of cameras, 90 degrees was a proper angle for calibration. However, the wide angles were better as the using of small angles brings a small field of view



and therefore, the leaves are not detected very well. The wider angles (larger than 90 degrees) provide better field of view and the plant can be seen clearly by the cameras. However, using only two images of left and right does not sufficiently cover all parts and branches of the plant.

The result of segmentation of the point clouds could count the leaves with the success rate of 96.053 %. The technique makes the segmentation much easier as the leaves that are overlaid in 2D images are in different height levels in 3D images. However, the segmentation of leaves that are overlaid and touched each other is still a difficult work. Also, the small leaves that fall in blind zone of stereo cameras have the possibility not to be detected and segmented.

The accuracy of estimation of leaf area and perimeter by active stereo imaging was acceptable. The accuracy for the estimation of other physical properties such as rectangularity, form factor etc. are less and for some properties less than 0.8. This comes from the reason that these factors are extracted from other leaf properties like leaf area or diagonals and the error of estimation gets multiplied several times and the result is not precise enough. In this case, scaling factors or using other techniques for finding the true estimation is required.

It was observed that active stereo imaging works properly until the number of leaves is small and the complexity of the canopy is not much. For plants with complex canopies and lots of leaves, the correspondence will be difficult or even impossible to do. Also, some of the leaves will be in the blind spot of cameras leading to the wrong number of leaves to count and detect.

Passive stereo imaging was also used for the estimation of physical properties of plants. The disparity map was obtained based on the calibration information. The overall success rate of the passive stereo algorithm for leaf counting was 84.32 %.

For the objects with textures that enough correspondent points can be found, and the use of a uniform light is possible, the use of passive stereo imaging will be easier and more promising. Another advantage can be the amount of processing time which is much less than active stereo imaging.

Different matching algorithms and cost functions were tested and compared. It was observed that generally BP algorithm provides better results for disparity map development. For cost functions, SAD provided better results compared to MSE and NCC. It was observed that the choice of matching algorithm and cost function highly depends on the type of application. This comes from the details and patterns present on the images. For example for the images with less details and uniform regions, BM provides the least proper results while BP and HOG lead in better results.

The use of spectral imaging for the detection and monitoring of plants was studied. To see if the spectral imaging can make the difference of plants, the spectral responses of pepper and five different weeds were obtained. The discrimination of plants based on the spectral data was almost impossible as the data were so large. Wavelet transform was used for dimension reduction. A classifier based on CNN was developed to discriminate pepper from weeds. For the CNN the pre-trained network called GoogLeNet was used. This network has been trained for over 1000 classification groups. The classifier could completely discriminate pepper and weeds. It was also seen that in case of difficulties in preparation of large datasets with samples over a thousand that is very important for deep learning techniques, the use of pretrained networks including GoogleNet, AlexNet, VGG16 etc. can be a promising choice for image processing and classification applications.

Spectral data also were classified using *k*-means clustering technique. This technique works perfectly for the separation of pepper and one weed, however, the separation of all plants were difficult. *k*-means clustering could separate all plants by using the data in the range of 700 – 1000 nm.

## SUGGESTIONS

The study showed that the use of stereo imaging can provide an inexpensive technique for the geometrical measurements of plants in early growth stages. For the applications where only 3D information is needed, active stereo imaging is recommended as no illumination is required. However, the more complicated the structure of the plant, the more the error of measurement. In case of larger plants, the use of more cameras from different angles of view will be necessary.

Based on the conclusions mentioned in the previous chapter, for the measurements of plant features during the entire growth period, the following is recommended.

- Utilize multiview imaging by using a movable system to take images in different angles or use several cameras and take images at several different angles. This involves more processing costs for matching images, but it will provide higher accuracy in developing the 3D models. In addition, this will improve the ability of the system in constructing the hidden or overlaid leaves.
- Fusion of stereo imaging with another depth measuring technique such as laser or ultrasound can also be promising. This technique, which is more popular because of Kinect cameras, has been used in other research work, however, for plant monitoring, higher accuracies are required. For example, Kinect cameras normally work well in low resolutions with imaging applications such as rooms or large objects.
- Image rectification requires hard work with cameras positioned far from each other. When the cameras are far apart, the matrices in the virtual work coordination system become massively large which can cause problems in MATLAB. In this regard, if the application allows, smaller baselines can be chosen for the stereo imaging system.

- Calibration of the system is quite critical and if something is not defined well, it will result in system inaccuracy. One of the important points is to make sure that the calibrator application selects the correspondent images correctly. This occurs with some calibrators and the correspondent images should be checked by the operator.
- The use of deep learning for separation of crop and weeds based on spectral data is promising. It is proposed to make a larger database of spectral responses for more crops and weeds and perform the classification to construct a robust classifier of crops and weeds.
- The separation of crops and weeds based on spectral data can be more necessary under farm conditions. In this regard, the development of a database of spectral responses in sunlight is required. This can bring difficulties as the sun light varies during the daytime and from one day to another.
- The use of tracking algorithms for monitoring and tracking of each leaf is interesting and conducive to extracting more information on the growth of leaves and plants. This will also help to produce growth models or to utilize growth models for the prediction purposes.
- As stereo matching is a challenging task and depends on the image and application, it is probable that new stereo matching techniques including deep learning for stereo matching can be promising and improve the stereo matching accuracy.
- The use of spectral data was properly well for discriminating plants and weeds. This can help to easily detect the plants, segment them and monitor their growth. In this study wavelet transform was used for the classification of spectral data. However, the use of other dimension-reduction algorithms such as Missing Values Ratio, Low Variance Filter, Random Forests and PCA can be interesting or even more promising for other applications.
- Based on the spectral responses and the previous research, it is recommended to use multispectral imaging in the red area of visible light and NIR as these areas provide more information about plants.

- When using *k*-means clustering method for the classification of crops and weeds, the spectral range of data is important. The best region for the classification is NIR which gives more information about various plants. However, fusion of visible and NIR data can also be more effective in some applications. The selection of spectral data for clustering is of high importance as well. In this study, spectral data with steps of one nanometer were used and reported. However, the usage of data with steps of 5 nm or 10 nm also can help to cluster data into groups easier.

## REFERENCES

- Aboali, M., Abd Manap, N. and Yusof, Z.M., 2017. Performance analysis between basic block matching and dynamic programming of stereo matching algorithm. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 9(2-13), pp.7-16.
- Aboali, M., Abd Manap, N. and Yusof, Z.M., 2018. A Multistage Hybrid Median Filter Design of Stereo Matching Algorithms on Image Processing. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, 10(4), pp.133-141.
- Aksoy, Eren Erdal, et al. (2015), 'Modeling leaf growth of rosette plants using infrared stereo image sequences', *Computers and electronics in agriculture*, 110, 78-90.
- Alchanatis, V., Ridel, L., Hetzroni, A. and Yaroslavsky, L., 2005. Weed detection in multi-spectral images of cotton fields. *Computers and Electronics in Agriculture*, 47(3), pp.243-260.
- Alenya, Guillem, Dellen, Babette, and Torras, Carme (2011), '3D modelling of leaves from color and ToF data for robotized plant measuring', 2011 IEEE International Conference on Robotics and Automation (IEEE), 3408-14.
- Amandep, K (2014), 'Image Segmentation Using Watershed Transform', *International Journal of Soft Computing and Engineering (IJSCE)*. University Putiala, India, 4 (1).
- Amean, Zainab Mohammed, Low, Tobias, and Hancock, Nigel (2021), 'Automatic leaf segmentation and overlapping leaf separation using stereo vision', *Array*, 100099.
- Amir, Y.M. and Thörnberg, B., 2017. High precision laser scanning of metallic surfaces. *International Journal of Optics*, 2017.
- Anton, B., 2014, October. Real time gestures recognition based on Hidden Markov Models. In 2014 International Conference on Mechanical Engineering, Automation and Control Systems (MEACS) (pp. 1-3). IEEE.
- Babellahi, F., Amodio, M.L., Marini, F., Chaudhry, M.M., de Chiara, M.L., Mastrandrea, L. and Colelli, G., 2020. Using chemometrics to characterise and unravel the near infra-red spectral changes induced in aubergine fruit by chilling injury as influenced by storage time and temperature. *Biosystems Engineering*, 198, pp.137-146.

- Bagavathiappan, S., Lahiri, B.B., Saravanan, T., Philip, J. and Jayakumar, T., 2013. Infrared thermography for condition monitoring—A review. *Infrared Physics & Technology*, 60, pp.35-55.
- Bahrami, Ch. (2015), 'Dynamical simulation of spring canola growth under water limit conditions using Aqua Crop model', (Eurmieh University).
- Barzegar, Z. (2013), 'The effect of different growth media on growth of bell pepper plant', (Industrial University of Isfahan).
- Bell IE, Baranoski GV. Reducing the dimensionality of plant spectral databases. *IEEE Transactions on Geoscience and Remote Sensing*, 2004, 42(3): 570–576.
- Beraldin, J.A., Blais, F., Cournoyer, L., Godin, G., Rioux, M. and Taylor, J., 2003. Active 3D sensing. National Research Council of Canada.
- Besse, F., Rother, C., Fitzgibbon, A. and Kautz, J., 2014. Pmbp: Patchmatch belief propagation for correspondence field estimation. *International Journal of Computer Vision*, 110(1), pp.2-13.
- Bi, S., Gu, Y., Zou, J., Wang, L., Zhai, C. and Gong, M., 2021. High precision optical tracking system based on near infrared trinocular stereo vision. *Sensors*, 21(7), p.2528.
- Bianco, G., Gallo, A., Bruno, F. and Muzzupappa, M., 2013. A comparative analysis between active and passive techniques for underwater 3D reconstruction of close-range objects. *Sensors*, 13(8), pp.11007-11031.
- Bianco, Gianfranco, et al. (2013), 'A comparative analysis between active and passive techniques for underwater 3D reconstruction of close-range objects', *Sensors*, 13 (8), 11007-31.
- Billiot, B., 2013. Conception d'un dispositif d'acquisition d'images agronomiques 3D en extérieur et développement des traitements associés pour la détection et la reconnaissance de plantes et de maladies (Doctoral dissertation, Dijon).
- Billiot, B., Cointault, F. and Gouton, P., 2013b, September. Mesure de netteté basée sur les descripteurs généralisés de Fourier appliquée à la reconstruction 3D par Shape from Focus. In 24ème colloque GRETSI (pp. 4-pages).
- Billiot, B., Cointault, F., Journaux, L., Simon, J.C. and Gouton, P., 2013a. 3D image acquisition system based on shape from focus technique. *Sensors*, 13(4), pp.5040-5053.
- Bolkas, D., 2019. Terrestrial laser scanner intensity correction for the incidence angle effect on surfaces with different colours and sheens. *International Journal of Remote Sensing*, 40(18), pp.7169-7189.

- Bozokalfa, M Kadri and Kilic, Murat (2010), 'Mathematical modeling in the estimation of pepper (*Capsicum annum L.*) fruit volume', *Chilean journal of agricultural research*, 70 (4), 626-32.
- Cajal, C., Santolaria, J., Samper, D. and Garrido, A., 2015. Simulation of laser triangulation sensors scanning for design and evaluation purposes. *International Journal of Simulation Modelling*, 14(2), pp.250-264.
- Carter, G.A. and Knapp, A.K., 2001. Leaf optical properties in higher plants: linking spectral characteristics to stress and chlorophyll concentration. *American journal of botany*, 88(4), pp.677-684.
- Chaerle, L., Lenk, S., Leinonen, I., Jones, H.G., Van Der Straeten, D. and Buschmann, C., 2009. Multi-sensor plant imaging: Towards the development of a stress-catalogue. *Biotechnology Journal: Healthcare Nutrition Technology*, 4(8), pp.1152-1167.
- Chakravorty, Geetanjali, et al. (2015), 'An experimental method to estimate the growth-rate of a leaf using image processing and solving an inverse growth-problem', 2nd International and 17th National Conference on Machines and Mechanisms, INaCoMM.
- Chan, B.M.H., 2002. *A miniaturized 3-D endoscopic system using active stereo-vision* (Doctoral dissertation).
- Chang, J.R. and Chen, Y.S., 2018. Pyramid stereo matching network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5410-5418).
- Che, E. and Olsen, M.J., 2019. An efficient framework for mobile lidar trajectory reconstruction and Mo-norvana segmentation. *Remote Sensing*, 11(7), p.836.
- Chen, Y.S., Hung, Y.P. and Fuh, C.S., 2001. Fast block matching algorithm based on the winner-update strategy. *IEEE Transactions on Image Processing*, 10(8), pp.1212-1222.
- Chéné, Y., Rousseau, D., Lucidarme, P., Bertheloot, J., Caffier, V., Morel, P., Belin, É. and Chapeau-Blondeau, F., 2012. On the use of depth camera for 3D phenotyping of entire plants. *Computers and Electronics in Agriculture*, 82, pp.122-127.
- Chiang, P.J. and Lin, C.H., 2022. Active Stereo Vision System with Rotated Structured Light Patterns and Two-Step Denoising Process for Improved Spatial Resolution. *Optics and Lasers in Engineering*, 152, p.106958.
- Costa, Ana Patrícia, Pôças, Isabel, and Cunha, Mário (2016), 'Estimating the leaf area of cut roses in different growth stages using image processing and allometrics', *Horticulturae*, 2 (3), 6.



- Coste, A., 2013. Multiple Baseline Stereo. Computer Vision and Image Processing Course work. University Hospital of Saint-Etienne.
- Dalal, N. and Triggs, B., 2005, June. Histograms of oriented gradients for human detection. In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) (Vol. 1, pp. 886-893). Ieee.
- de Jong, S.M., Addink, E.A., Hoogenboom, P. and Nijland, W., 2012. The spectral response of *Buxus sempervirens* to different types of environmental stress—A laboratory experiment. *ISPRS Journal of Photogrammetry and Remote Sensing*, 74, pp.56-65.
- de Souza, Micael Felipe, et al. (2020), 'Spectral differentiation of sugarcane from weeds', *biosystems engineering*, 190, 41-46.
- De Vylder, Jonas, et al. (2011), 'Leaf segmentation and tracking using probabilistic parametric active contours', *International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications (Springer)*, 75-85.
- DEMİREL, K., Levent, G.E.N.Ç., MENDEŞ, M., SAÇAN, M. and KIZIL, Ü., 2012. Estimation of growth curve parameters for pepper (*Capsicum annuum* cv. Kapija) under deficit irrigation conditions. *Ege Üniversitesi Ziraat Fakültesi Dergisi*, 49(1), pp.37-43.
- DEMİREL, Kürşad, et al. (2012), 'Estimation of Growth Curve Parameters for Pepper (*Capsicum annuum* cv. Kapija) Under Deficit Irrigation Conditions', *Ege Üniversitesi Ziraat Fakültesi Dergisi*, 49 (1), 37-43.
- Denker, K., 2014. Acquisition and On-line Reconstruction of 3D Point Data from Hand-held Laser Scanners and Multi-camera Stereo-matching (Doctoral dissertation, Technische Universität Kaiserslautern).
- Di Stefano, L., Marchionni, M. and Mattocchia, S., 2004. A fast area-based stereo matching algorithm. *Image and vision computing*, 22(12), pp.983-1005.
- Dias, P., Matos, M. and Santos, V., 2006. 3D reconstruction of real world scenes using a low-cost 3D range scanner. *Computer-Aided Civil and Infrastructure Engineering*, 21(7), pp.486-497.
- Dipanda, A. and Woo, S., 2005. Efficient correspondence problem-solving in 3-D shape reconstruction using a structured light system. *Optical Engineering*, 44(9), p.093602.
- Dipanda, A., Woo, S., Marzani, F. and Bilbault, J.M., 2003. 3-D shape reconstruction in an active stereo vision system using genetic algorithms. *Pattern recognition*, 36(9), pp.2143-2159.

- Do, Y., 1999, July. Application of neural networks for stereo-camera calibration. In *IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339)* (Vol. 4, pp. 2719-2722). IEEE.
- Dong, Z., Sun, X., Liu, W. and Yang, H., 2018. Measurement of free-form curved surfaces using laser triangulation. *Sensors*, 18(10), p.3527.
- Donné, S., De Vylder, J., Goossens, B. and Philips, W., 2016. MATE: Machine learning for adaptive calibration template detection. *Sensors*, 16(11), p.1858.
- Dorsch, R.G., Häusler, G. and Herrmann, J.M., 1994. Laser triangulation: fundamental uncertainty in distance measurement. *Applied optics*, 33(7), pp.1306-1314.
- Durgante FM, Higuchi N, Almeida A, Vicentini A. Species spectral signature: discriminating closely related plant species in the Amazon with near-infrared leaf-spectroscopy. *Forest Ecology and Management*, 2013, 291: 240–248.
- Eigen, D., Puhersch, C. and Fergus, R., 2014. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27.
- Esteban, I., Dijk, J. and Groen, F., 2010, October. Fit3d toolbox: multiple view geometry and 3d reconstruction for matlab. In *Electro-Optical Remote Sensing, Photonic Technologies, and Applications IV* (Vol. 7835, pp. 206-214). SPIE.
- Fan, Xing-Rong, et al. (2015), 'A knowledge-and-data-driven modeling approach for simulating plant growth: A case study on tomato growth', *Ecological Modelling*, 312, 363-73.
- FAO 'FAO stats', <<http://www.fao.org/faostat/en/#>>, accessed.
- Farjon, Guy, et al. (2021), 'Leaf Counting: Fusing Network Components for Improved Accuracy', *Frontiers in Plant Science*, 12, 1063.
- Feng, J., Zeng, L. and He, L., 2019. Apple fruit recognition algorithm based on multi-spectral dynamic image analysis. *Sensors*, 19(4), p.949.
- Garibaldi-Márquez, F., Flores, G., Mercado-Ravell, D.A., Ramírez-Pedraza, A. and Valentín-Coronado, L.M., 2022. Weed Classification from Natural Corn Field-Multi-Plant Images Based on Shallow and Deep Learning. *Sensors*, 22(8), p.3021.
- Gélard, W., Devy, M., Herbulot, A. and Burger, P., 2017, February. Model-based segmentation of 3D point clouds for phenotyping sunflower plants. In *12. International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*.

- Geng, J., 2011. Structured-light 3D surface imaging: a tutorial. *Advances in Optics and Photonics*, 3(2), pp.128-160.
- Gigilashvili, D., Shi, W., Wang, Z., Pedersen, M., Hardeberg, J.Y. and Rushmeier, H., 2021a. The role of subsurface scattering in glossiness perception. *ACM Transactions on Applied Perception (TAP)*, 18(3), pp.1-26.
- Gigilashvili, D., Thomas, J.B., Hardeberg, J.Y. and Pedersen, M., 2021b. Translucency perception: A review. *Journal of Vision*, 21(8), pp.4-4.
- Giménez, C, et al. (2013), 'VegSyst, a simulation model of daily crop growth, nitrogen uptake and evapotranspiration for pepper crops for use in an on-farm decision support system', *Irrigation Science*, 31 (3), 465-77.
- Goudriaan, Jan and Van Laar, HH (2012), *Modelling potential crop growth processes: textbook with exercises (2: Springer Science & Business Media)*.
- Guo, Jingwei and Xu, Lihong (2017), 'Automatic segmentation for plant leaves via multiview stereo reconstruction', *Mathematical Problems in Engineering*, 2017.
- Gupta, S.D. and Ibaraki, Y. eds., 2014. *Plant image analysis: fundamentals and applications*. CRC Press, Chapter 9: 179-206.
- Gyaourova, A., Kamath, C. and Cheung, S.C., 2003. Block matching for object tracking (No. UCRL-TR-200271). Lawrence Livermore National Lab.(LLNL), Livermore, CA (United States).
- Hamdi, N., Auhmani, K. and Hassani, M.M., 2014. A comparative study of dimension reduction methods combined with wavelet transform applied to the classification of mammographic images. *International Journal of Computer Science & Information Technology*, 6(6), p.139.
- Hamzah, R.A., Abd Rahim, R. and Noh, Z.M., 2010, July. Sum of absolute differences algorithm in stereo correspondence problem for stereo matching in computer vision application. In *2010 3rd International Conference on Computer Science and Information Technology (Vol. 1, pp. 652-657)*. IEEE.
- Hartley, R. and Zisserman, A., 2003. *Multiple view geometry in computer vision*. Cambridge university press.
- Hashemi, S., Kiani, S., Noroozi, N. and Moghaddam, M.E., 2010. An image contrast enhancement method based on genetic algorithm. *Pattern Recognition Letters*, 31(13), pp.1816-1824.

- Havens, Kirk J and Sharp, Edward (2015), Thermal imaging techniques to survey and monitor animals in the wild: a methodology (Academic Press).
- Heise, P., Jensen, B., Klose, S. and Knoll, A., 2015, May. Fast dense stereo correspondences by binary locality sensitive hashing. In 2015 IEEE International Conference on Robotics and Automation (ICRA) (pp. 105-110). IEEE.
- Heo, Y.S., Lee, K.M. and Lee, S.U., 2010. Robust stereo matching using adaptive normalized cross-correlation. *IEEE Transactions on pattern analysis and machine intelligence*, 33(4), pp.807-822.
- Hirschmuller, H. and Scharstein, D., 2007, June. Evaluation of cost functions for stereo matching. In 2007 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.
- Hossain, S. and Lin, X., 2022. Efficient Stereo Depth Estimation for Pseudo LiDAR: A Self-Supervised Approach Based on Multi-Input ResNet Encoder. arXiv preprint arXiv:2205.08089.
- Huang, Y., Dong, W., Sanaeifar, A., Wang, X., Luo, W., Zhan, B., Liu, X., Li, R., Zhang, H. and Li, X., 2020. Development of simple identification models for four main catechins and caffeine in fresh green tea leaf based on visible and near-infrared spectroscopy. *Computers and electronics in agriculture*, 173, p.105388.
- Isa, M.A. and Lazoglu, I., 2017. Design and analysis of a 3D laser scanner. *Measurement*, 111, pp.122-133.
- Jang, W., Je, C., Seo, Y. and Lee, S.W., 2013. Structured-light stereo: Comparative analysis and integration of structured-light and active stereo for measuring dynamic shape. *Optics and Lasers in Engineering*, 51(11), pp.1255-1264.
- Jang, W., Je, C., Seo, Y. and Lee, S.W., 2013. Structured-light stereo: Comparative analysis and integration of structured-light and active stereo for measuring dynamic shape. *Optics and Lasers in Engineering*, 51(11), pp.1255-1264.
- Jang, W., Je, C., Seo, Y. and Lee, S.W., 2013. Structured-light stereo: Comparative analysis and integration of structured-light and active stereo for measuring dynamic shape. *Optics and Lasers in Engineering*, 51(11), pp.1255-1264.
- Jin, J. and Tang, L., 2009. Corn plant sensing using real-time stereo vision. *Journal of Field Robotics*, 26(6-7), pp.591-608.

- Kaasalainen, S., Åkerblom, M., Nevalainen, O., Hakala, T. and Kaasalainen, M., 2018. Uncertainty in multispectral lidar signals caused by incidence angle effects. *Interface Focus*, 8(2), p.20170033.
- Kacira, M. and Ling, P.P., 2001. Design and development of an automated and Non-contact sensing system for continuous monitoring of plant health and growth. *Transactions of the ASAE*, 44(4), p.989.
- Kandiannan, K, et al. (2002), 'Allometric model for leaf area estimation in black pepper (*Piper nigrum* L.)', *Journal of Agronomy and Crop Science*, 188 (2), 138-40.
- Kashani, A.G., Olsen, M.J., Parrish, C.E. and Wilson, N., 2015. A review of LiDAR radiometric processing: From ad hoc intensity correction to rigorous radiometric calibration. *Sensors*, 15(11), pp.28099-28128.
- Kendall, A., Martirosyan, H., Dasgupta, S., Henry, P., Kennedy, R., Bachrach, A. and Bry, A., 2017. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE international conference on computer vision* (pp. 66-75).
- Khojastehnazhand, Mostafa, Mohammadi, Vahid, and Minaei, Saeid (2019), 'Maturity detection and volume estimation of apricot using image processing technique', *Scientia Horticulturae*, 251, 247-51.
- Kihlström, H., 2019. Active Stereo Reconstruction using Deep Learning. Master thesis. Linköping University. Sweden.
- Laba M, Tsai F, Ogurcak D, Smith S, Richmond ME. Field determination of optimal dates for the discrimination of invasive wetland plant species using derivative spectral analysis. *Photogrammetric Engineering and Remote Sensing*, 2005, 71(5): 603–611.
- Latimer, W., 2015. Understanding laser-based 3D triangulation methods. *Vision Systems Design*, 20(6).
- Lee, Jong Whan (2008), 'Machine vision monitoring system of lettuce growth in a state-of-the-art greenhouse', *Modern Physics Letters B*, 22 (11), 953-58.
- Lelas, M. and Pribanić, T., 2016. Accurate stereo matching using pixel normalized cross correlation in time domain. *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije*, 57(1), pp.46-57.
- Li X, He Y. Discriminating varieties of tea plant based on Vis/NIR spectral characteristics and using artificial neural networks. *Biosystems Engineering*, 2008, 99(3): 313–321.
- Li, D., Xu, L., Tang, X.S., Sun, S., Cai, X. and Zhang, P., 2017. 3D imaging of greenhouse plants with an inexpensive binocular stereo vision system. *Remote Sensing*, 9(5), p.508.

- Li, Dawei, et al. (2020), 'A leaf segmentation and phenotypic feature extraction framework for multiview stereo plant point clouds', *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 2321-36.
- Li, Lei, Zhang, Qin, and Huang, Danfeng (2014), 'A review of imaging techniques for plant phenotyping', *Sensors*, 14 (11), 20078-111.
- Li, Xiaobin, Wang, Yushun, and Fu, Lihong (2016), 'Monitoring lettuce growth using K-means color image segmentation and principal component analysis method', *Transactions of the Chinese Society of Agricultural Engineering*, 32 (12), 179-86.
- Li, Y.F. and Chen, S.Y., 2003. Automatic recalibration of an active structured light vision system. *IEEE Transactions on Robotics and Automation*, 19(2), pp.259-268.
- Lin, F., Zhang, D., Huang, Y., Wang, X. and Chen, X., 2017. Detection of corn and weed species by the combination of spectral, shape and textural features. *Sustainability*, 9(8), p.1335.
- Liu, H., Wang, R., Xia, Y. and Zhang, X., 2020. Improved cost computation and adaptive shape guided filter for local stereo matching of low texture stereo images. *Applied Sciences*, 10(5), p.1869.
- Loch, B.I., Belward, J.A. and Hanan, J.S., 2005. Application of surface fitting techniques for the representation of leaf surfaces. In *Proceedings of the International Congress on Modelling and Simulation (MODSIM05)* (pp. 1272-1278). Modelling and Simulation Society of Australia and New Zealand Inc..
- Longchamps L, Panneton B, Samson G, Leroux GD, Thériault R. Discrimination of corn, grasses and dicot weeds by their UV-induced fluorescence spectral signature. *Precision Agriculture*, 2010, 11(2): 181–197.
- Louargant M, Jones G, Faroux R, Paoli JN, Maillot T, Gée C, Villette S. Unsupervised classification algorithm for early weed detection in row-crops by combining spatial and spectral information. *Remote Sensing*, 2018, 10(5): 761–779.
- Luisa España-Boquera, M., Cárdenas-Navarro, R., López-Pérez, L., Castellanos-Morales, V. and Lobit, P., 2006. Estimating the nitrogen concentration of strawberry plants from its spectral response. *Communications in soil science and plant analysis*, 37(15-20), pp.2447-2459.
- Marcelis, LFM, et al. (2006), 'Modelling dry matter production and partitioning in sweet pepper', *III International Symposium on Models for Plant Growth, Environmental Control and Farm Management in Protected Cultivation* 718, 121-28.

- Massimiliano, P.E.P.E., Alfio, V.S., Costantino, D. and Scaringi, D., 2022. Data for 3D reconstruction and Point Cloud classification using Machine Learning in Cultural Heritage environment. *Data in Brief*, p.108250.
- Memon, Q. and Khan, S., 2001. Camera calibration and three-dimensional world reconstruction of stereo-vision using neural networks. *International Journal of Systems Science*, 32(9), pp.1155-1159.
- Moeslund, Thomas B, Aagaard, Michael, and Lerche, Dennis (2005), '3d pose estimation of cactus leaves using an active shape model', 2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05)-Volume 1 (1: IEEE), 468-73.
- Morris, J.A., 2006. Design of an Active Stereo Vision 3D Scene Reconstruction System Based on the Linear Position Sensor Module.
- Moshou D, Vrindts E, De Ketelaere B, De Baerdemaeker J, Ramon H. A neural network based plant classifier. *Computers and Electronics in Agriculture*, 2001, 31(1): 5–16.
- Nguyen, M., Chan, Y.H., Delmas, P. and Gimel'farb, G., 2013, November. Symmetric dynamic programming stereo using block matching guidance. In 2013 28th International Conference on Image and Vision Computing New Zealand (IVCNZ 2013) (pp. 88-93). IEEE.
- Nguyen, Thuy Tuong, et al. (2016), 'Comparison of structure-from-motion and stereo vision techniques for full in-field 3d reconstruction and phenotyping of plants: An investigation in sunflower', 2016 ASABE Annual International Meeting (American Society of Agricultural and Biological Engineers), 1.
- Nichols, S.A., 2001. *Improvement of the Camera Calibration Through the Use of Machine Learning Techniques* (Doctoral dissertation, Division of Colleges & Universities, Florida Board of Education).
- Nidamanuri RR. Hyperspectral discrimination of tea plant varieties using machine learning, and spectral matching methods. *Remote Sensing Applications: Society and Environment*, 2020, 19: 100350.
- Nidamanuri RR. Hyperspectral discrimination of tea plant varieties using machine learning, and spectral matching methods. *Remote Sensing Applications: Society and Environment*, 2020, 19: 100350.
- Niku, S.B., 2001. Introduction to robotics: analysis, systems, applications (Vol. 7). New Jersey: Prentice hall.

- Noble SD, Brown RB. August. Plant species discrimination using spectral/spatial descriptive statistics. In Proceedings of the 1<sup>st</sup> International Workshop on Computer Image Analysis in Agriculture held in Potsdam Germany, 2009, 27–28.
- Patil, Smita, Soma, Shridevi, and Nandyal, Suvarna (2013), 'Identification of growth rate of plant based on leaf features using digital image processing techniques', *Int J Emerg Technol Adv Eng*, 3.
- Paturkar, A., Sen Gupta, G. and Bailey, D., 2022. Plant trait measurement in 3D for growth monitoring. *Plant Methods*, 18(1), pp.1-15.
- Pears, Nick, Liu, Yonghuai, and Bunting, Peter (2012), *3D imaging, analysis and applications* (3: Springer).
- Peiravi, A. and Taabbodi, B., 2010. A reliable 3D laser triangulation-based scanner with a new simple but accurate procedure for finding scanner parameters. *Journal of American Science*, 6(5), pp.80-85.
- Peiravi, A. and Taabbodi, B., 2010. A reliable 3D laser triangulation-based scanner with a new simple but accurate procedure for finding scanner parameters. *Journal of American Science*, 6(5), pp.80-85.
- Pinggera, P., Breckon, T. and Bischof, H., 2012. On cross-spectral stereo matching using dense gradient features. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (Vol. 2, p. 3).
- Poggi, M., Pallotti, D., Tosi, F. and Mattoccia, S., 2019. Guided stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 979-988).
- Polder, G, et al. (2007), 'Correcting and matching time sequence images of plant leaves using penalized likelihood warping and robust point matching', *Computers and Electronics in Agriculture*, 55 (1), 1-15.
- Rusinkiewicz, S., Hall-Holt, O. and Levoy, M., 2002. Real-time 3D model acquisition. *ACM Transactions on Graphics (TOG)*, 21(3), pp.438-446.
- Rusinkiewicz, S., Hall-Holt, O. and Levoy, M., 2002. Real-time 3D model acquisition. *ACM Transactions on Graphics (TOG)*, 21(3), pp.438-446.
- Sánchez-Molina, JA, et al. (2015), 'Support system for decision making in the management of the greenhouse environmental based on growth model for sweet pepper', *Agricultural Systems*, 139, 144-52.



- Sandoval, J., Gor, S., Ramallo, J., Sfer, A., Colombo, E., Vilaseca Ricart, M. and Pujol Ramo, J., 2010. Spectral signatures: A way to identify species and conditions of vegetables.
- Santos, Thiago T and De Oliveira, Alberto A (2012), 'Image-based 3D digitizing for plant architecture analysis and phenotyping', Embrapa Informática Agropecuária-Artigo em anais de congresso (ALICE) (In: CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES, 25., 2012, Ouro Preto ...).
- SANTOS, TT and Ueda, Julio (2013), 'Automatic 3D plant reconstruction from photographs, segmentation and classification of leaves and internodes using clustering', Embrapa Informática Agropecuária-Resumo em anais de congresso (ALICE) (In: INTERNATIONAL CONFERENCE ON FUNCTIONAL-STRUCTURAL PLANT MODELS, 7., 2013 ...).
- Sarıbaşı, Hatice Şeyma, Saka, Andac Kutay, and Özer, Harun (2018), 'Mathematical growth model for organically grown pepper transplants', *Biological Agriculture & Horticulture*, 34 (1), 10-17.
- Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X. and Westling, P., 2014, September. High-resolution stereo datasets with subpixel-accurate ground truth. In *German conference on pattern recognition* (pp. 31-42). Springer, Cham.
- Schnee, J. and Futterlieb, J., 2011, August. Laser line segmentation with dynamic line models. In *International Conference on Computer Analysis of Images and Patterns* (pp. 126-134). Springer, Berlin, Heidelberg.
- Schuldt, D., Thiem, J. and Mustedanagic, S., 2016, July. Stereo Vision on an ARM/DSP Multicore Platform based on Code Generation using the MATLAB Embedded Coder. In *ICINCO* (2) (pp. 558-562).
- Serati, G., Sedaghat, A., Mohammadi, N. and Li, J., 2022. Digital Surface Model Generation from High-Resolution Satellite Stereo Imagery Based on Structural Similarity. *Geocarto International*, (just-accepted), pp.1-22.
- Shirzadifar, Alimohammad, et al. (2018), 'Weed species discrimination based on SIMCA analysis of plant canopy spectral data', *Biosystems Engineering*, 171, 143-54.
- Shrestha, DS, Steward, Brian L, and Birrell, Stuart J (2004), 'Video processing for early stage maize plant detection', *Biosystems engineering*, 89 (2), 119-29.

- Silva, LOLA, et al. (2013), 'Comparative assessment of feature selection and classification techniques for visual inspection of pot plant seedlings', *Computers and electronics in agriculture*, 97, 47-55.
- Song, Yu, et al. (2007), 'Surface modelling of plants from stereo images', *Sixth International Conference on 3-D Digital Imaging and Modeling (3DIM 2007) (IEEE)*, 312-19.
- Stöckle, Claudio O, Donatelli, Marcello, and Nelson, Roger (2003), 'CropSyst, a cropping systems simulation model', *European journal of agronomy*, 18 (3-4), 289-307.
- Strothmann W, Ruckelshausen A, Hertzberg J, Scholz C, Langsenkamp F. Plant classification with in-field-labeling for crop/weed discrimination using spectral features and 3d surface features from a multi-wavelength laser line profile system. *Computers and Electronics in Agriculture*, 2017, 134: 79–93.
- Su, Huihuang and He, Bingwei (2011), 'Stereo rectification of calibrated image pairs based on geometric transformation', *International Journal of Modern Education and Computer Science*, 3 (4), 17.
- Sun, Y., Paik, J.K., Koschan, A. and Abidi, M.A., 2002, August. 3D reconstruction of indoor and outdoor scenes using a mobile range scanner. In *Object recognition supported by user interaction for service robots (Vol. 3, pp. 653-656)*. IEEE.
- Symonds P, Paap A, Alameh K, Rowe J, Miller C. A real-time plant discrimination system utilising discrete reflectance spectroscopy. *Computers and Electronics in Agriculture*, 2015, 117: 57–69.
- Szczepanski, M., 2019. *Online stereo camera calibration on embedded systems* (Doctoral dissertation, Université Clermont Auvergne(2017-2020)).
- Szeliski, R., 2010. *Computer vision: algorithms and applications*. Springer Science & Business Media.
- Teng, Chin-Hung, Kuo, Yi-Ting, and Chen, Yung-Sheng (2011), 'Leaf segmentation, classification, and three-dimensional recovery from a few images with close viewpoints', *Optical Engineering*, 50 (3), 037003.
- Teutsch, C., Isenberg, T., Trostmann, E., Weber, M., Berndt, D. and Strothotte, T., 2005, January. Evaluation and correction of laser-scanned point clouds. In *Videometrics VIII* (Vol. 5665, pp. 172-183). SPIE.
- Tošić, I. and Frossard, P., 2010. Dictionary learning for stereo image representation. *IEEE Transactions on Image Processing*, 20(4), pp.921-934.

- Trujillo-Romero, F.J, 2008. Modélisation et reconnaissance active d'objets 3D de forme libre par vision en robotique (Doctoral dissertation, Institut National Polytechnique de Toulouse-INPT).
- Tu, D., Jin, P. and Zhang, X., 2019. Geometrical model of laser triangulation system based on synchronized scanners. *Mathematical Problems in Engineering*, 2019.
- Türker-Kaya, Sevgi and Huck, Christian W (2017), 'A review of mid-infrared and near-infrared imaging: principles, concepts and applications in plant tissue analysis', *Molecules*, 22 (1), 168.
- Uematsu, Y., Teshima, T., Saito, H. and Honghua, C., 2007, September. D-calib: Calibration software for multiple cameras system. In *14th International Conference on Image Analysis and Processing (ICIAP 2007)* (pp. 285-290). IEEE.
- Urquhart, C.W., 1997. The active stereo probe: the design and implementation of an active videometrics system. University of Glasgow (United Kingdom).
- Veitch-Michaelis, J.L., 2017. Fusion of LIDAR with stereo camera data: an assessment (Doctoral dissertation, University College London).
- Wang X, Zhang M, Zhu J, Geng S. Spectral prediction of *Phytophthora infestans* infection on tomatoes using artificial neural network (ANN). *International Journal of Remote Sensing*, 2008, 29(6): 1693–1706.
- Wang, J., 2018. *High resolution 2D imaging and 3D scanning with line sensors* (Doctoral dissertation, Carnegie Mellon University).
- Wang, Jianlun, et al. (2013), 'An adaptive thresholding algorithm of field leaf image', *Computers and electronics in agriculture*, 96, 23-39.
- Wang, T.M. and Shih, Z.C., 2021. Measurement and analysis of depth resolution using active stereo cameras. *IEEE Sensors Journal*, 21(7), pp.9218-9230.
- Wang, Y., Liu, K., Hao, Q., Wang, X., Lau, D.L. and Hassebrook, L.G., 2012. Robust active stereo vision using Kullback-Leibler divergence. *IEEE transactions on pattern analysis and machine intelligence*, 34(3), pp.548-563.
- Weisenfeld, N.I., 2002. *A non-contact, active stereo imaging system for intraoperative surface measurements* (Doctoral dissertation, Massachusetts Institute of Technology).
- Werner, Tomas and Zisserman, Andrew (2002), 'New techniques for automated architectural reconstruction from photographs', *European conference on computer vision* (Springer), 541-55.

- Won, C., Ryu, J. and Lim, J., 2019. Omnimvs: End-to-end learning for omnidirectional stereo matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 8987-8996).
- Wu, D., Chen, J., Lu, B., Xiong, L., He, Y. and Zhang, Y., 2012. Application of near infrared spectroscopy for the rapid determination of antioxidant activity of bamboo leaf extract. *Food chemistry*, 135(4), pp.2147-2156.
- Wu, Stephen Gang, et al. (2007), 'A leaf recognition algorithm for plant classification using probabilistic neural network', 2007 IEEE international symposium on signal processing and information technology (IEEE), 11-16.
- Xiang, B., Yao, J., Lu, X., Li, L., Xie, R. and Li, J., 2018. Segmentation-based classification for 3D point clouds in the road environment. *International Journal of Remote Sensing*, 39(19), pp.6182-6212.
- Xiaoping, W., Fei, Z., Hsiang-te, K. and Haiyang, Y., 2017. Spectral response characteristics and identification of typical plant species in Ebinur lake wetland national nature reserve (ELWNNR) under a water and salinity gradient. *Ecological Indicators*, 81, pp.222-234.
- Xing, Y.J., Xing, J., Sun, J. and Hu, L., 2007. An improved neural networks for stereo-camera calibration. *Journal of Achievements in Materials and Manufacturing Engineering*, 20(1-2), pp.315-318.
- Xu, H.R., Ying, Y.B., Fu, X.P. and Zhu, S.P., 2007. Near-infrared spectroscopy in detecting leaf miner damage on tomato leaf. *Biosystems Engineering*, 96(4), pp.447-454.
- Yedidia, J.S., Freeman, W. and Weiss, Y., 2000. Generalized belief propagation. *Advances in neural information processing systems*, 13.
- Yedidia, J.S., Freeman, W.T. and Weiss, Y., 2003. Understanding belief propagation and its generalizations. *Exploring artificial intelligence in the new millennium*, 8(236-239), pp.0018-9448.
- Zarco-Tejada PJ, Camino C, Beck PSA, Calderon R, Hornero A, Hernández-Clemente R, Kattenborn T, Montes-Borrego M, Susca L, Morelli M, Gonzalez-Dugo V. Previsual symptoms of *Xylella fastidiosa* infection revealed in spectral plant-trait alterations. *Nature Plants*, 2018, 4(7): 432–439.
- Zbontar, J. and LeCun, Y., 2016. Stereo matching by training a convolutional neural network to compare image patches. *J. Mach. Learn. Res.*, 17(1), pp.2287-2318.

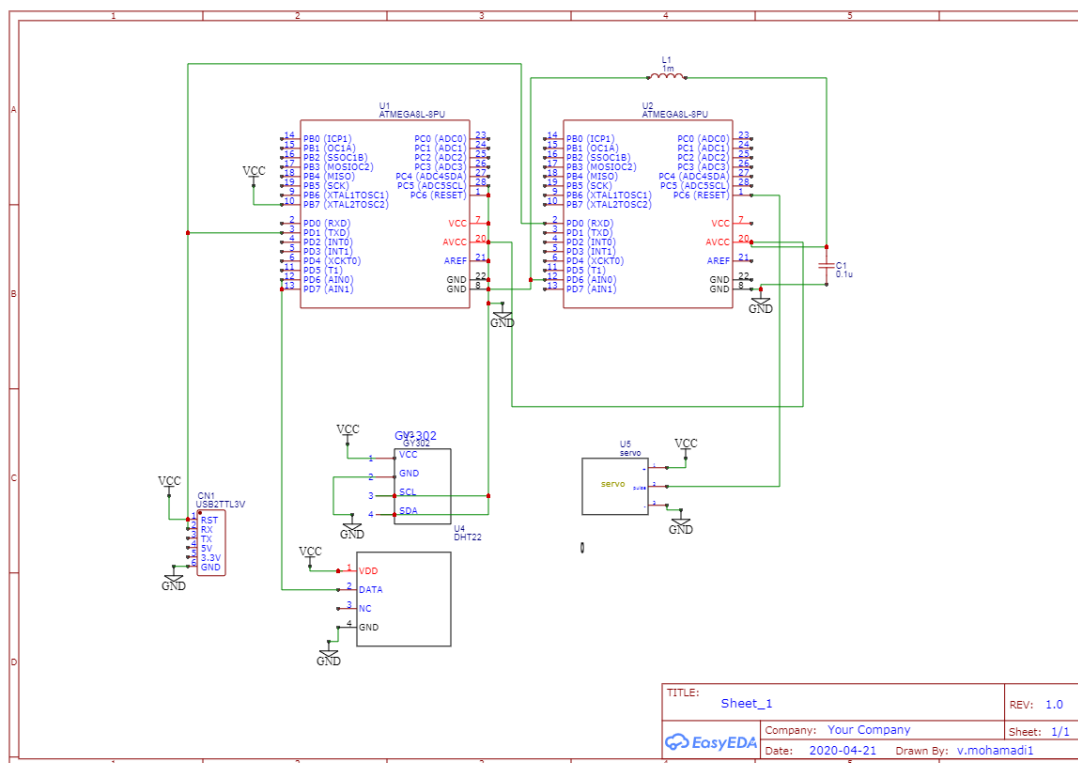
- Zhang M, Liu X, O'Neill M. Spectral discrimination of *Phytophthora infestans* infection on tomatoes based on principal component and cluster analyses. *International Journal of Remote Sensing*, 2002, 23(6): 1095–1107.
- Zhang, B., Yu, H. and Sun, L., 2013, April. Peak Detection Algorithm for Laser Induced Breakdown Spectroscopy. In *2nd International Symposium on Computer, Communication, Control and Automation* (pp. 453-455). Atlantis Press.
- Zhang, J., Huang, Y., Pu, R., Gonzalez-Moreno, P., Yuan, L., Wu, K. and Huang, W., 2019. Monitoring plant diseases and pests through remote sensing technology: A review. *Computers and Electronics in Agriculture*, 165, p.104943.
- Zhang, K., Lu, J., Lafruit, G., Lauwereins, R. and Van Gool, L., 2009, November. Robust stereo matching with fast normalized cross-correlation over shape-adaptive regions. In *2009 16th IEEE International Conference on Image Processing (ICIP)* (pp. 2357-2360). IEEE.
- Zhao, Y., Lei, S., Yang, X., Gong, C., Wang, C., Cheng, W., Li, H. and She, C., 2020. Study on Spectral Response and Estimation of Grassland Plants Dust Retention Based on Hyperspectral Data. *Remote Sensing*, 12(12), p.2019.
- Zude-Sasse, M., Truppel, I. and Herold, B., 2002. An approach to non-destructive apple fruit chlorophyll determination. *Postharvest Biology and Technology*, 25(2), pp.123-133.

# IV

# APPENDICES

A

# SCHEMATIC OF THE CIRCUIT



## B

### PASSIVE STEREO IMAGE PROCESSING PROGRAM

```
%% Copyright
%{
%   Leaf Properties Detection using Passive Stereo Imaging

%   Copyright (C) 2022  Vahid Mohammadi
%
%   This program has been developed for a PhD dissertation
%   No distribution is allowed unless by the permission
%   from the author or the supervisor of the dissertation
%}

clc
clear
close all

%% Preprocessing of the image      =====

tic(); % This is for having the duration of processing
% Load the stereo images.
left0 = imread('D:\Academic Works\PhD Files\Dissertation\Results\Data-
2019\Day 29 - Shahrivar 26\Depth\P3-L.jpg');
left = imresize(left0, 0.2);
right0 = imread('D:\Academic Works\PhD Files\Dissertation\Results\Data-
2019\Day 29 - Shahrivar 26\Depth\P3-R.jpg');
right = imresize(right0, 0.2);

% Crop the image to the region of interest
I = imcrop(left0, [1250 600 1600 1100]);

% Setting a threshold

threshold = 0.5;

% Let's convert the image from uint8 to double (this may make the
following
% processes easier to handle

h = rgb2hsv(I);
h1 = h(:, :, 1);
h2 = 0.21 < h1 & h1 < 0.29;

SE = strel('disk', 4, 4);
img4 = imopen(h2, SE);

figure(1), imshow(img4)
% Averaging Filter (To reduce noise and smoothen the image)
```



```

H = fspecial ('average',15);
img5 = imfilter(img4,H,'conv','symmetric','same');

img6 = imfill(img5,'holes');
figure, imshow(img6)

%% Tophat & leaf detection =====

Red = I(:,:,1); R = immultiply(Red,img6);
Green = I(:,:,2); G = immultiply(Green,img6);
Blue = I(:,:,3); B = immultiply(Blue,img6);
Out = cat(3,R, G, B);
figure, imshow(Out)
I = rgb2gray(Out);

imshow(I)

se = strel('disk',30);
top = imtophat(I,se);
figure, imshow(top)

thr = top > 80;
hats = bwareaopen(thr, 130);
figure, imshow(hats)

BB = bwconncomp(hats);
Cntrd = regionprops(BB, 'centroid');
Size = size(Cntrd,1);
    figure, imshow(hats)
    axis on
    hold on
for i=1:Size
    Ce = Cntrd(i).Centroid; Ce = round(Ce);

    plot(Ce(1),Ce(2),'Color','r','Marker','+');

end
fprintf('Number of leaves = %d \n',Size)

%% --- Obtain watershed image -----

I = imadjust(I,[0.3;0.7],[,]);

% Use the Gradient Magnitude as the Segmentation Function
% Compute the gradient magnitude. The gradient is high at the borders
of the
% objects and low (mostly) inside the objects.

E = entropyfilt(I);
EE = round(stdfilt(I));
t = mat2gray (EE);
ER = imadjust(t,[0.1 0.3],[,]);

```

```

gmag = imgradient(I,'roberts');
L = watershed(gmag);
Lrgb = label2rgb(L);

% A variety of procedures could be applied here to find the foreground
markers,
% which must be connected blobs of pixels inside each of the foreground
objects.
% In this example you'll use morphological techniques called "opening-
by-reconstruction"
% and "closing-by-reconstruction" to "clean" up the image. These
operations will
% create flat maxima inside each object that can be located using
imregionalmax.
%
% Opening is an erosion followed by a dilation, while opening-by-
reconstruction
% is an erosion followed by a morphological reconstruction. Let's
compare the two.
% First, compute the opening using imopen.

se = strel('disk',20);
Io = imopen(I,se);

% Next compute the opening-by-reconstruction using imerode and
imreconstruct.

Ie = imerode(I,se);
Iobr = imreconstruct(Ie,I);

% Following the opening with a closing can remove the dark spots and
marks. Compare a regular morphological closing with a closing-by-
reconstruction. First try imclose:

Ioc = imclose(Io,se);

% Now use imdilate followed by imreconstruct. Notice you must
complement the image inputs and output of imreconstruct.

Iobrd = imdilate(Iobr,se);
Iobrcbr = imreconstruct(imcomplement(Iobrd),imcomplement(Iobr));
Iobrcbr = imcomplement(Iobrcbr);
fgm = imregionalmax(Iobrcbr);

% Notice that some of the mostly-occluded and shadowed objects are not
marked, which means that these objects will not be segmented properly in
the end result.

se2 = strel(ones(5,5));
fgm2 = imclose(fgm,se2);
fgm3 = imerode(fgm2,se2);

```

% This procedure tends to leave some stray isolated pixels that must be removed. You can do this using `bwareaopen`, which removes all blobs that have fewer than a certain number of pixels.

```

fgm4 = bwareaopen(fgm3,20);

bw = imbinarize(Iobrcbr);

D = bwdist(bw);
DL = watershed(D);
bgm = DL == 0;

% Compute the Watershed Transform of the Segmentation Function.
% The function imimposemin can be used to modify an image so that it
has regional minima only in certain desired locations. Here you can use
imimposemin to modify the gradient magnitude image so that its only
regional minima occur at foreground and background marker pixels.

gmag2 = imimposemin(gmag, bgm | fgm4);
% Finally we are ready to compute the watershed-based segmentation.

L = watershed(gmag2);
figure
imshow(I)
hold on
himage = imshow(Lrgb);
himage.AlphaData = 0.3;
title('Colored Labels Superimposed Transparently on Original Image')

%% ----- Calculate Geometrical Features ----- %%

.
.
.

%% Calculate the disparity map =====

% Load calibration information
load('SParams1.mat');

% Rectify images
[J1, J2] = rectifyStereoImages(left, right, stereoParams);

% Display the images after rectification.

figure(2); imshow(cat(3, J1(:,:,1), J2(:,:,2:3)),
'InitialMagnification', 50);

% For building the disparity map, 'SemiGlobal' matching method is used
% which is more accurate than 'BlockMatching'. The default is
% 'SemiGlobal'.
% The default BloxkSize is 15.
% The default range of 'DisparityRange' is [0 64]
% The default value of 'ContrastThreshold' is 0.5

```

```

% The default value of 'UniquenessThreshold' is 15
% The default value of 'DistanceThreshold' is []
disparityMap = disparity(rgb2gray(J1),
rgb2gray(J2), 'Method', 'SemiGlobal', ...
    'BlockSize', 7, 'DisparityRange', [0 128], 'ContrastThreshold', 0.6, ...
    'UniquenessThreshold', 4, 'DistanceThreshold', 100);
figure(3); imshow(disparityMap, [0, 67], 'InitialMagnification', 50);

axis image;

% Use the 'jet' color map.
% You might also consider removing this line to view the disparity map
in
% grayscale.
colormap('jet');

% Display the color map legend.
colorbar;

%% 3D reconstruction =====

points3D = reconstructScene(disparityMap, stereoParams);

% Convert to meters and create a pointCloud object
points3D = points3D ./ 1000;
ptCloud = pointCloud(points3D, 'Color', J1);

% Create a streaming point cloud viewer
player3D = pcplayer([-3, 3], [-3, 3], [0, 8], 'VerticalAxis', 'y', ...
    'VerticalAxisDir', 'down');

% Visualize the point cloud
figure(4), view(player3D, ptCloud);

Z = points3D(:, :, 3);
mask = repmat(Z > 1.9 & Z < 2.2, [1, 1, 3]);
J1(~mask) = 0;
figure(5), imshow(J1, 'InitialMagnification', 50);

%% Basic Block Matching =====

% Calculate the disparity using basic block matching with sub-pixel
% estimation.
% The original Mathworks example code utilized the TemplateMatcher from
their
% vision toolbox; I've modified the code to work without this
dependency.
.
.
.

```

```
%% ||||| - Calculate the features and the distance of each leaf -|||  
%%  
  
%{  
    For each leaf, calculate the selected features. Then for the pixel x  
    find the distance of the leaf from the camera and then  
    calculate the real size of the leaf  
    The pixel number for the scale is 34 for the distance 80 cm.  
%}  
.  
.  
.
```

## K-MEANS CLUSTERING USING SILHOUETTE

### METHOD

```

% Select optimal number of clusters (K value) using the
specified range
fh = @(X,K) (kmeans(X,K));
eva = evalclusters(i,fh,"silhouette","KList",2:6);
clear fh
K = eva.OptimalK;
clusterIndices = eva.OptimalY;

% Display cluster evaluation criterion values
figure
bar(eva.InspectedK,eva.CriterionValues);
xticks(eva.InspectedK);
xlabel("Number of clusters");
ylabel("Criterion values - Silhouette");
legend("Optimal number of clusters is " + num2str(K))
title("Evaluation of Optimal Number of Clusters")
disp("Optimal number of clusters is " + num2str(K));
clear K eva

% Calculate centroids
centroids = grpstats(i,clusterIndices,"mean"); % grpstats
returns a table or dataset array with the means for the
data groups

% Display results

% Display 2D scatter plot (PCA)
figure
[~,score] = pca(i);
clusterMeans = grpstats(score,clusterIndices,"mean");
h2 =
gscatter(score(:,1),score(:,2),clusterIndices,colormap("
lines"));
for i2 = 1:numel(h2)
    h2(i2).DisplayName =
strcat("Cluster",h2(i2).DisplayName);
end
clear h2 i2 score
hold on

```

```
h2 =  
scatter(clusterMeans(:,1),clusterMeans(:,2),50,"kx","LineWidth",2);  
hold off  
h2.DisplayName = "ClusterMeans";  
clear h2 clusterMeans  
legend;  
title("First 2 PCA Components of Clustered Data");  
xlabel("First principal component");  
ylabel("Second principal component");
```

## LIST OF DATASETS OF 3D POINT CLOUDS











No	Name	Details	Address
1	ShapeNet	<ul style="list-style-type: none"> <li>• 3D CAD models</li> <li>• over 300M models</li> </ul>	<a href="https://shapenet.org/">https://shapenet.org/</a>
2	ModelNet	<ul style="list-style-type: none"> <li>• synthetic object point clouds</li> <li>• 12,311 CAD-generated meshes in 40 categories</li> </ul>	<a href="https://modelnet.cs.princeton.edu/">https://modelnet.cs.princeton.edu/</a>
3	nuScenes	<ul style="list-style-type: none"> <li>• a large-scale autonomous driving dataset</li> <li>• 3D bounding boxes for 1000 scenes</li> </ul>	<a href="https://www.nuscenes.org/">https://www.nuscenes.org/</a>
4	SUN RGB-D	<ul style="list-style-type: none"> <li>• 10335 real RGB-D images of room scenes</li> <li>• 700 object categories</li> </ul>	<a href="https://rgbd.cs.princeton.edu/">https://rgbd.cs.princeton.edu/</a>
5	S3DIS	<ul style="list-style-type: none"> <li>• Stanford 3D Indoor Scene Dataset</li> <li>• 6 large-scale indoor areas with 271 rooms</li> </ul>	<a href="http://buildingparser.stanford.edu/dataset.html">http://buildingparser.stanford.edu/dataset.html</a>
6	SemanticKITTI	<ul style="list-style-type: none"> <li>• a large-scale outdoor-scene dataset</li> <li>• dataset provides 23201 point clouds</li> </ul>	<a href="http://www.semantic-kitti.org/">http://www.semantic-kitti.org/</a>
7	Argoverse	<ul style="list-style-type: none"> <li>• dataset with 3D tracking annotations for 113 scenes</li> </ul>	<a href="https://www.argoverse.org/data.html">https://www.argoverse.org/data.html</a>
8	SUN3D	<ul style="list-style-type: none"> <li>• a large-scale RGB-D video database</li> <li>• 415 sequences captured in 254 different spaces</li> </ul>	<a href="http://sun3d.cs.princeton.edu/">http://sun3d.cs.princeton.edu/</a>
9	ScanObjectNN	<ul style="list-style-type: none"> <li>• 2902 3D objects in 15 categories</li> </ul>	<a href="https://hkust-vgd.github.io/scanobjectnn/">https://hkust-vgd.github.io/scanobjectnn/</a>
10	Semantic3D	<ul style="list-style-type: none"> <li>• scanned outdoor scenes with over 3 billion points</li> <li>• 15 training and 15 test scenes</li> </ul>	<a href="http://www.semantic3d.net/">http://www.semantic3d.net/</a>
11	Completion3D	<ul style="list-style-type: none"> <li>• dataset for evaluating 3D Object Point Cloud Completion methods</li> </ul>	<a href="https://completion3d.stanford.edu/">https://completion3d.stanford.edu/</a>
12	Dex-Net 2.0	<ul style="list-style-type: none"> <li>• dataset of 1,500 3D object models</li> </ul>	<a href="http://berkeleyautomation.github.io/dex-net/">http://berkeleyautomation.github.io/dex-net/</a>
13	SensatUrban	<ul style="list-style-type: none"> <li>• urban-scale photogrammetric point cloud dataset</li> </ul>	<a href="https://github.com/QingyongHu/SensatUrban">https://github.com/QingyongHu/SensatUrban</a>
14	SemanticPOSS	<ul style="list-style-type: none"> <li>• dataset for 3D semantic segmentation contains 2988 various and complicated LiDAR scans</li> </ul>	<a href="http://www.poss.pku.edu.cn/semanticposs.html">http://www.poss.pku.edu.cn/semanticposs.html</a>



15	SHREC'19	<ul style="list-style-type: none"> <li>Human body models</li> </ul>	<a href="http://profs.scienze.univr.it/~marin/shrec19/">http://profs.scienze.univr.it/~marin/shrec19/</a>
16	ModelNet40-C	<ul style="list-style-type: none"> <li>dataset to benchmark the corruption robustness of 3D point cloud recognition</li> <li>contains 185,000 distinct point clouds</li> </ul>	<a href="https://sites.google.com/umich.edu/modelnet40c">https://sites.google.com/umich.edu/modelnet40c</a>
17	KITTI-Depth	<ul style="list-style-type: none"> <li>depth maps from projected LiDAR point clouds</li> <li>dataset has 86k training images, 7k validation images</li> </ul>	<a href="http://www.cvlibs.net/datasets/kitti/eval_depth.php?benchmark=depth_prediction">http://www.cvlibs.net/datasets/kitti/eval_depth.php?benchmark=depth_prediction</a>
18	MVP	<ul style="list-style-type: none"> <li>multi-view partial point cloud dataset</li> <li>containing over 100,000 high-quality scans</li> </ul>	<a href="https://paul007pl.github.io/projects/VRCNet">https://paul007pl.github.io/projects/VRCNet</a>
19	Toronto-3D	<ul style="list-style-type: none"> <li>a large-scale urban outdoor point cloud dataset</li> <li>dataset covers approximately 1 km of road and consists of about 78.3 million points</li> </ul>	<a href="https://github.com/WeikaiTan/Toronto-3D">https://github.com/WeikaiTan/Toronto-3D</a>
20	VOID	<ul style="list-style-type: none"> <li>contains 56 sequences in total, both indoor and outdoor</li> </ul>	<a href="https://github.com/alexklwong/void-dataset">https://github.com/alexklwong/void-dataset</a>
21	3DCSR	<ul style="list-style-type: none"> <li>indoor working environment, which contains the popular objects in the working space</li> </ul>	<a href="http://multimediauts.org/3D_data_for_registration/">http://multimediauts.org/3D_data_for_registration/</a>
22	BLVD	<ul style="list-style-type: none"> <li>a large scale 5D semantics dataset</li> <li>249,129 3D annotations</li> </ul>	<a href="https://github.com/VCCIV/BLVD/">https://github.com/VCCIV/BLVD/</a>
23	RadarScenes	<ul style="list-style-type: none"> <li>real-world radar point cloud dataset for automotive applications</li> </ul>	<a href="https://radar-scenes.com/">https://radar-scenes.com/</a>
24	Sydney Urban Objects	<ul style="list-style-type: none"> <li>contains a variety of common urban road objects</li> <li>631 individual scans of objects</li> </ul>	<a href="http://www.acfr.usyd.edu.au/papers/SydneyUrbanObjectsDataset.shtml">http://www.acfr.usyd.edu.au/papers/SydneyUrbanObjectsDataset.shtml</a>
25	ChangeSim	<ul style="list-style-type: none"> <li>photo-realistic simulation environments with the presence of environmental non-targeted variations, such as air turbidity and light condition changes</li> </ul>	<a href="https://github.com/SAMMiCA/ChangeSim">https://github.com/SAMMiCA/ChangeSim</a>
26	Freiburg Spatial Relations	<ul style="list-style-type: none"> <li>546 scenes each containing two out of 25 household objects</li> </ul>	<a href="http://spatialrelations.cs.uni-freiburg.de/">http://spatialrelations.cs.uni-freiburg.de/</a>
27	WADS	<ul style="list-style-type: none"> <li>LiDAR scans collected in severe winter weather</li> <li>Over 26 TB of multi modal data</li> </ul>	<a href="https://bitbucket.org/autonomymtu/wads/src">https://bitbucket.org/autonomymtu/wads/src</a>

28	3D-BSLS-6D	<ul style="list-style-type: none"> <li>• real captures from Photoneo PhoXi structured light scanner</li> <li>• 6D pose estimation</li> </ul>	<a href="https://www.skeletex.xyz/portfolio/datasets">https://www.skeletex.xyz/portfolio/datasets</a>
29	CODD	<ul style="list-style-type: none"> <li>• Cooperative Driving Dataset</li> <li>• contains lidar data from multiple vehicles navigating simultaneously</li> </ul>	<a href="https://github.com/eduardohenriquearnold/CODD">https://github.com/eduardohenriquearnold/CODD</a>
30	DurLAR	<ul style="list-style-type: none"> <li>• a high-fidelity 128-channel 3D LiDAR dataset</li> <li>• full 360 degree depth, range accuracy to <math>\pm 2</math> cm at 20-50m</li> </ul>	<a href="https://github.com/11997i/DurLAR">https://github.com/11997i/DurLAR</a>
31	Ford Campus Vision and Lidar Data Set	<ul style="list-style-type: none"> <li>• collected by an autonomous ground vehicle testbed</li> <li>• consists of the time-registered data</li> </ul>	<a href="http://robots.engin.umich.edu/SoftwareData/Ford">http://robots.engin.umich.edu/SoftwareData/Ford</a>
32	MVHand	<ul style="list-style-type: none"> <li>• multi-view hand posture dataset</li> <li>• 3D point clouds of the hand</li> </ul>	<a href="https://github.com/ShichengChen/multiviewDataset">https://github.com/ShichengChen/multiviewDataset</a>
33	TERRA-REF	<ul style="list-style-type: none"> <li>• datasets for the study of plant sensing, genomics, and phenomics</li> </ul>	<a href="https://datadryad.org/stash/dataset/doi:10.5061/dryad.4b8gtht99">https://datadryad.org/stash/dataset/doi:10.5061/dryad.4b8gtht99</a>

## SAMPLE IMAGES OF WEEDS

Weed	Weed photo	Leaf photo
Bindweed		
Nutsedge		
Plantago lanceolata		
Potentilla		
Sorrel		

**Title:** Design, Development and Evaluation of a System for the Detection of Aerial Parts and Measurement of Growth Indices of Bell Pepper Plant Based on Stereo and Multispectral Imaging

**Keywords:** Stereo Image processing; 3D Reconstruction; Point Cloud; Surface reconstruction; Stereo camera calibration; Multispectral data; Crop/weed separation.

**Abstract:** Monitoring plants growth including the measurement of physical properties, counting plants leaves, detection of plants and their separation from weeds brings much benefits to the producers. Preferable techniques for this task are those that are non-destructive because plant is a very sensitive creature and any manipulation of which can affect its growth or lead to losing leaves or branches. Imaging techniques are of the best solutions for plant growth monitoring and geometric measurements. This project involves the use of stereo imaging and multispectral data analysis for examining, separation, and classification of crop plant/weeds. Active and passive stereo imaging were employed for the estimation of physical properties and counting leaves and multispectral data was utilized for the separation of crop and weed. Use of Active Stereo Imaging (ASI) during the primary stage of growth provides acceptable results, however, it is unable to detect and reconstruct all leaves and plant parts during the later stages. Using ASI, yielded  $R^2$  values of 0.978 and 0.967 for estimation leaf area and perimeter, respectively. The result of segmentation of the point clouds could count the leaves with the success rate of 96.053 %. However, the overall success rate of the passive stereo algorithm for leaf counting was 84.32 %. The results of separation of crop and weeds using spectral data were very promising and the classifier—which was based on deep learning—was capable of completely separating pepper plants from five weeds.

**Titre:** Conception, développement et évaluation d'un système de détection de parties aériennes et de mesure d'indices de croissance de poivrons basés sur l'imagerie stéréo et l'analyse spectrale

**Mots-clés:** Traitement d'image stéréo ; reconstruction 3D ; Nuage de points; reconstruction de surface ; Étalonnage de la caméra stéréo ; Données multispectrales ; Séparation cultures/mauvaises herbes.

**Résumé:** Le suivi de la croissance des plantes, y compris la mesure des propriétés physiques, le comptage des feuilles des plantes, la détection des plantes et leur séparation des mauvaises herbes apporte de nombreux avantages aux producteurs. Les techniques préférables pour cette tâche sont celles qui sont non destructives car la plante est une créature très sensible et dont toute manipulation peut affecter sa croissance ou entraîner la perte de feuilles ou de branches. Les techniques d'imagerie sont des meilleures solutions pour le suivi de la croissance des plantes et les mesures géométriques. Ce projet implique l'utilisation de l'imagerie stéréo et de l'analyse de données multispectrales pour l'examen, la séparation et la classification des plantes cultivées/mauvaises herbes. L'imagerie stéréo active et passive a été utilisée pour l'estimation des propriétés physiques et le comptage des feuilles et des données multispectrales ont été utilisées pour la séparation des cultures et des mauvaises herbes. L'utilisation de l'imagerie stéréo active (ISA) au cours de la phase primaire de croissance fournit des résultats acceptables, cependant, elle est incapable de détecter et de reconstruire toutes les feuilles et parties de la plante au cours des étapes ultérieures. L'utilisation de l'ISA a donné des valeurs  $R^2$  de 0,978 et 0,967 pour l'estimation de la surface foliaire et du périmètre, respectivement. Le résultat de la segmentation des nuages de points a pu compter les feuilles avec un taux de réussite de 96,053 %. Cependant, le taux de réussite global de l'algorithme stéréo passif pour le comptage des feuilles était de 84,32 %. Les résultats de la séparation des cultures et des mauvaises herbes à l'aide de données spectrales étaient très prometteurs et le classificateur, qui était basé sur l'apprentissage en profondeur, était capable de séparer complètement les plants de poivron de cinq mauvaises herbes.