



The University of Adelaide
The School of Mechanical Engineering

Image Processing and Analysis for Autonomous Grapevine Pruning

by

Ming GAO

*Thesis submitted for the degree of
Master of Engineering Science on 07/2011*

Table of Contents

| | |
|---|-----|
| Abstract..... | III |
| List of Figures..... | V |
| List of Tables..... | VI |
| Statement of Originality..... | VII |
| Acknowledgements..... | IX |
| Publications..... | XI |
| Chapter 1 Introduction..... | 1 |
| 1.1 Motivation..... | 1 |
| 1.2 Introduction of Vine Pruning..... | 2 |
| 1.2.1 The Grapevine..... | 2 |
| 1.2.2 The Annual Growth Cycle..... | 3 |
| 1.2.3 Reasons for Grapevine Pruning and Automated Pruning..... | 4 |
| 1.2.4 Two-Bud Spur Pruning..... | 8 |
| 1.3 Autonomous Grape Pruning..... | 11 |
| 1.4 Outline..... | 12 |
| Chapter 2 Literature Review..... | 15 |
| 2.1 Image Processing..... | 15 |
| 2.2 Image Analysis..... | 17 |
| 2.3 Stereo Vision..... | 19 |
| 2.4 Relevant Projects..... | 21 |
| 2.5 Aim of This Research..... | 22 |
| Chapter 3 Obtaining 2D Cutting Positions..... | 25 |
| 3.1 Introduction..... | 25 |
| 3.2 Image Processing..... | 26 |
| 3.2.1 RGB image to Gray-Scale Image..... | 27 |
| 3.2.2 Analysing the Image..... | 29 |
| 3.2.3 Gray-Scale Image into Binary Image..... | 32 |
| 3.3 Image Analysis..... | 35 |
| 3.3.1 Locating the Cordon..... | 36 |
| 3.3.1.1 Procedures..... | 37 |
| 3.3.1.2 Results..... | 39 |
| 3.3.1.3 Discussion..... | 39 |
| 3.3.2 Locating the Branches..... | 41 |
| 3.3.2.1 Method One-Segmentstion the Brasnch..... | 41 |
| 3.3.2.2 Method Two-Moving the Branch..... | 43 |
| 3.3.2.3 Method Three-sloping Changing..... | 45 |
| 3.3.2.4Results and Discussions..... | 47 |
| 3.3.3 Locating the Nodes..... | 49 |
| 3.4 Conclusion..... | 50 |
| Chapter 4 Experiments on Obtaining Two-Dimensional Pruning Positions..... | 51 |

| | |
|---|-----|
| 4.1 Introduction..... | 51 |
| 4.2 Procedures..... | 51 |
| 4.3 Discussion..... | 58 |
| 4.4 Concluding Remarks | 59 |
| Chapter 5 Stereo Vision and Experiments for 3D Pruning Positions..... | 61 |
| 5.1 Introduction of the Projection Models..... | 61 |
| 5.1.1 Basic Information of the Pinhole Camera Model..... | 61 |
| 5.1.2 Basic Information of the Linear Camera Model..... | 63 |
| 5.2 Camera Calibration..... | 64 |
| 5.2.1 Hardware Calibration..... | 65 |
| 5.2.2 Software Calibration..... | 68 |
| 5.3 Space Coordinates Calculations | 72 |
| 5.3.1 Coordinate Calculation in the Pinhole Camera Model | 72 |
| 5.3.2 Coordinate Calculation in the Linear Camera Model..... | 76 |
| 5.3.3 Relationship of Calculations in the Two Models | 77 |
| 5.4 Obtaining of the Three Dimensional Information | 78 |
| 5.5 Concluding Remarks | 83 |
| Chapter 6 Conclusions and Recommendations | 85 |
| 6.1 Conclusions..... | 85 |
| 6.2 Some Recommendations | 86 |
| References..... | 90 |
| Appendix A: iBot Camera | 95 |
| Appendix B: The Ten Images Captured for the Experiments | 100 |

Abstract

In recent years, efforts are made to automate vineyard operations to cap the ever increasing labour cost. However, one of the operations that have not been completely automated is grapevine pruning. A robotic machine for grapevine pruning needs to respond to the changing physical characteristics of the environment, and to date, no algorithm can accurately identified appropriate positions for grapevine pruning in a variety of environmental conditions. The aim of this research was therefore to develop a new algorithm using image processing, image analysis and stereo vision system to determine pruning positions and making automatic grapevine pruning possible.

In order to get the pruning positions accurately and automatically, images taken from two cameras are processed and analysed. Utilizing the latest computer vision techniques, the algorithm takes three steps before the final cutting positions are derived. First, the uploaded images are pre-processed by the so called image processing phase during which binary image is obtained from the original image. Second, image analysis technique is employed to identify different parts of grape vine and obtain the 2D positions of the cutting points. Novel algorithms are proposed to locate the cordon, the branch and finally the nodes step by step with high accuracy. Both the locating precision and computing complexity of the algorithms are considered, so that the images could be processed incessantly. Finally, cameral calibration and stereo vision were also performed after that to determine the 3D pruning point, so the system can inform the robotic machine to perform the pruning.

For the experiments, images were taken from the vineyard which located in the Adelaide National Wine Centre and the University of Adelaide Waite Campus. Ten images full of canes were analysed, and an 85% success rate for pruning positions was achieved. 85% of the cutting positions found by the proposed

method were consistent with those found by experts, while the rest 15% of the cutting positions found by the proposed method would not affect the growth of the grapevine significantly in the next year although they don't match the positions found by experts perfectly. The experiment verified the accuracy and application potential of proposed algorithm.

The contribution of this research has three folds:

- a) The latest research and application of automatic grapevine pruning is reviewed;
- b) A novel automatic grapevine pruning algorithm using computer vision technique to automatically identify important grapevine features, locate the pruning points, and derive their three dimensional coordinates is proposed, researched and developed;
- c) The proposed algorithm can be also applicable to other agricultural operation automation other than grapevine pruning.

List of Figures

| | |
|--|----|
| Figure 1-1: Structure of a grapevine [Crossen97] | 3 |
| Figure 1-2: Annual growth cycle of a grapevine [Svensson02] | 4 |
| Figure 1-3: Two types of secateurs..... | 5 |
| Figure 1-4: The second generation of pruning machine..... | 7 |
| Figure 1-5: Two-bud spur pruning | 9 |
| Figure 1-6: Spur pruning. In the pruning, the mature shoot which is most distant from the permanent cordon is generally removed and the closest one is cut back to a 2 node spur [Ludvigsen07] | 10 |
| Figure 1-7: Flowchart of proposed approach..... | 11 |
| | |
| Figure 2-1: Thermal image of a grapefruit | 17 |
| Figure 2-2: From two-dimensional to three-dimensional..... | 20 |
| | |
| Figure 3-1: The flow chart to get the two-dimensional pruning positions | 26 |
| Figure 3-2: The RGB colour space model [Gonzalez04] | 27 |
| Figure 3-3: Image (No. 002) (a) RGB image. (b) Gray-scale image..... | 29 |
| Figure 3-4: (a) Image viewer window. (b) Pixel region window | 30 |
| Figure 3-5: (a) Image viewer window. (b) Pixel region window | 31 |
| Figure 3-6: The histogram of the RGB image in Figure 3-3 (a)..... | 32 |
| Figure 3-7: Comparison of the original image and image after thresholding..... | 34 |
| Figure 3-8: Black and white image..... | 35 |
| Figure 3-9: The flow chart of the three steps in image analysis | 36 |
| Figure 3-10: Flowchart of the cordon determining..... | 38 |
| Figure 3-11: Bottom of the cordon | 39 |
| Figure 3-12: Locating the cordon | 39 |
| Figure 3-13: Image (No.061)..... | 41 |
| Figure 3-14: The cordon | 42 |
| Figure 3-15: Parts of the branch | 42 |
| Figure 3-16: (a) Locating the branches; (b) The distance from Y1 to Y2..... | 42 |
| Figure 3-17: The flow chart of locating the branch..... | 43 |
| Figure 3-18: Three steps to locate the branch..... | 45 |
| Figure 3-19: From RGB to black and white..... | 47 |
| Figure 3-20: Part of the cane and its slope, a) Part of a cane, b) Slope changing of this cane | 47 |
| Figure 3-21: The workflow for method two..... | 48 |
| | |
| Figure 4-1: Original RGB image | 52 |
| Figure 4-2: From RGB to black and white image | 52 |
| Figure 4-3: The iterative process of locating the nodes..... | 57 |

| | |
|---|----|
| Figure 5-1: Example for a perspective projection [Thomas03]..... | 62 |
| Figure 5-2: Geometry of the pinhole camera model [Thomas03] | 62 |
| Figure 5-3: Geometry of the linear camera model [Thomas03]..... | 63 |
| Figure 5-4: Simplified linear camera model for determination of the horizontal view angle..... | 64 |
| Figure 5-5: Checking parallel and frontal view of the two cameras [Thomas03]..... | 67 |
| Figure 5-6: An exemplary object that meets the conditions listed above..... | 67 |
| Figure 5-7: Before calibration | 68 |
| Figure 5-8: After calibration..... | 68 |
| Figure 5-9: Perspective transformation model of the camera..... | 70 |
| Figure 5-10: Frame of reference for space co-ordinate calculation on pinhole camera model [Thomas03] | 73 |
| Figure 5-11: Relation between the two camera co-ordinate systems [Thomas03]..... | 73 |
| Figure 5-12: Frame of Reference for Space Coordinate Calculation on Linear Camera Model [Thomas03] | 77 |
| Figure 5-13: Flow chart of the 3D position calculation..... | 78 |
| Figure 5-14: a) Left image, b) Right image | 80 |
| Figure 5-15: The procedure of cutting the whole four branches | 83 |

List of Tables

| | |
|---|----|
| Table 3-1: The attribute of RGB and gray-scale images | 29 |
| Table 4-1: Precision of 2D positioning..... | 53 |
| Table 5-1: Outcome of the software calibration..... | 71 |
| Table 5-2: Actual values (Unit: mm)..... | 83 |
| Table 5-3: Calculated values by the program (Unit: mm)..... | 83 |
| Table 5-4: ErrorPrecision (Unit: %) | 84 |

Statement of Originality

Name: Ming GAO

Program: Master by Engineering Science

This work contains no material which has been previously accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in this text.

I give consent to this copy of my thesis, when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library catalogue, the Australasian Digital Theses Program (ADTP) and also through web search engines, unless permission has been granted by the university to restrict access for a period of time.

Signature:

Date:

Acknowledgements

I would like to acknowledge the effort of colleagues and staff in the School of Mechanical Engineering, the University of Adelaide, who have made it possible to do this research and helped in some way during my studies.

In particular, I wish to express sincere appreciation to Dr. Tien-Fu Lu for his supervision, guidance and encouragement in this research and preparation of this thesis.

I am also grateful for the assistance and encouragement provided by Dr. Milton Woods. Special thanks are also extended to Mr Chao Wang for his dedicating works in language assistance.

Last but not the least, I would like to thank my beloved parents who have given me a great deal of support and encouragement to conquer the problems I experienced during my academic and personal life while studying and living in Australia.

Publications

1. Ming Gao and Tien-Fu Lu, “Image Processing and Analysis for Autonomous Grapevine Pruning”, proceedings of the 2006 IEEE International Conference on Mechatronics and Automation June 25 - 28, Luoyang, China.
2. Ming Gao and Tien-Fu Lu, “Autonomous Grapevine Pruning- A Computer Vision Approach”, February 2011 submitted to Australian Journal of Agricultural Engineering

Chapter 1 Introduction

1.1 Motivation

Grapevine pruning is the final operation in the vineyard to be automated in winter. Just like other kinds of fruit trees or vines, the grapevine needs to be pruned. If not been pruned or been improperly pruned, the excess branches would derive more nutrients and cannot make full use of the nutrients, impeding the growth of other branches. In order to control the number, position, and vitality of branches, as well as the yield and the quality of the fruits, the operation of pruning is required to be conducted manually[Michael09]. However, to date, grapevine pruning has not been completely automated because of the variations of grape varieties, climates and soils which make automation difficult. With ever increasing cost of skilled hand labour in Australia motivation comes naturally to develop a more efficient and economic way in the operation of grapevine pruning.

Semi-automatic pruning machine, which is compatible with standard tractors, has been used in the pruning of vines. This kind of machine can cut the grapevine to a certain extent under the help of human operations. But it cannot perform detailed cutting which is required by optimum pruning. This raises a need for robotic machinery that could use image analysis and processing techniques to provide powerful, useful and flexible tools to do this detailed work. Image analysis was first applied in this kind of situation by [McFarlane97]. He used image analysis algorithms in the measurements relating to the winter pruning of the long wood grapevines, and tested them on 60 images, with an 80% of success rate. However, there are still problems such as inaccuracy indetermining the bottom position of the branches. That is because the visualization system can only get very little information about the structure of the vine. So far, no accurate algorithm has

successfully identified the particular branch location of the grapevines. This study proposes to investigate, research and develop appropriate algorithms using image processing and image analysis techniques to obtain the two-dimensional positions for the cutting point. Stereo vision system is also used in order to get the three-dimensional pruning positions. This process can be subdivided into three steps: First, the images captured by the two cameras on the robot are uploaded to the computer. Image processing techniques are then used in the preliminary treatment of the images in order to prepare for the image analysis. Second, image analysis algorithms are developed to locate the two-dimensional pruning positions step by step utilizing the structure characteristic of the grapevine. Finally, using the 2D positions obtained from the images captured by the two cameras from two different angle images (two-dimensional), the cutting point is stereo reconstructed from two-dimensional to three-dimensional. Then the computer will inform the robotic system to perform the pruning accurately.

1.2 Introduction of Vine Pruning

1.2.1 The Grapevine

Figure 1-1 illustrates the structure of a grapevine. The visible part consists of the main trunk, arms (also called cordons or canes) and the shoots growing from the arms.

NOTE:

This figure is included on page 3 of the print copy of the thesis held in the University of Adelaide Library.

Figure 1-1: Structure of a grapevine [Crossen97]

1.2.2 The Annual Growth Cycle

The only permanent parts of a grapevine which seldom change from season to season above the ground are the trunk and arms. Figure 1-2 illustrates the growth cycle of a grapevine. Since the main purpose of this study is to prune using robotic machinery, only the pruning part is involved. In autumn, the vine needs to be pruned. Thus, when winter comes, only a predetermined number of buds on each shoot are left on the vine. In spring, the buds burst. They become shoots bearing new leaves and bunches of grape flowers.

NOTE:

This figure is included on page 4 of the print copy of the thesis held in the University of Adelaide Library.

Figure 1-2: The annual growth cycle of a grapevine [Svensson02]

1.2.3 Reasons for Grapevine Pruning and Automated Pruning

Just like other kinds of fruit trees or vines, the grapevine needs to be pruned. Fruit trees without pruning usually have large crowns and too many branches. Thus, there would be less illumination on the leaves and the atmospheric condition of the grapevine would be bad and thus impede its normal growth. Moreover, resistance to plant diseases and insects and pests of the grapevine would get weakened. The fruit quality and output of the grapevine would also be reduced. Another important reason to operate the grapevine pruning is to avoid and lighten the detrimental effects caused by apical dominance. Apical dominance stands for the phenomenon in which the central stem of a tree or branch dominates over other stems and gains more nutrients than other stems. Apical dominance can hinder the further growth of the whole plant and lead to reduction of output. For

grapevine, it usually needs to be pruned in winter so that plant can be more vigorous in summer and have a good harvest.

There are several methods that have been used in the grapevine pruning: manual pruning and semi-automatic pruning. The first generation of pruning is the manual pruning using trimming tools such as secateurs and saw. It is still widely used in many situations especially in the developing countries that have a lower labour price. Figure 1-3 shows two pairs of common used secateurs.



Figure 1-3: Two types of secateurs

The second generation of pruning method makes the use of a series of mechanisms such as rotary saw, electric secateurs and large mechanical systems. The most crucial characteristic of these mechanisms that distinguish them from tools used in the first generation of pruning is that they are driven by electric, diesel or other sources of impersonal forces. Figure 1-4 lists several kinds of mechanisms that are used in the grapevine pruning. They can help people perform pruning and liberate them from the hard physical works. However, they still have some drawbacks. For example, the first tool shown in picture 1 of Figure 1-4 is a long-arm rotary saw that can be used in the grapevine pruning.

But if a worker is pruning using it, it will take him a very long time to finish the work. That is simply because that the vineyards are usually very large nowadays and there are millions of grapevines in them. And for the second pruning machine in Figure 1-4, it is a commercial product developed by Spagnolo Engineering. It can operate pruning quite fast and precisely. However, the accuracy of the operation depends on the experiences and capability of the driver of this machine to a large extent. It is the driver who commands the machine and makes the decision.

Another commercial product developed by Pellenc Ltd. shown in picture 3 of Figure 1-4 has a better performance because of its cordon tracking system. Although this system can make the operation progress faster, it will not help when determining which part to be trimmed. It is also the driver who makes the decision of which part to be trimmed and which part to be remained. And in practical operation, the machine can only do the pruning roughly. Many stems and branches that need to be remained get trimmed. Therefore, the pruning effects are not comparable to pruning manually by experienced workers.





2



3

Figure 1-4: The second generation of pruning machine

From the discussions above, it is easy to find out that the first generation of pruning is time-consuming and labour-intensive. It is not suitable in countries that have a high labour cost. As for the second generation of pruning machine, they are more automatic. But these machines need to be manipulated by people. And the accuracy of the pruning operation depends on the experience of the drivers and operators to a larger extent. That means the vineyardists have to find some experienced drivers or operators to do the pruning and pay a lot every time the drivers or operators are employed. And the rough pruning of these machines

will have negative influences on the next year's harvest. That is why an automatic pruning system is desirable in the grapevine pruning. This system will be completely automatic and operate the pruning with minimum intervention of people. Thus, the costs of pruning can be largely reduced. Moreover, the automatic pruning system will be equipped with computer vision system which can do the measurement and help the pruning blade to locate the pruning spot accurately through image processing and analysing.

1.2.4 Two-Bud Spur Pruning

Pruning consists of the cuttings of shoots, spurs and canes. It mainly refers to the pruning of dormant canes and summer shoots, usually carried out manually using pruning shears and thus has a high precision. In this study, the 2-bud spur pruning method will be followed to locate the pruning position. Spur pruning can be applied on any variety but this technique is ideally suited to varieties that are fruitful in the basal bud area [Michael199]. Normally a two-bud spur will produce two shoots in the spring. In most cases, the bottom shoots are retained as a spur and the top canes are removed, remaining 2 to 3 buds on last season's canes, which can be seen in Figure 1-5 and Figure 1-6.

Figure 1-5 portrays the cutting sequence. The four images show the 2-bud spur pruning operated from the first year to the fourth year. In the first year, there is only one main branch above the cordon and two subordinate branches, which are pullulating at different sides. The 2-bud spur pruning first takes place at the middle position of the main branch. After that, it is operated between the second spur and third spur on the subordination branches (Figure 1-5, left top). There will be two spurs on the subordinate branch after one year's pullulating. During the second year, (Figure 1-5, right top) the last year's subordinate branch has become a main branch. First, prune at the middle position of the second year's branch. Then, the pruning operation will be conducted between the second spur

and the third spur of the subordinate branch. There will be another two spurs on the subordinate branch after the third year's pullulating. The pruning operation is also carried out as described above, as shown in Figure 1-5 (right bottom, left bottom).

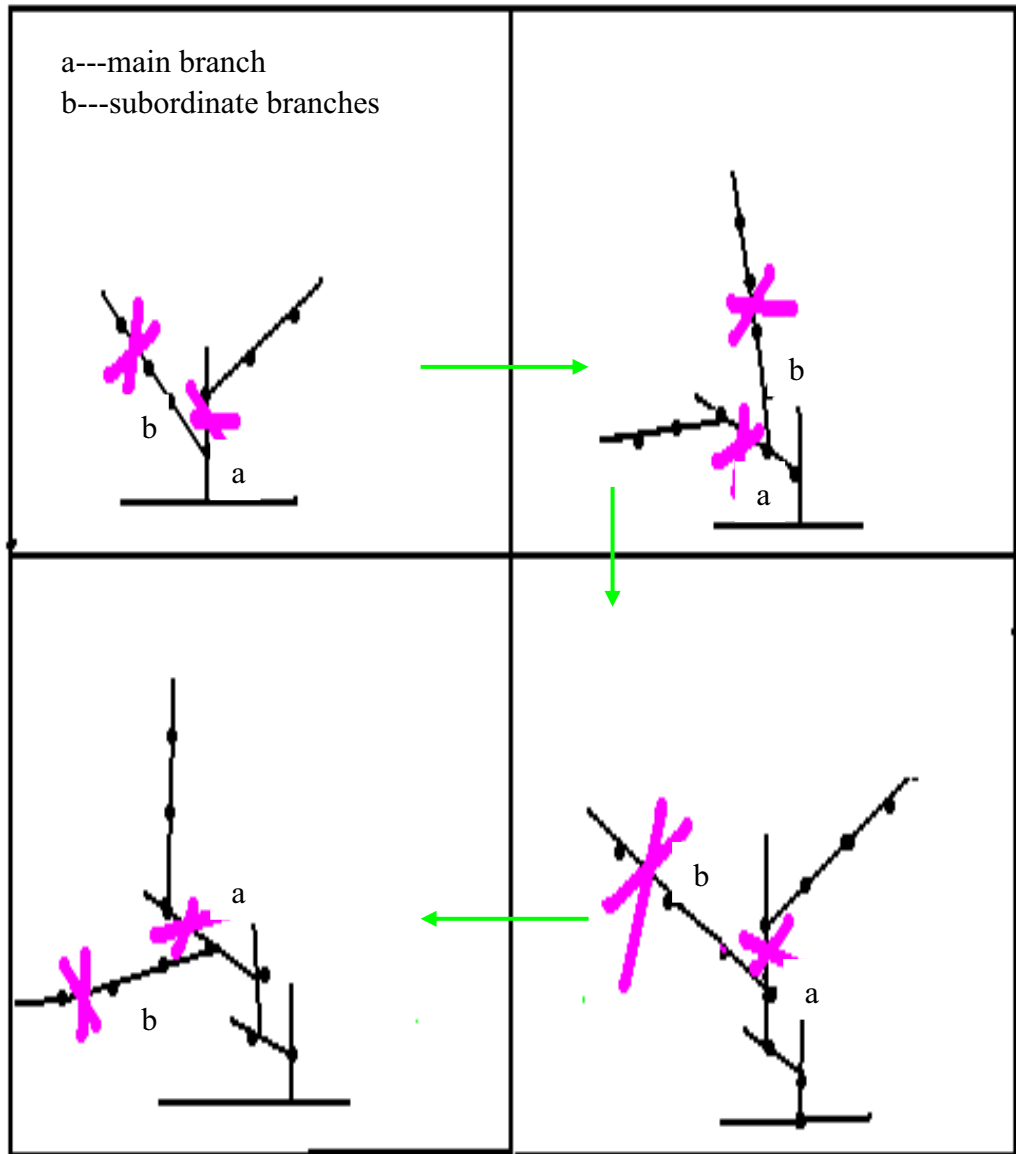


Figure 1-5: Two-bud spur pruning

NOTE:

This figure is included on page 10 of the print copy of the thesis held in the University of Adelaide Library.

Figure 1-6: Spur pruning. In the pruning, the mature shoot which is most distant from the permanent cordon is generally removed and the closest one is cut back to a 2 node spur [Ludvigsen07]

1.3 Autonomous Grape Pruning

Therefore the main challenge of this study is to identify appropriate cutting positions (the second node on the branch) on the grapevines using digital image processing techniques. The whole process can be divided into the following three steps (Figure 1-7):

- a) The images captured by two cameras on the robot are uploaded to the computer. Image processing techniques are then applied for the preliminary treatment of the images in order to prepare for the image analysis.
- b) Following the devised principles, image analysis will be performed on each of the images to identify the desired features and extract the two-dimensional pruning positions.
- c) According to the two different angle images (two-dimensional) captured by the two cameras, the cutting points are stereo reconstructed from two-dimensional to three-dimensional. Finally, the computer will inform the robotic system to perform the pruning accurately.

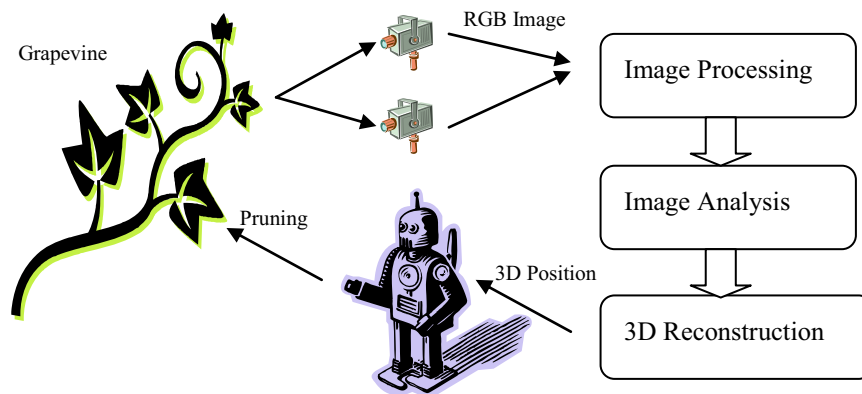


Figure 1-7: Flowchart of proposed approach

The most challenging part in this study is the localisation of the branches and the nodes because the structures of the branches and the nodes are very complex and not easy to be identified and located. In this study, according to the position of the cordon and the changing slope of the branches, a novel algorithm is

developed to locate the cutting position of the nodes. The 2-bud spur pruning method is then used to locate the pruning position. In order to locate the cutting point, 2-bud spur pruning rules will be followed to locate the canes. After that, the cutting position is obtained through the implementation of certain image processing and analysis techniques. Ten images full of canes have been analysed in this study, and an 85% success rate for pruning positions is achieved.

1.4 Outline

The outline of this thesis is briefly summarised (chapter by chapter) as the following.

A literature review which focuses on the investigations of image processing, image analysis and stereo vision systems relating to the grapevine pruning is presented in **Chapter 2**. Previous studies on autonomous grapevine pruning that help to determine the position of the cutting points in this study are introduced and discussed. Several literatures regarding image processing and analysis system are also reviewed and studied. A number of related case studies on autonomous robot application are also touched upon and reviewed. Through the review of these studies and documented methods, several uncertainties regarding the image processing, analysis and stereo vision parameters in this project is clarified. The research gap for this study is also identified.

Chapter 3 elaborates the research methodology of image processing, image analysis separately. The flow chart of the image processing used in the localisation of the cutting point in this thesis is summarised. Different methods of locating the cordon, the branch and the nodes are discussed. Comparisons are made between them, trying to find the most suitable locating method.

In **Chapter 4** a new algorithm for grapevine pruning is presented, using the 2-bud spur pruning. This algorithm is derived and based on the discussion and comparison made in **Chapter 3**. First, the images captured by the camera will be processed. Then the algorithm is employed to locate the branches, canes and nodes. An 85% success rate has been achieved in the experiment set up for locating the cutting nodes.

Chapter 5 expound the theory of stereo vision system and the experimental mock up for three-dimensional cutting locations. Stereo vision system is a computer vision application which is used in the detection of the objective position information. It has a similar principle as the human eyes do. The two cameras of this system need hardware and software calibration before the detection. After that the three dimensional position information of the object can be calculated based on the images the cameras captured.

In **Chapter 6**, a summary of main achievements, contributions and conclusions of this thesis are presented. Several possible directions that can be further studied to perfect the existing system are listed.

Chapter 2 Literature Review

Computer vision has been applied in many fields, including autonomous vehicle driving, human face recognition, industrial inspection, medical images, remote sensing, surveillance and transport. Image processing, image analysis and stereo vision are three main aspects of computer vision. Actually, they are not exclusive to each other. Rather, they interweave in many subfields and common techniques. These three techniques are distinguished explicitly in order to make the description of the proposed method more clear and fluent. Image processing is the precursor for image analysis in the following research, and stereo vision information cannot be attained without the other two steps. In this chapter the latest research status of image processing, image analysis, stereo vision and their applications in modern agriculture industry are reviewed respectively. The motivation of this study is also discussed.

2.1 Image Processing

Image processing technology is used to process the pictures which are taken from the vineyard. After image processing, a raw RGB coloured image is abstracted and transformed into some other information formats that is more convenient for further analysis. There are many techniques in the field of image processing. Among all of these techniques, image restoration and image segmentation are two crucial procedures in this research. The purpose of image restoration is to "compensate for" or "undo" defects which degrade an image; while the segmentation process divides an image into constituent segments or objects [Gonzalez04].

[McFarlane97] brought up several unique methods including the obtaining of the threshold for image segmentation, noise filtering, image compression and

skeleton extraction. However, by this means, the purpose of noise removal cannot be fully achieved. There are several disadvantages such as:

- a) The loss of information due to the merging of adjacent branches;
- b) The loss of gray level information, which was potentially useful in the resolving of the vine structure problem;
- c) The distortion of the skeleton in the region of the junctions and crossovers.

[Svensson02] adopted image segmentation techniques in the obtaining of the threshold of image, trying to minimise incorrect classification and locate the branches and spurs accurately. However, [Bulanon02] employed a segmentation algorithm in the automatic recognition of *Fuji* apples. The fruit image was enhanced using the red colour difference because results showed that the fruit had the highest red colour difference among the objects in the image. However, in the grapevine pruning research, the image of the grapevine doesn't have a well-distributed gray level pixel number because in winter the trunk and the branches mostly become brown. It is really a complex task to distinguish the grapevine from the background based on the colour difference.

According to [Stajnko03], an estimation of number and diameter of apple fruits in an orchard during the growing season can be obtained through thermal imaging analysis. A thermal camera captured images of apple trees for five times during the growth. Each time 120 images of twenty apple trees were taken late in the afternoon to achieve a temperature gradient between fruits and their background. However, thermal imagery is not suitable here in the vineyard. First of all, the price of IR camera is much higher than the normal camera. IR cameras are generally employed in many particular situations in which the normal cameras cannot work properly. Furthermore, the pictures captured by thermal imaging are not clear in the complex context. Taking Figure 2-1 for example, it is an image of

grapefruit. It is obvious that the borderlines of the branches are hard to distinguish.

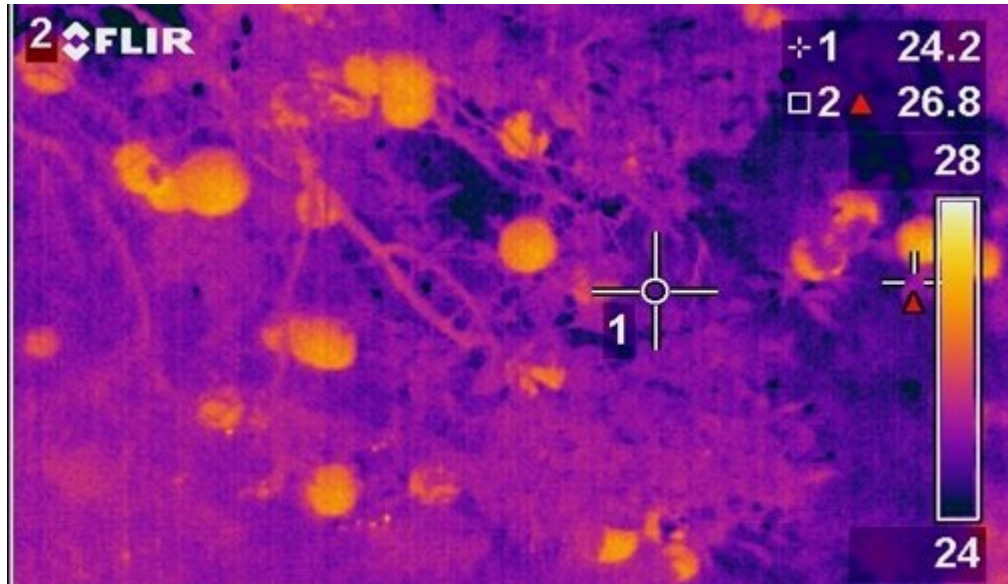


Figure 2-1: Thermal image of a grapefruit

Many image-processing techniques have been adopted in the field of agriculture such as using the differences in colour of the original image caused by many reasons, e.g. differences in temperature presented above. However, these technologies are the immediate application of the basic techniques in image processing. They only process the images independently in a certain aspect. Some of them are not suitable for our purpose. There is a need to find the most suitable technologies and combine them together, in a more organic way. Only by doing this, the system can have the images better processed and then analyses them with ease. This study chooses several methods combined together in the processing phase. This will be further described in **Chapter 3**.

2.2 Image Analysis

Image analysis techniques have been widely used in many fields to analyse the processed images and to extract information from the images. [Naugle97] stated

that image analysis had already been applied in cordon grapevine pruning. Several algorithms have been developed. Four steps are required to carry out the image analysis developed by [McFarlane97]: locating of wire and trunk, noise removal, locating of branch segment, the joining of segments into branches.

The wire can be located through Hough transform [McFarlane97]. Hough transform aims at finding and linking line segments in an image. [Rudolf 98] explained that Hough transform is a standard tool in image analysis which allows recognition of global patterns in an image space through the recognition of local patterns (ideally a point) in a transformed parameter space. It is particularly useful when the patterns one is looking for are sparsely digitized and have 'holes'. It also helps when the pictures are noisy. The image is segmented at different thresholds, and pixels belonging to horizontal line elements no more than two pixels thick and no less than three pixels long can be identified. Since the wire is the lowest part visible in the image, the peak ones corresponding to the lowest can then be selected from the transforming. The trunk can be found as follows:

- a) [McFarlane97] defined a height halfway between the bottom of the image and the wire since the trunk was nearly at the bottom of the image;
- b) A histogram was built from the image according to each horizontal coordinate x , which represented the number of pixels in a vertical line corresponding to this height.
- c) Peaks were identified in the histogram, which were more than two standard deviations above the mean. Then the axis of the trunk was found.

There is also a need to carry out the noise removal at this stage of segmentation. The noise is generally introduced by the reflections of the vines from the background and the dark part of the vine. Smoothing by pre-blurring, dilation or erosion could not help and sometimes may merge or destroy the image of branches. One efficient method of noise removal developed by [McFarlane97] is

the use of the size filter. In this process all blobs and holds less than 10 pixels were removed. The last step was to join segments together to form complete branches. Fuzzy logic criteria were used to assign values to the cubic curves, indicating the degree of possibility of the branches being physically joined. However, the result of the process showed that the bottom location of the branches can still not be identified accurately enough for pruning purpose. This is because the system has difficulties in distinguishing the complex branches.

Moreover, some studies reported sensor application of computer vision for robot guidance and demonstrated the feasibility of this approach [Sarig93]. This study analyses the image in a similar way as McFarlane did. However, the height and many other detailed settings are different from his method.

2.3 Stereo Vision

When people look at the world around them using their two eyes, their brains are able to combine the two slightly different views got from different eyes and produce three dimensional (3D) perceptions. Each eye captures its own view and the two separate images are sent on to the brain for processing. In a similar way, stereo vision has been used to obtain the three demonstrational position of an object [Iocchi98]. The word "stereo" comes from the Greek word "stereos" which means firm or solid [Cooper04]. With stereo vision one can see an object as solid as it is in a three dimensional space. The position of this object can describe with width, height and depth (or x, y and z). It is the added perception of the depth dimension that makes stereo vision so real and special.

The technique called stereo vision, similar to the mechanism of human binocular vision in conception, is to use two cameras to obtain two images simultaneously from which the distance information can be calculated. Stereo vision takes the

advantage that it can achieve the three-dimensional acquisition without energy emission or parts moving. In a particular application, the key issue in making stereo vision practical is to find the most suitable combination of algorithms that can provide reliable estimates of distances.

Currently, stereo vision has become a very important and active topic in the field of computer vision because of the rising needs of real world applications. [Scharstein01] used stereo correspondence to assess different components and design decisions made by individual stereo algorithms. Also, [Scharstein03] and [Szelilski03] described a method for acquiring high-complexity stereo image pairs with pixel-accurate correspondence information using structured light.

Moreover, stereo vision can also be used on mobile robot. [Iocchi 98] developed a multi-resolution stereoscopic vision system for a Pioneer mobile robot, by using low-cost STH-V1 Integrated Stereo Head and the SRI's stereo algorithm running on a conventional PC architecture. Since the stereo vision can extract information from the environment with ease and it is a passive sensor, it can be easily integrated with other computer vision systems (object recognition, tracking, etc.).

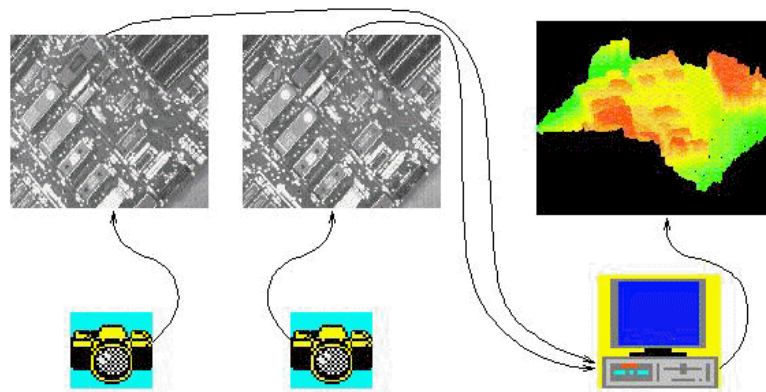


Figure 2-2: From two-dimensional to three-dimensional

As shown above (Figure 2-2), two cameras are used to obtain the three-dimensional information about the object. More cameras can be used

depending on detailed situation. In this research, there are two cameras on the robot to acquire the distance information required by the grapevine pruning.

2.4 Relevant Projects

In the field of agricultural automation, many computer vision technologies have been successfully applied. The Agriculture Department of Illionis University devepoped an automatic system to seed the fields with tomat base on machine vision and “crop-dusting system based on the difference GPS”. [Srivastava10] made “Robo Kisan” trying to help farmers by detecting the symptoms of diseases on grapevine automatically. The diagnosis process of Robo Kisan was mainly based on the image processing and analysis. [Zhao&Zheng03] of University of California-Davis presented a system that can be used in the target crop-dusting, and this system is based on the detecting and analyzing of the vision system.

According to [Niku01] and [Wiley99], image-processing techniques are used at the very beginning of the process in order to enhance, improve, or otherwise alter an image and thus to prepare it for image analysis. Image analysis techniques are then adopted to extract information from images. Finally, three-dimensional information of the object are obtained employing stereo vision system.

As mentioned above, [McFarlane97] had presented an algorithm that employs image analysis for the pruning of long wood grape vines. There are several important factors in deciding whether to remove a branch or not during pruning, including its direction, diameter, length, and the position of its bottom end relating to the wire and the trunk. There are several disadvantages in his research, such as the merging of some adjacent branches, the loss of gray level information which can probably be used in resolving the vine structure problems, and the distortion of the skeleton in the region around branch junctions and crossovers. The bottom position of the branches were not measured accurately enough for

pruning purposes, but could have been extrapolated to their correct positions if the location of the long wood had been known with higher precision.

[Svensson01] presented the possibility of using image processing as a tool to facilitate vine management, in particular shoot counting and assessment of the grapevine canopy. In his research, the images were captured under different lighting and background conditions; [Svensson01] described the methods of segmenting different parts of a vine in his paper. He also developed different approaches on how shoot counting can be done and had them evaluated. In canopy assessment, the emphasis was on the way of the estimating of canopy density. However, it was difficult to make the methods work on images captured on different occasions without calibration and it was not accurate enough to estimate the shoot counting and canopy assessment.

2.5 Aim of This Research

A lot of quantitative work and study have been done in the area of fruit tree pruning and computer vision. However, the research works reviewed in previous sections are only able to find the approximate positions of the branches in the pruning of the long wood grapevine. It is far from satisfactory for precision automatic grapevine pruning. It also makes fully automated pruning much inferior to manual pruning. Moreover, none of them could find 3D positions which are essential for automatic operation in practical pruning situations. The bottom position of the branches couldn't be found accurately. They are not useful in finding the particular positions of the nodes. There is still an urgent need for automatic grapevine pruning algorithm with higher accuracy and adaptability. It seems like that using the approach similar to [McFarlane97]'s, but with a different algorithm, might help to locate the nodes on grapevine for winter pruning. A method which combines image processing, image analysis and stereo

vision with an appropriate algorithm will make automatic grapevine pruning practical and thus save much labour work.

This research project proposes to investigate and develop methods using image processing, image analysis and stereo vision for grapevine pruning. In order to improve pruning accuracy, the three dimensional position information of the cutting point is needed to be obtained, helping the kinematic scheme of the robot operate the trimming more accurately.

Chapter 3 Obtaining 2D Cutting Positions

3.1 Introduction

To prepare for the grapevine pruning, the following steps need to be carried out:

- Image processing: to prepare the image for analysis.
- Image analysis: to located the cordon, the branches and the nodes.
- Stereo vision: to derive the three-dimensional coordinates for the robotic system to perform the pruning.

This chapter describes the methods used in this study to process and analyse the images. After the treatment of techniques described in this chapter, 2D pruning positions will be obtained. Three-dimensional methodology and experimental setup will be detailed in **Chapter 5**.

There are three main steps in order to locate the two-dimensional pruning positions. First, the images from the camera are uploaded to the computer. Then image processing techniques which can perform colour transforming is applied to process the image and prepare it. Finally, image analysis algorithm is designed to extract the pruning positions. Figure 3-1 shows the flow chart of 2D analysis phase.

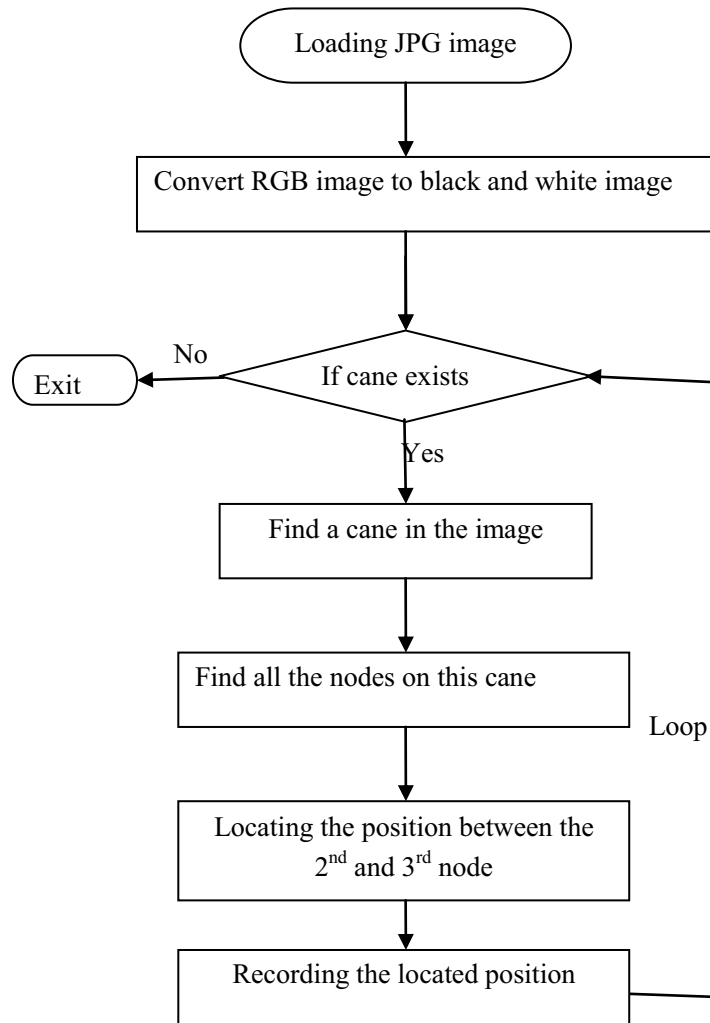


Figure 3-1: The flow chart to get the two-dimensional pruning positions

To simplify our discussion, two assumptions which are reasonable in the real world have been made for the methodology study presented in this research:

1. The background is processed and simplified in this study to make the image analysis algorithm more robust
2. Illumination level is assumed to be constant.

3.2 Image Processing

In the image processing step, the raw RGB image is converted into a gray-scale image, and then the converted image is further converted into a binary image

(black and white image) which largely reduces the computational power required for further analysis. Specifically, the image analysis algorithm could distinguish the cordons, the branches and thereafter the nodes based on the binary image.

3.2.1 RGB image to Gray-Scale Image

A CCD camera (Olympic IR300) was used with a resolution of 2048×1536 to capture 100 images and the weather was bright sunshine. The RGB colour images contained an $M \times N \times 3$ array of colour pixels, in which each colour at a specific spatial location can be represented by a certain constituent of red, green and blue components [Gonzalez04]. This is an application of an optical principle in the computer vision field. This principle states that any colour can be described by giving its red, green and blue values. Figure 3-2 shows the RGB colour space model.

NOTE:
This figure is included on page 27 of the print copy of the thesis held in the University of Adelaide Library.

Figure 3-2: The RGB colour space model [Gonzalez04]

However, a gray-scale (or gray-level) image is one in which there is only one kind of basic colour—gray and the objects in this kind of image can be distinguished by the differences of gray level. The reason differentiating this kind of images from any other sorts of colour images is that there is less information to

be provided for each pixel. In fact a “gray” colour is one in which the red, green and blue components all have equal intensities in RGB space (Figure 3-2). Thus, the only thing required is to specify a single intensity value for each pixel, rather than collecting three intensity values to specify a pixel in a full colour image.

If an image has been converted from a RGB image to a gray-scale one, it is much easier and faster to perform calculations for image processing. By the same token, in this study, a white curtain was used to separate the background while the image was captured. Figure 3-3 illustrates the image changing from a RGB one to a gray-scale image.

A white curtain is used as the background so that the foreground can be better distinguished. Figure 3-3 (a) and (b) illustrate the process of changing a RGB image to a gray-scale image and the conversion formula is as follows:

$$gray_scale = 0.299 \times red + 0.587 \times green + 0.114 \times blue$$



(a)



(b)

Figure 3-3: Image (No. 002) (a) RGB image. (b) Gray-scale image

Table 3-1 compares the attributes of these two images. Only the image class of the input and the output remains the same throughout the processing, while the size and the bytes of the image are both reduced to 1/3 of the original image.

Table 3-1: The attribute of RGB and gray-scale images

| | Size | Bytes | Class |
|-------------------------|-------------|---------|-------------|
| RGB Image | 1536×2048×3 | 9437184 | Unit8 array |
| Gray-Scale Image | 1536×2048 | 3145728 | Unit8 array |

3.2.2 Analysing the Image

Once the image has been changed into gray-scale, the range of colour value for one pixel is [0, 255]. By using the Matlab *imview* command in the image viewer window (Figure 3-4, a), a more accurate pixel located in the image can be viewed in the pixel region window (Figure 3-4, b).

It is noticeable that in Figure 3-4 (b), the gray level pixel values are all around 150. This indicates that the gray level value of the pixel of the background is

around 150. Repeat the operation on a different pixel, Figure 3-5 (a) and (b) show that the gray level pixel value of the branch is less than 100.

In this study, image histogram is adopted to show the distribution of intensities of an intensity image. Figure 3-6 illustrates the gray-scale image after using the Matlab function *imhist*.



(a)



(b)

Figure 3-4: (a) Image viewer window. (b) Pixel region window

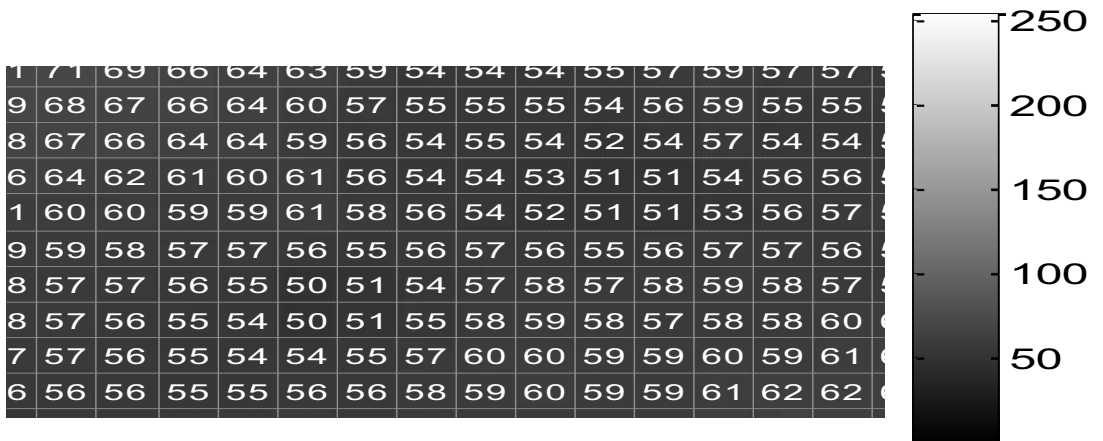


Figure 3-5: (a) Image viewer window. (b) Pixel region window

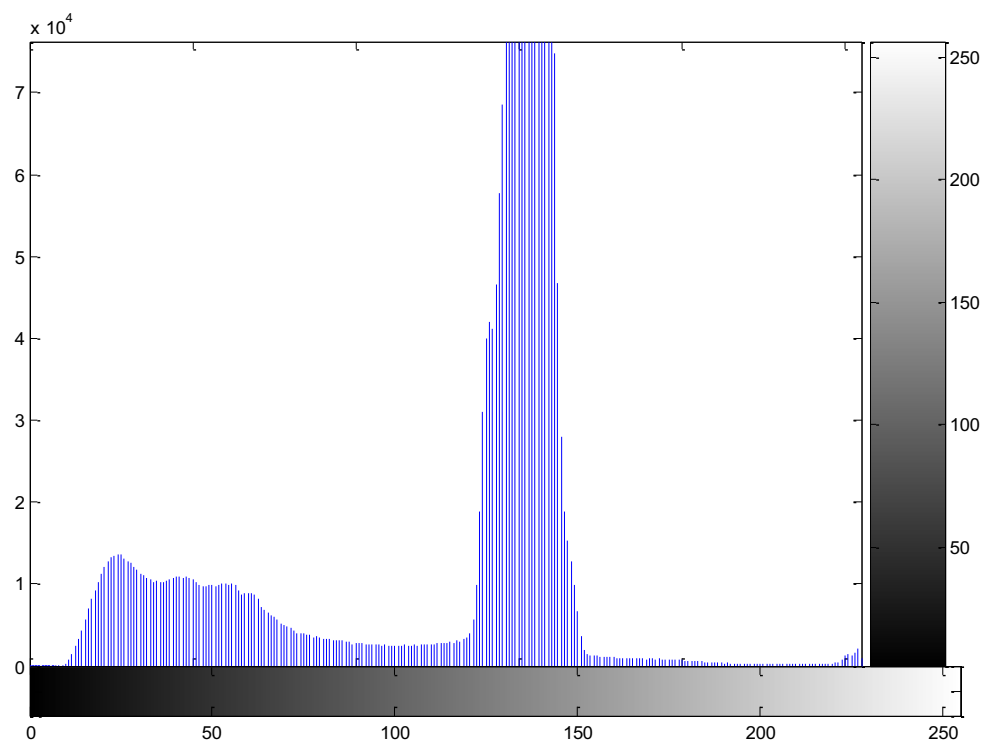


Figure 3-6: the histogram of the RGB image in Figure 3-3 (a)

Figure 3-6 illustrates the histogram of the RGB images in Figure 3-3 (b). From Figure 3-5 and 3-6 it is easy to find out that the pixels of which the gray level values are less than 100 ($p < 100$) contained almost all the cordons and branches. The pixels which have gray level values higher than 100 ($p > 100$) are the background (Figure 3-4 and 3-6). It has been tested on all the 100 images. In order to separate the cordons and branches from the background, there is a need to take the advantages of the binary image which will be introduced in the next section.

3.2.3 Gray-Scale Image into Binary Image

The binary (black and white) image is a special kind of intensity image [Thompson95]. The gray level values of pixels in a binary images can only be 0 (black) or 1 (white). That means there are only two colours in the image: the colour used for the objects (foreground colour) and the colour used for the rest parts of the image (background colour). Binary image, also known as bi-level or two-level, is often used in the image processing. It can also be considered as the result of some operations. Here, in this study, it can be seen as the result of the thresholding operation. Image thresholding, just as the name implies, stands for a

process in which a threshold value is chosen and set as the demarcation point. There are several steps in a common thresholding operation as listed below.

- a). Choose a threshold value T depending on the situation;
- b). Segment the image into the objects (foreground) and background according to the threshold value. Two sets of pixels are created in this process:

$$\begin{cases} P_1 = \{f(m,n) : f(m,n) > T\} \text{ (foreground pixels)} \\ P_2 = \{f(m,n) : f(m,n) \leq T\} \text{ (background pixels)} \end{cases}$$

Where $f(m,n)$ is the pixel value of the point located in column m and row n .

- c). Assign a new value to all the pixels in set P_1 . In a similar way, assign a new value to all the pixels in set P_2 .

There are many algorithms such as iterative algorithm that can be used in the thresholding process to make the operation more accurate. The thresholding steps above change correspondingly when they get used. Figure 3-7 illustrates the thresholding operation through the comparison between the original image and the image after the thresholding operation.



Original image



Image after thresholding operation

Figure 3-7: Comparison of the original image and image after Thresholding operation

In this study, the common usage of thresholding is chosen. Here the thresholding value is chosen to be 100, based on the outcome of Figure 3-6. In the images of this study, the pixels which have values less than 100 (contain 100) in the former images are reset to be 0 (black) in the binary images, whereas pixels with values higher than 100 in the former images are reset to be 1 (white) in the binary images. It is much easier to distinguish the cordons and branches from the background in the binary images. Consequently, it is much more convenient to find the cordons for the next step. Figure 3-8 illustrates the black and white image which only contains 0 and 1.



Figure 3-8: Black and white image

3.3 Image Analysis

Image analysis techniques have been used in many fields to analyse the processed images and extract information from the images. In this study, to obtain further information, it is required to determine the positions of the cordons, the canes and the nodes. The application of image analysis in this study mainly focuses on these three parts. Figure 3-9 illustrates the flow chart of three main steps in image analysis.

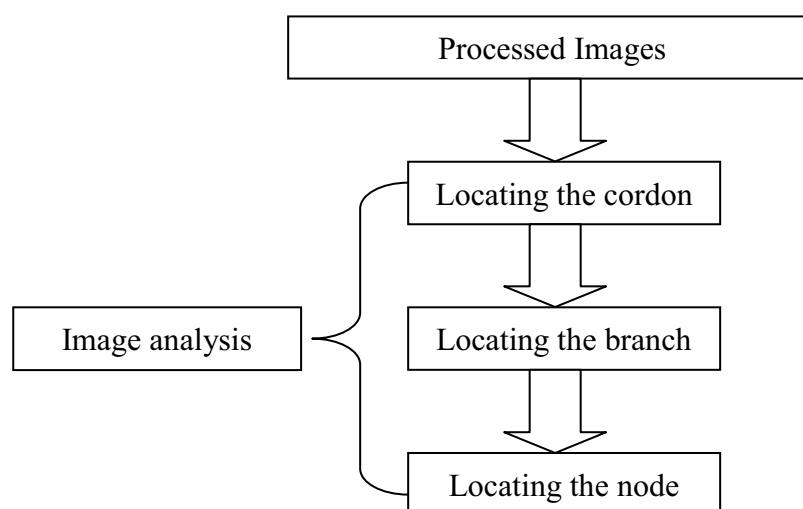


Figure 3- 9: The flow chart of the three steps in image analysis

3.3.1 Locating the Cordon

A major challenge of this research is to find the positions of the nodes on the canes. This is difficult because of the changing of the underlying slopes of the canes, i.e. some of the nodes are considerably smoother on the cane than others. According to the 2-bud spur pruning method, the pruning point depends on the positions of the nodes. If the node cannot be identified by the program as a result of the changing of the underlying cane slope or the system finds more nodes than that actually existed due to the noises, the cutting point could not be accurately determined. This would lead to incorrect pruning which will affect the vine growth in the following year. Moreover, the growth processes of different grapevine varieties vary.

Locating the cordon is the first step in image analysis phase. This is decided by the vegetative cycle of the grapevine. In the growth process of a grapevine, the branches and nodes appear only after the outgrowth of the cordon. And the cordon is the only perennial part on a grapevine among these three parts. What is more, the cordon is the lowest part in the whole image if the background is not taken into consideration. Once the position of the cordon is determined, it can be

seen as the base line to find the branch, which is above it. The area under this line would not be considered and it will save time for the calculating of the computer.

3.3.1.1 Procedures

According to Figure 3-8, the branches and the cordon have already been set to be black (pixel value= 0) and the background is all white (pixel value = 1). Since the original images are captured by a CCD camera with a resolution of 2048×1536 , which means that the $Y_{\max} = 1536$ and the $X_{\max} = 2048$. It can also be seen that, in the direction paralleled to the X-axis, the cordon contains more black pixels. On the basis of this discovery, in Figure 3-10, check the number of the pixels of which the values are 0 in X direction which has a certain Y value, counting from the bottom line in the image. Record the Y position while the counting number is greater than $X_{\max} / 2 = 1024$, and reset all the pixels values of this line to be 1(white) when the counting number is less than $X_{\max} / 2 = 1024$. The flowchart of this program is as follows:

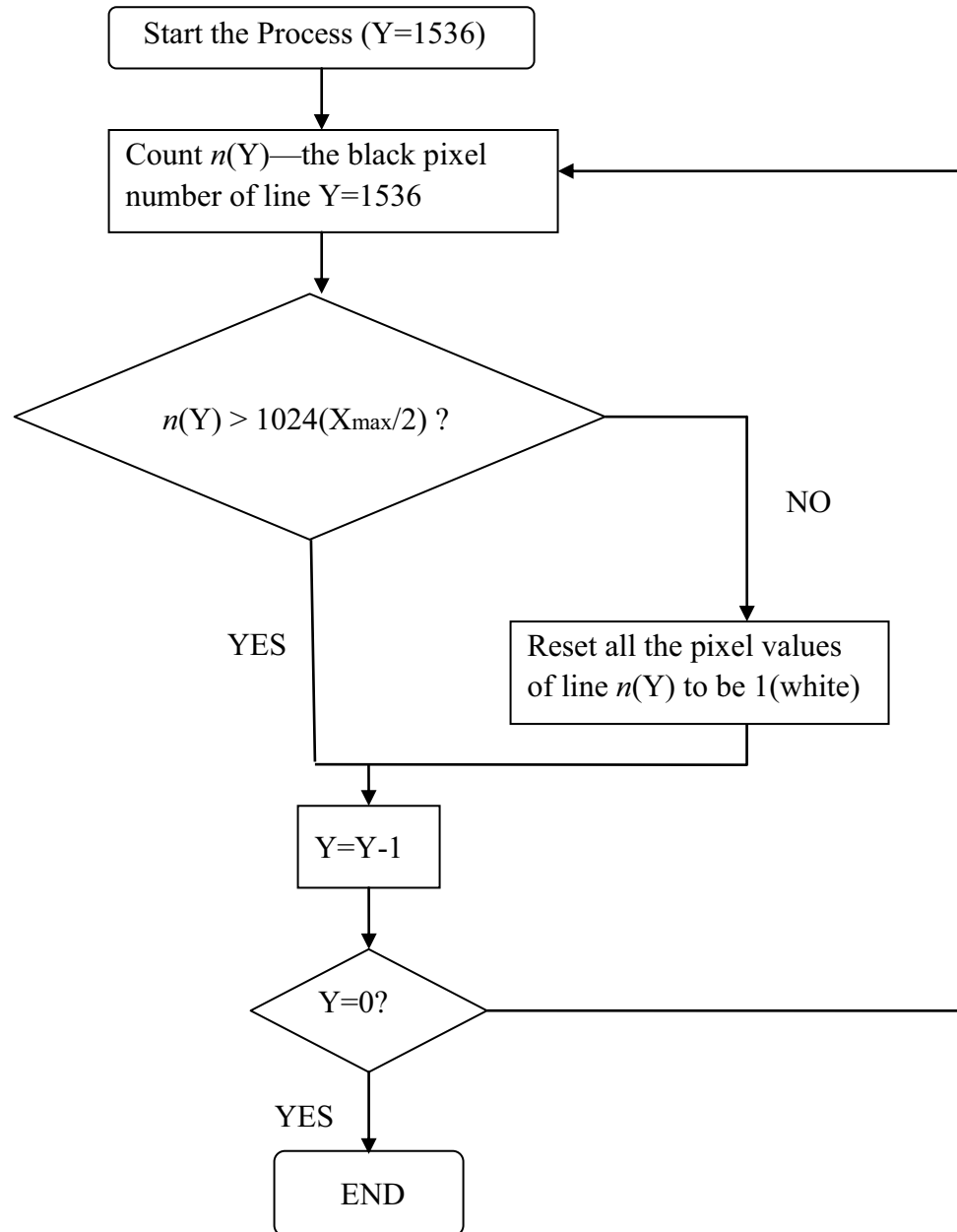


Figure 3-10: Flowchart of the cordon determining

The bottom position Y can be found in the procedures finally. Figure 3-9 displays the image of the cordon, and the corresponding coordinate Y is 1250. This result can be seen in Figure 3-11.

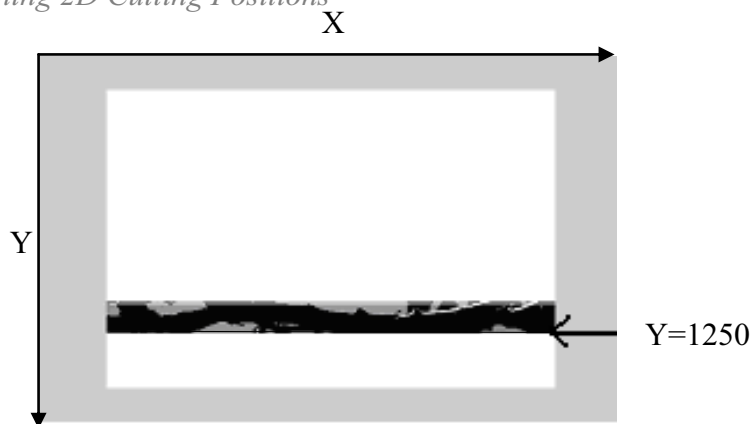


Figure 3-11: The Cordon

3.3.1.2 Results

At the end of the procedure described in Section 3.3.1.1, for the image shown in Figure 3-11 the cordon can be seen clearly, the lowest position of Y can also be known ($Y=1250$). Figure 3-12 shows another example using the same method. In this case, the position of the cordon is located to be 582 (in Y direction).

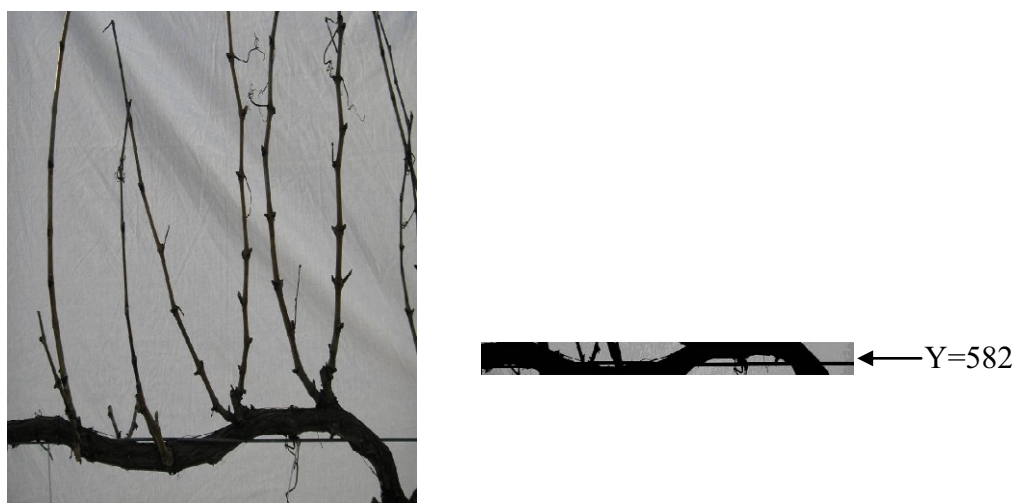
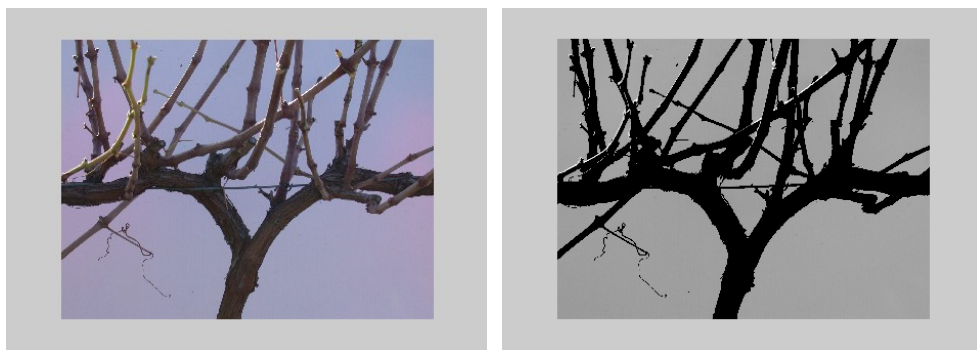


Figure 3-12: Locating the cordon

3.3.1.3 Discussion

The cordons have been found successfully on those two images shown above. However, the particular conditions of the images vary a lot in different situations. Another problem arises: there may be two parts of a cordon in the image after the processing of the image. Figure 3-13(c) illustrates an image that has such an issue.

This is because there are too many branches near the cordon. In 3.3.1.1, as have been explained, on the same X direction, only those of which the black pixel value is more than half of the X_{max} can be shown on the images and considered as part of the cordon. However, if there are too many branches near the cordon just as shown in Figure 3-13 (b), it is possible that the program would take the pixel lines which have dense branches on the X directions as part of the cordon after the program manipulation. That is because that the gray level values of the branches and cordon are both 0 (black) and once the proportion of black pixels is higher than 50% of the X_{max} , they will be recognized as the cordon by the program. On the same X direction line, if the threshold of the black pixels values proportion is changed to be more than 70% (lines of which the black pixels values proportions are higher than 70% of the X_{max} will be considered as a part of the cordon), such issue won't happen again (Figure 3-13, Y=877). It has been tested on another 10 images, and a 100% success rate is achieved. After the accurate locating of the cordon, the next step is to locate the branches.



(a)

(b)



Figure 3-13: Image (No.061)

3.3.2 Locating the Branches

In order to find the coordinates of the cutting point for the measurement in the image, it is necessary to locate the branches first. In this part, three different methods are described and discussed in order to locate the branches.

3.3.2.1 Method One-Segmentation the Branch

The shape and the gray level pixel values are used to locate the branches in Method One. The procedure is as follows:

- a). Locating one part of the branch;
- b). Segmenting the points into a branch;
- c). Finding the whole branch based on the part of the branch that has been located.

The cordon is found and considered as the starting line above which the branches are located. Figure 3-14 shows the result of the cordon found in the last step and Figure 3-15 shows the result of locating parts of the branches growing on the cordon.

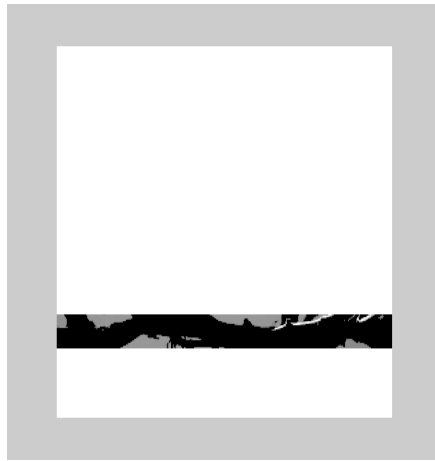


Figure 3-14: The cordon



Figure 3-15: Parts of the branch

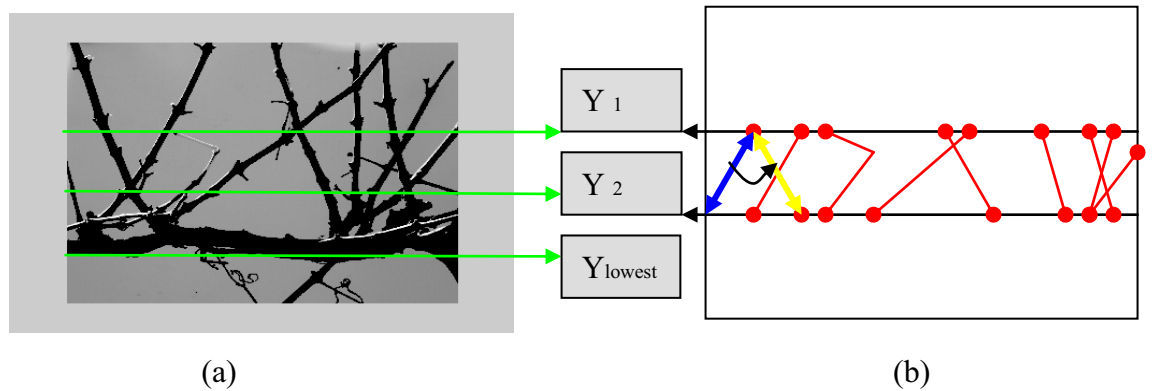


Figure 3-16: (a) Locating the branches; (b) the distance from Y_1 to Y_2

After finding the cordon, the lowest position of the cordon is known, denoted as Y_{lowest} (Figure 3-16, a). Marking two lines randomly above Y_{lowest} , i.e. trisect the distance from $Y=0$ to Y_{lowest} . There will be three lines: Y_1 , Y_2 and the Y_{lowest} (Figure 3-16, a). Parts of the branches can be shown between lines $Y=Y_1$ and $Y=Y_2$ (Figure 3-16, b). There will be some points on lines $Y=Y_1$ and $Y=Y_2$. Take the first point on line Y_1 , and try to find the corresponded point on line Y_2 . As shown in Figure 3-16 (b), there are two points on line Y_2 in this area. Connect the points and it turns out to be that the first line (blue) does not exist because there are no pixels on this line which have similar gray level values in Figure 3-16 (a). The yellow line is part of the branches in this area because there are pixels on this line which have similar gray level values in Figure 3-16 (a).

As described above, the image can be segmented and the whole branch can be determined by giving the small points and their connecting lines. Then the position of the branches will be obtained in any possible areas above the Y_{lowest} .

3.3.2.2 Method Two-Moving the Branch

In this method, the robot is assumed to have two “arms”. One is fitted with a high-speed rotating blades trimmer for pruning, and the other one can part the branches. This parting movement requires the arm to hold one branch and move it to a certain distance, so that the camera can take two photos: one before the moving of the branch and another one after the branch is moved. This allows the robot to isolate an individual branch for pruning and identify the position on it.

Figure 3-17 illustrates the flow chart followed by the robot to process the images.

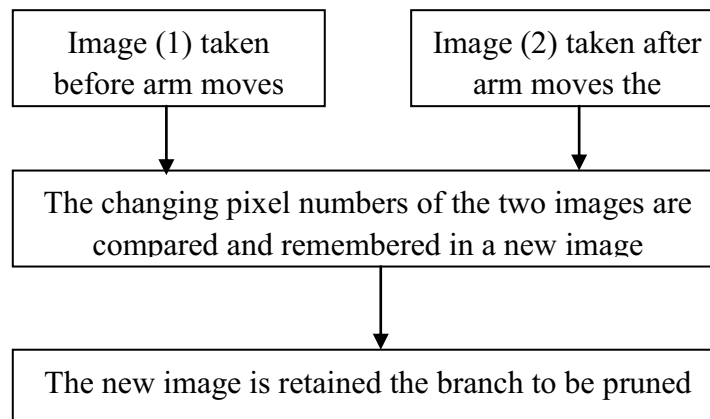
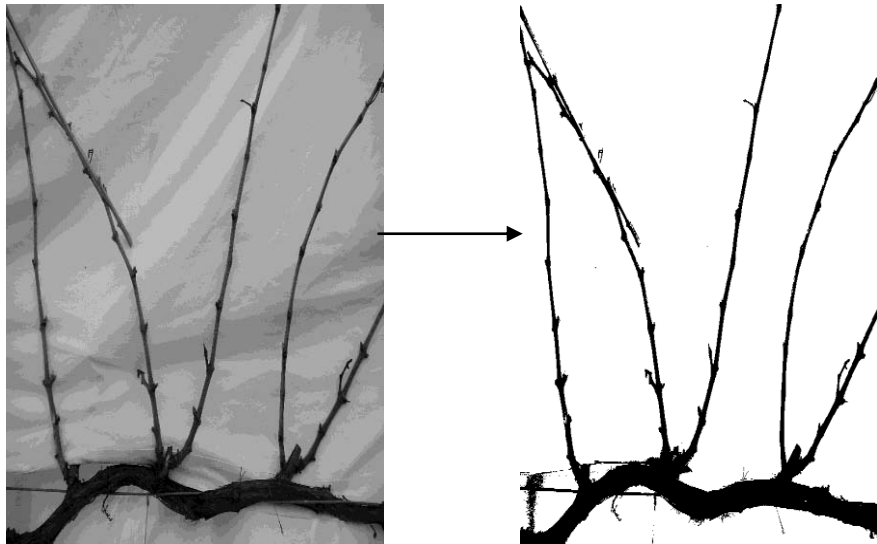


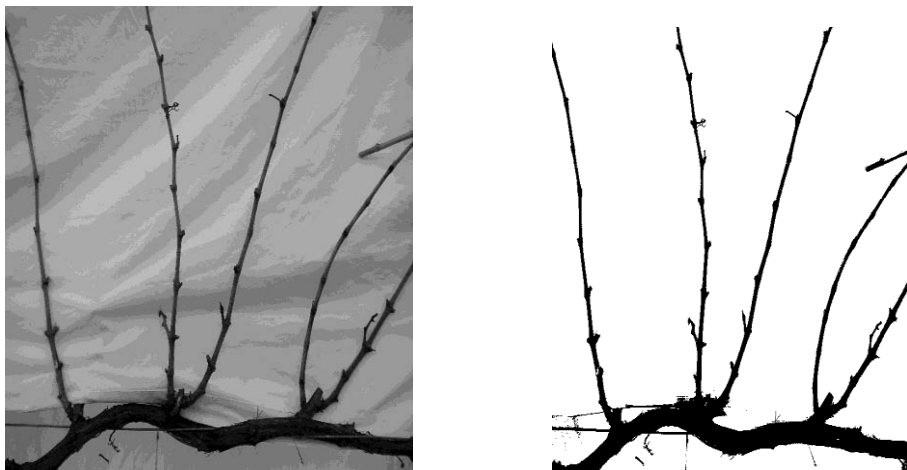
Figure 3-17: The flow chart of locating the branch

Following this, here is an example of the execution of these steps on an image of a vine. First, image (1) was uploaded and changed from RGB to black and white (Figure 3-18, a). After that, image (2) was uploaded and it was changed from RGB format to black and white (Figure 3-18, b). Finally, the gray level values of pixels in these two images were compared. At the same position, some of the

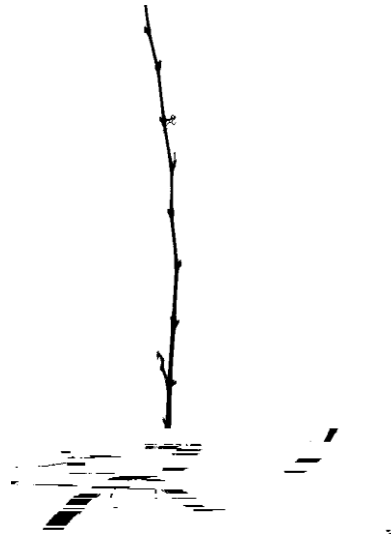
gray level values of the pixels have changed because the branch has been moved. On the basis of this process, a new image that only contained the pixels which have changed can be produced (Figure 3-18, c).



(a) Image 1: before the moving of the branch, form RGB to black and white



(b) Image 2: after the moving of the branch, form RGB to black white



(c) A new image contained only pixels which have changed

Figure 3-18: Three steps to locate the branch

3.3.2.3 Method Three-Slope Changing

Figure 3-19 (b) illustrates the canes which have a pixel value of 0 (black) in the image. Suppose that the top left is the original point of the whole image. In order to locate all the canes in the image including the shorter ones, the program can start searching from $1/3$ of the image height (Figure 3-19, the red line) of the image. Note that $1/3$ of image height is an empirically constant that is chosen after trial and error. In our experiment settings, if the starting point is higher than $1/3$, some branches may be missed which located under this starting line. Meanwhile, there's no need to scan from a position lower than $1/3$. Other than computation efficiency considerations, starting from a lower position increases the false recall possibilities.

The program searches from the starting point until it finds the first pixel of which the value changes from 1 to 0 in the horizontal direction. This pixel is considered to be the left side boundary point of the first cane in the image. The width of the cane could be regarded as a small line in the same horizontal direction. It is also

necessary to find the other side of the cane. As all the canes' gray level pixels has been set to be black, it is easy for the program to find the other side of the cane where the gray level values of pixels change from 0 to 1 (0 is black, 1 is white).

After the boundary points of the cane get located, the middle point of the cane in the same horizontal direction could then be identified (Y_{middle}, X_{middle}) . The program moves vertically down to the next position $(Y_{middle} + 1, X_{middle})$ and searches again trying to find the boundary of the cane. It uploads the middle point of the cane and moves vertically down again to find next point. Eventually, the whole shape of the cane can be determined, and all the middle points of the cane will be recorded.

As the connection of middle points of the cane are not likely in a smooth line, it is important to modify the changing slope of the cane after having scanned for a short distance, Figure 3-20 illustrates the slope changing of the cane. After having scanned a short distance of the mid-line of the cane, Linear Least-Squares Error Fit, given by formula (3-1), is used to simulate the new line, where 'a' is the slope and 'b' is the position that crossing the Y coordinate, X and Y are the x, y coordinates of the middle points of the cane. Using the slopes of this new line, the process moves vertically down from the middle point pixel and continues the locating of the boundary of the cane. Once the boundaries of the cane get identified, the middle points of the cane are obtained.

$$Y = a \times X + b, \quad (3.1)$$

$$X = X_i \text{ for } i = 1, 2, 3, \dots, n,$$

$$Y = Y_i \text{ for } i = 1, 2, 3, \dots, n.$$

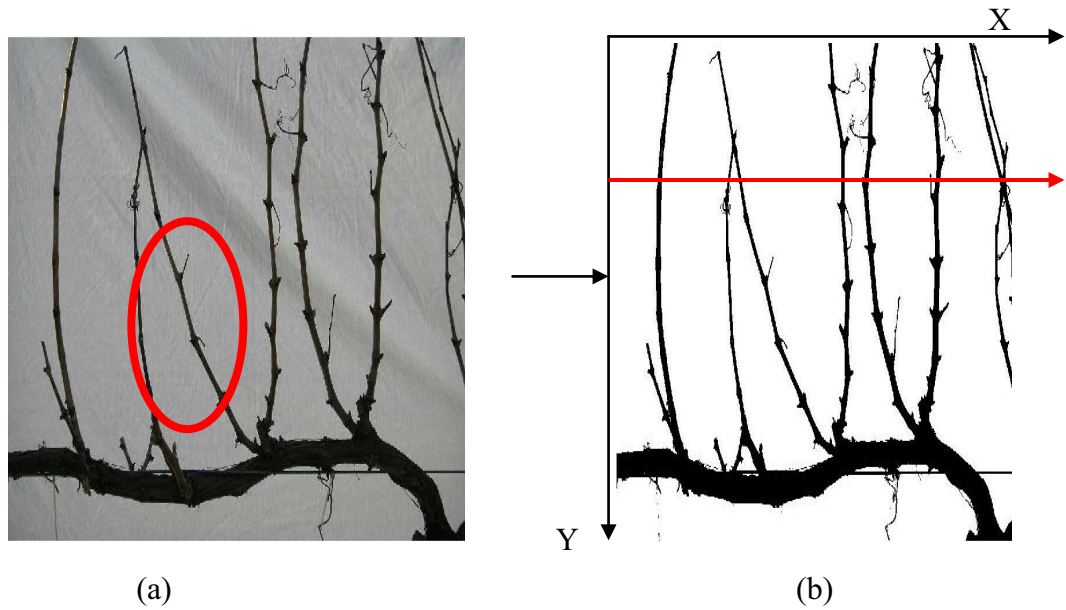


Figure 3-19: From RGB to black and white

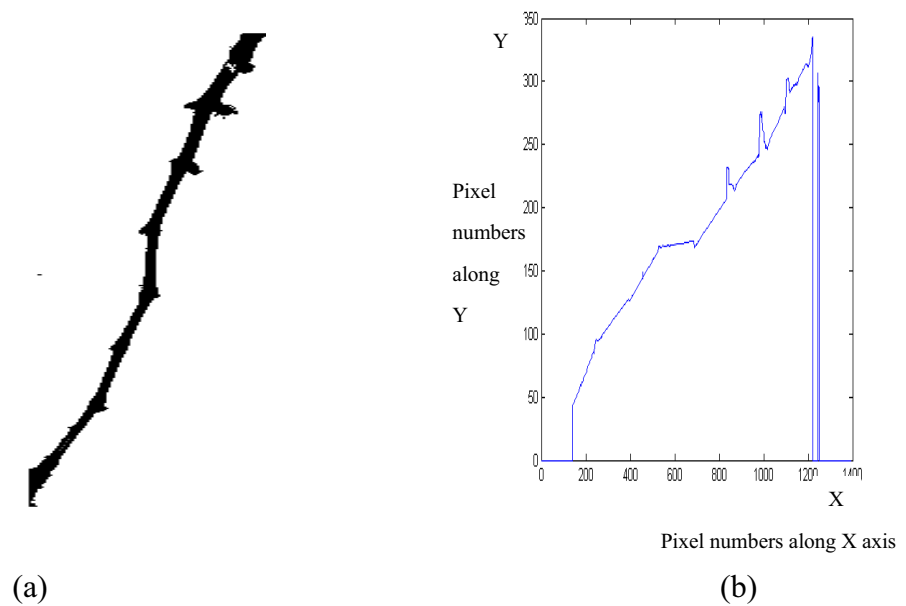


Figure 3-20: Part of the cane and its slope, a) part of a cane, b) slope changing of this cane

3.3.2.4 Results and Discussions

Three methods have been discussed in this chapter. Comparing all these three methods, the last one is the best option to locate the branch. That is because:

For method one, first it segments the image and try to recollect the points and connect it into a new line. Then it needs to determine whether the line is a real branch or not. However, this method has to depend on the position of the cordon.

If two or more cordons were found, this method could fail. Furthermore, during the connection of the parts of branches, there are quite a few choices in $Y=Y_2$ line for a part in $Y=Y_1$ to connect to. The number of line sections a part should be in is therefore hard to determine since it depends on the situation. Finally, the determination of the branch needs to recheck the original binary image and try to find out whether there are black pixels along the line connected in the later image. However, pixels of other branches may be mistaken as the expected one. Therefore, the program would make an incorrect determination.

According to method two, a two-arm robot is required, with one arm holding the branch and the other one pruning. In the spirit of this method, it is quite easy to locate the branches. However, many pictures need to be uploaded to the computer for being processed and therefore a delay before operation is unavoidable before the positions are finally located. Also, when the robot arm grips one branch and has it moved to a certain position, it affects other branches by producing some displacement. If this method is adopted, the pruning procedure will be:

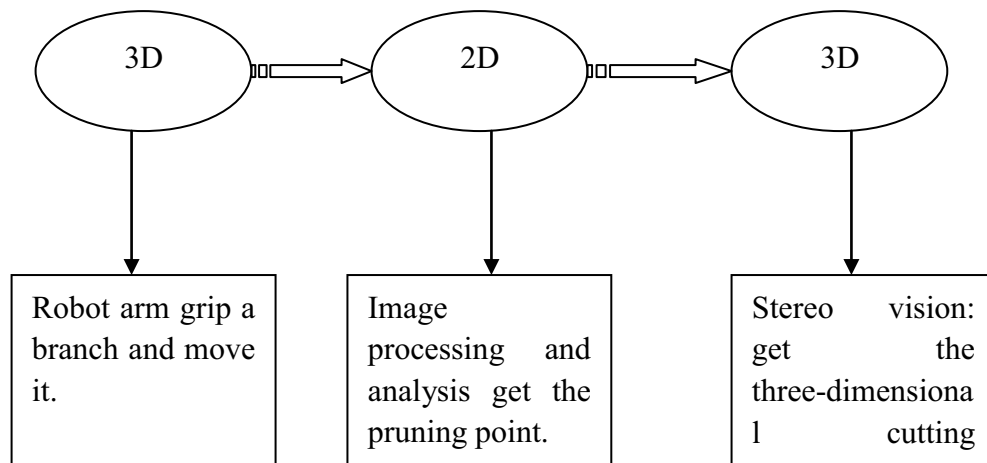


Figure 3-21: The workflow for method two

Compared with method one and three, method two has one more step to do—constructing the three-dimensional branch position grip and moving it. To construct the three-dimensional branch position, the system has to capture images and have them processed and analysed. It is obvious that the program will have to

locate the branch again. That means the program should employ another method to locate the branch. The stereo vision will get applied here as well. That needs a lot of unnecessary work which could be avoided through other methods.

In method three, the location of the branches is more accurate when looking through the entire image from the starting point. Besides, the whole program is operating on only one image, there is no need to check the original image or process the image repeatedly which will save much time for the arithmetic unit of the system. Also, it got the information of the slope changing of the branches which is very useful and convenient for the system to locate the nodes.

3.3.3 Locating the Nodes

Once the branches have been located, the program will focus on the locating of the node. According to the 2-bud spur pruning method, the pruning point is in the middle region between the second and the third nodes back from the bottom of the cane. According to the study of all the images captured, it is obvious that the node is much wider on the same horizontal direction than any other parts of the cane. This means that the method used to find peaks according to the slope changing could be used to locate the nodes, where the slope changing is more distinct than any other parts of the cane. Several steps need to be done in the locating process:

- a). Starting the program from the side near the cordon;
- b). Checking the slope and slope changes of the branch;
- c). Recording the peak slope change rates and the corresponding coordinates;
- d). Match the coordinates and the corresponding nodes so that the position information of the nodes can be determined.

3.4 Conclusion

The flow chart of the image processing has been thoroughly described at the beginning of this chapter. Thresholding operation should be conducted in order to convert gray scale image into binary image. It makes the information acquisition of the image much faster. Image processing and analysis have also been introduced in this chapter. The program checks the pixels which have a value of 0 (black pixel) along each horizontal line paralleled to the X-axis in the image and reset all the pixels' value on this line to 0 to locate the cordon. Three different methods on how to locate the branches have been expatiated. In Section 3.3, method three is finally chosen for locating of the branch. According to the sloping changing of the cane, the cutting point can be located as well.

Chapter 4 Experiments on Obtaining Two-Dimensional Pruning Positions

4.1 Introduction

In this chapter the method employed in obtaining the two-dimensional pruning position is experimented on a real world dataset. The algorithm used in calculating the two-dimensional cutting position is based on the method described in **Chapter 3**. Two digital cameras were used to capture the images at the National Wine Centre, South Australia. There is a research vineyard containing several kinds of grapevine varieties. The distance between the camera and the vine is 1000mm horizontally. Several studies have investigated the problems caused by the relative movement of the camera and the vine [McFarlane97], [Norman91], [Throop91], but in this research the camera was held stationarily with a tripod. A white curtain was used to separate the whole vine from its disorderly background.

4.2 Procedures

Figure 4-1 is the original RGB image taken from the National Wine Centre with a white curtain as the background. There are six canes in this image. As described in **Chapter 3**, each time the program searches from the starting point (top left corner) to locate a cane. However, in order to locate the shorter cane that has been pointed out in a red ellipse in Figure 4-1, the search progress starts from a lower position. After a comparing of all the 10 images (see Appendix B), the search starting point is chosen to be from the 1/3 of the height of the image.

After the image has been uploaded, according to Section 3.2.2, the colour of the image is transformed from RGB into black and white. Figure 4-2 shows the black and white image after transformation.

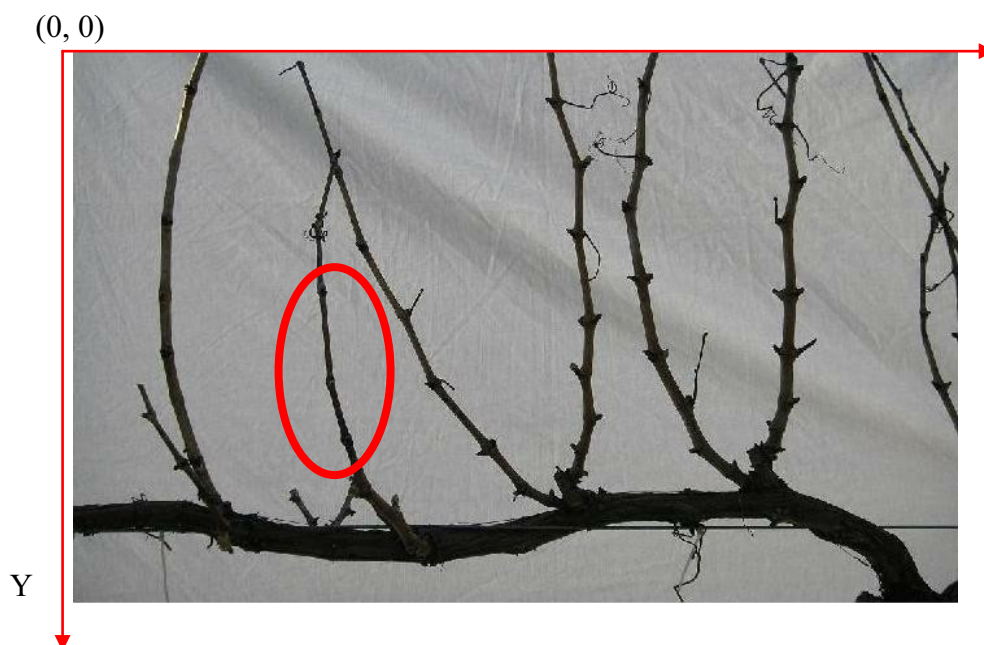


Figure 4-1: Original RGB image



Figure 4-2: From RGB to black and white image

Figure 4-3 shows the results of the 2-bud spur pruning using the algorithm. Figure 4-3(a) illustrates one cutting point that has been located according to the

2-bud spur pruning method. After that, using stereo vision technology which will be introduced in **Chapter 5**, the robot gets three-dimensional positions for the cutting points. Then the robot arm can cut off unnecessary branches on the basis of the position information of the cane (Figure 4-3(b)). From Figure 4-3(c) to Figure 4-3(l), the whole pruning process is explained.

Through all the 10 images (see Appendix B), the success rate in finding the canes is 98%, and a success rate of 90% is achieved in locating the nodes. Besides, the success rate in finding the pruning position is 85%, which is listed in Table 4-1.

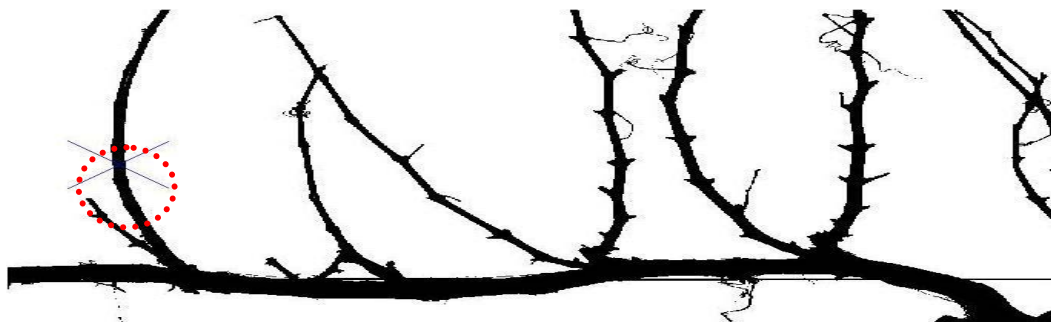
Success rate (Precision) here stands for a ratio which obtained the following formula:

$$\text{success rate} = \frac{\text{The number of successful confirmatory experiments}}{\text{The number of all the confirmatory experiments}}$$

Success here means that the outcomes of the locating are consistent with the actual pruning positions.

Table 4-1: Success Rate (Precision) of 2D positioning

| | Success Rate (10 images, Appendix B) |
|-----------------------|---|
| Canes | 98% |
| Nodes | 90% |
| Cutting Points | 85% |



(a): Locating the first node



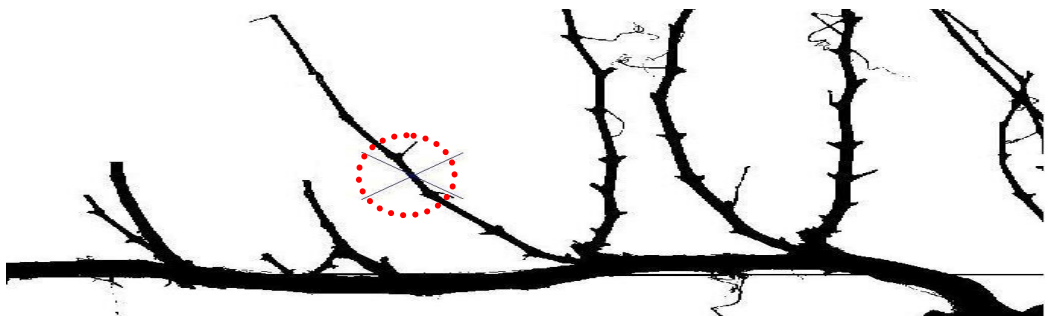
(b): Cutting the first cane, prepare to find the next node



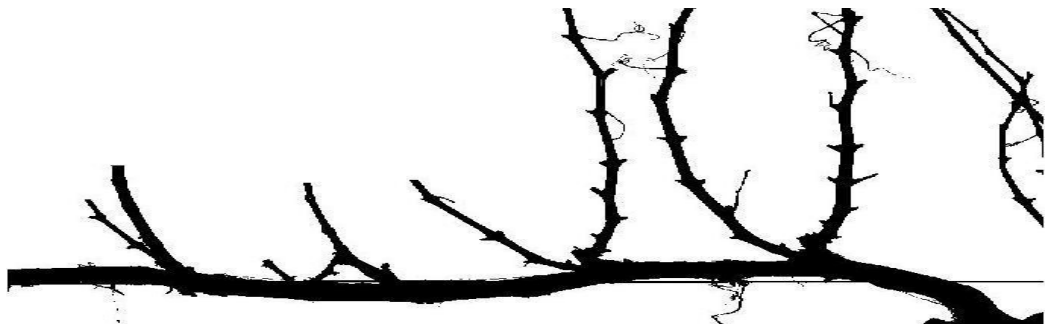
(c): Locating the second node



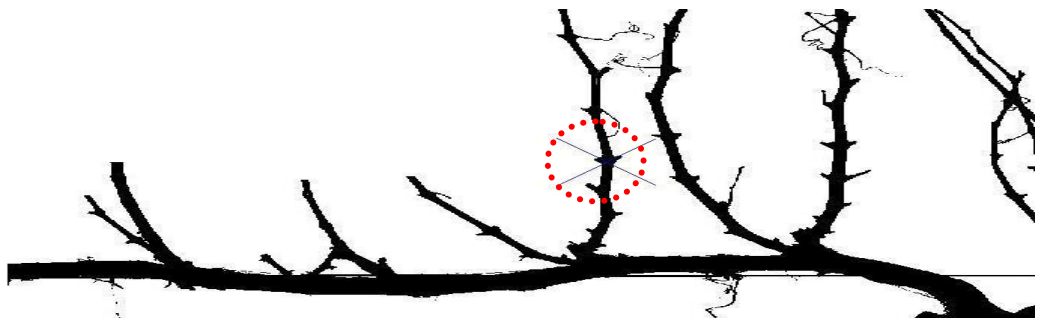
(d): Cutting the second cane, prepare to find the next node



(e): Locating the third node



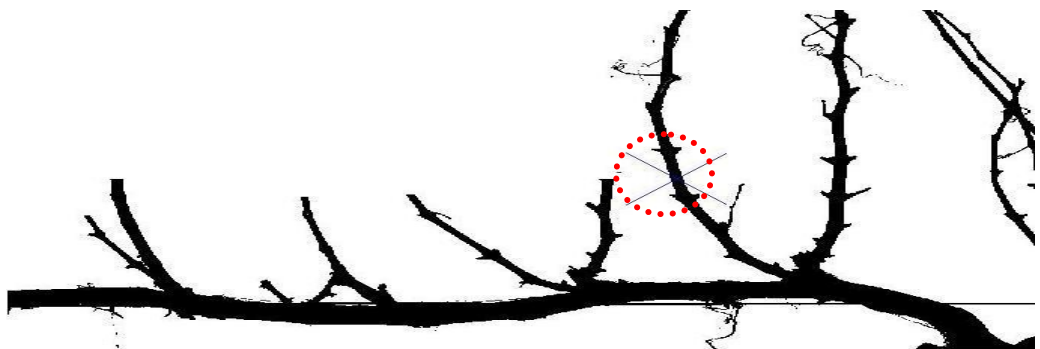
(f): Cutting the third cane, prepare to find the next node



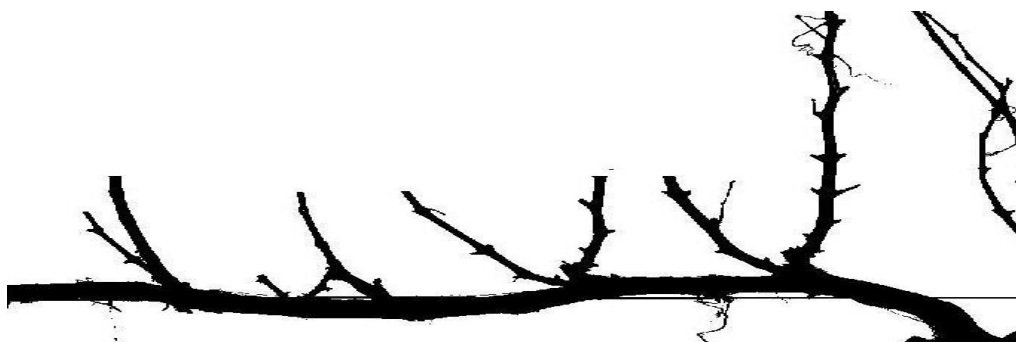
(g): Locating the forth node



(h): Cutting the forth cane, prepare to find the next node



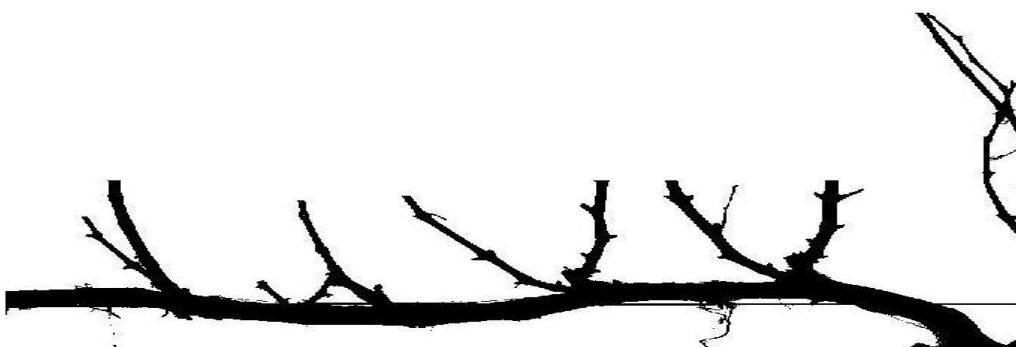
(i): Locating the fifth node



(j): Cutting the fifth cane, prepare to find the next node



(k): Locating the sixth node



(l): Cutting the sixth cane, prepare to find the next node

Figure 4-3: The iterative process of locating the nodes

4.3 Discussion

A major challenge of this research is to find the positions of the nodes on the canes. This is difficult because of the changing of the underlying slopes of the canes, i.e. some of the nodes are considerably smoother on the cane than others. According to the 2-bud spur pruning method, the pruning point depends on the positions of the nodes. If the node cannot be identified by the program as a result of the changing of the underlying cane slope or the system finds more nodes than the actually existed ones due to noise, the cutting point could not be accurately determined. This would lead to incorrect pruning which would affect the vine growth in the following year. For example, if the robot cut between the third and fourth nodes on a branch because of the noises, more branches will come up in the next year. They will derive more nutrients and cannot make full use of the nutrients, impeding the growth of other branches. Moreover, the growth processes of different grapevine varieties vary. Thus, a database containing information about the distances between nodes on canes of particular varieties could be used to reset the algorithm according to grapevine variety. This method has already been implemented in this study.

The precision measure is used to evaluate the efficacy of the pruning algorithm. It is defined as the percentage of positions which perfectly match those found by experts in all the positions found by the algorithm. The overall precision of cutting position is 85% which means 85% of the cutting positions match with experts' choices. The rest 15% of positions found by the algorithm is not very satisfactory (within 10mm to the standard cutting position) but won't do harm to the growth of the grapevine in the next year.

In this study, a white curtain needs to be placed behind the vines during daytime. Also, the curtain could be replaced by a light source which would be more convenient and can also be used in the evening when the images are captured.

Other sources could also be used to provide uniform illumination on the entire vine [McFarlane97]. Alternatively, three-dimensional stereo vision techniques can be applied to locate and remove objects further than a pre-assigned distance away from the foreground canes. This will be introduced in **Chapter 5**.

4.4 Concluding Remarks

On the basis of the last chapter, this chapter details the grapevine pruning algorithm in winter and experiment it on real world images. The algorithm is developed to locate the cutting positions, following the 2-bud spur pruning method. In this experiment, an example of process for a single image (taken from the Nation Wine Centre in Adelaide, South Australia) has been given and an overall success rate of 85% was achieved. Moreover, in order to perform the pruning in the real world, the robot needs three-dimensional information on the positions of the nodes and branches, rather than the two-dimensional positions. This will get explained in **Chapter 5: Stereo Vision and Experiments for 3D Pruning Positions**.

Chapter 5 Stereo Vision and Experiments for 3D Pruning Positions

Human's two eyes take slightly different views on the same object, thus producing the depth information for the object. Human beings are able to estimate the distance by comparing the two different pictures perceived by their two eyes and combining them to one namely the cyclopean view [Cooper04]. This can be imitated by using two cameras placed next to each other and capturing images of the same object with them simultaneously. The system takes advantage of the differences between the two images and combines the two images into a single one from which the depths of all points in the scene are inferred. In this research, according to the two images (two-dimensional) taken from different angles, the stereo vision is adopted to calculate the three-dimensional position of the cutting point on each cane of the grapevines. This chapter describes the hardware and software calibration steps required by the "pinhole" camera model; space coordinates calculations and all the experiments for stereo vision.

5.1 Introduction of the Projection Models

5.1.1 Basic information of the Pinhole Camera Model

The stereo vision camera placement and the image processing approximation can be illustrated by a model called pinhole camera model as shown in the following Figures, which is often called perspective projection. Pinhole camera is very popular in image processing and stereo vision calculation [Iocchi99], [Bouguet99]. That is mainly because that the optical structure of pinhole camera is quite similar to the structure of human eyes and it can be simulated by computer in practices. Besides, the price of pinhole camera is acceptable.

NOTE:
This figure is included on page 62 of the print copy of
the thesis held in the University of Adelaide Library.

Figure 5-1: Example for a perspective projection [Thomas03]

NOTE:
This figure is included on page 62 of the print copy of
the thesis held in the University of Adelaide Library.

Figure 5-2: Geometry of the pinhole camera model [Thomas03]

The Figures above show the geometry of perspective projection, which can be described by the following equations (5.1 and 5.2) which transforms 3D information in the real world to coordinates in an image plane.

If f is denoted as the focal length, sz is the distance between the camera and the object, ix and sx are the image size and the object size, respectively. We have:

$$\frac{ix}{f} = \frac{sx}{sz} \quad (5.1)$$

$$\frac{iy}{f} = \frac{sy}{sz} \quad (5.2)$$

Affected by the non-linear projection, only a small area around the image centre is accurate and the size of this area depends on the focal length. Distortions in the image occur due to the nonlinearity: a straight line in the real world may appear as a curve line in the image world [Thomas03]. To set up a stereo vision, two cameras have to be calibrated to minimize the disparity or misalignment between the images captured by them, followed by further software calibration.

5.1.2 Basic Information of the Linear Camera Model

The linear camera model is based on the assumption that there is no distortion in the camera image. In reality this is only true for object at infinity or a focal length of which the value is approximated to zero [Thomas03]. The geometry of the linear camera model is shown in Figure 5-3.

NOTE:
This figure is included on page 63 of the print copy of
the thesis held in the University of Adelaide Library.

Figure 5-3: Geometry of the linear camera model [Thomas03]

Figure 5-3 shows the simple relation between the two distance s_{-ob} and i_{-ob} by

$$\frac{s_{-ob}}{len} = \frac{i_{-ob}}{length} \quad (5.3)$$

To make the model more useful for the subsequent calculations in stereo vision, another simplification is made: the CCD sensor is regarded as a single point in the model because of its small size. This simplification can be seen in Figure 5-4. This assumption gives rise to the equation for the camera view angle as shown in equation (5.4):

$$\tan \alpha = \frac{len}{2 \cdot dist} \quad (5.4)$$

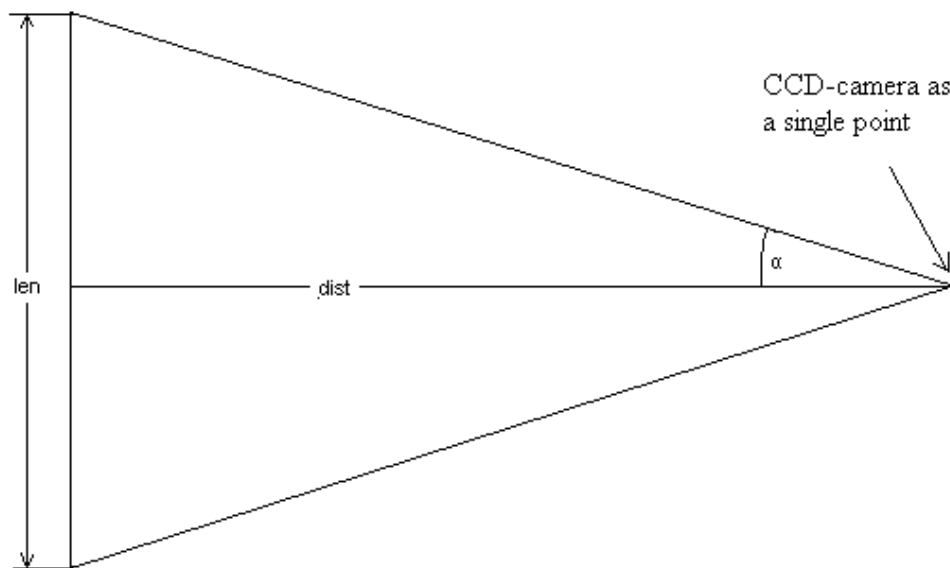


Figure 5-4: Simplified linear camera model for determination of the horizontal view angle

The non-linear distortion can be reduced by using the Matlab calibration toolbox. This method will be introduced later in this chapter.

5.2 Camera Calibration

As mentioned above, there are two assumptions in Section 5.1 to fulfil the requirements of the Linear Camera Model:

- a) There is no non-linear distortion in camera image;
- b) The CCD sensor is regarded as a point.

According to Assumption (a), to make the Linear Camera Model work, one must

adopt some measures to eliminate the non-linear projection and the consequent distortion. The non-linear projection has a lot to do with the relative position of the two cameras. When the cameras are placed in the wrong direction or their CCD sensor panels are unparallel, non-linear projection would happen. That is why there is a need to do some hardware calibration. Moreover, since the hardware calibration can only eliminate part of the effects of non-linear projection, more adjustments must be made as a compensation of the non-linear distortion. During the software calibration, the distortion parameter can be got. The most important reason to conduct software calibration is that the intrinsic parameters and the extrinsic parameters can both be recovered or determined in this process. It provides more useful information for the calculation of the three dimensional coordinates of the object.

5.2.1 Hardware Calibration

Stereo vision depends on the position and the orientation of the two cameras to a great extent. A small distortion of the camera orientation angle would cause great error of the object position. The longer the distance between the camera and the object is the more errors the camera misalignment would be resulted in [Thomas03]. The hardware is calibrated to ensure that both cameras are parallel and share the same frontal plane. Thus, since their CCD sensor panels are on the plane, there is less probability for the non-linear projection to take place. There are two possible ways to improve the measuring accuracy. Method one is to make an appropriate adjustment of the two cameras, using a reference object. Another method is to determine the orientation against each other during a more complex camera software calibration, rather than focusing on the adjustment of the camera. Because of the presence of a reference substance in method one, the calibration of each camera would not have influence on the other one. Thus it makes the calculation of the space co-ordinates easier and quite a few error recalculations are required. Without a reference substance, method 2 might end up with a deflection angle between the cameras' sensor panel and the expected plane. That

is why the first method is chosen in this research.

To calibrate the cameras, first of all, an object is placed in the centreline of the two cameras. The idea is to ensure that both cameras are parallel and frontal while the distance to the object is varying. Adjust the cameras or the orientation angles of them depends on the misalignment to the centreline of the camera. Fine-tuning is required until the results of calibration get consistent. The whole process can be better explained by the steps listed below.

- a) Place an object in the centreline of the two cameras horizontally;
- b) Open the images captured by the two cameras and check whether the object image in the two images share the same height and are identically paralleled to the X-axis. If the practical situation does not meet the two requirements, adjust the camera;
- c) Have the cameras fixed preliminary;
- d) Make some fine-tunings and make sure that the practical situation meets the two requirements;
- e) Move the object along the centreline and repeat step (a)(b)(c)(d) until the two requirements are identically met at all times.

Note: the object used here should be coloured in its frontal plane vertically and horizontally. Thus it helps the operator to make a comparison between the two images captured. An ideal object can be a cuboid which has its frontal plane spray-painted vertically and horizontally and it should be placed in the centreline of the two cameras and had its frontal plane perpendicular to the centreline. Figure 5-6 shows an object which meets the conditions listed above.

NOTE:
This figure is included on page 67 of the print copy of
the thesis held in the University of Adelaide Library.

Figure 5-5: Checking parallel and frontal view of the two cameras [Thomas03]

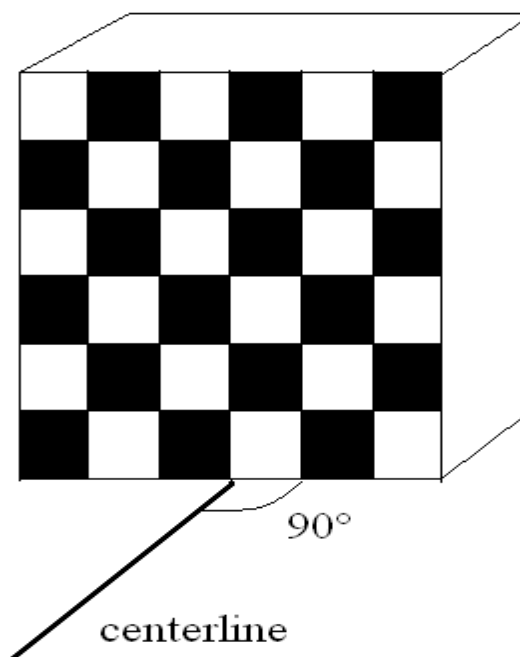


Figure 5-6: An exemplary object that meets the conditions listed above

Figure 5-7 shows the image before calibration. There are two images: the left image captured by the left camera and the right one captured by the right camera. It can be seen that the blue horizontal lines of the two images are not lying on the same level in Figure 5-7. Thus there was a need to adjust the camera, making sure that both blue horizontal lines are lying on the same level.

The steps listed above were followed in the adjustments. Some coarse adjustments were firstly conducted on the two cameras. When the blue horizontal lines of the two images became very close and parallel to the base line, the positions and space angles of the two cameras are fixed preliminarily. Following this, some fine-tunings were made to make sure the blue horizontal lines of the two images on the same height and parallel to the base line identically. Then the object was moved along the centerline and the steps were followed repeatedly until the outcome kept consistent. Figure 5-8 shows the result of the hardware calibration.

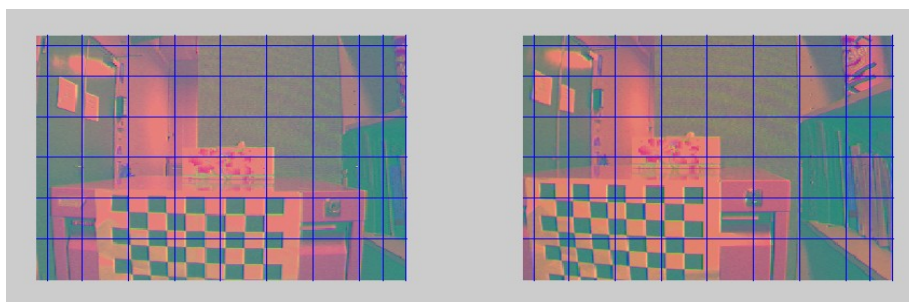


Figure 5-7: Before Calibration

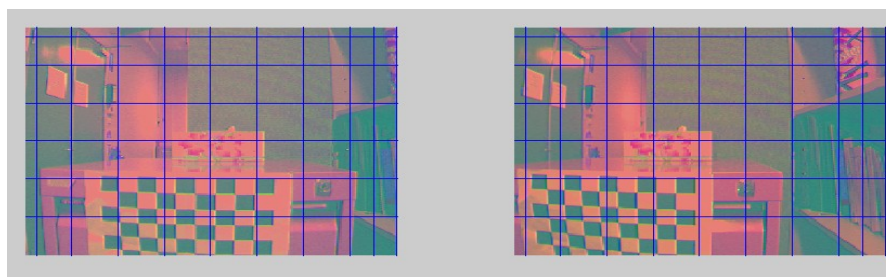


Figure 5-8: After calibration

5.2.2 Software Calibration

Software calibration is required to determine the parameters for the calculation of the space coordinates of the observed object. The parameters can be divided into two categories: the intrinsic parameter and the extrinsic parameter [Thomas03]. The intrinsic parameters includes: focal length, principal point, skew and

distortion. The extrinsic parameters consist of the rotation vector and the translation vector.

Tsai(87) developed a calibration method based on RAC (Radial alignment constraint). Radial alignment constraint means that the two vectors $O_I P_I$ and PP_C in Figure 5-9 share the same direction i.e.

$$O_I P_I \times PP_C = 0 \quad (5.5)$$

Figure 5-9 is the perspective transformation model of the camera. Oc stands for the CCD point of camera and P is the object point. There are four coordinate systems in this picture: object coordinate system $O_w X_w Y_w Z_w$, camera coordinate system $O_c X_c Y_c Z_c$, image plane coordinate system $O_I X_I Y_I$ and the Computer image coordinate system OUV .

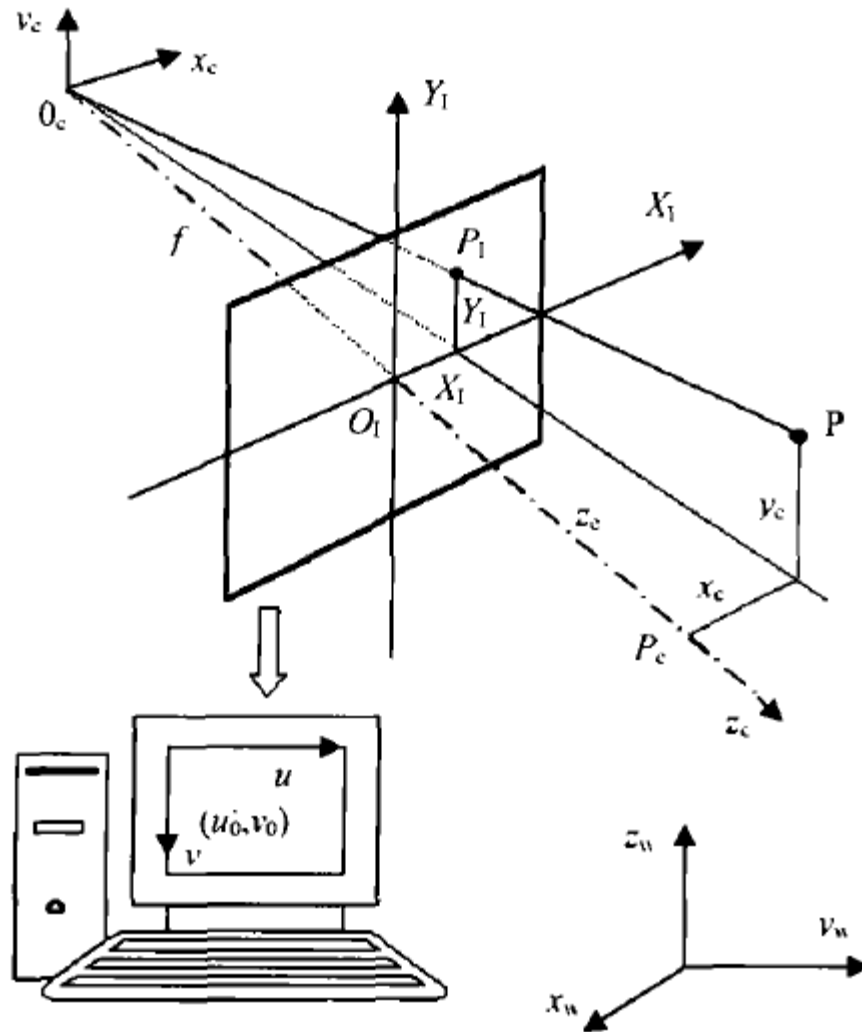


Figure 5-9: Perspective transformation model of the camera

This method first solves the over determined linear equations employing the least squares method, obtaining the extrinsic parameters. For example, if image point (X_d, Y_d) exists in the image plane coordinate system $O_I X_I Y_I$, an equation can then be established between the image coordinate (X_d, Y_d) and its corresponding point in computer image coordinate system $O_U V$. With the presence of the radial alignment constraint equation in formula 5.4 and the equation of the image point mentioned above, extrinsic parameters such as the focal length and principal point can then be got. After that, another equation can be founded based on the mathematic relation between $P(X_w, Y_w, Z_w)$ and

$P(u, v)$ of Figure 5-9. Then, the intrinsic parameters can be substituted into this equation and the extrinsic parameters can be got.

On the basis of the RAC method presented by Tsai, [Bouguet03] developed a Matlab toolbox which can produce the parameters needed by analysing some basic image information. This toolbox can also be used for corner extraction, stereo image rectification and 3D stereo triangulation. This makes the whole work of the software calibration much easier. <https://www.vision.caltech.edu/bouguetj/calib-doc/index.html> (Jean-Yves 2010).

In this study, the author used the toolbox directly in the software calibration. Several steps were followed in this process:

- a) Four images of the cheeseboard pattern were taken and uploaded to the computer;
- b) The Matlab command `calib_gui` was conducted for both the left and right cameras;
- c) The parameters were finally got, following the procedure of the toolbox.

The parameter got in the software calibration operation discussed above can be seen in Table 5-1.

Table 5-1: Outcome of the software calibration

| | | |
|--|-----------------|--|
| Intrinsic Parameters of left camera | Focal Length | $fc_left = [477.75 \quad 680.36] \quad ; ; [627.10 \quad 454.88]$ |
| | Principal point | $cc_left = [494.90 \quad 174.93] \quad ; ; [532.41 \quad 133.07]$ |
| | Skew | $alpha_c_left = [0.00] \quad ; ; [0.00] \quad \Rightarrow$ angle of pixel axes = 90.00 ; ; 0.00 degrees |
| | Distortion | $kc_left = [-0.98 \quad 0.24 \quad -0.012 \quad -0.33 \quad 0.00]$; ; |

| | | |
|---|-----------------------|--|
| | | [3.26 4.02 0.075 1.13 0.00] |
| Intrinsic parameters of left camera | Focal Length | fc_right = [445.22 653.00] ;i[623.98 480.67] |
| | Principal point | cc_right = [578.44 182.49] ;i[623.87 144.50] |
| | Skew | alpha_c_right = [0.00 ;i [0.00] => angle of pixel axes = 90.00 ;i0.00 degrees |
| | Distortion | om = [-0.0374 -0.0960 -0.0246] ;i [0.2832 1.4997 0.0988] |
| Extrinsic parameters (position of right camera & left camera) | Rotation vector | fc_right = [445.22 653.00] ;i[623.98 480.67] |
| | Translation vector | T = [-111.68 -40.23 -51.16] +/- [81.05 24.60 94.05] |

In this study, only the focus length and the translation vector will be used:

$$f = \text{focus length};$$

translation vector:

$$[192.73 = 111.68 + 81.05, -15.63 = 40.23 + 24.60, 145.21 = 51.16 + 94.05]$$

Once the objective focal length has been changed by turning the end of the lens to get a clearer picture, the value of the focal length will change correspondingly and the determination has to be done again.

5.3 Space Coordinates Calculations

5.3.1 Coordinate Calculation in the Pinhole Camera Model

Humans are able to estimate distances of objects using stereo vision. It is also possible to determine an objects' space co-ordinates (SCO) based on two camera

picture that have been set up in a stereo vision system. The relation between the image and real world has already been given by equation (5.1) and (5.2).

The base point of the frame of reference for the objects' SCO is set right in the left camera as shown in Figure 5-10. The sz -axis is perpendicular to the image plane. Moreover, the image plane (ix, iy) is parallel to the space plane that can be determined by axis sx and axis sy (Figure 5-10).

NOTE:
This figure is included on page 73 of the print copy of
the thesis held in the University of Adelaide Library.

**Figure 5-10: Frame of reference for space co-ordinate calculation on pinhole camera model
[Thomas03]**

NOTE:
This figure is included on page 73 of the print copy of
the thesis held in the University of Adelaide Library.

Figure 5-11: Relation between the two camera co-ordinate systems [Thomas03]

The co-ordinates can then be described in the frame of reference (SCO) (sx , sy , sz). According to Figure5-11, the relationship between the two cameras and the object point is as follows:

$$q_R = q_L + v \quad (5.6)$$

On the basis of [Bouguet99], a difference can be made between the vertical and the horizontal focal lengths of each camera. The following four equations are developed from the pinhole camera projection formula 5.1 and 5.2 (Section 5.1).

Right camera:

$$ix_R = f_{Rx} \cdot \frac{q_R(1)}{q_R(3)} \quad (5.7)$$

$$iy_R = f_{Ry} \cdot \frac{q_R(2)}{q_R(3)} \quad (5.8)$$

Left camera:

$$ix_L = f_{Lx} \cdot \frac{q_L(1)}{q_L(3)} \quad \Leftrightarrow \quad q_L(1) = \frac{ix_L}{f_{Lx}} \cdot q_L(3) \quad (5.9)$$

$$iy_L = f_{Ly} \cdot \frac{q_L(2)}{q_L(3)} \quad \Leftrightarrow \quad q_L(2) = \frac{iy_L}{f_{Ly}} \cdot q_L(3) \quad (5.10)$$

(5.6) in (5.7):

$$ix_R = f_{Rx} \cdot \frac{q_{-L}(1) - v(1)}{q_{-L}(3) - v(3)} \quad (5.11)$$

(5.6) in (5.8):

$$iy_R = f_{Ry} \cdot \frac{q_{-R}(2) - v(2)}{q_{-R}(3) - v(3)} \quad (5.12)$$

(5.9) in (5.11):

$$ix_R = \frac{f_{Rx}}{f_{Lx}} \cdot \frac{ix_L \cdot q_{-L}(3) - f_{Lx} \cdot v(1)}{q_{-L}(3) - v(3)} \Leftrightarrow q_{-L}(3) = \frac{f_{Rx} \cdot v(1) - ix_R \cdot v(3)}{\left(\frac{f_{Rx}}{f_{Lx}} \cdot ix_L\right) - ix_R}$$

So the space co-ordinates can be determined by:

$$sx = q_{-L}(1) = \frac{q_{-L}(3) \cdot ix_L}{f_{Lx}} \quad (5.13)$$

$$sy = q_{-L}(2) = \frac{q_{-L}(3) \cdot iy_L}{f_{Ly}} \quad (5.14)$$

$$sz = q_L(3) \quad (5.15)$$

Where:

$$q_L(3) = \frac{f_{Rx} \cdot v(1) - ix_R \cdot v(3)}{\left(\frac{f_{Rx}}{f_{Lx}} \cdot ix_L \right) - ix_R} \quad (5.16)$$

Assume that there is an ideal calibrated camera system and the two camera co-ordinates systems can only be represented by a vector in sx-direction (Figure 5-5), following conclusions can be made for q_L(3):

$$q_L(3) = \frac{f_{Rx} \cdot v(1)}{\left(\frac{f_{Rx}}{f_{Lx}} \cdot ix_L \right) - ix_R} \quad (5.17)$$

5.3.2 Coordinate Calculation in the Linear Camera Model

The calculation of the space coordinates in the linear camera model is more complex than the pinhole camera model. The advantage of the linear camera model is that the focal length is not needed. Instead, the camera's view angle α must be known [Thomas03]. The process can be divided into two steps. First, the distance (sz-coordinate) between the object point and the cameras is calculated. After the distance from the camera to the z axis gets known, the other two coordinates (sx and sy) can be determined. The base point of the frame of reference is between the two cameras as shown in Figure 5-12.

NOTE:
 This figure is included on page 77 of the print copy of
 the thesis held in the University of Adelaide Library.

Figure 5-12: Frame of Reference for Space Coordinate Calculation on Linear Camera Model
 [Thomas03]

5.3.3 Relationship of Calculations in the Two Models

In different camera models, the distance between camera and object point is calculated in different ways [Thomas03].

Pinhole camera model:

$$q_L(3) = \frac{f_{Rx} \cdot v(1)}{\left(\frac{f_{Rx}}{f_{Lx}} \cdot ix_L \right) - ix_R} \quad (5.18)$$

Linear camera model:

$$x = \frac{k}{\frac{2 \cdot \tan(\alpha)}{length} \cdot (e_L - e_R)} \quad (5.19)$$

where $x = q_L(3)$ is the distance between camera and object point, $k = v(1)$ is the distance between the two cameras, $e_L = ix_L$ is the position in the left image and $e_R = ix_R$ is the position in the right image.

If a simplification is made: $f_{Rx} = f_{Lx}$ (it is allowed if the cameras are focused on to the same plant), an direct relationship between the two different intrinsic camera parameters α (camera view angle, linear camera model) and f (focal length, pinhole camera model) can be made through:

$$f = \frac{\text{length}}{2 \cdot \tan(\alpha)} \quad (5.20)$$

This equation means that the connection between the two models is established and the pinhole camera model can be considered as a linear model after some simplifications have been made. Equation (5.20) also provides a possible way for the calculation of the three-dimensional information. The program can be written based on both the pinhole camera model and the linear model.

5.4 Obtaining of the Three Dimensional Information

The program used in the calculating of the object space coordinates is based on the method described in Section 5.3: solve the equations given in **Section 5.3** according to the two dimensional position identified in **Chapter 4** and the parameters described in **Section 5.3**. The following flow chart of the program illustrates the process in a clear way.

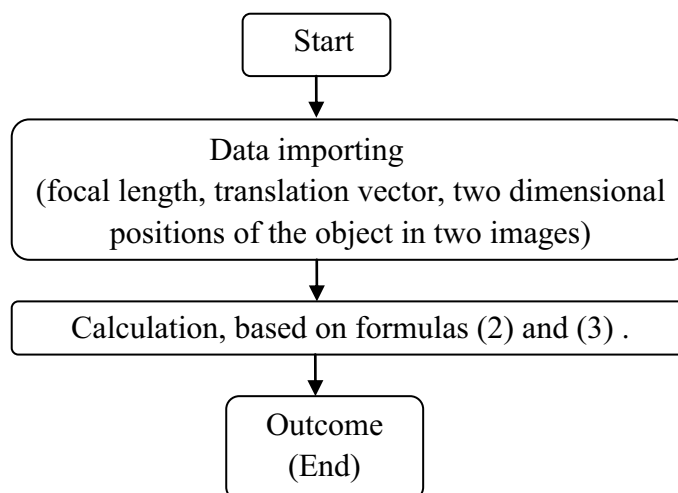


Figure 5-13: Flow chart of the 3D position calculation

The focus length and the translation vector were obtained from the software. The study used a mock grapevine in the lab. The distance between the camera and the target was 1000mm according to the measurement. Two images were captured using the two cameras, as shown in Figure 5-14 and Figure 5-15.

After the picture got captured, the images were uploaded onto the computer and the positions of the first cutting point were located using the method discussed in Chapter 3 and Chapter 4. Then the images were further examined and the calculation was conducted by the program. The result turned out to be:

$$distance_z=848.7543mm,$$

$$distance_x=2.5705mm,$$

$$distance_y=2.2727mm.$$

This is the three-dimensional position of the grapevine cutting point. Compare it with the real world result, which is

$$real_distance_z=845mm,$$

$$real_distance_x=2.5mm,$$

$$real_distance_y=2mm.$$

The outcome of the comparison is quite satisfactory, with an error rate of only 2.4%. After the locating of the pruning point of the grapevine, the system will inform the robot arm to perform the pruning. As shown in Figure 5-14, the distance between the camera and the target is 1000mm.

The experiment above is only set up on one branch. Experiments with more branches are needed to be performed to validate the program, and Figure 5-15 illustrates the procedure of cutting the whole four branches.

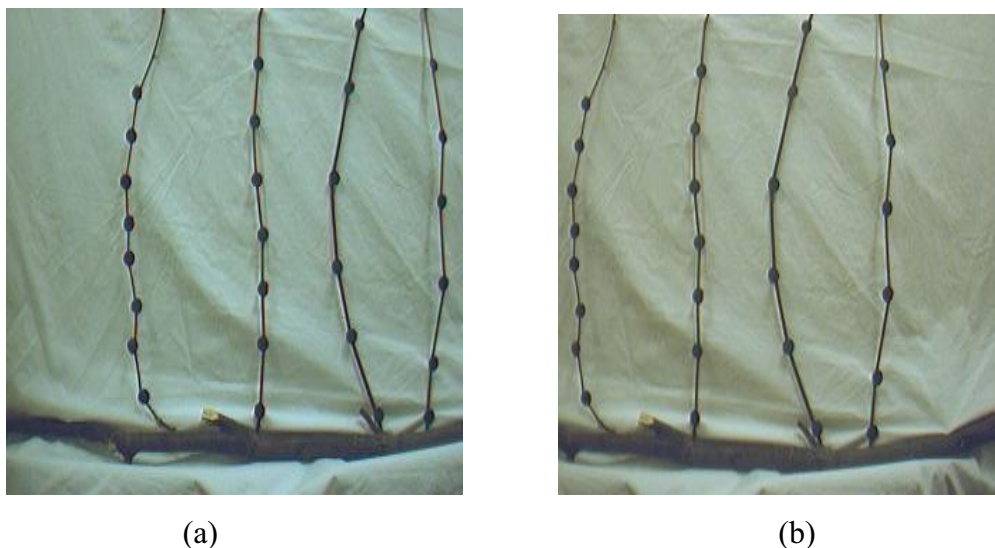


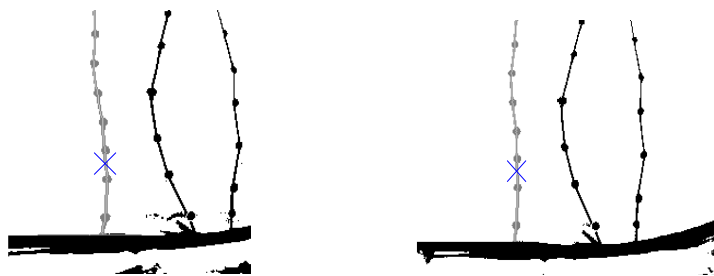
Figure 5-14: a) Left image, b) Right image



(a): First branch cutting points for both left and right images.



(b): After cutting the first branch for both left and right images.



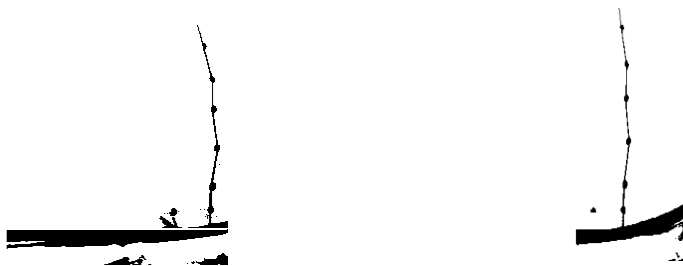
(c): Second branch cutting points for both left and right images.



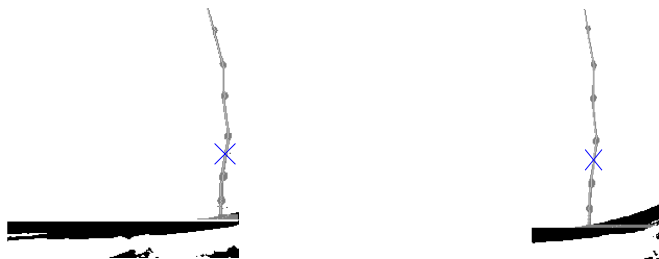
(d): After cutting the second branch for both left and right images.



(e): Third branch cutting points for both left and right images.



(f): After cutting the third branch for both left and right images.



(g): Fourth branch cutting points for both left and right images.



(h): After cutting the fourth branch for both left and right images.

Figure 5-15: The procedure of cutting the whole four branches

Table 5-1 shows the actual values of the grapevine. Table 5-2 shows the calculated values of the set up grapevine and Table 5-3 shows the successful rate in locating the 3D position.

Table 5-2: Actual Values (Unit: mm)

| | First Cane | Second Cane | Third Cane | Forth Cane |
|----------|-------------------|--------------------|-------------------|-------------------|
| Z | 682 | 593 | 584 | 653 |
| X | 123 | 245 | 325 | 425 |
| Y | 298 | 267 | 278 | 225 |

Table 5-3: Calculated Values by the program (Unit: mm)

| | First Cane | Second Cane | Third Cane | Forth Cane |
|----------|-------------------|--------------------|-------------------|-------------------|
| Z | 628.0935 | 564.0612 | 562.5079 | 585.4183 |
| X | 158.9039 | 280.7509 | 366.606 | 492.6179 |
| Y | 267.7186 | 235.7721 | 222.7479 | 185.1342 |

Table 5-4: Error percentage (Unit: %)

| | First Cane | Second Cane | Third Cane | Forth Cane | Row Average |
|-----------------------|-------------------|--------------------|-------------------|-------------------|--------------------|
| Z | 7.90 | 4.88 | 3.68 | 10.35 | 6.70 |
| X | 29.19 | 14.59 | 12.80 | 3.70 | 25.48 |
| Y | 9.83 | 11.70 | 19.87 | 17.71 | 14.78 |
| Column Average | 15.64 | 10.39 | 12.12 | 10.59 | 12.19 |

5.5 Concluding Remarks

From the experiments above, it can be seen that the hardware and software calibrations results are closely related to the space coordinates calculation. In

Figure 5-7 and Figure 5-8, a zoom between the two images reveals that the two cameras are not on the same height level. That is why the hardware calibration needs to be conducted. Before the images of the chessboard pattern are captured, the distances between the two cameras are 150mm. According to the hard calibration result, the cameras are separate at a distance of 138 mm. Also the directions of the two camera's CCD panels have been adjusted. As for the software calibration, the outcome of this process, namely the parameters got in the software calibration, can be used in the final calculation of the three dimensional position of the located nodes. The two parameters used are the focus length and the transformation vector determined through the Matlab toolbox developed by Bouguet.

The pinhole camera model is employed in the 2D to 3D transformation. The linear camera model and the pinhole camera model are exchangeable using equation (5.18), after some simplification have been made. The basic formulas of this model are used in program. With the focus length, transformation vector and the two dimensional positions of the nodes got in Chapter 3 and Chapter 4, the program can then have the 3D position of the nodes calculated.

Section 5.5 showed that the indoor grapevine setup is quite successful. Ten images (Appendix B) full of canes have been analysed, and an 85% successful rate of pruning position is achieved. While the rest 15% of the cutting positions found by the proposed method would not affect the growth of the grapevine in the next year significantly although they don't match the positions found by experts perfectly.

Chapter 6 Conclusions and Recommendations

The purpose of this chapter is to conclude this thesis by making an evaluation on its original objectives and contributions. Some suggestions for further study are also made.

6.1 Conclusions

In this study, a novel method is presented for grapevine pruning in winter. Specifically, algorithms utilizing image process, image analysis and stereo vision technologies are developed to locate the cutting positions on the grapevine, following the 2-bud spur pruning method.

At the beginning of this thesis, a literature review was presented, followed by the description of the methodology in this research. The methodology includes the introduction of image processing, image analysis techniques and stereo vision system. The purpose of the adoption of the image processing techniques is to pre-process the images for image analysis. The stereo vision system is used to locate the three-dimensional positions of particular notes. An example of the signal image processing (taken at the National Wine Centre in Adelaide, South Australia) is given in Chapter 4, and an overall success rate of 85% in locating the pruning positions is achieved. An indoor model grapevine was set up in the lab, and the three-dimensional cutting positions have been successfully located.

This research has successfully made a combination of the image processing, image analysis algorithm and stereo vision system. A new image analysis algorithm is developed in the location of the two dimensional position of the cutting point. Also, it is the first time that the stereo vision is applied in the

grapevine pruning. The success rate in the locating of the branches and nodes is high. The 3D calculation also proved to be quite success. These achievements prove that the methods and technologies developed in this study can be used in the pruning robot and do grapevine pruning automatically.

6.2 Some Recommendations

However, there is still a long way to go to improve this system to let it work more accurately in a more complex context. In this study, a white curtain is used to better distinguish the background and the objectives. This is because the background is usually very complex. The cameras may capture grapevines behind the objective one and mix them together without a curtain. However, in the practical situation, it is relatively costly to get the grapevines separated by curtains. It seems quite likely that a light source might work here. Under the illumination of the light source, the intensities of reflected light from braches in different positions changes greatly. It might work in distinguishing the objective grapevine and the background grapevines.

More work still needs to be done. For example, the method used in the 2D location of this study can be further developed. That is because that the method used in the location of the branches and nodes still have some errors, especially in the location of the nodes. The images taken by the camera need to be better processed and a better method should be used in the locating to reduce the error rate.

Also, the calculation based on the pinhole camera model is not accurate enough. That is because some simplifications have been made in this model to make the calculation easier. However, in the practical situation, the non-linear and some other effects do exists. It is better to use the non-linear formula in the calculation

though it might be more complex. The further work of this study will focus on this method to gain more accurate 3D position information.

References

Bulanon, D. M., Kataoka, T., Ota, Y., and Hiroma, T. (2002). "A segmentation algorithm for the automatic recognition of Fuji apples at harvest."

Burger Wilhelm, Burge Mark J., "Digital Image Processing". ISBN: 978-1-84628-379-6, 2008, pp. 13.

Chen, B., Tojo, S., and Watanabe, K. (2003). "Machine vision for a micro weeding robot in a paddy field." *Biosystems Engineering*, 85(4), 393-404.

Cho, S. I., Lee, D. S., and Jeong, J. Y. (2002). "Weed-plant discrimination by machine vision and artificial neural network." *Silsoe research institute*, 83(3), 275-280.

Cooper, R. (2004). "Magic eye how to see 3D." Magic Eye Inc.

Davies, E. R. (1990). "Machine vision: theory, algorithms, practicalities." 191-206.

Dubois, D., and Prada, H. (1985). "A review of fuzzy set aggregation connectives." *Inform Sci.*, 36, 85-121.

Erdogan, D., Guner, M., Dursun, E., and Gezer, I. "Mechanical harvesting of apricots." *Biosystems Engineering*, 19-28.

Faugeras, O. (1993). "Three dimensional computer vision."

References

- Forsth, D., and Ponce, J. (2001). "Computer vision: a modern approach, 3rd draft version." Prentice-Hall.
- <http://www.cmis.csiro.au/iap/RecentProjects/stereoEE02.htm> (2004). "Image analysis: 3D image using stereo vision."
- Gonzalez, R. C., and Woods, R. E. (1992). "Digital image processing." 491-494.
- Gonzalez, R. C., Woods, R. E., and Eddins, S. L. (2004). *Digital image processing using matlab*, Prentice Hall.
- Grabisch, M. (1996). "The application of fuzzy integrals in multicriteria decision making." *European J. Operational Res.*, 89.
- Intrieri, C., and Silvestrono, O. (1983). "Advances on mechanical pruning of grapevine: equipments and training systems." *First International workshop on mechanical pruning of the grapevine*.
- Iocchi, L. "Multiresolution stereo vision system for mobile robots." *Ai center- sri international 1*, Italy.
- Jean-Yves Bouguet (2010), Camera Calibration Toolbox for Matlab:
https://www.vision.caltech.edu/bouguetj/calib_doc/index.html
- Kondo, N., Ahmad, U., Monta, M., and Murase, H. (2000). "Machine vision based quality evaluation of iyokan orange fruit using neural networks." *Computers and electronics in agriculture*, 29, 135-147.

References

- Marchant, J. A. (1996). "Tracking of row structure in three crops using image analysis." *Computers and electronics in agriculture*, 15, 161–179.
- McFarlane, N. J. B., Tisseyre, B., Sinfort, C., Tillett, R. D., and Sevila, F. (1997). "Image analysis for pruning of long wood grape vines." *Silsoe research institute*, 111-119.
- Mercurio J F ; Gunkel W W ; Sobel T A ; Throop J A ; Norman D W
Vision-guided block-type robotic grapevine pruner . ASAE paper no . 89
– 7519 , New Orleans , USA , 12 – 15 Dec 1989
- Michael W. Kilby , “Evaluation of various pruning methods for maximum production of wine grape cultivars in southern Arizona”, 1999 Wine Grape Research Report.
- Ming Gao, Tien-Fu Lu. “Image Processing and Analysis for Autonomous Grapevine Pruning” Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation, 2006, pp. 922 – 927.
- Nash, J. C. “Compact Numerical Methods for Computers: Linear Algebra and Function Minimisation”. 2nd ed. Bristol, England: Adam Hilger, pp. 21-24, 1990.
- Naugle, J. A., Rehkugler, G. E., and Throop, J. A. (1989). "Grapevine cordon following using digital image processing." *Trans.ASAE*, 32(1).
- Norman D W., Throop J A., Gunkel W W.. “Electrohydraulic robot design for mechanized grapevine pruning,” ASAE paper pp. 91-1598,Chicago, USA, December 1991.

References

- Niku, S. B. (2001). *Introduction to robotics: analysis, systems, applications*, Prentice Hall.
- Perez, A. J., Lopez, F., Benlloch, J. V., and Christensen, S. (2000). "Colour and shape analysis techniques for weed detection in cereal fields." *Computers and electronics in agriculture*, 25, 197-212.
- Peterson, D. L., Anger, W. C., Bennedsen, B. S., and Wolford, S. D. (1999). "A system approach to robotic bulk harvesting of apples." *ASAE*, 99-1075.
- Pistre, R. (1994). "Head of vines farming methods." *Chapitre Domain, INRA, Internal communication*.
- Rudolf, K. (1998). "Hough transform.":
http://ikpe1101.ikp.kfa-juelich.de/briefbook_data_analysis/node122.html
- Sarig, Y. (1993). "Robotics of fruit harvesting: a state of art review." *Agricultural engineering research*, 54(4), 265-280.
- Scharstein, D., and Szelilske, R. (2001). "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms." *MSR-TR-2001-81*, Microsoft research technical report.
- Scharstein, D., and Szelilski, R. "High-accuracy stereo depth maps using structured light." *IEEE computer society conference on computer vision and pattern recognition*, Madison, WI, 195-202.
- Sevila, F. (1992). "A robot to prune the grapevine." 190-199.

References

- Sevila, F., Baylou, P., and Piquet, J. C. (1990). "Computer vision systems and growth modelization softwares applied to grapevine woods, as pruning robots design and control tools." *AgEng, Berlin (Germany)*, 24-26.
- Solis, L., and Wets, J. (1981). "Minimization by random search techniques." *Math. Oper. Res.*, 6, 38-52.
- Srivastava, A. (2010)"Robo Kisan- A Helping Hand to the Farmer." *Proceedings of the International MultiConference of Engineers and Computer Scientists*, Vol III.
- Stajanko, D., and Lakota, M. (2001). "Using image processing and analysis techniques for counting apple fruits in the orchard." *Horticultural Sci.*, 28(3), 95-99.
- Stajnlko, D., Lakota, M., and Hocevar, M. (2004). "Estimation of number and diameter of apple fruits in an orchard during the growing season by thermal imaging." *Computers and electronics in agriculture*, 42, 31-42.
- Straten, G. V. (2004). "Field robot event, Wageningen, 5-6 June 2003." *Computers and electronics in agriculture*, 42, 51-58.
- Svensson, J.. "Assessment of grapevine vigour using image processing". Master's thesis, 2002.
- Throop J A.,Ochs E., Gunkel W W., "Positioning with changing ground speed of mobile robots," ASAE paper pp91-7031, Albuquerque, USA,23-26 June 1991.

References

- Tian, L., Slaughter, D. C., and Norris, R. F. (1997). "Outdoor field machine vision identification of tomato seedlings for automated weed control." *Trans.ASAE*, 40(6), 1761-1768.
- Tisseyre, B., McFarlane, N. J. B., Sinfort, C., Tillett, R. D., and Sevila, F. (1997). "Fuzzy multicriteria decision-making for long cane pruning: a system for standard and complex vine configurations." *International journal of intelligent systems*, 12, 877-889.
- Tsai, R.Y. (1987). A versatile camera calibration technique for high accuracy 3d machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE J. Robotics Automat*, RA-3(4):323–344.
- Wiley, J., and Ltd, S. (1999). "Image processing: the fundamentals." *Maria Petrou*, 265-322.
- Wilson, G. B. (1983). "Five years of machine pruning: a grower's experience." *Am.J. Enol. Vitic*, 34(1), 40-41.
- Zhang, Z., Deriche, R., Faugeras, O., and Luong, Q. (1995). "A robust technique for matching two uncalibrated image through the recovery of the unknown epipolar geometry." *Artificial intelligence*, 78-87.
- Zhao, M C., Zheng, J Q. (2003).Tree crown recognition and precision to yard target pesticide application[J]. *Transactions of the CSAE*, 19(6): 150-153.

Appendix A: iBot Camera

NOTE:

This appendix is included on pages 95-96 of the print copy of the thesis held in the University of Adelaide Library.

Appendix B: The Ten Images Captured for the Experiments

Image One



Image Two

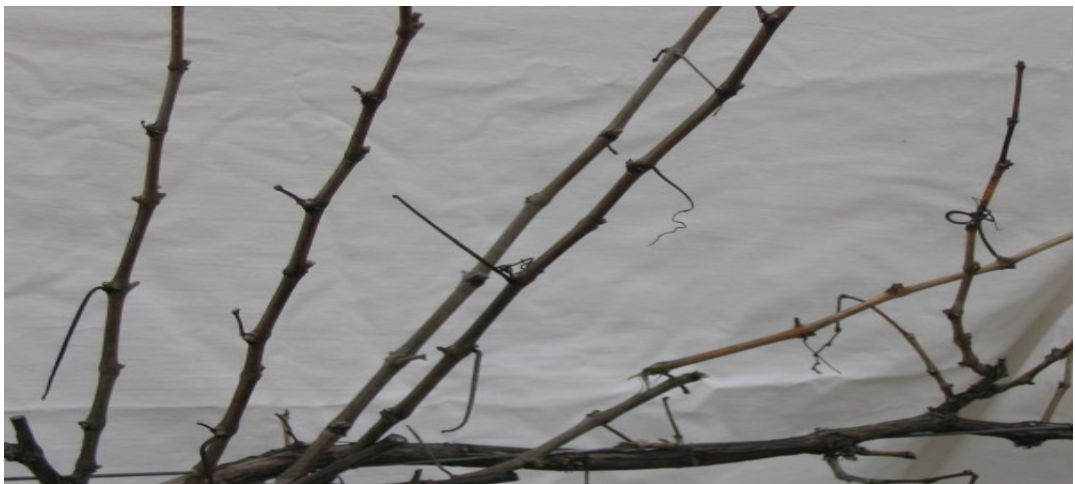


Image Three



Image Four



Image Five



Image Six

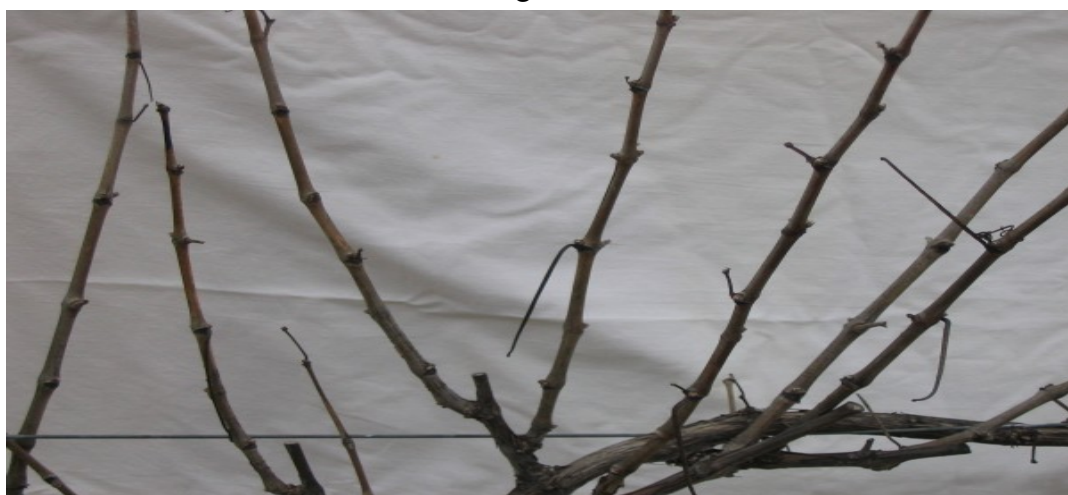


Image Seven

Appendix B: The Ten Images Captured for the Experiments



Image Eight



Image Nine



Image Ten

Appendix B: The Ten Images Captured for the Experiments