DETECTION OF PLANT CHARACTERISTICS AND A COMPARISON OF EFFECTIVENESS BETWEEN 2D AND 3D DATA VISUALIZATION IN SUPPORTING HUMAN PERCEPTION OF PLANT CHARACTERISTICS

A Thesis

by

THANH VAN PHAM

BS, Hanoi University of Science and Technology - Vietnam, 2010

Submitted in Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in

COMPUTER SCIENCE

Texas A&M University-Corpus Christi
Corpus Christi, Texas

May 2018

©THANH VAN PHAM

All Rights Reserved

May 2018

DETECTION OF PLANT CHARACTERISTICS AND A COMPARISON OF EFFECTIVENESS BETWEEN 2D AND 3D DATA VISUALIZATION IN SUPPORTING HUMAN PERCEPTION OF PLANT CHARACTERISTICS

A Thesis

by

THANH VAN PHAM

This thesis meets the standards for scope and quality of
Texas A&M University-Corpus Christi and is hereby approved.

Scott King, PhD
Chair

Byung Cheol Lee, PhD
Committee Member

Alaa Sheta, PhD
Committee Member

May 2018

ABSTRACT

Efficient agriculture requires the assessment of plant characteristics. A higher crop yield can be achieved with good quality plant characteristic data. In this research, a system was developed using the algorithms presented here to automatically extract plant characteristics. The automatically extracted values were compared with ground-truth data to evaluate the accuracy of the system. As well, the effectiveness of using 2 or 3 dimensional data visualization for determining these characteristics is studied. An experiment was conducted to investigate how effective plant characteristics are evaluated when using 2D or 3D data visualizations. Participants were presented with either plant pictures (2D) or 3D plant models and tasked with identifying plant height and the number of leaves. Task completion times and accuracy rates were gathered for performance analysis.

TABLE OF CONTENTS

CONTENTS PAGE

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Motivation

Efficient assessment of plant characteristics can lead to productive agriculture. Crop yields can be increased with good quality plant characteristic data. For the plant data collection phase, the conventional ways for a farmer or agricultural researcher to collect and examine plant status and characteristics in the field are time-consuming and labor-intensive. As well, the weariness from a long day standing in the field can negatively influence the degree of accuracy of the data collection. Therefore, with the purpose of supporting the farmers and agricultural researchers in getting rid of laborious methods, this research is aimed to find an effective approach to collect plant data and improve quality of plant characteristics analysis.

1.2 Problem Description

In order to find a way to collect plant data effectively, an automatic system was developed to extract plant characteristics from a 3D model. The system can segment the 3D model to separate the plant model from the background and then accumulate the plant data. As well, with the awareness of the fact that visualization can affect human perception and understanding of data, choosing a suitable visualization type can improve the quality of data analysis. Hence, in this research, an experiment was conducted to investigate the impact of 3-Dimensional (3D) or 2-Dimensional (2D) visualization on human performance in specifying plant characteristics.

1.3 Contributions

The followings are the main contributions:

- Introduce an approach to collect plant data by developing a system to automatically extract plant characteristics from a 3D model.
- Evaluate the effectiveness of 2D vs. 3D representation of plants in determining plant characteristics.

1.4 Description of Work

Overall, the work is split into two efforts. One is an automatic plant feature extraction system. And the other is a 2D vs. 3D data visualization experiment. In order to implement the research, plant characteristics were measured, images of plants were collected, and a 3D model was created.

Ground-truth data of plant was gathered manually. Characteristics from five replications of species were measured. These five replications were chosen randomly at the AgriLife research center in Weslaco, Texas. There are two parameters, which are plant height and number of leaves, need to be taken into consideration. Plant height plays an important role in plant growth analysis, the plant height is measured from ground to the highest point of the plant. Number of leaves gives information about the state of the plant growth at one specific time. Leaves were manually counted on the plants and the height of the plant was measured by using a yardstick. These data were saved for analysis and comparison in the experiment and automatic system later on.

A database of 2D images of plants was created. Each plant was placed in a chamber with two RGB cameras used to take pictures of the plant (see Figure 1).

The cameras were hung on a post that could move around the plant. The cameras took a picture every 10 degrees. Two cameras were installed in different positions to capture all angles of the plant. There were 72 pictures (36 pictures from each camera) taken for each plant. In the chamber, four light bulbs were set up in the four corners of the chamber to supply enough light for the image collection process.

Figure 1. Inside the chamber at AgriLife center, Weslaco, TX.

A 3D model of the plant was generated using a commercial software (AutoDesk Recap Pro) from the 72 pictures. The 3D model files include a geometry and topology file (.obj file), a material file (.mtl file) and a texture file (.jpg file). The images taken and the 3D model created were used in the experiment and the automatic extraction system.

Human subject experiment was conducted to investigate the effect of the 2D images and 3D model of the plant regarding plant feature perception by viewers and the ability of viewers to process data effectively. In the experiment, viewers have to answer questions about plant features including plant height and number of leaves by using 2D images or a 3D model of the plant.

A system was developed to extract plant features automatically. The features that the system extracts are the height of the plant, the volume coverage of the plant, and the green index. The accuracy of the automatic system is determined by comparing the extracted data by system with the measured data collected from the beginning of this research.

CHAPTER 2

PRIOR WORK

This chapter discusses the prior work related to automatic plant feature extraction and comparison of effectiveness between 2D and 3D visualization on human performance. In the past, to improve the yield and quality of the crop, reduce the time-consuming, tedious, and backbreaking work in the agricultural area, many automatic systems were created to support farmers in determining features and state of soil, fruit, plant, etc. With the data collected from the automatic system, the farmer could have a better decision in farming, irrigating, and harvesting.

Yang et al. [2] developed a Fuzzy logic decision-making system for precision farming. The system processed the color images of the agricultural fields affected by weeds and estimating the amount of herbicide needed. Digital cameras and a personal computer were included to adjust the decision-making for using herbicides in the fields. In some cases, the system could reduce the amount of herbicide use by 15-64%.

A system for detecting defects on apples was designed by Puchalski et al. [3]. The system processed apple images that was taken while an apple was rotating in front of the camera. After the system combined multiple images and adjusted the rotation, dark areas generated by defects would appear in the same shape and same place in more than three frames. This proposed system had a 96% classification correctness in finding bruises, frost damage, and scab.

Jin et al. [4] presented a system using adaptive and fixed intensity interception and Otsu segmentation, to detect defects on yellow-skin potatos. With the observation that most of the defects lie through dark or black spots having low proportion and no significant peak in gray level histogram, the system's accuracy reached 92.1%

in classification, 91.4% in recognition and 100% in inspection.

Weis and Gerhards [5] build a system to detect weeds. Their system could not only separate plant and background but also detect weed density and species variations. The weed classification based on shape features revealed the type and number of weeds per image.

Reis et al. [6] built a grape recognition system. The intention of the system was to automatically detect a bunch of white and red grapes from color images. By applying color mapping, morphological dilation, stem and black area detection, the system obtained 97% accuracy for red grapes and 91% accuracy for white grapes.

Sansao et al. [7] propose a method of analysis for green index images using Gabor filters to determine weed coverage percentage. This approach used excess green index images for filtering crops region. The purpose of the system was to monitor the growth of weed and control the usage of herbicides.

An automatic and robust expert system was developed by Romeo et al. [8] to analyze the greenness in an agricultural image. The system implemented image histogram analysis to reflect the direct effect caused by the illumination, and the contrast is an index for greenness identification. The system was applied in large fields such as maize or barley fields to identify the loss of greenness compared to their healthy states. Then the farmer could use the results to make effective treatments such as carrying out a more appropriate irrigation method to supply enough water to the field.

Kelman and Linker [9] proposed an apple detection algorithm. This algorithm was used in detecting mature apples in tree images. By applying 3D convexity analysis and a Canny filter, the system achieved 94% correctness in mature apple detection.

Many studies were conducted to compare the effectiveness between 2D and 3D data visualizations and find a suitable way to visualize information to support human perception. The studies have varying conclusions for 2D and 3D data visualization. Some studies concluded that 3D visualization is better in improving the participant comprehension of data but others said that 2D visualization is better.

Risden et al. [10] tested the ease of use for 2D and 3D information visualization in web content. In this study, they mentioned the strengths and weaknesses of two traditional 2D browsers (a standard collapse/expanse tree browser and web-based hierarchical categorization) and XML3D. XML3D was a novel browser integrating an interactive 3D hyperbolic graph view with a traditional 2D list view of the data. The participants were asked to do searching tasks in the web interfaces. The results showed outstanding performance in task completion time and accuracy rate of viewers when researchers used the XML3D type.

Ware et al. [11] tried to design an interface using 2.5D attitude. The participants were asked to finish a set of tasks such as reading, entering, and editing. For this type of task, 2D interface seemed to be more appropriate since the text was a naturally 2D medium and fonts could be designed more clearly on a flat screen. There were also other tasks where the information was essentially 3D such as interior design, 3D character design, and molecular model. For these types of task, the right approach was adding depth cues as much as was practical. However, in most cases where large different types of information were displayed, a 2.5D design attitude was the best choice. The 2.5 design, which used 3D depth selectively and focused more on a 2D layout, seemed to be the best visualization type.

An experiment was conducted by Althoff et al. [12] to compare between 3D and 2D user interfaces for vehicle infotainment applications. Based on a general design for multimodal architecture, both types of design (2D and 3D) were implemented by typical key-console and touch-screen or by natural speech and active hand and head gestures. In general, most of the car users required a simple and clear design that did not distract them. The study indicated that neither of the two types of the interface design were preferred, but the 3D design obtained more votes regarding the joy of use.

Barfield et al. [13] researched the impact of 2D or 3D graphics presented on paper or computer on the effectiveness of participant's problem-solving. After the research, results showed that the duration for finding a solution was faster with the computer than with paper. Moreover, researchers found that novice participants gave out more correct solutions using 2D paper presentation, while experienced participants generated more exact answers using the 3D graph on a computer.

A study was conducted by Stewart et al. [14] to investigate if students from a university could interpret information from graphs in 2D or 3D format. The study concluded that 2D graphs could create better perception especially with complicated data demonstrated. Additionally, 3D graph rendering might negatively influence the participant's comprehension about graph.

In research on cell phones, a comparison between 2D and 3D menus was carried out by Kim et al. [15]. The starting point of this research was to find a way to convey more data in the limited area of the phone screen. 3D menus were examined with the intention of determining if it could display more items and give out natural and intuitive interface. Some tasks were done using menus with different breadths displayed on the small screen of iPhone or Android phone using 3D visualization to check the effectiveness. The results showed that 3D menus received more interest from the viewers but the 2D menus had more performance with low memory loading.

Demian et al. [16] ran a test to discover the degree of information retrieval using 3D geometry and model topology. In this test, 3D visualization was utilized for creating queries, calculating the relevance of information and finally displaying the search results. The searching took into account the relationship between 3D objects and also relevant information attached to a related 3D object. The conclusion from the test was that 3D visualization is more complicated but very effective in terms of information retrieval.

Dull et al. [17] tried to explore the difference between three visual representations (two-dimensional, three-dimensional, three-dimensional rotatable) regarding the performance and information comprehension in accounting information. Participants were asked to make a prediction based on graphical data from four companies. The graphical data were displayed in 3 types of visual representation. Finally, participants using rotatable function 3D data could produce the most exact predictions.

CHAPTER 3

AUTOMATIC FEATURE EXTRACTION SYSTEM

This chapter discusses in detail the automatic feature extraction system (AFES). The main functions of this system are to process 3D model files and extract features of the plant including plant height, plant volume, and the greenness. The system is a step forward to a replacement with the manual features collection in conventional ways. Before the plant features extraction step, AFES preprocesses to segment some parts of the 3D model in order to separate the plant and the background (the chamber, the floor, and the pot). First, the system design of AFES is introduced and then the analysis and evaluation of each function in the system will be mentioned.

3.1 AFES Design

In AFES, state of the art algorithms are applied to process the 3D model file to automatically extract plant features. These extracted results are then compared with the data collected earlier (described in chapter 1) to check the accuracy of the system. AFES has three 3 main phases shown in Figure 2:

1. Read file and display: the system reads the 3D files and displays the 3D model.

2. Extract background: the system extracts the plant out of the background (the floor, the pot, the chamber).

3. Feature calculation: the system calculates features of the plant. The features includes the volume coverage of the plant, the plant height, the greenness of the plant (green index).

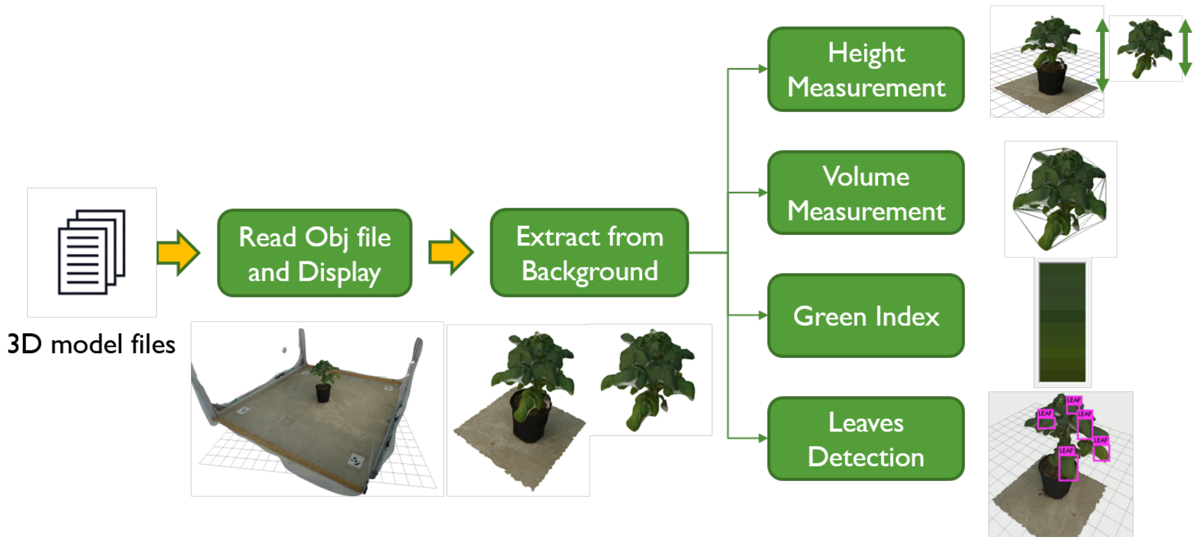


Figure 2. AFES design.

3.1.1 Read File and Display

There are many file formats that can support storing 3D information such as .obj file, .stl file, .fbx file. But in AFES, only .obj file format is accepted. One 3D model must include three main files which are topology and geometry file, material file, and texture file.

Topology and geometry file (.obj file): this file format was developed by Wavefront Technologies for its Advanced Visualizer animation package [18]. A sample of an OBJ file is shown in Appendix A. An obj file stores 3D geometry and topology information including vertex coordinates, texture coordinates, vertex normal coordinates, and face construction data. Vertex coordinates are the position information of vertex in x,y,z dimensions. Texture coordinates are used in mapping process which projects a texture map onto a 3D object. Vertex normal coordinates are the normal vector coordinates of each vertex. Face construction data includes 3 indexes of vertices, 3 indexes of normal vectors and 3 indexes of UV coordinate that could make up a triangle face in the 3D model.

Material file (.mtl file): this file has material definition including color, texture, reflection map of each material. This information is applied to the surfaces and vertices of objects. MTL file has the format shown in Appendix B.

Texture file (.jpeg file): this image file and material file provide all texture information to the display process.

AFES will open all three files to read and transfer all their information to another algorithm to display the 3D model. OpenGL is used to process 3D model data and display it.

3.1.2 Background Extraction

The purpose of this phase is to extract the plant from the background in the original 3D model. As mentioned previously, the commercial software processes 72 images taken in the chamber to create an original 3D model file. This 3D model not only has the plant but also has the chamber and floor model. Therefore, in order to effectively extract features from the plant, the system must segment the plant from the background. This background extraction includes plant (with pot) extraction phase, green part extraction phase, and plant (without pot) extraction phase. This phase also includes floor extraction and floor measurement to calculate the size of the floor. This floor size is used in the evaluation later to assess the accuracy of algorithms. The flow of phases is shown in Figure 3.

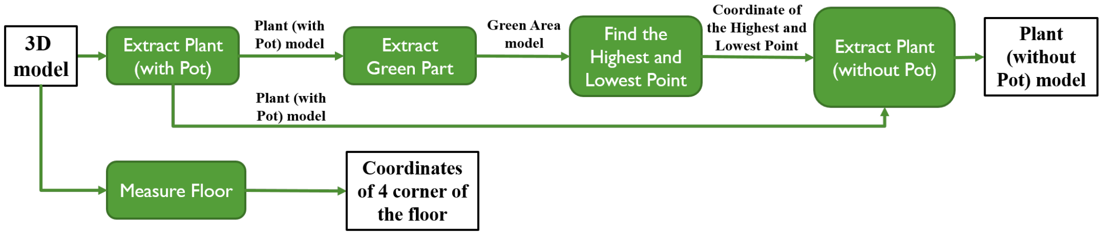


Figure 3. Flow of phases in background extraction.

Extract plant (with pot): Because the plant was placed inside the chamber at the fixed position which is the middle of the chamber, it is easy to determine the location of the plant in the chamber. The 3D model is scanned and checked within an area around the center point of the chamber. The parts of the model that are inside this area will appear as the plant, otherwise as the chamber (see Figure 4).

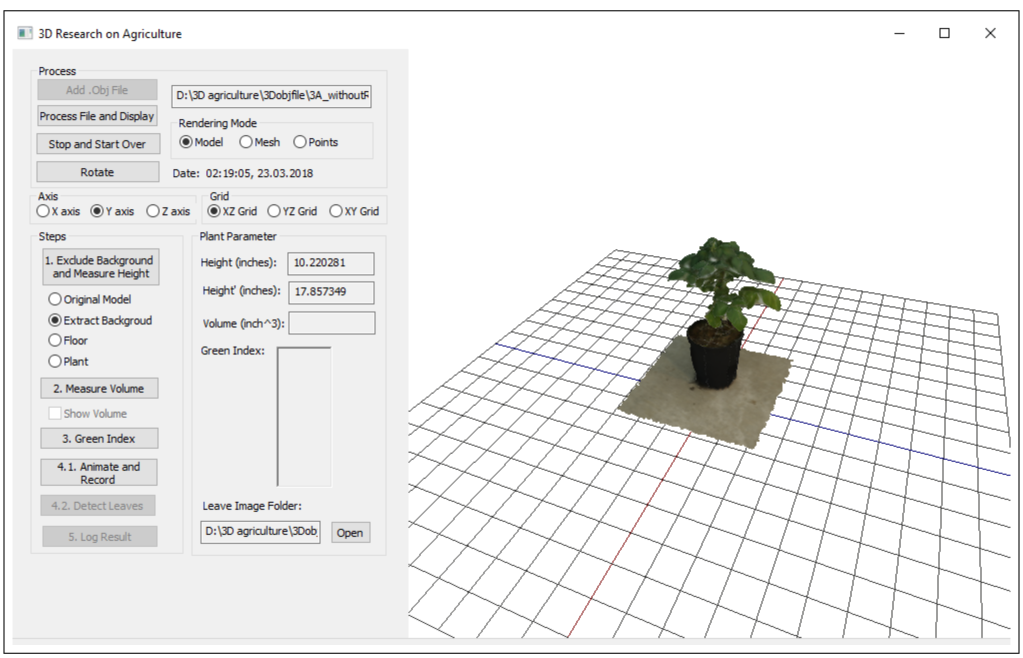


Figure 4. Screen shot of the plant (with pot) extraction step.

Extract green part: After having the plant (with pot) extracted during the previous step, a scanning process is performed to detect the green portion of the 3D model. The HSV index is used to detect green colors ranging from yellow-green to blue-green. Unlike the RGB color space in which color is formed in relation to primary colors (red, green, blue), HSV is named after three values: hue, saturation, and value. The HSV color space describes colors (hue) in relation to the shade (saturation) and the brightness value (see Figure 5). Hue is the color part; its value ranges from 0-360. Saturation indicates the gray level in the color; its value is from 0-100 percent. Value (or brightness) represent the brightness degree in the color, from 0-100 percent. Because HSV shows how people perceive color in real life in a better way than RGB, and in HSV there is a fixed range for each color, so HSV color space is usually used in effective color detection. See Figure 6 for a representation of HSV space. The Hue index used for detecting the green part of the 3D model of the plant is in the range: 70 - 170. The algorithm for green detection in AFES first converts the texture file of the 3D model of plant from RGB to HSV. Then, the system filters all pixels in the texture file (in HSV format) with an H value between 70 and 170, an S value between 100 and 255, and a V value from 45 to 255. The result of this step is a new texture file having only the green pixels. Finally, AFES uses the pixel value from the new texture file to filter the vertices of the 3D model creating a 3D model containing only the green parts.

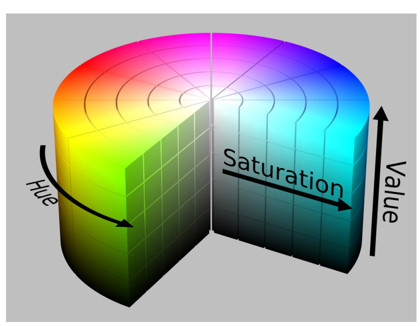


Figure 5. HSV cylinder.

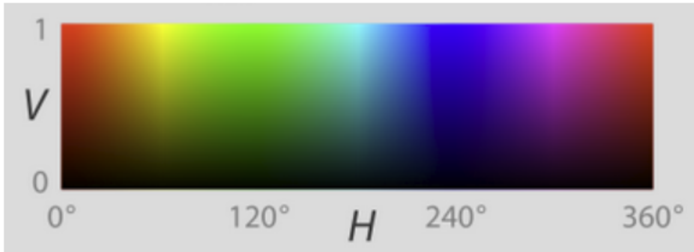


Figure 6. HSV range.

Find extreme points: From the green model extracted in the previous phase; the system scans the green model to find the highest and lowest points of the model. From this result, the height of the plant is calculated by: highest point - lowest point.

Extract plant (without pot): when the highest and lowest points of the green part is achieved, the 3D model of the plant (with pot) is scanned again with the restriction that all vertices must be within the range from the lowest and the highest point to extract only the plant without the pot.

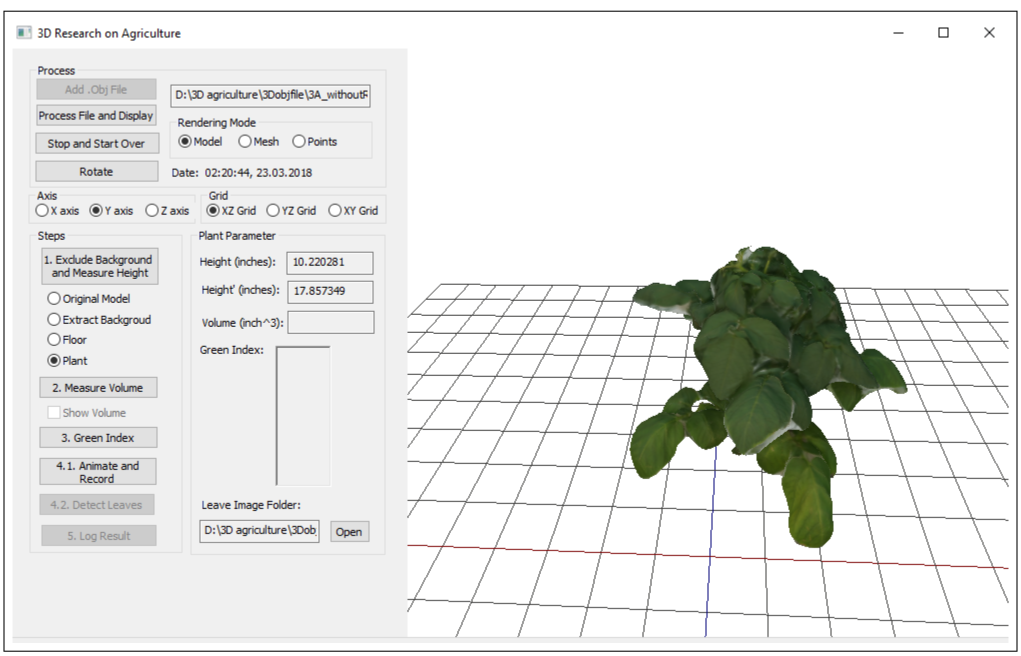


Figure 7. Screen shot of plant extraction (without pot) step.

Measure floor: Input of this phase is the original 3D model consisting of the chamber and a plant inside it. The method for extracting the floor uses a height threshold. This height threshold is the height of the floor. The 3D model is scanned to keep only the parts of the model that are below or equal to the height threshold (see Figure 8). Then, the 3D model of the floor is scanned to detect the coordinates of the 4 corners. From the 4 corners, the size of the floor (the width, the length) is determined. The reason for extracting the floor is to establish the size of the floor in model coordinates with the real size of the floor to determine the conversion between the 3D world and the real world.

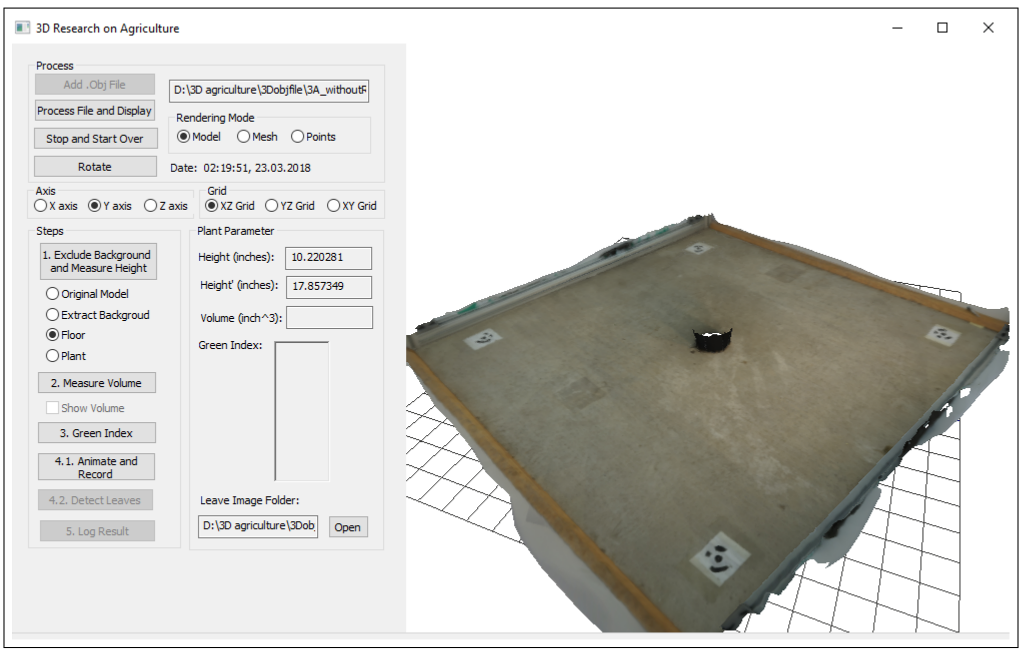


Figure 8. Screen shot of the floor extraction step.

3.1.3 Height measurement

Height is important information of the plant that needs to be collected in the growth analysis process. From the background extraction step previously, the highest point and lowest point of the plant is detected. These two points are used to calculate the height of the plant. There are two ways to measure the height of the plant in AFES. The first one is calculated from the highest point of the plant to the ground, the second is measured from the highest point of the plant to the lowest point of the plant (see Figure 3.1.3).

H1 = Highest point of plant - ground.

H2 = Highest point of plant - Lowest point of plant.

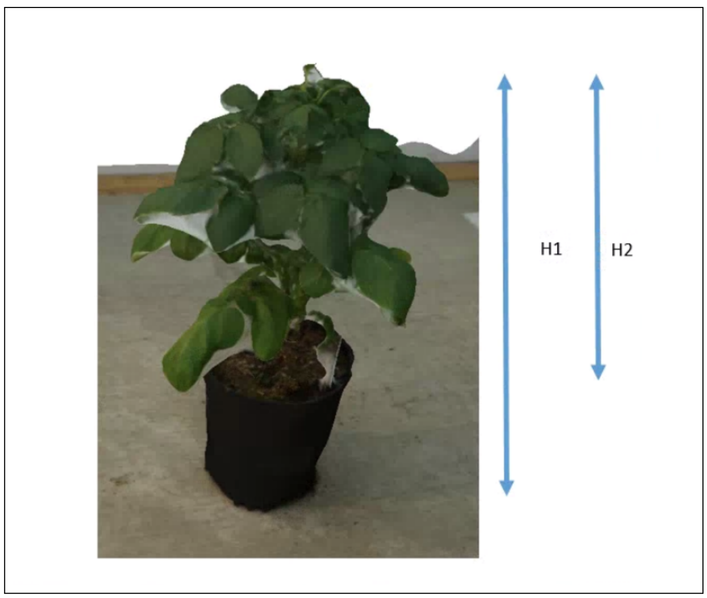


Figure 9. Two ways of plant height measurement.

3.1.4 Volume measurement

The volume of the plant is one feature that shows the status of plant growth at a specific time. In this step, AFES determines the smallest convex hull containing the plant. There are many algorithms to create the convex hull including quick hull algorithm [19], and gift wrapping algorithm [20]. In AFES, quick hull algorithm is chosen to support the volume calculation because of its fast implementation [19]. Figure 10 shows an example of a plant and its convex hull. The volume of the plant is approximated by the volume of the convex hull. The volume of a convex hull is calculated by following the steps shown in Figure 11. First, the center point of the convex hull is determined by taking the average of all vertex coordinates. Then, the center point together with each face on the convex hull create a tetrahedron. Finally, calculation of volume of a complex shape is converted into the sum of the volumes of the simple shape. For example, in Figure 12, Volume(convex hull) = Volume(C$E_1E_2E_5$) + Volume(C$E_1E_5E_6$) + Volume(C$E_6E_5E_4$) +...+ Volume(C$E_2E_4E_3$).

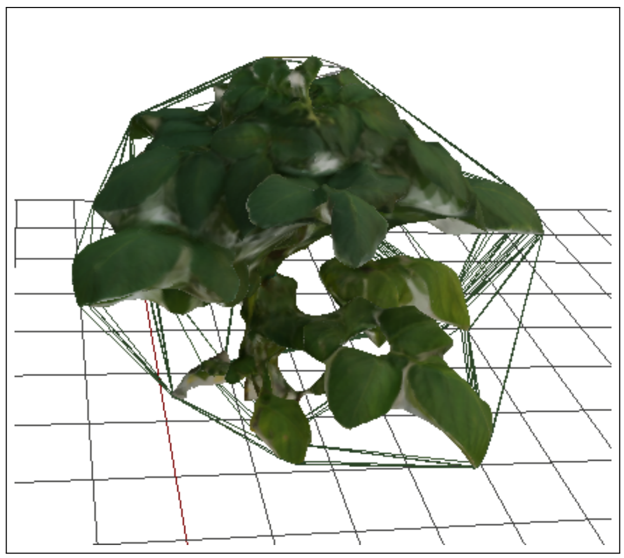


Figure 10. Screen shot of the convex hull that covers the plant.

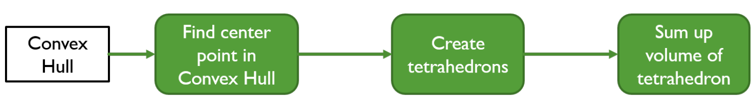


Figure 11. Volume calculation steps.

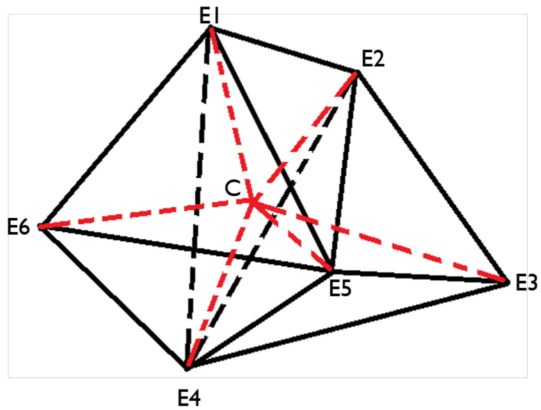


Figure 12. Volume measurement for Convex Hull.

3.1.5 Green Index

Information about the green index of the plant at a specific time will be useful in assessing the growth of a plant. The green index indicates how green the plant is. The green degree of the plant depends on many factors such as irrigation, photosynthesis, etc. However, in one plant, the green index is not the same at different heights. The green index at the top of the plant is lighter than at the bottom of the plant. To solve this problem, the plant height is divided into 10 segments, and the green color average degree is calculated for each segment. Figure 13 shows the calculated green color of a plant at different heights. In detail, by using HSV scanning, all green color pixels are detected from the texture image file of the 3D model. Instead of getting the average number of all green pixels of the whole plant, only the green pixels in each segment is considered to determine the average. In the system, the default number of segments is 10. If the number of segment increases, the quality of green index will be improved.

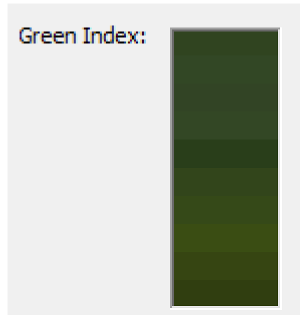


Figure 13. Screen shot of green index calculation step.

3.1.6 Leaf Detection

Another function of this system is detecting leaves on the plant, cropping an image of each leaf and saving them in a folder for further analysis. This functions include 2 main phases. The first phase is to record a video by moving a virtual camera around the plant from the bottom to the top of the plant. In the system, the virtual camera is automatically moved in a fixed route (this can be improved in the future to enable the camera to move in a changeable route). FFmpeg framework [21] is used to support converting and streaming the animation on rendering context of the automatic system into a .mp4 video file. In the second phase, the recorded video from phase 1 is sent to YOLO system to detect leaf on the plant and save images of leaf to a specific folder.

YOLO (You only look once) is a state-of-the-art, real-time object detection system developed by Redmon et al. [1]. YOLO was developed in 2015 and is widely used. The YOLO detection system includes 3 main processes. First, it resizes the input image to 448 x 448 resolution. Second, it applies a single convolutional network on the image. Finally, YOLO filters the detecting results by the model's confidence (see Figure 14). Previous detection systems adjust the function of classifiers or localizers to do the detection. They apply the model to an image at different locations and scales. Areas of the image with the high scoring are considered as a detection. YOLO utilizes a completely different method in which YOLO uses a single neural network on the whole image. This network breaks the image into smaller areas. The neural network then predicts bounding boxes and probabilities for each area. These bounding boxes are graded by the predicted probabilities.

YOLO has two main advantages over previous detection systems. First, YOLO runs very fast, YOLO version 1 can operate at 45Hz which is better than real time speed [22]. With YOLO version 2, the speed increases up to 91Hz depends on the resolution. YOLO's faster version can achieve up to 155Hz but with lower accuracy [1]. Second, YOLO can be trained the network on real-world images database with adequate accuracy because its neural network can learn generalized object detection. YOLO investigates the images as a whole during test time. Hence its predictions are established from the global context in the image. The YOLO detection network has 24 convolutional layers and two connected layers (see Figure 15). The 1x1 convolutional layers decrease the features space from previous layers. One restriction of YOLO is that it requires a high-end GPU and CUDA for fast operation [23].

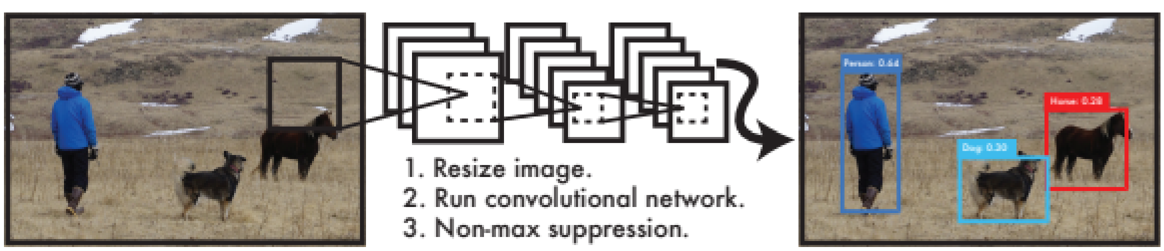


Figure 14. Three main processes of YOLO detection system [1].

Type	Filters	Size/Stride	Output
Convolutional	32	3×3	224×224
Maxpool		$2 \times 2/2$	112×112
Convolutional	64	3×3	112×112
Maxpool		$2 \times 2/2$	56×56
Convolutional	128	3×3	56×56
Convolutional	64	1×1	56×56
Convolutional	128	3×3	56×56
Maxpool		$2 \times 2/2$	28×28
Convolutional	256	3×3	28×28
Convolutional	128	1×1	28×28
Convolutional	256	3×3	28×28
Maxpool		$2 \times 2/2$	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Convolutional	256	1×1	14×14
Convolutional	512	3×3	14×14
Maxpool		$2 \times 2/2$	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	512	1×1	7×7
Convolutional	1024	3×3	7×7
Convolutional	1000	1×1	7×7
Avgpool		Global	1000
Softmax			

Figure 15. YOLO network layers [1].

Leaf detection training: YOLO must first be enabled to detect a custom object (leaf) by training it with a leaf database. To create the training database, many leaf images were collected from the internet and from the plant images taken in the chamber at the beginning of this research. 213 images of leaves were collected for the training database (see Figure 16). In this database, 90% of the images were used for training and 10% for testing. YOLO was trained for about 13,000 iterations to reach the smallest error (see Figure 17). In each iteration, 64 images were used for each batch of training. These 64 images were then divided by 32 to decrease GPU Video RAM (VRAM) requirements. The computer used for the training and detection step has a nVIDIA GeForce GTX 680 card and 2GB of VRAM. The trained YOLO was tested on the recorded video of the plant with an example result shown in Figure 18.

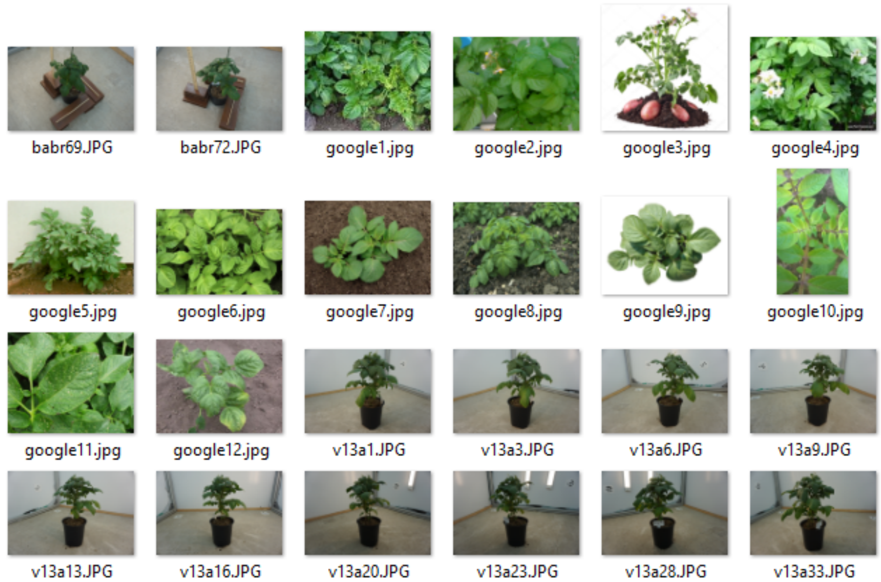


Figure 16. Example images from the leaf training database.

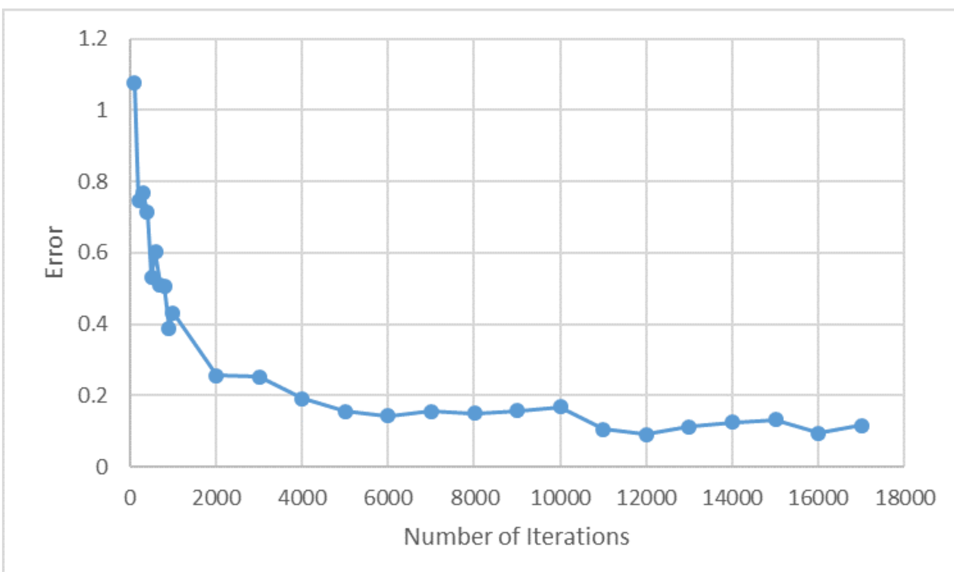


Figure 17. Learning curve of YOLO system for leaf training database.

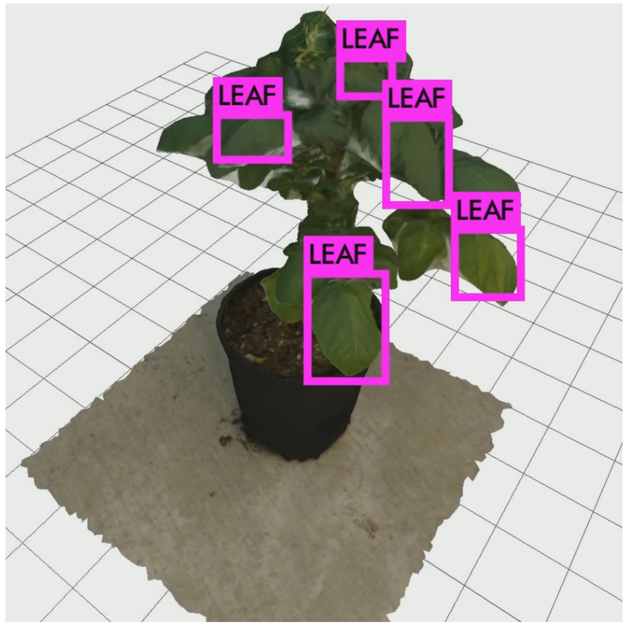
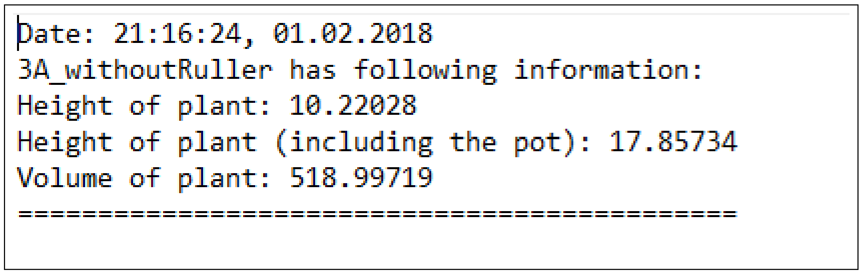


Figure 18. Screen shot of leaf detection using YOLO.

3.1.7 Data logging

In AFES, when a 3D model file is selected to process, a folder with a name including the 3D model name and a time stamp is automatically created. All information of features extracted from previous steps is logged into a text file with a time stamp (see Figure 19). The image of the green index, images of detected leaves, the recorded video of the plant, and the leaf detection video are also saved in this folder (see Figure 20). The purpose of this step is to keep a log of plant status over time. These logs will be useful for tracking and analyzing plant characteristics in the future.



Date: 21:16:24, 01.02.2018
3A_withoutRuller has following information:
Height of plant: 10.22028
Height of plant (including the pot): 17.85734
Volume of plant: 518.99719
===

Figure 19. Plant characteristics report.

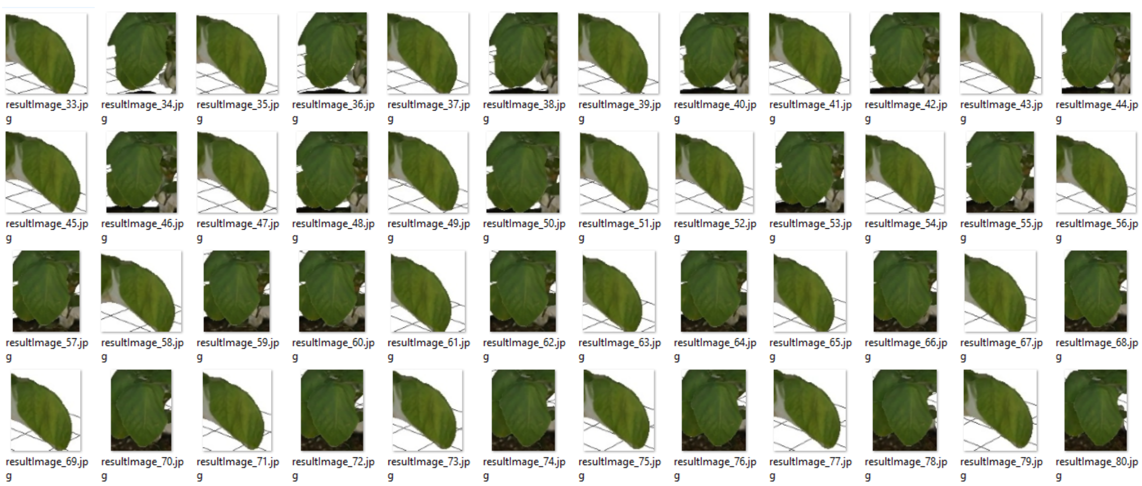


Figure 20. Leaves image logging.

3.2 Evaluation

One of the most important things that must be solved to achieve a good accuracy of the automatic system is a conversion factor between the real world and the 3D model world. This factor is used to convert a distance in the 3D world into a distance in the real world. The more correct the factor is, the more accurate the system is. At the beginning of this research, ground-truth information about plant height, leaves number, and the chamber size was collected. This data was used later on for the accuracy testing of the algorithms. In AgriLife center at Weslaco, Texas, plant images were taken from the indoor phenotyping chamber. In the chamber design (see Figure 21), the size of the floor is mentioned in detail. The floor's size in the real world was measured to map to the floor size extracted from the 3D model by AFES. From this mapping, the conversion factor between the real world and the 3D model world is obtained. This factor is used to convert the height of plant measured by system into inch unit. The result of height calculation is shown in Table I.

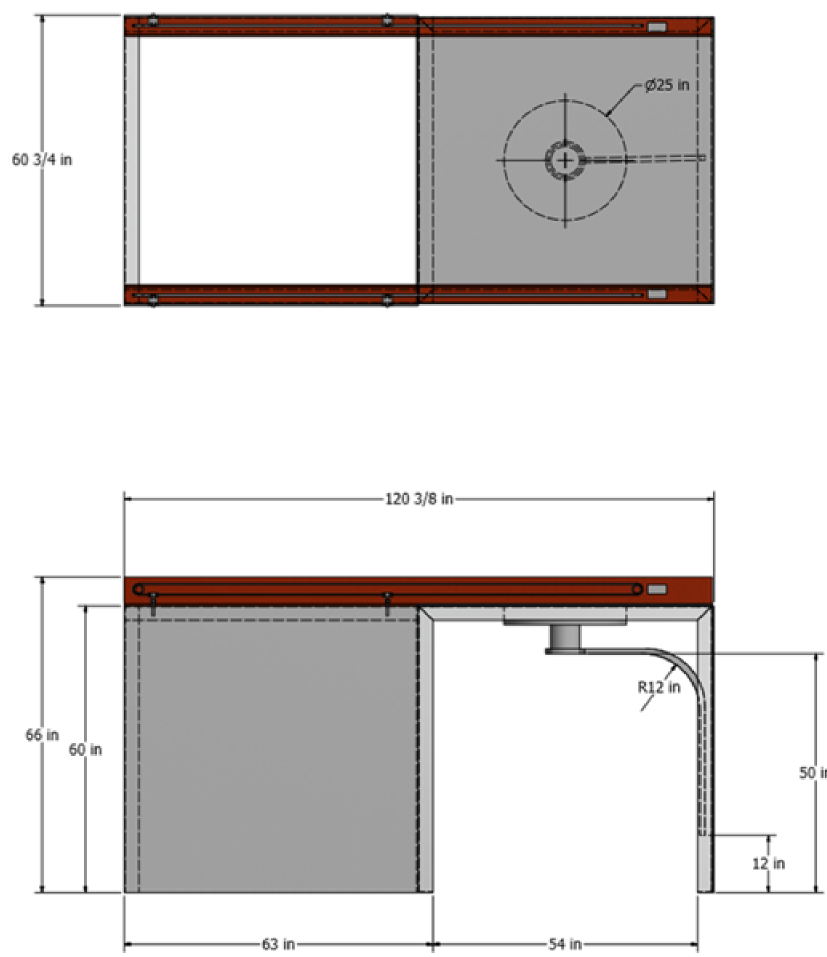


Figure 21. The chamber design from AgriLife center at Weslaco, Texas.

	Extracted Height (H1)	Accurate Height	Error	% Percent Error
Plant 1	17.85	17.8	0.05	0.28
Plant 2	17.97	18.5	0.53	2.86
Plant 3	17.68	17.25	0.43	2.49
Plant 4	20.78	21.25	0.47	2.21
Plant 5	16.77	18	1.23	6.83

Table I. Extracted height results

As being shown in Table I, the extracted height of plant 1 from the automatic system is nearly equal to the real plant height. For plant 2, 3 and 4, the differences between extracted heights and the accurate heights are about 0.5 inch (about 2.7% compared to its height). And for plant 5, the difference is a bit bigger, about 1.23 inches (6.8% compared to its height). The average of all differences was 0.44 inch (about 2.9% compared to the correct plant height).

A convex hull was used to approximate the volume of the plant. There are other methods to estimate the volume of the plant. One method approximates the volume of the plant to the volume of the smallest cylinder covering the plant. Another method considers the volume of the plant as the volume of an ellipsoid covering the plant (see Figure 22). These 3 methods were applied to calculate the volume of 5 plants, the results are shown in table II. The convex hull method had the smallest volume value which means the convex hull method approaches nearer to the correct volume of the plant than the others, since the volume of the plant can be no bigger than the convex hull which completely encloses the plant.

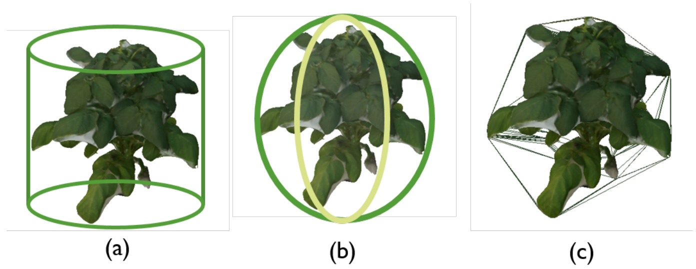


Figure 22. Methods for measuring volume of the plant: (a) Cylinder method (b) Ellipsoid method and (c) Convex Hull method.

	Cylinder Method	Ellipsoid Method	Convex Hull
Plant 1	1147.827	765.218	518.997
Plant 2	1079.726	719.817	587.444
Plant 3	1069.732	713.088	494.708
Plant 4	1532.695	1021.796	765.882
Plant 5	776.593	517.210	381.684

Table II. Summary of plant volume calculated from 3 methods (unit in $inch^3$).

For evaluating the leaf detection function, one of the advantages of this research is that the detection function only takes place with the 3D model including a chamber and a plant inside. There is no other thing that can confuse the detection function. The detection function may not detect all the leaves on the plant, but it did not detect a non leaf as a leaf; the false positive rate equals 0. To measure the false negative rate, the detection function was applied on 36 images of one plant. The percentages of leaves not detected in 36 images are shown in Table III. From the table, the average false negative rate was 0.76. The reasons for this problem could be the low quality of the 3D model, the low resolution of video input to YOLO detection system, or the size of leaf in the training database. To test the leaf size problem, the 213 images in the training databases were converted into their half size and quarter size. YOLO was trained again with these resized images for about 20,000 iterations but the average false negative rate was still the same with previous one. Hence, the leaf size is not a problem with YOLO detection. To improve the leaf detection function, the solutions may include raising the quality of the 3D model, increasing the power of computer running YOLO, enhancing the resolution of the recorded video, and having better training database.

	# miss/# leaves		# miss/# leaves		# miss/# leaves
Image 1	25/30	Image 13	27/30	Image 25	22/27
Image 2	21/24	Image 14	17/23	Image 26	18/22
Image 3	16/21	Image 15	18/21	Image 27	15/20
Image 4	18/24	Image 16	12/20	Image 28	12/20
Image 5	16/19	Image 17	15/18	Image 29	16/21
Image 6	19/25	Image 18	16/22	Image 30	20/25
Image 7	19/25	Image 19	20/25	Image 31	17/23
Image 8	17/23	Image 20	15/22	Image 32	20/25
Image 9	21/24	Image 21	23/26	Image 33	19/24
Image 10	11/19	Image 22	14/18	Image 34	9/14
Image 11	16/22	Image 23	18/23	Image 35	19/26
Image 12	16/23	Image 24	17/25	Image 36	18/24

Table III. Miss rate of leaf detection in 36 images.

CHAPTER 4

COMPARISON OF 2D AND 3D DATA VISUALIZATION

A proper data visualization can support data analysis and decision-making process and how data is presented graphically can influence the way a human perceives and understands data [24]. In order to find a suitable data visualization to improve the quality of plant characteristics understand and analysis, an experiment was conducted to explore the efficiency between 2D visualization and 3D visualization on human performance in data analysis. Volunteers in the experiment were required to answer questions about plant characteristics by looking at plant 2D images or a 3D model. Their performance is analyzed based on their task completion time and accuracy rate. The purpose of the experiment is to provide an appropriate approach of data visualization for determining plant characteristic.

4.1 Experiment Design

The participants in the experiment were required to be over 18 years old and have a science or engineering major at Texas A&M University-Corpus Christi. There was no restriction for gender of the participants. These inclusion criteria were to make sure that the participants were familiar with simple scientific questions and requirements. There were a total of 23 volunteers that took part in the experiment. At first, participants would have to decide whether to enter the experiment or not by reading and signing the research consent form. A survey questionnaire was then provided to them to collect demographic information including age, major of study, etc. The participant then logged into a web page to start the experiment. The flow of phases that the participant had to go through in the experiment is shown in Figure 23.

Figure 23. Phases that the participant has to go through in the experiment.

Before going into the main part of the experiment which is answering questions about plant characteristics, the participants needed to have instructions and practice to get used to how the 3D model and the 2D images are presented and how to control the 2D images and 3D model in the experiment. The participants were required to practice zoom in and out the images and turn the 3D model using a mouse until they felt confident with the control function provided in the experiment (see Figure 24 and 25). After that, introductions of 2 types of questions in the experiment were presented to the participants. They asked questions about the number of leaves and to estimate the height of a plant. The participants were introduced to how the questions were to be presented and how to measure the features required. With questions using the 3D data visualization type, there was a 3D model that the participant needed to use the mouse to look around the plant or zoom out/in the plant. With questions using 2D images, they were provided a several images taken from various viewpoints of the plant. In the main part, the participants started answering questions about plant characteristics. They had to finish 16 questions in total. In each question, their answer and completion time were stored for performance analysis.

The questions focused on 2 main characteristics of the plant including the height of the plant and the number of leaves. Every question was presented with 2D images or a 3D model. The order of the questions was randomized to eliminate the chance of any learning to affect results. Figure 26 shows an example of a question on leaf counting. Under each image is the angle information where the camera took the picture compared to the starting point. Figure 27 shows an example of a height estimation question. A yardstick was placed beside the plant so that the viewer could use the information on the yardstick to make a decision. All answer options provided per question were in terms of range.

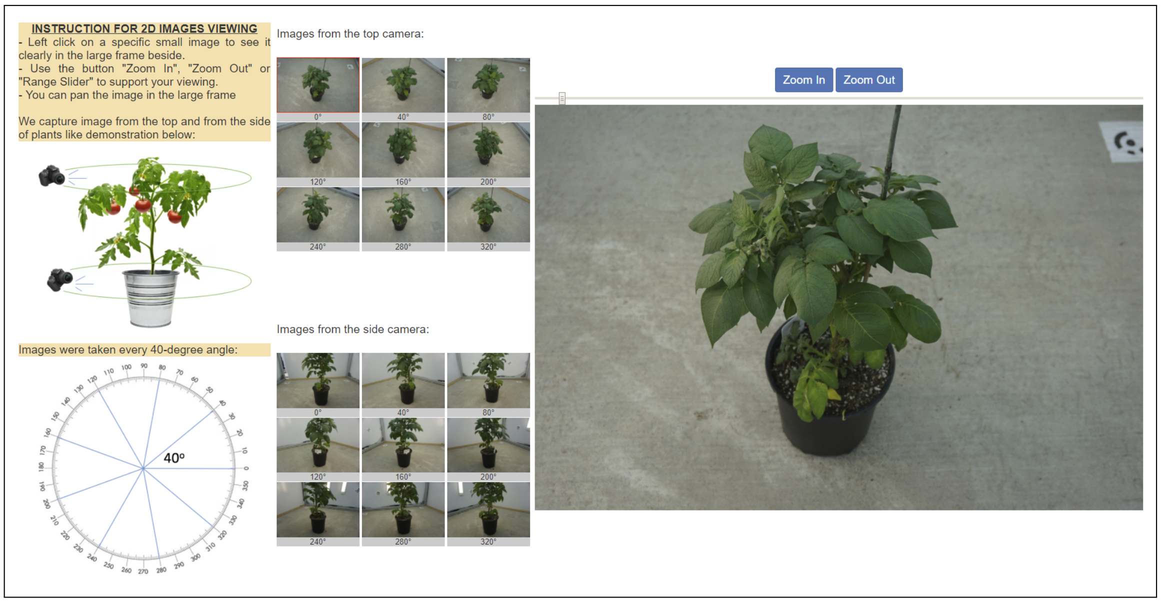


Figure 24. Screen shot of practising 2D images control function.

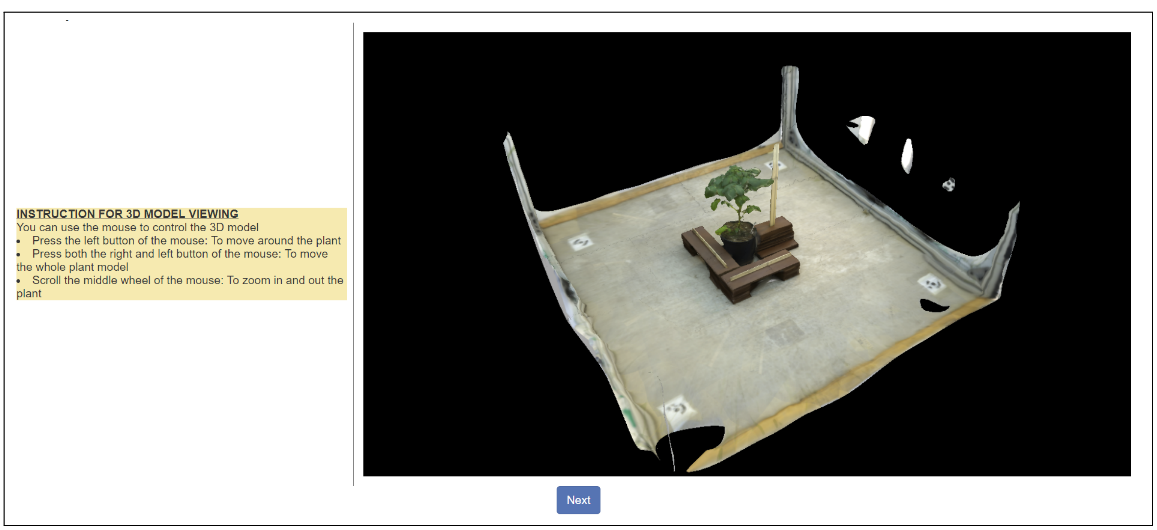


Figure 25. Screen shot of practising 3D model control function.

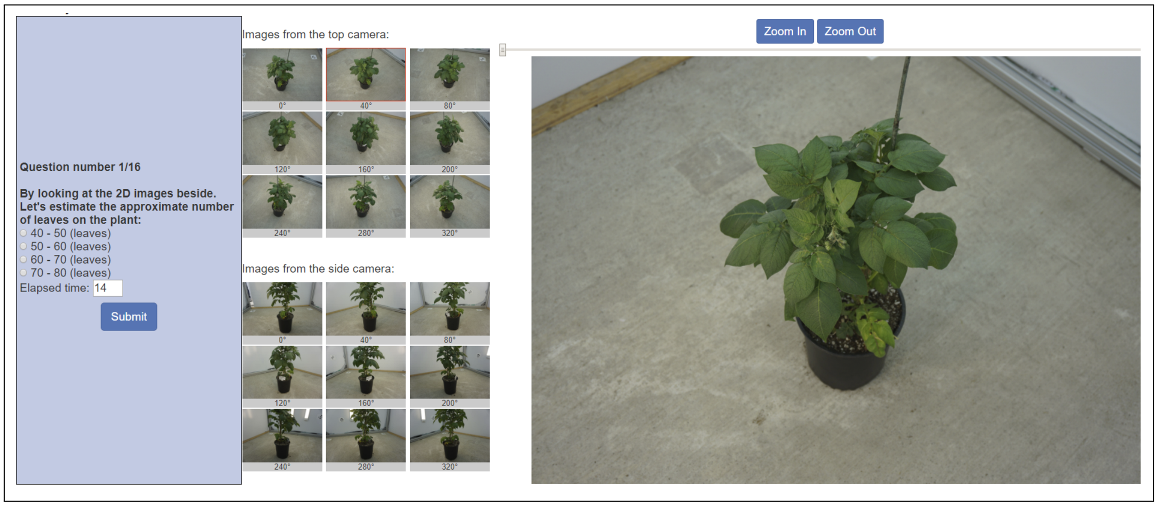


Figure 26. Presentation of leaves counting question (2D visualization type).

Figure 27. Presentation of height estimating question (3D visualization type).

After finishing the experiment, the participants had to fill in the form to show their opinion about the task load in the experiment. There are five sub-scales will be taken into consideration in task load index: mental demands, physical demands, performance, effort, and frustration. Each scale can be graded from 0 to 100 points. Based on the participant's thoughts and feelings, they decided the scores for each scale. I referenced the NASA Task Load Index (TLX) for the content of the form. Below is the scale descriptions:

Mental demand: How much mental and perceptual activities was required?

Physical demand: How much physical activity were required?

Performance: How successful do you think you were in accomplishing the goals of the task?

Effort: How hard did you have to work to accomplish your level of performance?

Frustration: How insecure, discouraged, irritated, stressed, and annoyed were you?

4.2 Results and Evaluation

23 students of Texas A&M University-Corpus Christi participated in the experiment counting number of leaves and measuring height. The average completion time and average accuracy rate of the participants were compared. To confirm the difference of the average values was stable and not random, a t-test was performed. Table IV summarizes the average value and standard deviation of task accuracy and task completion time.

	Height Estimation				**Leaf Counting**			
	2D images		**3D model**		**2D images**		**3D model**	
	Mean	σ	**Mean**	σ	**Mean**	σ	**Mean**	σ
Task Accuracy	0.696	0.199	0.704	0.169	0.449	0.258	0.261	0.265
Task Completion Time (seconds)	42.739	15.052	53.539	17.601	84.681	31.231	82.710	33.969

Table IV. Average (mean) and standard deviation (σ) values of task accuracy and task completion time.

The t-test method is a commonly used statistical method to check whether or not the difference between two sets of data most likely reflects a true difference in the population from which the groups were sampled. If the t-test value is less than 0.05 then the difference is true and stable; otherwise, there is a possibility that the difference is just random. The t-test formula is below:

$$t = \frac{m_A - m_B}{\sqrt{\frac{S_A^2}{n_A} - \frac{S_B^2}{n_B}}}$$

Consider A and B represent the two groups to compare, where:

m_A and m_B represent the means of groups A and B, respectively,

n_A and n_B are the sizes of group A and B, respectively, and

S^2 is the variance of the two samples. Its formula is shown below:

$$S^2 = \frac{\sum(x - m_A)^2 + \sum(x - m_B)^2}{n_A + n_B - 2}$$

With the plant height estimation, the average accuracy rate that participants obtained from 2D and 3D data visualization are approximately the same (see Figure 28). Specifically, the accuracy rate when using 2D images or using 3D model equaled 0.7 approximately. However, average duration spent for answering question with 3D model is longer than with 2D images (see Figure 29). Completion time with 2D images was 43 seconds, completion time with 3D model equaled 53 seconds. Therefore, on average, the participants achieved the same accuracy rate with both 2D and 3D visualization but they spent more time on the question using a 3D data visualization than with a 2D data visualization. The t-test value for the completion time between 2D visualization (mean 42.739 seconds, σ 15.052 seconds) and 3D visualization (mean 53.539 seconds, σ 17.601 seconds) was 0.0001 showing that completion time with 3D model is statistically longer than with 2D images.

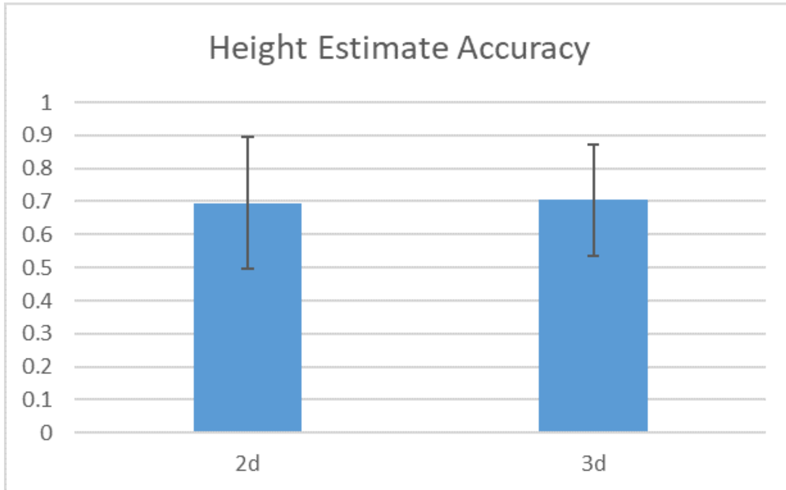


Figure 28. Average answer accuracy for plant height estimating question.

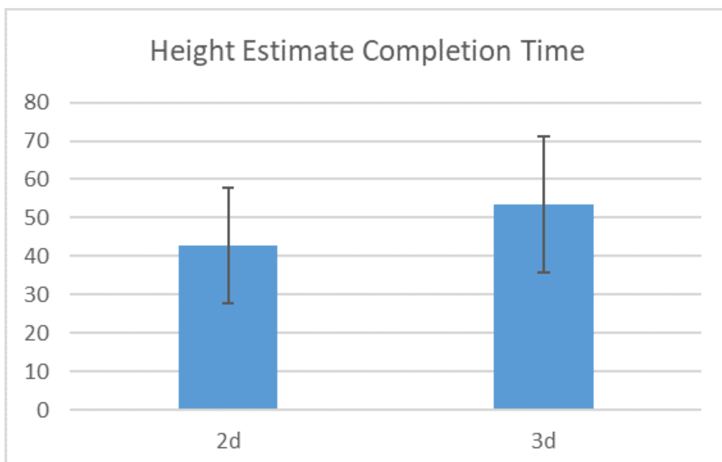


Figure 29. Average completion time for plant height estimating question.

For leaves counting, the difference in performance of participants between 2D and 3D data visualization is clearer (see Figure 30). The average accuracy rate achieved from 2D images is much higher than 3D model. In detail, the accuracy rate when using 2D images was 0.43 while the accuracy rate when using 3D model was 0.27. However, average time spent for 2D images and 3D model had small difference (see Figure 31). The completion time with 2D images equaled 84 seconds and the completion time with a 3D model equaled 82.9 seconds. For testing the accuracy rate between 2D images (mean 0.449, σ 0.258) and 3D plan model (mean 0.261, σ 0.265), the t-test value was 0.024 showing that accuracy attained from 2D data visualization is statistically better than 3D data visualization. The t-test value for the completion time between 2D images (mean 84.681 seconds, σ 31.231 seconds) vs. 3D model (mean 82.710 seconds, σ 33.969 seconds) was 0.7 showing no statistical difference.

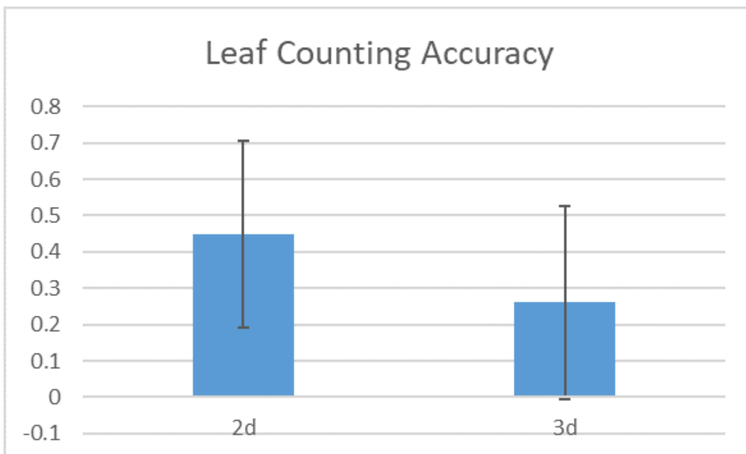


Figure 30. Average answer accuracy for counting leaves.

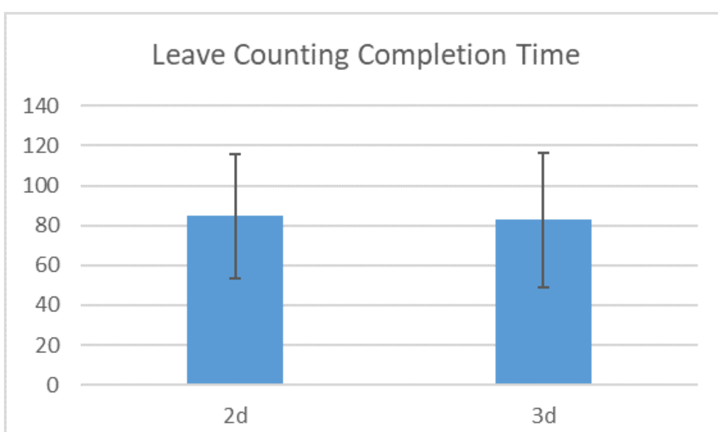


Figure 31. Average completion time for leaves counting question.

From previous conclusion of participant performance, in the height measurement task, both 2D and 3D visualization brought the same accuracy rate to the participants but they had to spent more time on questions using the 3D model. For counting leaves, the participants spent roughly the same duration to finish, but the 2D visualization gave much better results. All in all, in our experiment, 2D visualization brings better results than a 3D visualization.

We also analyzed workload for the experiment using the participant subjective assessments from the NASA TLX index. These data indicate the degree of difficulty experienced when performing a task. There were five sub-scales to be analyzed including mental demand, physical demand, performance, effort, and frustration. The sub-scale performance shows the degree of confidence that the participant felt about their answer, the mental demand indicates the participant's opinion about the mental difficulty of the task, level of irritation and stress from them is shown in the frustration sub-scale. The percentage values for each sub-scale was compared for the 2D and 3D visualization. For estimating height, task load index summary is shown in Figure 32, the participants preferred the 3D model because they had the overall view of the plant. They felt more confident with the results using a 3D model although they had to spend more time and effort to process and control the 3D model. However, in fact, they attained the same accuracy rate in 3D model and 2D images. With the leaves counting task load summary, as being seen in Figure 33, the participants preferred the 2D-image type and they had more confidence with the result using 2D images type, they said they liked the familiarity and simplicity with 2D images although they also felt frustrated with 2D images sometimes.

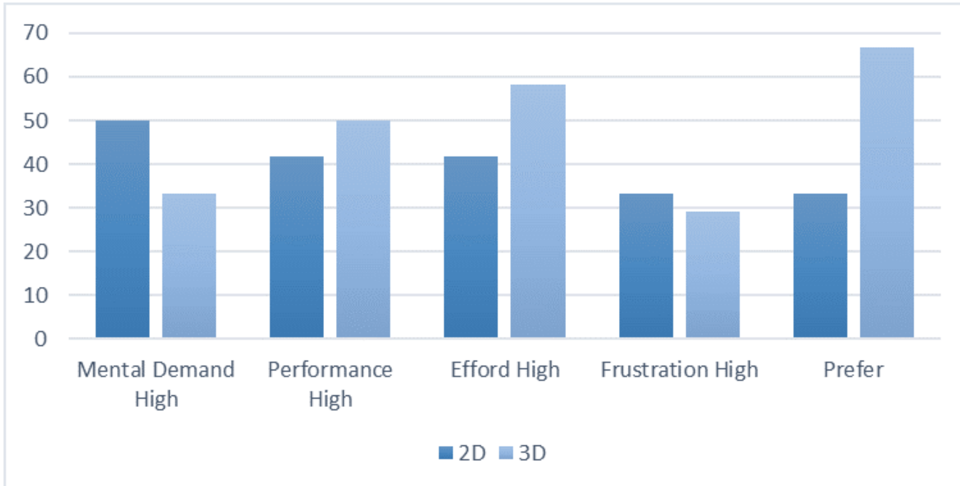


Figure 32. Summary of task load indexes for height estimating question.

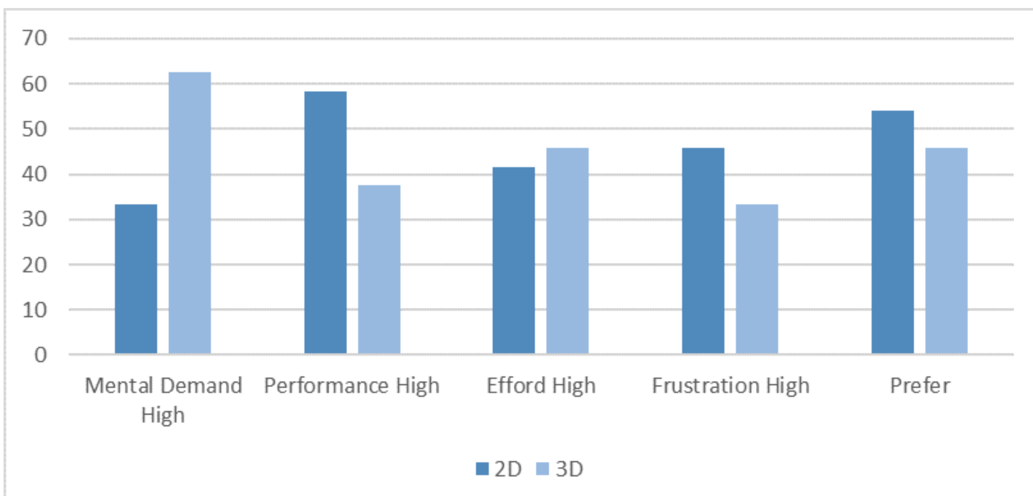


Figure 33. Summary of task load indexes for leaves counting question.

CHAPTER 5

CONCLUSION AND FUTURE WORK

In this thesis, an automatic system to calculate the characteristics of a plant is presented. The system allows users to determine the plant height, the volume of the plant, and the greenness at different height of the plant. YOLO is used to implement leaf detection in the system. The leaf detection has the false positive rate of 0, but it can not detect all the leaves on the plant at one time. The leaf detection ability depends on the quality of the system of the 3D model, and the resolution of the images or video input to YOLO system.

An experiment to investigate the effectiveness of 2D and 3D data visualization on human performance was conducted. In the experiment, participants were asked to answer questions about plant characteristics including plant height and number of leaves. The answers and task completion time for each question were collected for the performance analysis. A t-test was performed to analyze the data collected from the experiment. We found that 2D data visualization gives better results than using 3D but only minimally. The experiment also took into consideration the analysis of cognitive workload; Participant opinions about the physical demand, the mental demand, the performance, and the frustration level in the experiment were collected. For the height estimation, the participants preferred the 3D model although they attained the same accuracy rate in both 2D and 3D, for the counting leaves, participants preferred 2D images and they also obtained better accuracy rate with 2D images.

There are many modifications that can be performed to improve the quality of the system. There are some factors that affect the accuracy of the system. One is the quality of the 3D model. If the quality of the 3D model is low, the data extracted from the 3D model will have low accuracy. Another is the problem of confusing parts in the 3D model. Some plants have a stick placed right beside the plants to straighten the plant's stem. If the stick has a green color, it will make the system confuse it with the plant. Therefore, having a better 3D model and avoiding the confusing part in the background during the image collection step will help the AFES improves its accuracy. Moreover, a better 3D model can be obtained by changing the light bulbs in the chamber to provide better lighting condition or modifying the aperture and angle of the cameras to increase the quality of images taking phase. Another thing is to make sure the stick used with the plant has a different color from the plant so the algorithm will not mistake it with the plant.

YOLO can be trained to detect other parts of the plant such as flower or fruit. Knowing the state of plant flower and fruit can support more on the plant analysis. Some functions can also be developed to allow the users to do the segmentation process, pre-define the route of flight of the virtual camera (inside the 3D world) or use the computer mouse to measure any distance in the 3D model world. Furthermore, for now, AFES just does the leaf detection function and can not count the leaves on the plant. AFES could be upgraded in a way that every time a leaf is detected on the 3D model of plant, AFES can remove 3D the detected leaf from the the plant model. This can make the counting leaf function feasible because it allows YOLO to function more effectively and not confuse with the same leaf over again.

REFERENCES

[1] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. *arXiv preprint*, 2017.

[2] CC Yang, SO Prasher, JA Landry, J Perret, HS Ramaswamy, et al. Recognition of weeds with image processing and their use with fuzzy logic for precision farming. *Canadian Agricultural Engineering*, 42(4):195–200, 2000.

[3] Czesław Puchalski, Józef Gorzelany, Grzegorz Zagula, and Gerald Brusewitz. Image analysis for apple defect detection. *Biosystems and agricultura engineering*, 8:197–205, 2008.

[4] Jing Jin, Jinwei Li, Guiping Liao, Xiaojuan Yu, and Leo Christopher C Viray. Methodology for potatoes defects detection with computer vision. In *International symposium on information processing*, pages 346–351, 2009.

[5] Martin Weis and Roland Gerhards. Detection of weeds using image processing and clustering. *Bornimer Agrartechnische Berichte*, 69:138–144, 2009.

[6] Manuel JCS Reis, Raul Morais, Emanuel Peres, Carlos Pereira, Olga Contente, S Soares, António Valente, José Baptista, Paulo Jorge SG Ferreira, and J Bulas Cruz. Automatic detection of bunches of grapes in natural environment from color images. *Journal of Applied Logic*, 10(4):285–290, 2012.

[7] JP Sansao, MS Jnior, Leonardo A Mozelli, FA Pinto, and Daniel M Queiroz. Weed mapping using digital images. In *Proceedings of International Conference on Agricultural Engineering-CIGR*, number 1931, 2012.

[8] J Romeo, Gonzalo Pajares, M Montalvo, José Miguel Guerrero, María Guijarro, and Jesús Manuel de la Cruz. A new expert system for greenness identification in agricultural images. *Expert Systems with Applications*, 40(6):2275–2286, 2013.

[9] Eliyahu Efim Kelman and Raphael Linker. Vision-based localisation of mature apples in tree images using convexity. *Biosystems engineering*, 118:174–185, 2014.

[10] Kirsten Risden, Mary P Czerwinski, Tamara Munzner, and Daniel B Cook. An initial examination of ease of use for 2d and 3d information visualizations of web content. *International Journal of Human-Computer Studies*, 53(5):695–714, 2000.

[11] Colin Ware. Designing with a 21/2d attitude. *Information Design Journal*, 10(3):258–265, 2000.

[12] Dipl-Inform Frank Althoff, Gregor McGlaun, Manfred Lang, and Gerhard Rigoll. Comparing an innovative 3d and a standard 2d user interface for automotive infotainment applications. In *Mensch & Computer 2003*, pages 53–63. Springer, 2003.

[13] Woodrow Barfield and Robert Robless. The effects of two-or three-dimensional graphics on the problem-solving performance of experienced and novice decision makers. *Behaviour & Information Technology*, 8(5):369–385, 1989.

[14] Brandie M Stewart, Jessica M Cipolla, and Lisa A Best. Extraneous information and graph comprehension: Implications for effective design choices. *Campus-Wide Information Systems*, 26(3):191–200, 2009.

[15] Kyungdoh Kim, Robert W Proctor, and Gavriel Salvendy. Comparison of 3d

and 2d menus for cell phones. *Computers in Human Behavior*, 27(5):2056–2066, 2011.

[16] Peter Demian, Kirti Ruikar, Tarun Sahu, and Anne Morris. Three-dimensional information retrieval (3dir): exploiting 3d geometry and model topology in information retrieval from bim environments. 2016.

[17] Richard B Dull and David P Tegarden. A comparison of three visual representations of complex multidimensional accounting information. *Journal of Information Systems*, 13(2):117–131, 1999.

[18] Kenton McHenry and Peter Bajcsy. An overview of 3d data content, file formats and viewers. *National Center for Supercomputing Applications*, 1205:22, 2008.

[19] C Bradford Barber, David P Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software (TOMS)*, 22(4):469–483, 1996.

[20] Kokichi Sugihara. Robust gift wrapping for the three-dimensional convex hull. *Journal of Computer and System Sciences*, 49(2):391–407, 1994.

[21] Ffmpeg. `https://www.ffmpeg.org/`. Accessed: 2018-04-26.

[22] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[23] NVIDIA Developer. Cuda toolkit. `https://developer.nvidia.com/cuda-downloads`. Accessed: 2018-04-25.

[24] Melanie Tory and Torsten Moller. Human factors in visualization research. *IEEE transactions on visualization and computer graphics*, 10(1):72–84, 2004.

APPENDIX A

SAMPLE OF A OBJ FILE

```
# List of geometric vertices, with (x, y, z [, w]) coordinates, w is optional and
defaults
to 1.0.
v 0.123 0.234 0.345 1.0
...
# List of texture coordinates, in (u, v [, w]) coordinates, these will vary between 0
and 1, w is optional and defaults to 0.
vt 0.500 1 [0]
...
# List of vertex normals in (x, y, z) form; normals might not be unit vectors.
vn 0.707 0.000 0.707
...
# Polygonal face element, with format:  f v1/vt1/vn1 v2/vt2/vn2 v3/vt3/vn3
f 6/4/1 3/5/3 7/6/5
```

APPENDIX B

SAMPLE OF A MTL FILE

```
#Material name statement:
newmtl my_mtl
#Material color and illumination statements:
#To specify the ambient, diffuse and specular reflectivity of the current material
using RGB values in the range 0.0 to 1.0:
Ka 0.0435 0.0435 0.0435
Kd 0.1086 0.1086 0.1086
Ks 0.0000 0.0000 0.0000
#To specify the specular exponent for the current material.  A high exponent results
in a tight, concentrated highlight.  Ns values normally range from 0 to 1000.
Ns 10.0000
#To specify the illumination model to be used in shading calculations, the "illum"
statement is used.  The illumination model number can be a number from 0 to 2,
described below
illum 2
#Texture map statements:
map_Ka mapfile.jpeg
map_Kd mapfile.jpeg
```