



## University of Bradford eThesis

This thesis is hosted in [Bradford Scholars](#) – The University of Bradford Open Access repository. Visit the repository for full metadata or to contact the repository team



© University of Bradford. This work is licenced for reuse under a [Creative Commons Licence](#).

**A REAL TIME 3D SURFACE  
MEASUREMENT SYSTEM USING  
PROJECTED LINE PATTERNS**

**ANQI SHEN**

Submitted for the degree of

Master of Philosophy

**Department of Computing**

**UNIVERSITY OF BRADFORD**

**2010**

# Abstract

This thesis is based on a research project to evaluate a quality control system for car component stamping lines. The quality control system measures the abrasion of the stamping tools by measuring the surface of the products. A 3D vision system is developed for the real time online measurement of the product surface.

In this thesis, there are three main research themes. First is to produce an industrial application. All the components of this vision system are selected from industrial products and user application software is developed. A rich human machine interface for interaction with the vision system is developed along with a link between the vision system and a control unit which is established for interaction with a production line. The second research theme is to enhance the robustness of the 3D measurement. As an industrial product, this system will be deployed in different factories. It should be robust against environmental uncertainties. For this purpose, a high signal to noise ratio is required with the light pattern being produced by a laser projector. Additionally, multiple height calculation methods and a spatial Kalman filter are proposed for optimal height estimation. The final research theme is to achieve real time 3D measurement. The vision system is expected to be installed on production lines for online quality inspection. A new 3D measurement method is developed. It combines the spatial binary coded method with phase shift methods with a single image needs to be captured.

The proposed algorithm for 3D surface fitting has been published in:

*Anqi Shen, Ping Jiang (2010) An Optimized 3D Surface Reconstruction Method Using Spatial Kalman Filtering of Projected Line Patterns. Advances in Soft Computing, SOCO 2010, Volume 73/2010, 179-186, Springer. DOI: 10.1007/978-3-642-13161-*

# Table of Contents

<b>CHAPTER 1 .....</b>	<b>1</b>
<b>INTRODUCTION .....</b>	<b>1</b>
1.1 Background .....	2
1.2 Problems and Objectives .....	3
1.3 Thesis Contributions .....	5
1.4 Thesis Organization .....	6
<b>CHAPTER 2 .....</b>	<b>8</b>
<b>BACKGROUND KNOWLEDGE AND LITERATURE REVIEW .....</b>	<b>8</b>
2.1 History of Stereo Vision .....	9
2.2 Binocular Stereo Vision.....	9
2.3 Structured light techniques .....	13
2.3.1 Basic Principle .....	13
2.3.2 Temporal Binary Coded Method .....	17
2.3.3 Spatial Binary Coded Method.....	19
2.3.4 Colour Coded Method.....	22
2.3.5 Phase Shift Method .....	25
2.3.6 Existing Applications .....	29

2.4 Summary .....	30
<b>CHAPTER 3 .....</b>	<b>33</b>
<b>SYSTEM DESIGN .....</b>	<b>33</b>
3.1 Introduction .....	34
3.2 Requirement Analysis .....	34
3.3 Technical Analysis .....	37
3.3.1 2D or 3D Vision Systems .....	37
3.3.2 Binocular or Structured Light .....	37
3.3.3 Which kind of structured light method .....	38
3.4 System Setup .....	42
3.5 Summary .....	44
<b>CHAPTER 4 .....</b>	<b>46</b>
<b>TECHNICAL DEVELOPMENT .....</b>	<b>46</b>
4.1 Introduction .....	47
4.2 Line extraction .....	47
4.3.1 Segmentation .....	47
4.3.2 Line Extraction .....	53
4.4 Height Calculation .....	55

4.4.1 Height Calculation at One Point .....	55
4.4.2 Height Calculation along a Line .....	56
4.4.3 Height Calculation in the Whole Image.....	59
<b>4.5 3D reconstructions .....</b>	<b>62</b>
4.5.1 Introduction of Kalman Filter .....	62
4.5.2 Spatial Kalman Filter for 3D Surface Fitting.....	64
4.5.3 Surface Generation.....	69
<b>4.6 Summary .....</b>	<b>70</b>
 <b>CHAPTER 5 .....</b>	 <b>72</b>
 <b>SOFTWARE IMPLEMENTATION AND EXPERIMENTS.....</b>	 <b>72</b>
5.1 Introduction.....	73
5.2 Developing the Software for 3D Measurement.....	73
5.2.1 Software overview .....	76
5.2.2 User Rights Control .....	77
5.2.3 Image Capturing.....	78
5.2.4 Calibration.....	79
5.2.5 3D Measurement .....	80
5.2.5 Product Quality Control .....	81
5.2.6 Parameter Settings.....	82

5.3 Experiments .....	83
5.3.1 Approach and Implementation.....	83
5.3.2 Robustness Test.....	86
5.3.3 Precision Test.....	89
5.3.4 Real Time Performance.....	89
5.3.5 Tests on Different Objects .....	91
5.4 Summary .....	93
<b>CHAPTER 6 .....</b>	<b>94</b>
<b>CONCLUSIONS AND FUTURE WORKS.....</b>	<b>94</b>
6.1 Conclusions.....	95
6.2 Further Works .....	98
<b>REFERENCE .....</b>	<b>100</b>

# List of Tables

<i>Table 2.1 Comparison among different methods</i> .....	30
<i>Table 3.1 Device list</i> .....	42
<i>Table 4.1 Data Result after Kalman filter</i> .....	68
<i>Table 5.1 IO definitions</i> .....	74
<i>Table 5.2 User rights control</i> .....	78
<i>Table 5.4 Measure results in different conditions</i> .....	88
<i>Table 5.3 Height measured after calibration (unit: mm)</i> .....	89
<i>Table 5.5 Processing time with different ROI and iterative calculation times</i> .....	89



# List of Figures

<i>Figure 2.1 Feature points and Epipolar lines.....</i>	<i>10</i>
<i>Figure 2.2 Epipolar Geometry.....</i>	<i>11</i>
<i>Figure 2.3 Basic Stereoscope Equations .....</i>	<i>12</i>
<i>Figure 2.4 Correspondent points finding with Epipolar line.....</i>	<i>14</i>
<i>Figure 2.5 Height calculation using light projection .....</i>	<i>15</i>
<i>Figure 2.6 Calibration model .....</i>	<i>15</i>
<i>Figure 2.5 Binary code projection model .....</i>	<i>18</i>
<i>Figure 2.6 Grid patterns with dot landmarks .....</i>	<i>20</i>
<i>Figure 2.7 2x3 pixels coded pattern.....</i>	<i>21</i>
<i>Figure 2.8 Colour strip pattern.....</i>	<i>23</i>
<i>Figure 2.9 Colour coded grid .....</i>	<i>24</i>
<i>Figure 2.10 Grey scale sinusoidal pattern.....</i>	<i>26</i>
<i>Figure 2.11 Sinusoidal and Gray code patterns .....</i>	<i>28</i>
<i>Figure 2.12 Branch-cut barriers in phase unwrapping.....</i>	<i>29</i>
<i>Figure 3.1 Stamping Plants .....</i>	<i>35</i>
<i>Figure 3.2 Stamping flow chart .....</i>	<i>35</i>
<i>Figure 3.3 Light pattern finally used .....</i>	<i>41</i>
<i>Figure 3.4 Actual devices.....</i>	<i>43</i>
<i>Figure 3.5 Light pattern views.....</i>	<i>43</i>
<i>Figure 3.6 System setup .....</i>	<i>44</i>
<i>Figure 4.1 Select ROI with raw image data.....</i>	<i>48</i>
<i>Figure 4.2 Plot chart from raw image data.....</i>	<i>49</i>
<i>Figure 4.3 Global threshold results by using different methods.....</i>	<i>49</i>
<i>Figure 4.4 Local threshold results by using different methods.....</i>	<i>51</i>

<i>Figure 4.5 Local threshold results with mean filter.....</i>	<i>52</i>
<i>Figure 4.6 Line extraction definition .....</i>	<i>53</i>
<i>Figure 4.7 Extracted line images .....</i>	<i>54</i>
<i>Figure 4.8 Height measurement model.....</i>	<i>55</i>
<i>Figure 4.9 Height calculation along line.....</i>	<i>57</i>
<i>Figure 4.10 Line searching sequence .....</i>	<i>58</i>
<i>Figure 4.11 Broken lines and unreliable points .....</i>	<i>59</i>
<i>Figure 4.12 Sample image clip with an object.....</i>	<i>60</i>
<i>Figure 4.13 Plot charts along vertical direction .....</i>	<i>60</i>
<i>Figure 4.14 Period calculation model .....</i>	<i>61</i>
<i>Figure 4.15 Line correction by period calculation.....</i>	<i>62</i>
<i>Figure 4.16 Points on the lines and the 3D surface.....</i>	<i>67</i>
<i>Figure 4.17 3D surface after applying the Kalman filter .....</i>	<i>69</i>
<i>Figure 4.18 Error variance and height plot chart .....</i>	<i>69</i>
<i>Figure 4.17 Height interpolation .....</i>	<i>70</i>
<i>Figure 4.18 3D surface after interpolation.....</i>	<i>70</i>
<i>Figure 5.1 Software running flow chart.....</i>	<i>76</i>
<i>Figure 5.2 Software overview .....</i>	<i>77</i>
<i>Figure 5.3 Camera monitor screen.....</i>	<i>79</i>
<i>Figure 5.3 Calibration screen.....</i>	<i>80</i>
<i>Figure 5.4 Measurement screen.....</i>	<i>81</i>
<i>Figure 5.5 Production screen .....</i>	<i>82</i>
<i>Figure 5.6 Settings screen.....</i>	<i>83</i>
<i>Figure 5.7 Sample Object .....</i>	<i>84</i>
<i>Figure 5.8 Measurement process.....</i>	<i>85</i>
<i>Figure 5.9 Measured error map.....</i>	<i>86</i>

<i>Figure 5.10 Measured result with paper on the object.....</i>	<i>86</i>
<i>Figure 5.11 Extracted line images with different illumination.....</i>	<i>87</i>
<i>Figure 5.12 Error variance versus iteration times .....</i>	<i>90</i>
<i>Figure 5.13 Measured results of different objects .....</i>	<i>92</i>

# List of Abbreviation

2D	Two-Dimensional
3D	Three-Dimensional
BGA	Ball-Grid-Array
LCD	Liquid-Crystal Display
DLP	Digital Light Processing
CCD	Charge-Coupled Device
ROI	Region of Interest
SNR	Signal to Noise Ratio
RGB	Red, Green and Blue
IO	Inputs and Outputs
GUI	Graphic User Interface
PLC	Programmable Logic Controller
OPC	OLE for Process Control
OLE	Object Linking and Embedding
STD	Standard Deviation
CPU	Central Processing Unit
SVR	SHRIS Vision Research

# Acknowledgements

First of all, I must give my gratitude to my primary supervisor Dr.Ping Jiang. His thoughtful ideas and constructive suggestion lead me into my research field. I must give my special thanks to his patient guidance on technique development and thesis writing up. Many thanks given to my second supervisor Dr.John Baruch for his support on my research work, thesis writing up and my life in UK. Special thanks to Miss Rona Wilson for her kind support for my study in University of Bradford.

Then I give my thanks to partners in Shanghai Tongji University, Pro.Jin Zhu, Dr.XiaoNian Wang, Chao Liu. They gave me a lot of useful ideas. I had many nice discussions with them. I also would like to thank my boss Mr.Xun Wang and colleagues Jiansong Wan, Jiangzhen Hu, Quoqiang Wu, Chengzhong Ruan and Huili Zhu in SHRIS (Shanghai Ro-Intelligent System,co.,Ltd.). Thank for their support in my research and giving me the chance to study in UK.

I would like thank my friends in Bradford, Yongqiang Chen, Weiping Jiang, Scott, Qian Zhen, Ivey and Wan Ting. They always bring me a lot of fun and make my life here never boring.

To my family, thank you for giving me so much love and support. At last, special thanks to my girl friend Huan. Her never end love and encouragement help me complete this thesis.

# Chapter 1

## Introduction

## 1.1 Background

As the technology develops, machine vision systems begin to play an important role in modern industry with manufacturing applications. Machine vision is simply to make a computer see and perceive what's behind the picture. A machine vision system usually works in the following stages. The first is to acquire images from a camera or a group of cameras. The secondly, using low-level image processing methods, is to reduce the noise and enhance what we are interested in. The third step is to extract meaningful information from these images. The last step is to analyse and interpret all the information to understand the scene behind the images. The whole process tries to mimic the human vision system and simulate how the brain processes images from the eyes. Usually, a machine vision system will work with other manufacturing equipment such as motors, cylinders or robotic arms to achieve production automation. As a result, machine vision is considered to be a subfield of engineering at the cross section of computer science, optics, mechanical engineering and industrial automation.

During recent years, in modern industry, a large number of applications based on machine vision systems are being implemented in: product assembly [1,2], appearance defect inspection [3,4], robot control [5,6], gesture recognition [7,8]. They are all existing applications and more are under development. The existing applications are based on 2D (Two-Dimensional) vision technology. With increasing demands for 3D (Three-Dimensional) measurement in industries, 2D machine vision systems cannot fulfil the advanced applications, which require detection, measurement and reconstruction in the stereo space. Recently, 3D applications have received extensive attention from researchers and engineers. The technical revolution from 2D vision to 3D vision can enhance production quality, reliability and precision. Rapid developments in

3D vision applications have emerged in diverse fields, for instance, surface deformation detection [9], terrain height measurement in geography [10], 3D spatial mounting and industry robot applications [11,12], surface roughness analysis [13]. Also 3D vision is widely used in the biomedical field, stereo shape measurement of vascular wall [14], oral dental measurement [15], anatomic structure segmentation [16] and scoliosis imaging and analysis [17]. These are the tip of the iceberg. Countless researchers are making efforts to build real vision worlds.

In this thesis, a 3D measurement system for the detection of product surface deformation is developed. It is intended to be used in a car-body stamping line. The shape of the produced car-bodies can be slightly changed due to abrasion of the stamping tools, which results in unqualified products with surface flaws. Presently, the measurement is done by an expensive 3D scanner or coordinate measuring machine. The measurement is a slow process and has to be conducted off-line. For this reason, only sampling tests are feasible for quality control. If any flaw is found from a test, the whole batch of car-bodies produced by this stamping machine must be recalled. If machine vision can do the surface measurement and defect detection on-line, an earlier alert can be raised to avoid wastage and improve product quality. A measurement system which can accomplish the measurements in real time is highly desired by industry. Here we designed a real time 3D measurement system for this application.

## **1.2 Problems and Objectives**

To develop such a system, we must consider the following problems we will have to face.



Firstly, the car manufacturing line is a mature system. It runs continuously hour after hour. Most of the automobile manufacturing suppliers, such as ABB, KUKA and COMAU, have standardised the design of their lines. So there is a potential request that this vision system must be designed as an accessory for an existing line and compatible to most of the automobile manufacturing suppliers' standards. It must avoid the use of proprietary components. Furthermore, all the components must be industry- standard compliant. This potential request imposes restrictions on parts selection. For example, a normal office LCD (Liquid-Crystal Display) projector is not suitable for this application, though it is widely used for stereo vision research. Selecting suitable components and techniques for the manufacturing industry is the first objective of this research.

Secondly, this system is not designed only to operate in a laboratory with an ideal environment. Different factories have different working environments. Illumination conditions can vary for different applications. Some companies will use this on a robot for dynamic and multiple region measurement and others just mount it on a fixed stand for a static and single view observation. Therefore, in order to be applicable to a wide range of products and working environments, a reliable and robust method in terms of varying surface reflectivity and background illumination has to be developed. This forms another objective of this research.

As well as these two problems, another important consideration is how to achieve rapid measurement. As we all know, efficiency is a critical factor for the success of a manufacturer. Saving more time on one car means the increasing on the efficiency of the production line. So a production line has its designed processing time for each step and limited total time for a cycle. The whole line has to run in accordance to the procedure and accomplish every single step in the predefined time. As required by the end-user of this project, the whole processing time of this vision system should be less

than 3s with a precision of less than 0.5mm for an on-line 3D measurement. A real time measuring system is required. In the last decade, structured light projection is the most widely used stereo vision technique in industry. This method usually requires the projection of a number of light patterns on to an object. There is a trade-off between the number of light patterns and the measuring precision. Reducing the number of light patterns results in decreased precision. The most important research objective of this research is to develop a solution to achieve real-time and high-precision 3D measurement.

In summary, the main objectives are:

- 1) To develop a 3D vision system suitable for use in industry. The system should be easily installed and compatible with the current production lines.
- 2) To develop a robust method suitable for various target applications and working environments.
- 3) To develop a measuring algorithm to fulfil the real-time requirement of this measurement application.

## **1.3 Thesis Contributions**

In the 3D machine vision field, this thesis has made its contributions to achieve the objectives presented above.

This vision system has a simple portable system setup based on industry standards. An automobile manufacturing company can use this for the quality control of its stamping line without large modification of the existing lines. Its simplicity also gives customers an economically attractive solution to enhance their quality control.

This thesis proposes the use of a light pattern using laser projection of parallel lines to enhance robustness and reliability. Moreover, this parallel light pattern reduces the complexity compared to other methods.

A real-time vision system is designed. In this thesis, only one image is used for the stereo measurement. The programme greatly decreases the quantity of light patterns.

A fine 3D reconstruction method is introduced combining with a spatial Kalman filter. This method contributes to the improvement of the measurement precision and error correction.

## **1.4 Thesis Organization**

The rest of the thesis is composed of five chapters. The main contents can be summarized as follows:

Chapter 2: Comprehensive literature review of 3D machine vision technologies. This chapter is divided into four parts. The first is about the history of stereo vision. A discussion on binocular vision and structured light techniques are presented in the second and the third parts. A summary of stereo vision is given in the final part..

Chapter 3: Description of the designed 3D measurement system. In this chapter, requirements are discussed and analysed. A vision system is designed to fulfil these requirements. Finally a short summary is provided.

Chapter 4: Technical development. The developed algorithms for real-time 3D measurement are presented in this chapter.

Chapter 5: System Evaluation. To achieve the desired task made-to-order software is developed. After a general introduction, the detailed experimental setup is described.

Then the same experiments are conducted on different targets and in different environments to verify its robustness and precision. Finally, the usability of the system is evaluated.

The last chapter, Chapter 6, is an overall summary of this thesis. It concludes the results of this developed vision system. Further work is discussed at the end.

# Chapter 2

## Background Knowledge and Literature Review

## **2.1 History of Stereo Vision**

As we all know, human beings' eyes are separated. What we see from each retina is slightly different [18]. The difference is called disparity. The disparity is minor for the objects far away and becomes greater for those nearby. Obviously, the distance of an object has a relationship with the disparity that we see in our eyes.

In 280 B.C., Euclid, the “father of geometry”, discovered the principle behind stereo vision [19,20]. He realized the images of one object simultaneously perceived by two eyes differ in their horizontal positions. In his book “Optical”, he constituted the visual rays from the eyes and established a theory of stereo vision from the view of geometry. One of the greatest artists and scientists, Leonardo da Vinci, conducted his research on the depth perception for his paintings and sketches [21]. He tried to establish a profound understanding of shading, texture and viewpoint projection. His success was not only in portraying a realistic depiction of the 3D world, but also left us knowledge about 3D modelling. Kepler summarized his theory of light and projection in [22]. Lindberg carried out a series of researches on early vision theory [23,24,25].

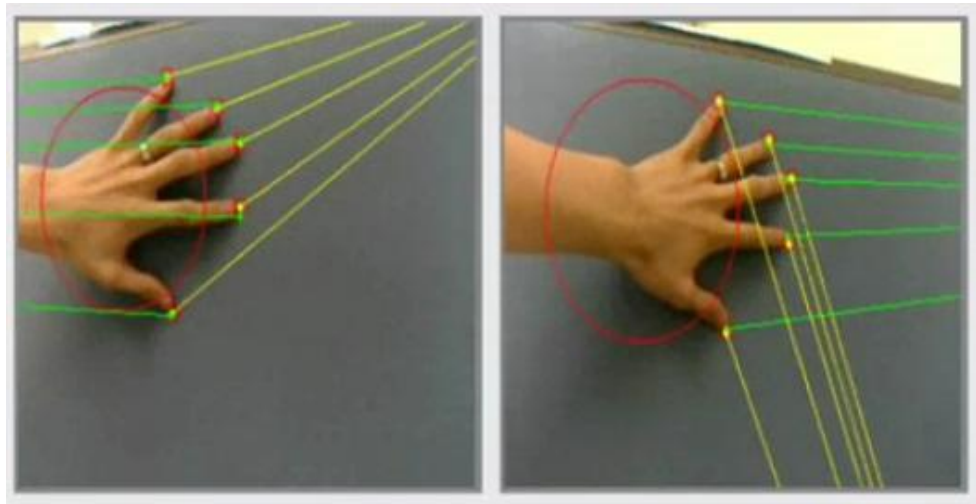
During Victorian era, stereo vision became an active research area. Many theories and innovations appeared. In 1851, Sir Charles Wheatstone [26] first came up with the idea of presenting slightly different images to the two eyes using a device he called a reflecting mirror stereoscope. Later, the Scottish scientist Sir David Brewster invented the first stereoscope named the Brewster Stereoscope, which became the template for all later stereoscopes.

## **2.2 Binocular Stereo Vision**

Our brain receives two separate images taken from our eyes. The relative position of an observed object will differ in the two eyes. Our brain is capable of measuring this disparity and extracting depth information. A binocular stereo vision system acts the same as human beings. There are three steps to form a depth image.

1) Feature points selection and determination in one image.

A feature point should be easily recognizable and precisely locatable, such as corners, light spots, edges, isolated areas [27,28] and so on. **Figure 2.1** gives an example; the finger tips are defined as feature points.

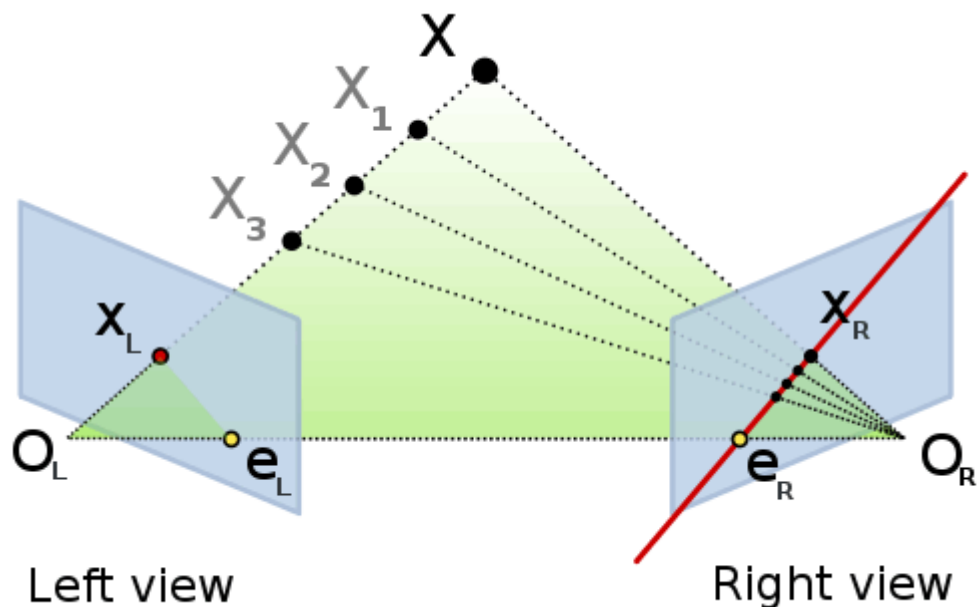


*Figure 2.1 Feature points and Epipolar lines*

2) Feature point correspondence in the other image.

In order to determine the correspondence for a given feature point selected in 1), a local template match is usually carried out. However, a 2D search for the match is computationally intensive and prone to spurious correspondence. Thanks to perspective projection, the 2D search can be avoided by searching only along a line, i.e. Epipolar lines, as shown in **Figure 2.1**. Epipolar Geometry is a general mathematic theory to describe two perspective images of a scene, see **Figure 2.2**.  $O_L$  and  $O_R$  are the focal

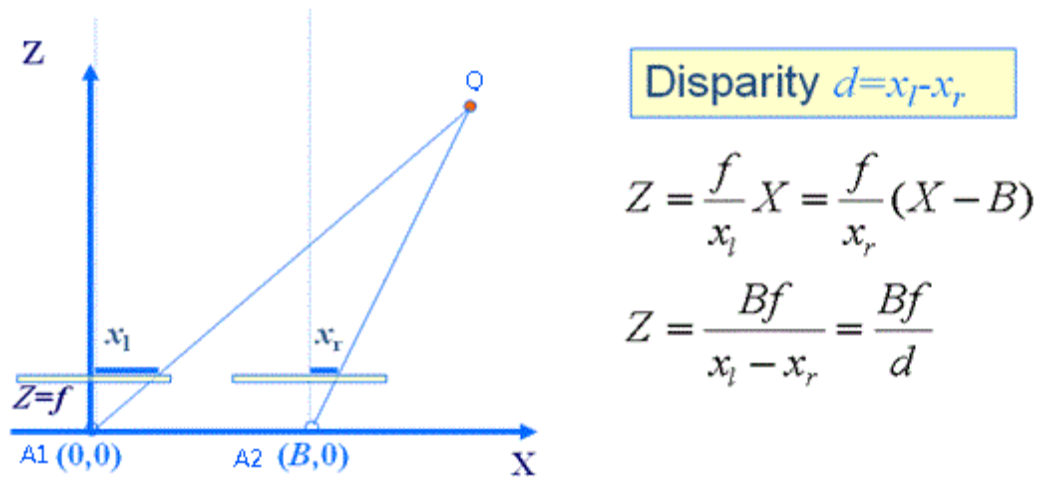
points of the camera.  $X$  is the target point which we are interested in. From the left view, any point along the line  $O_L X$  can be projected at the same point  $X_L$ , where  $O_L$  is the left focal point. Therefore, given a feature point at  $X_L$ , the possible correspondence must be on the projection of the line  $O_L X$  to the right view, which is called the Epipolar line. The points  $X$ ,  $O_L$  and  $O_R$  form a plane called the Epipolar plane. Epipolar geometry [29] can be used to constrain the correspondence of points. If  $X_L$  is known, the Epipolar line  $e_R$ - $X_R$  is defined, and the corresponding point  $X_R$  must locate on this Epipolar line. It is called the Epipolar constraint. On the other hand, if  $X_L$  and  $X_R$  are known, then the stereo position of point  $X$  can be calculated via triangulation. That is called Epipolar triangulation.



*Figure 2.2 Epipolar Geometry*

3) Disparity measurement and depth calculation.





**Figure 2.3 Basic Stereoscope Equations**

**Figure 2.3** depicts a basic stereoscope model. A1 and A2 is two

The distinct image points have constraints while a 3D point is projected onto 2D image planes.

The most difficult step in the aforementioned 3D measurement is determining the correspondence. Although the Epipolar line reduces the search space, an accurate matching is still difficult to achieve, especially for those smoother surfaces without distinct features. The stereo correspondence problem has attracted a lot of attention from researchers, such as Kanade [30], Belhumeur [31], Geiger [32]. Region-based matching [33,34,35] is a common method used for finding corresponding points. In [36], Mittal use this method to do the segmentation and to track people. Some approaches [37,38] are based on Epipolar geometry. Also there is some new arithmetic, Jenney [39] and Cernushi [40] proposed a Bayesian probability based approach to locate the best matched point with maximum probability. Roy [41] described a new algorithm with global 2D optimization which transformed it into a maximum-flow problem to produce the stereo correspondence. Though a lot of research has been done, finding

correspondence for a dense point cloud, to achieve high accuracy 3D measurement is still very difficult, if not impossible, with binocular stereo vision.

One way of getting round the correspondence problem in stereo vision systems is to employ an active light source, usually a laser, which puts a light spot into the scene. The identification of the spot in each of the two images can be accomplished by subtracting an image of the scene taken after the light source has been turned off from the corresponding image of the scene with the light spot. The position of the light spot in a difference image is then obtained by a simple threshold procedure. The alignment between the source and the cameras is not critical in active stereo systems since the source just puts an identifiable marker in the scene. That is the original version of structured light and the active stereo approach has become popular, especially in industry.

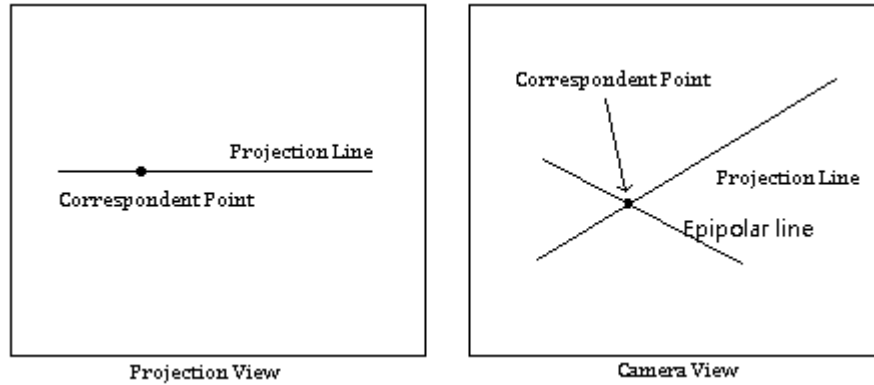
## **2.3 Structured light techniques**

### ***2.3.1 Basic Principle***

Stereo vision has been greatly improved by the improving technology. The method of using a single camera and a coded structured light pattern projection instead of binocular vision has become more popular in industry and research institutes.

At the end of the last section, we presented a simple active stereo method using a light spot to provide a clear feature to locate the correspondence point. For 3D measurement of a scene, scanning the full scene with single spots will be very time consuming. Using Epipolar constraints, the point on one image must be located on the relevant Epipolar line on the other image. If we project a straight line instead of a spot, the correspondence point can be easily identified by finding the intersection between the

detected straight line and the Epipolar line. In **Figure 2.4**, according to Epipolar geometry, one point on the projection line in the projection view must locate on the Epipolar line in the camera view. The intersection between the projection line and the Epipolar line in camera view which corresponds to that point in projection view. This approach can achieve a faster line-wise scan for 3D measurement.

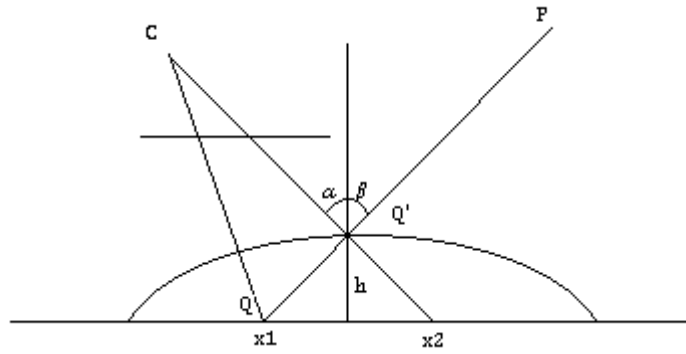


**Figure 2.4** Correspondent points finding with Epipolar line

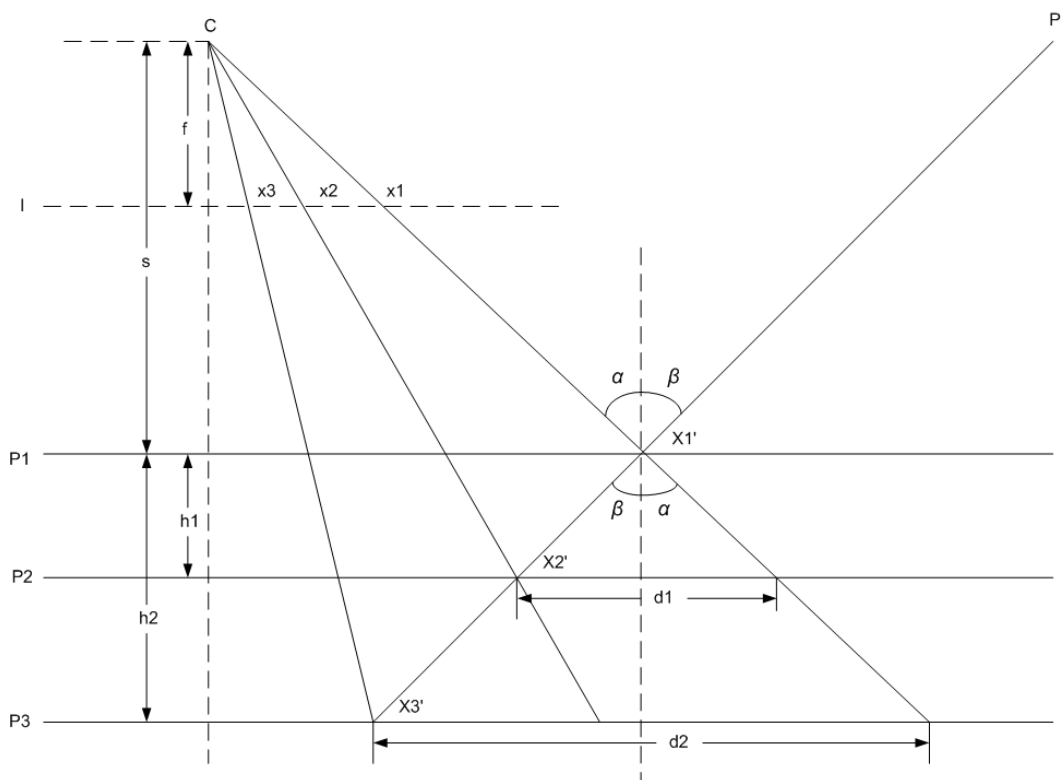
After we find the correspondence point, we can solve the height information by calculating the disparity. Here **Figure 2.5** depicts the basic principle to retrieve the height information using spot light projection. Point  $P$  denotes the exit pupil of the projector. Point  $C$  denotes the entrance pupil of the camera.  $Q$  is the original projected point on a reference height plane.  $Q'$  is the new projected point due to presence of the object.  $\alpha$  is the angle the camera light axis makes with vertical and  $\beta$  is the angle of the projector line with respect to the vertical line. Therefore, the relation between the height  $h$  and the shift of the projected points can be found as

$$h = \frac{k \cdot (x_2 - x_1)}{\tan \alpha + \tan \beta} \quad (2-3)$$

where  $x_2, x_1$  are the coordinates of the projected points perpendicular to the projection line;  $k$  is the scale factor from pixel distance to real world distance. For a given system,  $k, \alpha,$  and  $\beta$  are constants which can be identified by calibration.



**Figure 2.5** Height calculation using light projection



**Figure 2.6** Calibration model

**Figure 2.6** shows calibration model to find the values of these factors. According to the previous definition in **Figure 2.5**, firstly we put a flat board a defined distance from the camera. We mark this position  $P1$  and get coordinate  $x1$  on the image plane  $I$ . Secondly we move the flat board a short distance  $h1$  and mark as  $P2$  with coordinate  $x2$ . Then we move the flat board a short distance  $h2$  and mark as  $P3$  with coordinate  $x3$ . From similar triangles, we get the following equations:

$$\frac{x1}{x1'} = \frac{f}{s} \Rightarrow x1' = \frac{x1 \cdot s}{f}$$

$$\frac{x2}{x2'} = \frac{f}{s+h1} \Rightarrow x2' = \frac{x2 \cdot (s+h1)}{f}$$

$$\frac{x1-x2}{d1} = \frac{f}{s+h1} = \frac{x2}{x2'} \Rightarrow d1 = \frac{(x1-x2) \cdot (s+h1)}{f}$$

$$\frac{x1-x3}{d2} = \frac{f}{s+h2} = \frac{x3}{x3'} \Rightarrow d2 = \frac{(x1-x3) \cdot (s+h2)}{f} \quad (2-4)$$

From **Figure 2.5** and **Equation 2-3**, we can list following equations:

$$h1 = \frac{k}{\tan \alpha + \tan \beta} \cdot d1 \quad h2 = \frac{k}{\tan \alpha + \tan \beta} \cdot d2 \quad (2-5)$$

We make  $K$  as the total factor and solve the **Equation 2-4, 2-5**, then we get

$$\frac{h1}{(x1-x2)(s+h1)} = \frac{h2}{(x1-x3)(s+h2)}$$

$$\Rightarrow s = \frac{h1h2(x3-x2)}{h1(x1-x3)-h2(x1-x2)} \quad (2-6)$$

$$K = \frac{k}{\tan \alpha + \tan \beta} = \frac{h1}{d1} = \frac{h1 \cdot f}{(x1-x2)(s+h1)} \quad (2-7)$$

and  $x1, x2, x3, h1, h2$  and  $f$  have known values. From the calibration model, we build the relationship between image pixels and the real world.

However, the above equations are for a single light spot model. In order to get the height information of a certain surface, we must extend this model to the whole image. It will be a very slow process if points or lines are projected and images are captured in a serial manner. Obviously, we can project a group of points or parallel lines onto the surface simultaneously to reduce the number of pattern projections and image acquisitions. But actually, it is always difficult to find relevant points or lines. Therefore,

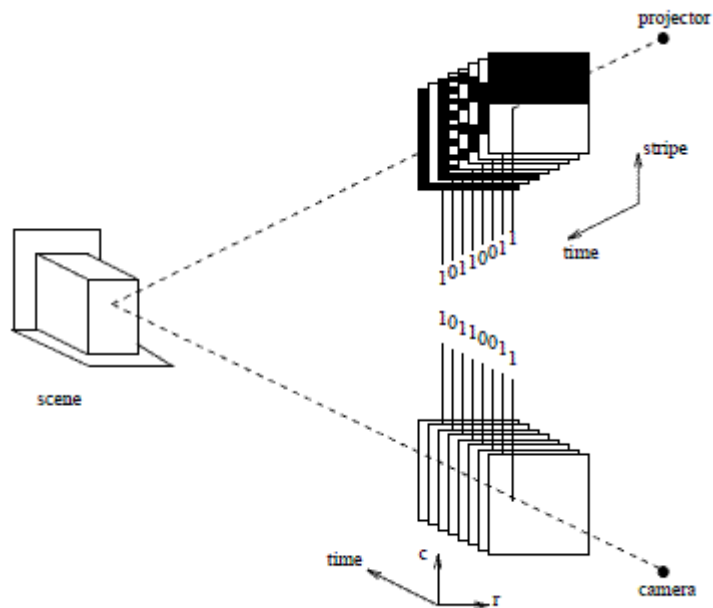
the projection pattern is encoded to ensure every projected point can be easily recognized and distinguished from the others.

Since the pattern is encoded, correspondence between image pixels and the projected pattern can be reliably detected. The decoded pixels can be triangulated and 3D information discovered. Salvi gives an overview of the coded structured light pattern methods [42,43]. Most of the structured light projection methods are based on a time-multiplexing strategy. A set of temporal coding patterns are projected onto the measuring surface. The correspondence of a given pixel is usually obtained by decoding the sequence of luminance values due to the projected patterns.

### ***2.3.2 Temporal Binary Coded Method***

An ideal light pattern for temporal projection is a binary coded pattern. In the binary coded structured light method, only two illumination levels are coded as 0 and 1, so that binary coded patterns have high accuracy and robustness especially for measurement. The primitive binary patterns can greatly raise the SNR (Signal to Noise Ratio). The pattern can be easily and precisely distinguished.

**Figure 2.5** gives the temporal binary code projection model used in Valkenburg's paper [44]. A pair of 9 banded light patterns is used. The first pattern is full illumination and the remaining eight form an 8 bit binary code. Each



*Figure 2.5 Binary code projection model*

Posdamer [45] presented the projection of a sequence of plain binary patterns which are coded by factors of 2. In total 16 frames of images are coded. With bright as 1 and dark as 0, a 16bit binary value will be generated on each pixel after a sequence of light pattern projections. That means that each pixel has a unique binary ID. From the ID, it is not difficult to find the correspondence between the projector and the image. The height information can be obtained by triangulation.

Inokuchi [46] improved the image pattern by introducing Gray code instead of plain binary. The consecutive coded patterns have Hamming evaluations, because there is only one bit difference between its neighbours. This method can correct mistakes when the grabbed pixel value is different from what it should be, caused by external noises. The application of Gray code makes the system more robust against noise.

Though the temporal binary coded light pattern techniques have been greatly improved during the last twenty years, they still cannot fulfil all kinds of applications. It has two drawbacks:

One is that it is difficult to reconstruct an object with a variable reflectance. The SNR will significantly decrease when projecting low illumination intensity patterns, while saturation will happen when projecting high illumination intensity patterns. Different parts of the surface give different reflectance performances. Binary coded patterns provide a global coded image. Unrecognized or misrecognized binary code in some parts of the image will cause failure in reconstructing the whole surface. For this reason, most binary coded methods require the objects to have a relatively uniform surface reflectance.

The other is its poor capability for real-time and dynamic measurement. Usually, to generate a binary serial code a series of light patterns will be projected and captured by a camera. Projection, image grabbing and image processing take too much time so that they are not suitable for real time applications with moving objects.

For variable reflectance, Skocaj et al [47] studied the issue to improve the performance. They used multiple light patterns with different illumination intensities and carried out a luminance calibration to correct the relative reflectance factors in an image. A matrix containing these factors is generated. The calibration of the illumination helps to correct mistakes. But on the other hand, this improvement requires more acquisition and processing time.

To work in real time, to reduce the number of light patterns, more identification information should be included in the patterns. Hence spatial binary coded and colour coded methods are introduced. The light pattern can have more recognizable features than just 0 and 1.

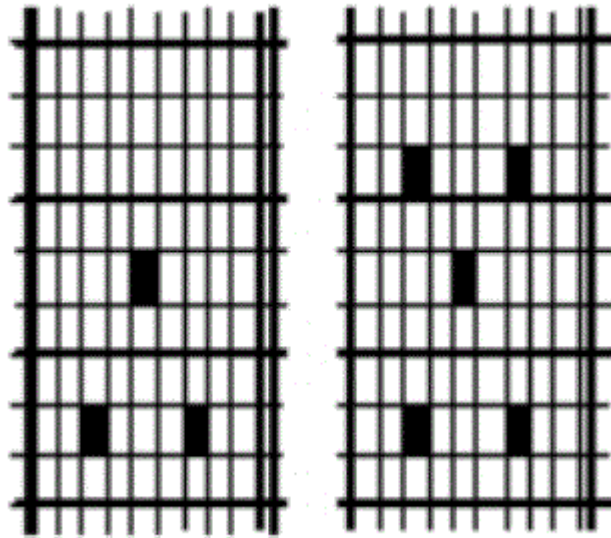
### ***2.3.3 Spatial Binary Coded Method***



In order to reduce the number of projection images, one solution is to use spatial information in addition to the temporal encoding.

The identity feature of each pixel in an image is not only encoded temporally, but also by its spatial relationship with its neighbours.

In [48], Moigne proposed a grid pattern for surface recovery. See **Figure 2.6**.



*Figure 2.6 Grid patterns with dot landmarks*

In the pattern, dots are added as landmarks grids. For different object, the number and position of the dots and the width of the grid are varied. Each intersection point has a fixed coordinate like latitude and longitude. With these coordinates, the corresponding point to a selected point can be found its relevant point whatever the surface.

This method has the advantage that only one image is used to determine the depth. That means real time 3D reconstruction becomes possible. But it also has a significant problem with surface breaks and the guidelines which happens on discontinuous surface. The discontinuous lines will cause correspondence failure.



(a) Pixel coded image pattern

$c_0$   $b_1$   $c_2$   $b_3$   $c_4$   $b_5$   $c_6$   $b_7$   $c_8$   $b_9$   $c_{10}$   $b_{11}$   $c_{12}$   $b_{13}$   $c_{14}$   $b_{15}$   $c_{16}$   $b_{17}$   $c_{18}$   
 $b_0$   $c_1$   $b_2$   $c_3$   $b_4$   $c_5$   $b_6$   $c_7$   $b_8$   $c_9$   $b_{10}$   $c_{11}$   $b_{12}$   $c_{13}$   $b_{14}$   $c_{15}$   $b_{16}$   $c_{17}$   $b_{18}$   $c_{19}$   
 $c_0$   $b_1$   $c_2$   $b_3$   $c_4$   $b_5$   $c_6$   $b_7$   $c_8$   $b_9$   $c_{10}$   $b_{11}$   $c_{12}$   $b_{13}$   $c_{14}$   $b_{15}$   $c_{16}$   $b_{17}$   $c_{18}$   
 $b_0$   $c_1$   $b_2$   $c_3$   $b_4$   $c_5$   $b_6$   $c_7$   $b_8$   $c_9$   $b_{10}$   $c_{11}$   $b_{12}$   $c_{13}$   $b_{14}$   $c_{15}$   $b_{16}$   $c_{17}$

$b_3$	$c_4$	$b_5$
$c_3$	$b_4$	$c_5$

 $((c_3, c_4, c_5), (b_3, b_4, b_5)) \rightarrow j = 4$

$c_3$	$b_4$	$c_5$
$b_3$	$c_4$	$b_5$

 $((c_3, c_4, c_5), (b_3, b_4, b_5)) \rightarrow j = 4$

(b) Binary sequence

Figure 2.7 2x3 pixels coded pattern

Figure 2.7 is a pixel coded method introduced in by Yuylsteke in [49]. In each of four chess-board blocks, the centre intersection point is labelled as “0” for dark and “1” for bright. Further, according to the positions of dark and bright blocks, additional marks “+” and “-” are added. Four primitives and 63 columns in total are used. Each row is assigned to two 63 bit binary sequence  $\{b_x\}$  and  $\{c_x\}$ . The grid point at column X with the modifier “+” is assigned to array  $\{b_x\}$  and “-” to  $\{c_x\}$ . In every 2x3 pixel window, a 6 bit codeword is generated that is a binary signature. With the underlying column

number which is in one-to-one correspondence, a 2x3 pixel window is sufficient for unique identification.

This modulation provides the following support. First, the provision of both a bright and a dark reference square to aid locating the feature points. In comparison with the method of detecting the feature points without the guidance squares, this method improves the feature recognition both in reliability and in accuracy. Secondly, by the inclusion of guidance squares, the adjacent squares provide additional information about their related direction. Encoding the detected feature points spatially, we can obtain a binary signature for a given window, e.g. 2\*3 in **Figure 2.7**, which provides a unique correspondence code. However, as with the other spatial binary code methods, the character of the unique identification is relative to its neighbours. The binary signature may become incorrect in a region with highly distorted neighbours.

#### ***2.3.4 Colour Coded Method***

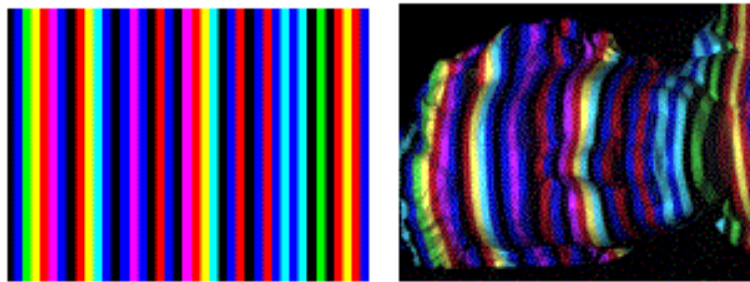
As described previously, the temporal binary code methods need a large number of pattern projections because only binary intensities are used. To reduce the number of light pattern projections, different coding methods can be applied.

Any colour can be described as a combination of three colours, R (Red), G (Green) and B (Blue). When we make each colour channel represent a binary code, the combination of these three channels will make a colour stripe. Each colour differs from that in neighbours. **Figure 2.8 (a)** is a coded colour stripe used in Zhang's paper [50]. **Figure 2.8 (b)** shows how the coded colour stripe is generated. When we project this coded colour stripe onto a surface, as shown in **Figure 2.8 (c)**, the colour in each pixel can form a binary value combined by R, G and B. Therefore, we can find that each coded

colour in the light pattern can actually represent a three bits binary code. This method greatly reduces the number of projected light patterns.



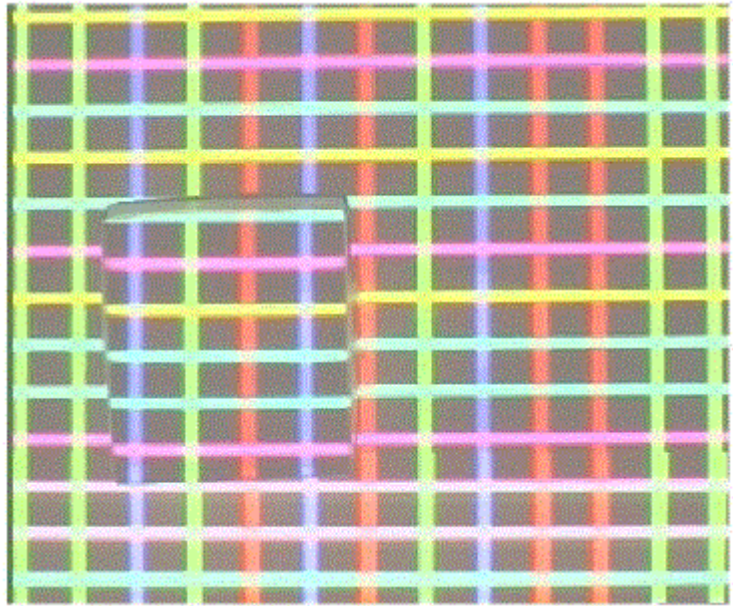
(a) *Coded colour stripe*



(b) *Light pattern*      (c) *Projection on a human statue*

**Figure 2.8** *Colour strip pattern*

In [51,52], a method using a colour coded grid is proposed, as shown in **Figure 2.9**. Different colours mark different guidelines. The intersection point between a pair of vertical and horizontal lines is uniquely determined in a selected region. The corresponding point is thus easily located when the object height is changed.



*Figure 2.9 Colour coded grid*

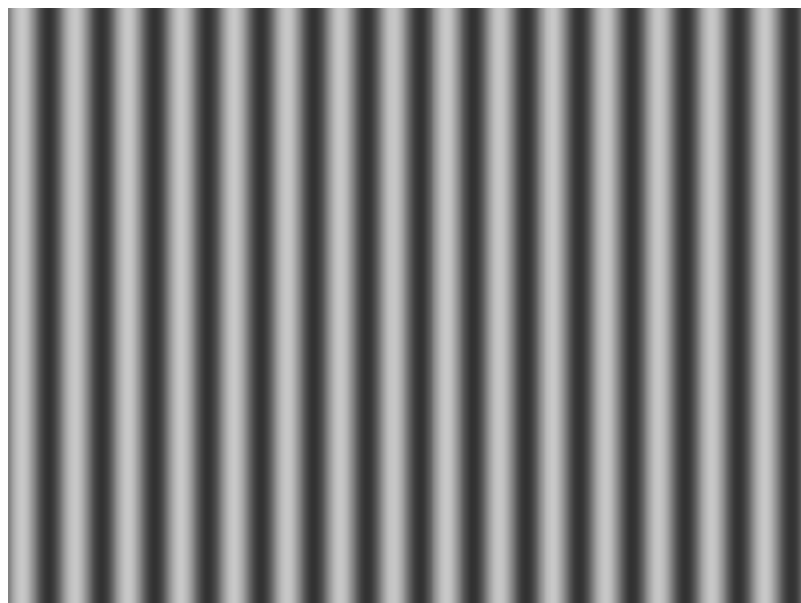
Caspi [53] compared the results of colour coded and binary coded structured light methods. He defined a colour scene using Gray code for each channel of R, G and B. He did his experiments on three objects: a planar colourful picture considering the effects of the background colours; an exponential staircase with a fixed height and a human face model for a complex 3D object. In order to achieve the same measurement resolution as achieved with the Gray code method, a single colour coded image pattern is not sufficient. His experimental results with 4 colour patterns produced a similar performance to using 10 Gray code light patterns. This method reduced the number of projected patterns.

Colour coded structured light techniques can greatly reduce the number of projected light patterns. There is a trade-off between robustness and the number of patterns in a structured light sequence. For embedding the same amount of information into a structured light sequence, a shorter sequence requires a single pixel to encode more information, with lower robustness against noise. The colour coded method relies on the

recognition of the R, G and B colour values for each pixel. For both projectors and cameras, the colours of neighbouring pixels will influence each other. Decoding is much more difficult than the black and white coding methods. So illumination, surface material, objects' texture and camera performance are all factors that can affect the final result. Using colour information with mixed R, G and B intensities, 3 temporal binary projections can be reduced to a single colour projection, which is important for real time measurement. In this way, we can regard colour coded methods as a combination of temporal binary code and spatial binary code. Due to interference between different colour channels, single colour projection cannot achieve the same robustness for decoding as 3 independent temporal binary codes.

### ***2.3.5 Phase Shift Method***

In the structured light stereo vision area, an important method is the phase shift approach. It is a kind of spatially coded method with a periodically changing light pattern. The observed intensity variation is related to its spatial shape. A commonly used periodic intensity pattern is a grey level sinusoidal pattern, **Figure 2.10** shows an example.



**Figure 2.10 Grey scale sinusoidal pattern**

In Bergmann's paper [54] four grey level sinusoidal patterns are used with a phase shift of a quarter period. The illumination model can be described as **Equation 2-8**.

$$I(x, y) = I_b(x, y) + I_m(x, y) \cos[\phi(x, y)] \quad (2-8)$$

Where  $I_b(x, y)$  is the background intensity at coordinates  $(x, y)$ ,  $I_m(x, y)$  is the projection modulation,  $\phi(x, y)$  is the phase distribution at point  $(x, y)$  and  $I(x, y)$  is obtained intensity from a CCD camera. There are three unknown variables  $I_b(x, y)$ ,  $I_m(x, y)$  and  $\phi(x, y)$ . To determine them, it requires at least three different intensity patterns to be projected on the surface and observed by the camera. But what we are really concerned is the phase  $\phi(x, y)$ . **Equation 2-9** presents a method to calculate the phase by projecting the four sinusoidal patterns with phase shifts of  $\pi/2$ .

$$I_1(x, y) = I_b(x, y) + I_m(x, y) \cos[\phi(x, y) + 0]$$

$$= I_b(x, y) + I_m(x, y) \cos\phi(x, y)$$

$$I_2(x, y) = I_b(x, y) + I_m(x, y) \cos\left[\phi(x, y) + \frac{\pi}{2}\right]$$

$$= I_b(x, y) - I_m(x, y) \sin\phi(x, y)$$

$$I_3(x, y) = I_b(x, y) + I_m(x, y) \cos[\phi(x, y) + \pi]$$

$$= I_b(x, y) - I_m(x, y) \cos\phi(x, y)$$

$$I_4(x, y) = I_b(x, y) + I_m(x, y) \cos\left[\phi(x, y) + \frac{3\pi}{2}\right]$$

$$= I_b(x, y) + I_m(x, y) \sin\phi(x, y)$$

(2-9)

By solving these four equations simultaneously, phase  $\phi(x, y)$  can be obtained as

$$\phi(x, y) = \tan^{-1} \left[ \frac{I_4(x, y) - I_2(x, y)}{I_1(x, y) - I_3(x, y)} \right] \quad (2-10)$$

The phase solution from the above equations is between  $-\pi/2$  to  $\pi/2$ , which cannot be used directly to evaluate the surface height. A periodic  $2\pi$  correction has to be carried out to produce the actual phase, which is called unwrapping. A simple unwrapping algorithm can be written as

$$\phi(x, y) = \begin{cases} \phi(x, y) + 2\pi & \text{if } \phi(x-1, y) - \phi(x, y) > \pi \\ \phi(x, y) & \end{cases} \quad (2-11)$$

From **Equation 2-5**, the height  $h$  is linear to the light shift  $d$  that corresponds to the phase shift in a phase map. As the result, we can obtain the height information of a pixel from its phase shift.

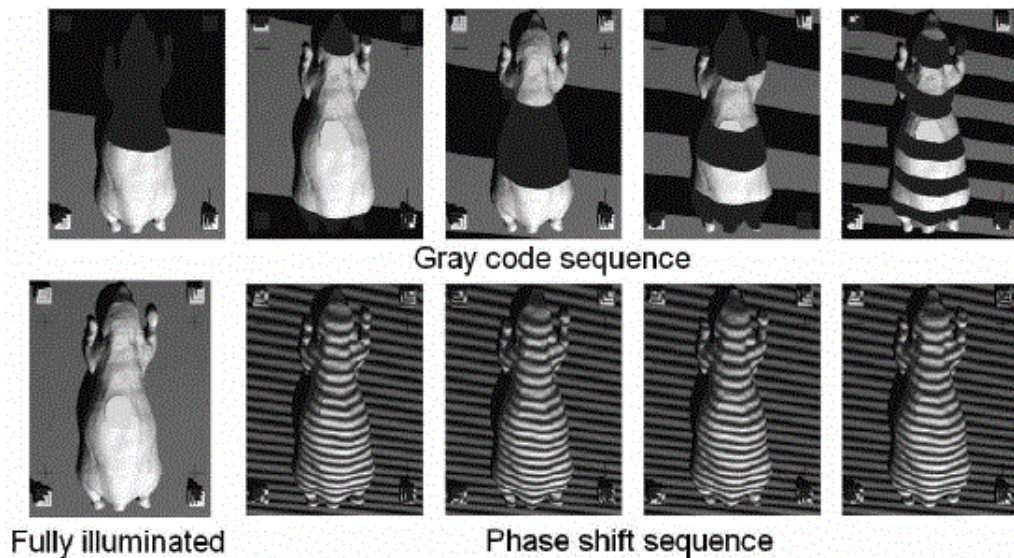
In Guhring's paper [55], it is suggested that using multi-stripe patterns which shift several periods is better than using the continuous sinusoidal patterns. To some extent, if we make the projected dark and bright line stripes thin, a similar image to a sinusoidal pattern is acquired in the camera. This phase shift method is called "line shift". The light pattern is much easier to make.

The phase shift method is a high quality measurement technique. There are three reasons: firstly, it covers the whole object. That means the height of any point can be calculated, that is different from grid or spot spatially coded methods. Those methods can only calculate height along the lines or on spots locations. Secondly, it uses the full range of grey scale values from 0-255 which produces higher resolution. The third is from **Equation 2-10**, with four phase shift patterns we can find that background and reflection factors are eliminated. This method is robust to the ambient light and variable surfaces.



But nothing is perfect; this method also has some drawbacks. Actually phase unwrapping is not that easy as mentioned previously. The phase cannot be recovered correctly when sharp changes in height or brightness, shadow and reflectance happen. The discontinuous phase map will produce confusion in the final result. Some sophisticated unwrapping algorithms exist to deal with those complex situations.

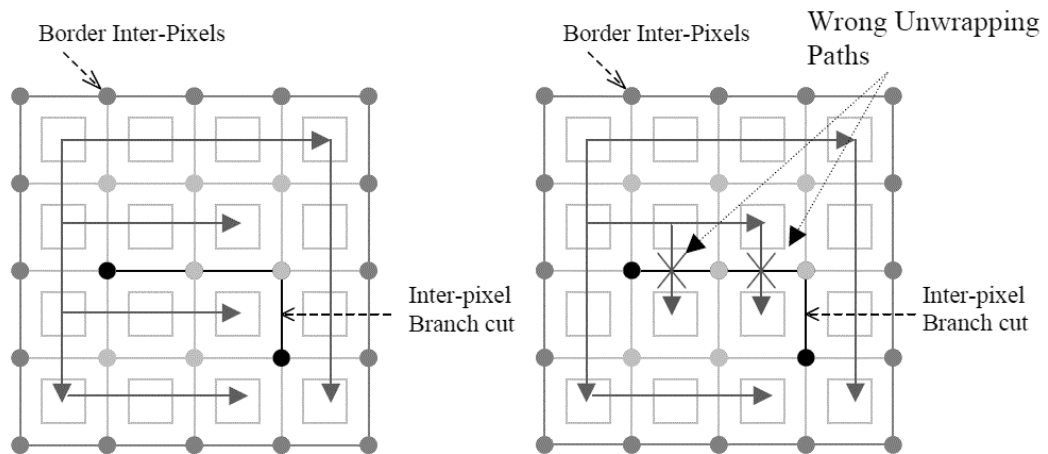
To solve the phase unwrapping difficulty, a binary coded light can provide a good aid for determining the correct phase. Ortuno [56] proposed a method where binary code patterns are used as an additional pattern supplementing the sinusoidal patterns, see **Figure 2.11**. The binary code patterns are used to avoid the unwrapping process by measuring coarsely which period a pixel belongs to. This combination solves the problem of a discontinuous phase map, but obviously the processing time is increased with more patterns involved. This unwrapping method is not suitable for the real time usage.



**Figure 2.11 Sinusoidal and Gray code patterns**

Besides the above unwrapping methods, the path-following method is also widely used. In [57], Karout defines some branch-cut barriers in a closed loop phase map. This

method is based on the theory that the summation of the gradient estimate of any closed path in the wrapped phase map must be equal to zero. A residue-vector is selected as the branch-cut barrier. Residues are defined to be local inconsistencies, at the beginning and the end of  $2\pi$  discontinuities. The residue counts how many  $2\pi$  periods are jumped. The residue can be divided into positive, negative and zero producing three groups. The residue-vector is defined as a positive point to a negative one. That forms the branch-cut barrier. The unwrapping path should not be across the branch-cut barriers as shown in **Figure 2.12**. Hence unwrapping paths are defined as well.



**Figure 2.12** Branch-cut barriers in phase unwrapping

This method can achieve high precision unwrapping especially for recovering small detail. This is the main contribution compared with other phase unwrapping methods. But also there is a significant weak point. Branch-cut barrier definition relies on the residue-vector finder. Residue can be affected by noise. The sign of the residue will be changed when the noise point is close to the correct one. It is a precise but not a robust solution.

### 2.3.6 Existing Applications

There have been a lot of applications of active stereo technologies. Yen and Tsia [58] used a sinusoidal phase shift method for BGA defect detecting. BGA defect detecting is an application in a very small field of view and very high precision is required.

Detection of a continuous surface with slight height changes means that the phase shift method will provide a suitable solution for the application, without using additional patterns or other unwrapping methods, it produced a satisfactory result.

Yu [59] and Wu [60] reconstructed 3D objects using the Gray code plus a line shift method. That is used for static 3D reconstruction. Even though quite a lot of images need to be taken, it is still a good method for this task.

Hall-Holt [61] used stripe boundary code for scanning moving objects. That is a spatial binary code method. Measurement is carried out on each frame by scanning a moving object.

## **2.4 Summary**

In this chapter, stereo vision techniques have been reviewed. Inspired by human eyes, binocular vision systems detect the disparity from two cameras to induce depth information. However, reliable correspondence has been a difficult challenge, especially with a smooth surface.

Structured light techniques are booming because of their capability to generate reliable correspondence. Temporal and spatial binary coded, colour coded and phase shift methods are the most popular methods used in active stereo. They are an essential technique for stereo vision systems. **Table 2.1** gives a general comparison of these methods.

*Table 2.1 Comparison among different methods*

<b>Methods</b>	<b>Temporal binary coded</b>	<b>Spatial binary coded</b>	<b>Colour coded</b>	<b>Phase shift</b>
<b>Pattern amount</b>	a lot	few	very few	4 or a lot
<b>Precision</b>	Middle	low	middle	high
<b>Mapping region</b>	whole image	edge and spot	whole image or edge and spot	whole image
<b>Robustness</b>	High	middle	low	middle
<b>Light pattern complex</b>	Low	middle	high	high
<b>real time performance</b>	Low	high	middle	middle
<b>Object requirement</b>	uniform surface	uniform surface	uniform surface	flat, smooth and uniform surface

In conclusion:

Temporal binary coding is a measurement method with reasonable precision and high robustness. It also has the advantage of supporting whole image reconstruction. But a series of light patterns are required so that it is not suitable for real time applications.

Spatial binary coded methods have a high cost-performance ratio. Compared with other methods, they slightly reduce the performance without increasing the device cost but greatly reduced the time required which makes it suitable for real time use. Low precision is a problem with height information only extracted at edge or spot positions using interpolation to provide the other heights.

The colour coded method is an eclectic method which increases precision for real time use. But the complex projection pattern generation and colour code detection increase the device cost. Robustness is the main drawback in this method. The SNR is much lower than in the binary coded methods.

The phase shift method has the highest precision. If applied to an ideal object surface, only 4 images are required, which makes it a suitable method for some applications requiring very high precision and less processing speed. However, on a surface with many discontinuities in depth, phase unwrapping becomes a big problem. Additional

patterns or a sophisticated unwrapping algorithm has to be applied. Therefore, it becomes a low speed method and not suitable for real time applications.

# Chapter 3

## System Design

## 3.1 Introduction

In the previous chapter, a literature review on 3D vision has been presented. It has been shown that structured light can be introduced for robust and accurate 3D reconstruction. In this chapter, we will discuss possible solutions for the 3D measurement of car-bodies produced by a stamping machine and presented to a laser projector based structured light system for this application. Firstly the application requirements are analysed and then the hardware components and architecture defined.

## 3.2 Requirement Analysis

In Chapter 1, a brief introduction of the application background is given. Quality of car-bodies produced by stamping machines may be affected seriously by the wear and tear of the pressing tools. Current quality control relies on an off-line sampling test by a 3D coordinate measurement system, which responds slowly and is expensive. In this project, an innovative 3D vision system is developed to meet the technical requirements of real-time, large field, accurate measurement.

**Figure 3.1** shows two kinds of stamping lines in the automobile industry. In a static stamping plant, one machine produces a product by mounting a certain stamping mould. Work pieces are usually loaded and unloaded by operators. Nowadays, more and more companies apply robots for the loading/unloading job. In a dynamic stamping plant, pre-designed stamping moulds and robots are used for flexible manufacturing. By selecting different moulds, various products can be made on these kinds of line, which can be embedded into a main flow production line. The system designed in this thesis is for quality control of the products from a dynamic stamping machine.

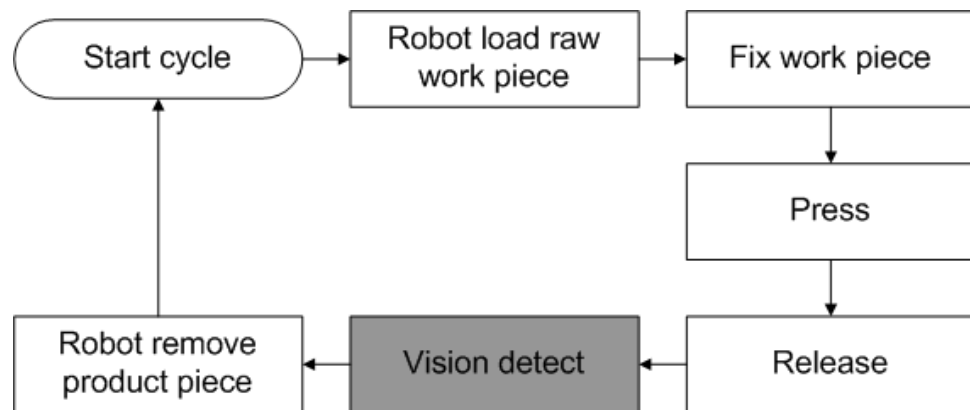


(a) *Static Stamping Plant*

(b) *Dynamic Stamping Plant*

**Figure 3.1 Stamping Plants**

Usually, the stamping process is shown as the following steps:



**Figure 3.2 Stamping flow chart**

The steps in white are commonly adopted in stamping lines. Because there is not a dedicated step for product quality inspection, products are sampled for quality control. In order to achieve real-time inspection, the ideal time is right after stamping tool release but before the product is removed by the robot. In that period, the product is still fixed in its position. If cameras are mounted on a robot arm, the robot first must reach the measurement position and wait for several seconds to carry out the 3D measurement. Then it removes the product piece into the good product loader or the flaw loader according to the measurement result.



The precision required for the shape of a car body piece is an error tolerance of 1mm for most automobile manufacturing companies. Considering other factors, the measuring precision should be less than 0.5mm.

The real time performance of such a system requires a minimum processing time. Processing time can be separated into two parts. The first part is the image grabbing time. This process must occur online. That means the fewer images grabbed the less time it takes. We consider using only one image to achieve this detection task. The second part is the image processing time. This should be finished after the image has been grabbed and before the robot starts to move the finished product. Normally, this period of time is less than 3 seconds. So the total processing time should take less than 3 seconds.

On the system setup; the 3D measurement system considered here will be integrated into the robot control system. The measuring components will be mounted on the robot arm. Therefore, the measuring system is required to be small and light. This requirement constrains the technical design. An office DLP (Digital Light Processing) or LCD (Liquid-Crystal Display) projector is not suitable, although most reported research on active stereo uses such a projector. Instead of an office projector, we consider using a small and reliable device such as laser projector. Furthermore, for industrial applications, a potential requirement is that all the parts should be suitable for industrial use. In the next section, the details of the technical analysis and the device selection are discussed. Robustness, reliability, compatibility and easy maintenance are all of concern.

One of the main contributions of this thesis is to embrace all the requirements listed about to build up a system to fulfil these requirements and make it suitable for industrial applications.

## **3.3 Technical Analysis**

### ***3.3.1 2D or 3D Vision Systems***

In a car stamping line, the abrasion of stamping tools is too tiny to be detected from the 2D image captured by a camera directly. It is difficult to make an explicit discrimination between image changes due to tiny height variation and due to signal noise. Further, product deformation in height is the most important indication of the stamping tool's abrasion. In order to detect it, the system has to be precise and sensitive enough to measure the height change. Even if a 2D vision system can detect the difference between a standard object and the measured one, quantisation of the height change is still undetermined. For these two reasons, the measurement is beyond the capability of 2D vision systems. A 3D stereo vision method is necessary for this application.

### ***3.3.2 Binocular or Structured Light***

Having decided that a 3D stereo vision technique should be used in this measurement application, a suitable approach has to be taken to facilitate the measurement. Chapter 2 gives a review of the stereo vision techniques. Both binocular vision and structured light methods have the potential to become a solution to the problem. Considering that the target object usually has a continuous surface with smooth optical characteristics, correspondence becomes difficult if a binocular vision system is used. As discussed previously, the binocular vision system needs to use salient features for matching between two images. . Anderson in [62], Geiger in [63] and Malik in [64] all mentioned junctions and edges as cues for finding these features. Therefore matching is difficult with a smooth surface with consistent albedo. For this reason, a binocular vision system is not a good solution for this application.

As shown in chapter 2, structured light techniques can produce a good performance with this application. As is discussed in Section 2.3, structured light techniques commonly have their limitations with an object with a discontinuous surface. They are more suitable for an object with a smooth and uniform surface. Luckily that is what we are facing in this application. Furthermore, unlike a binocular vision approach, a structured light approach relies on the decoding or deformation of the projected light patterns which are less sensitive to variable surface characters and ambient light. That has some advantages. Firstly, it's easy for one camera to measure different parts or objects without the recalibration required for a binocular vision system. Secondly, it is robust to environmental changes. Structured light patterns have higher SNR and clear features which are much easier to record than those in a binocular vision system. Of course, saving one camera and easy maintenance are other valuable points for customers. For all the reasons above, the structured light approach is the most suitable one for this application.

### ***3.3.3 Which kind of structured light method***

In Section 2.3, existing structured light methods are discussed. Here we need to find suitable light patterns for this application.

As we described in Section 3.2, this application should have a high real-time performance. The most critical factor we need consider is the number of images required. This number directly affects the online processing time. As concluded in **Table 2.1**, the temporal binary method is the first one to be excluded. From the experimental results, to project one pattern and acquire one stable image normally takes 0.5 second.

For this reason, the temporal binary code method is excluded. Usually, an 8 bit or 16 bit binary codeword is needed in the temporal binary code method. That means 8 or 16 light patterns and 4 or 8 seconds for acquiring multiple images. The processing time would be too long to be accepted as a real-time solution.

Next, the phase shift method is also excluded. Even when the pattern number has been reduced to 4, it is still not suitable for real-time applications. Another reason is that the light pattern is quite difficult to generate without a digital projector. Precise pixel alignment during the shift is hard to achieve.

The colour coded method is one possible choice for real-time applications because it is possible to take only one image for measurement. However, as we know, different materials have different light reflectance. The colour coded method is a method based on pixel intensity values. In the pixel domain, different light levels accumulated at a pixel produce different pixel intensity values. Changes in pixel intensity value may be caused by variable light reflection but not changed height. This may lead to different results with different object surfaces compared to the binary coded method which has only two intensities: black and white. Also the colour coded method has the same problem as the phase shift method in that the pattern image is quite difficult to generate without using a digital projector. The nonlinearity of the illumination, its cost and its size hinders the use of digital projectors in industry.

The spatial binary coded method is more suitable for this project for its real time performance and its robustness. But low precision is the main drawback associated with the spatial coded method. In spatial binary coded methods, usually grid and serialized spots are used as target features. Unlike the phase shift method and the colour coded method, the spatial binary code method doesn't map to a globally continuous code. The

measurement only takes on the target features; fitting height information only for target features.

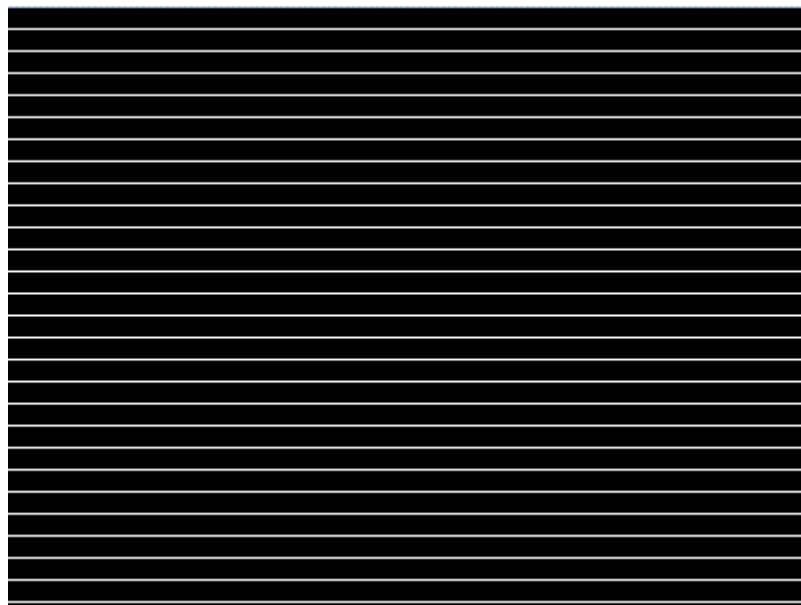
In conclusion, the temporal binary coded and the phase shift method failed in real-time performance. The colour coded method succeeds in real-time performance but fails in robustness. The spatial binary coded method fulfils the real-time performance and the robustness requirements, but fails in precision. The classical structured light methods are not suitable for this application.

In this thesis, a new approach is presented. It is a combination of the phase shift and the spatial binary coded method. Precision performance is inherited from the phase shift method; Real-time and robustness performances are both inherited from the spatial binary coded method. In short, it can be called a one image line shift method. One image with spatially distributed lines for height calculation and the height correction is carried out using the pattern period. In this method, a set of horizontal paralleled lines are used but not as a grid. Though a grid has one more set of lines in the vertical direction for increasing precision, horizontal parallel lines have three advantages. Firstly, we find that horizontal parallel lines which are seen from a camera will always be horizontally parallel even if the angle between the laser projector and the camera is changed. This angle is necessary for finding the height disparity that is discussed in Section 2.3.1. With this angle, a perspective transform is produced. As we can imagine, when a square pattern is projected onto a flat plane, it will be deformed into a quadrangle from the view point of the camera. Perspective correction has to be carried out for the height measurement, which reduces the precision and increases the computational burden. That is one of the reasons why we used horizontally parallel lines but not a grid.

The second reason is that feature extraction using a grid is much more difficult than with pure lines. The intersection points cause unreliable positioning on either the horizontal line or the vertical line. These two lines will affect each other. The pixels in this small area can be regarded as on the horizontal line, or on the vertical line, on both or on neither. Mistakes will happen at the intersection points especially where two lines do not cross perpendicularly. This situation produces more unreliable pixels.

The third reason is that if we consider using period calculations for the height correction, how clear the period information is in one direction is very important. The additional lines give additional information but also cause additional noises.

For these three reasons, we defined the light pattern as shown in **Figure 3.3**. The laser projector improves robustness. These lines with a high brightness against the background are easy identified.



*Figure 3.3 Light pattern finally used*

In addition, a spatial Kalman filter is applied for smoothing the surface.

Further experiments prove that the achieved measuring precision, real time performance and robustness are satisfactory for the application when using this new approach. The next chapter will give the detailed algorithm.

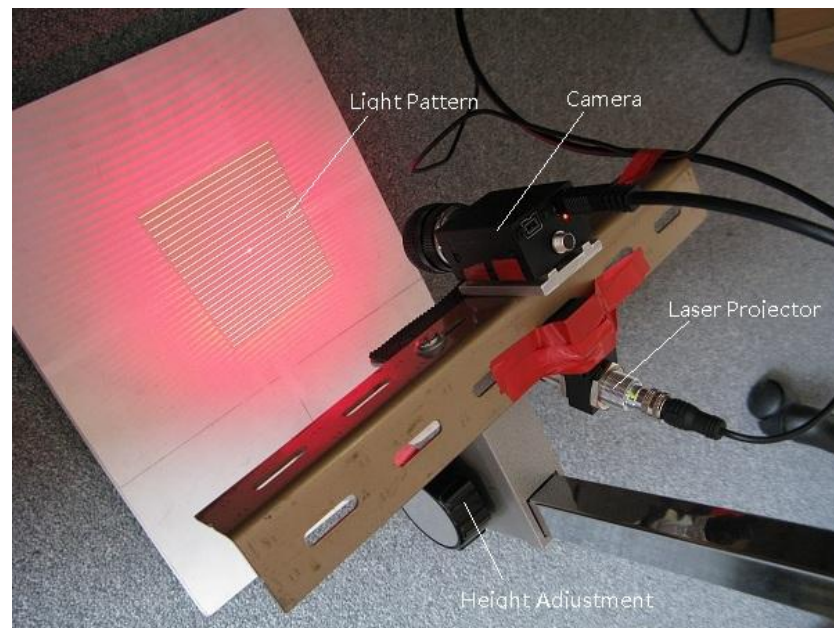
### 3.4 System Setup

In this thesis, the devices used are listed in **Table 3.1**.

*Table 3.1 Device list*

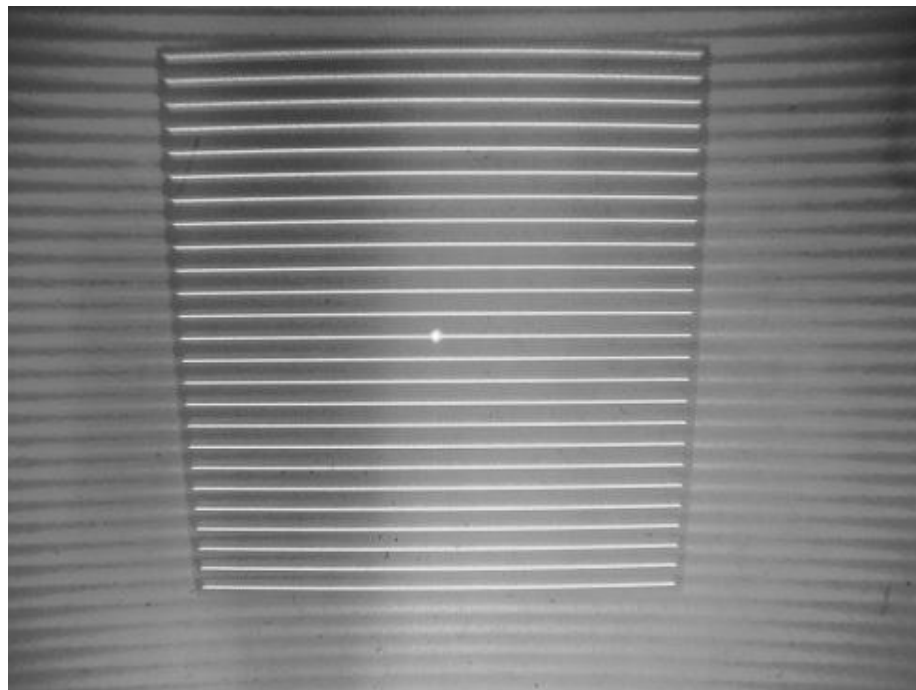
<b>Device name</b>	<b>Description</b>	<b>Select reason</b>
PointGrey GRAS-20S4C	resolution: 1624x1224 type: colour interface: 1394b bus speed: S800 pixel size: 4.4 $\mu\text{m}$ x4.4 $\mu\text{m}$	High resolution industry CCD camera. For testing different methods, a colour camera is selected but actually a mono camera is enough for the final proposed solution.
ZLA Z-Laser projector	pattern:25 lines type: lens filter colour: red	Industry based device. Laser project with lens filter ensures all the lines are static, while a laser scanner will often be sampled with missing lines by a camera The selected lens filter also has a centre spot light which is used as guidance for correspondent point finding.
ThinkPad T400 laptop	CPU: Intel Duo P8600 2.4G Memory: 4G Video Card: ATI 3400 Port: 1394a	For portable use. Equal performance comparing with normal desktop PC. 1394a only support 400Mbit/s as be max bus speed, but that's not affect this usage.
1394B hub	ports number: 4 max bus speed: S800 power supply: external	Addition power supply is given, because of lack of bus power supply on 1394a port. Flexible extension usage.
Height adjustment mounting stand	0-50 cm height adjustment screw bar. Mountable rack.	Height adjustment for calibration. Screw bar for easy height measurement. Rack is compatible to the robot mounting use.

**Figure 3.4** shows the configuration of the developed system.



***Figure 3.4 Actual devices***

**Figure 3.5** gives the light pattern observed from the CCD (Charge-Coupled Device) camera.

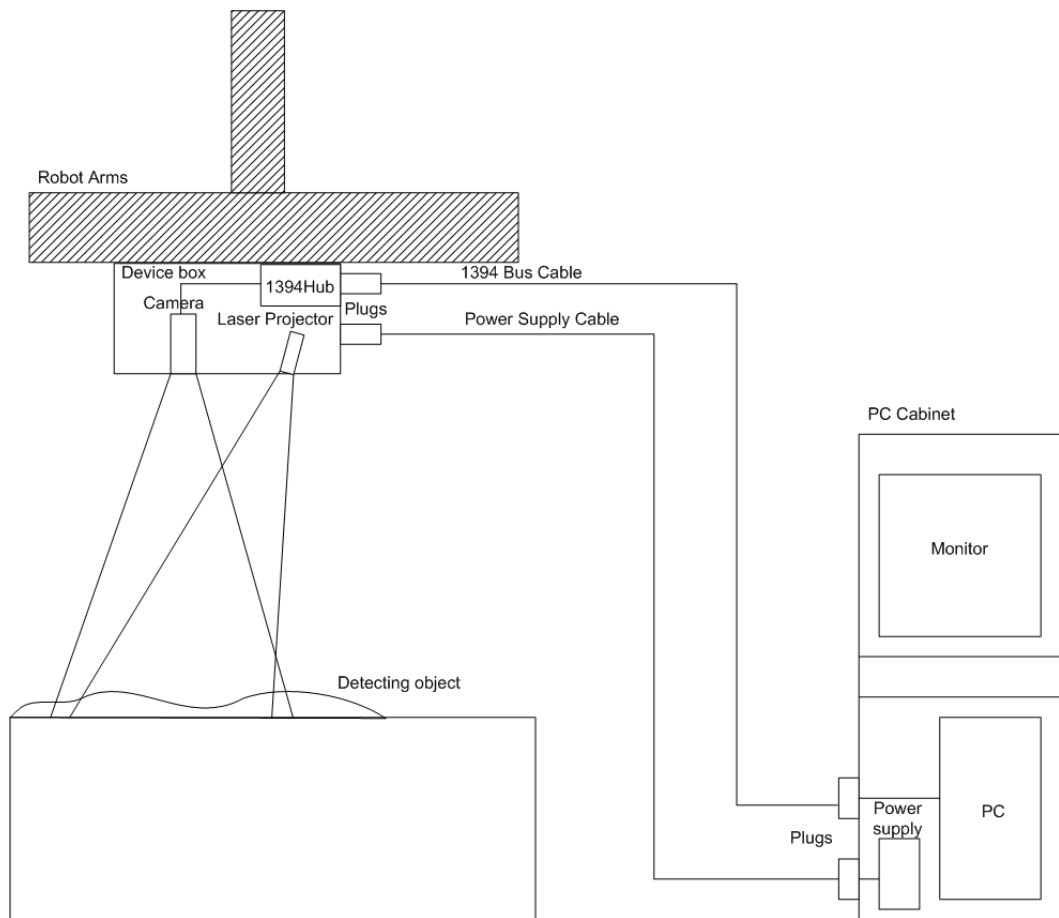


***Figure 3.5 Light pattern views***



The height information is obtained by measuring the deformation of the lines, which will be discussed in detail in the next chapter.

The above system configuration is only a prototype for technical research and development. For the further production, the system should have a more compact structure as shown in **Figure 3.6**.



*Figure 3.6 System setup*

### 3.5 Summary

In this chapter, the application requirements are defined and analysed. A brief summary of the requirements is listed below:

Device selection: industry based

Precision:	<0.5mm
Image grabbing time:	<0.5s
Processing time:	<3s
Robustness:	free of light illumination changes
Installation:	robot arm mountable

In Section 3.3, we made a technical analysis based on the requirements above. An industrial digital camera and a laser projector are selected to fulfil industry based device selection requirements. Parallel lines with the optimized height calculation method and a spatial Kalman filter are used for increasing the performance to meet the precision needs of the requirement. When only one image is grabbed and processed, time consumption is reduced to be within the time request. The laser projector has a high signal to noise ratio which makes it easy to extract features. This ensured system robustness against illumination changes. Small and light devices are used and they can be sealed in a device box with standard bus and power interfaces. This system setup makes it easy to mount it on a robot arm.

# Chapter 4

## Technical Development

## 4.1 Introduction

According to the system level design in the previous chapter, relevant algorithms need to be developed for real applications. To achieve this, there are three main challenges:

The first is how to extract line features. Obtaining clear and precise line features is the first step for height calculation. Furthermore, the extraction method should be robust against illumination changes.

Next is how to measure height from the extracted line features. An optimized height calculation method is proposed. This method is based on calculating the period along the vertical direction. It increases the precision of the 3D measurement.

The last is how to present the measured result. The measured result should be accurate. Spatial Kalman filler technique is introduced to further increase the precision.

In this chapter, detailed discussions are given on the approach and the algorithms.

## 4.2 Line extraction

### 4.3.1 Segmentation

The precision of the reconstruction relies on the quality of the line extracted from an image. The first step of the extraction is segmentation. Segmentation is an operation that takes an image as input and returns as a marked region the area that we are interested in.

Because in this application the input is a binary image, a simple way is to use a threshold operation to separate all the lines from the image background. In general, threshold operation is defined as:

$$S = \{(i, j) \in R \mid g_{min} < g_{i,j} < g_{max}\} \quad (4-1)$$

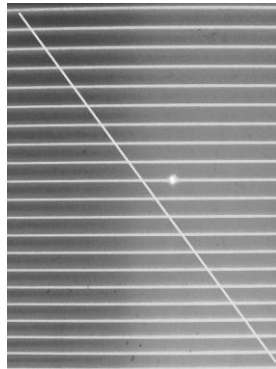
The points in the ROI (Region of Interest)  $R$  are selected for output region  $S$  if the grey value in point  $(i, j)$  is between predefined values  $g_{min}$  and  $g_{max}$ .

In particular, we find that the input image has bright lines and a dark background, as shown in **Figure 3.5**. In the camera view, laser projection produces a very high intensity compared to the background. Any intensity value above a selected threshold  $th$  can be regarded as line (marked as “1”). The rest are background (marked as “0”). Hence, we can simplify **Equation 4-1** as below:

$$S = \begin{cases} 0 & g_{r,c} < th \\ 1 & g_{r,c} \geq th \end{cases} \quad (r, c) \in R \quad (4-2)$$

In order to distinguish between lines and background, a suitable threshold value  $th$  should be determined for converting the original image to a binary line map.

There have been various techniques developed for adaptive threshold selection. Details of these methods can be found at [65]. **Figure 4.1** shows the selected ROI from **Figure 3.5** with the raw image data.

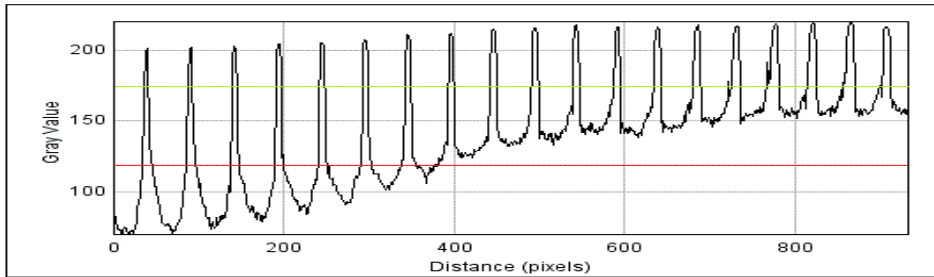


**Figure 4.1** Select ROI with raw image data

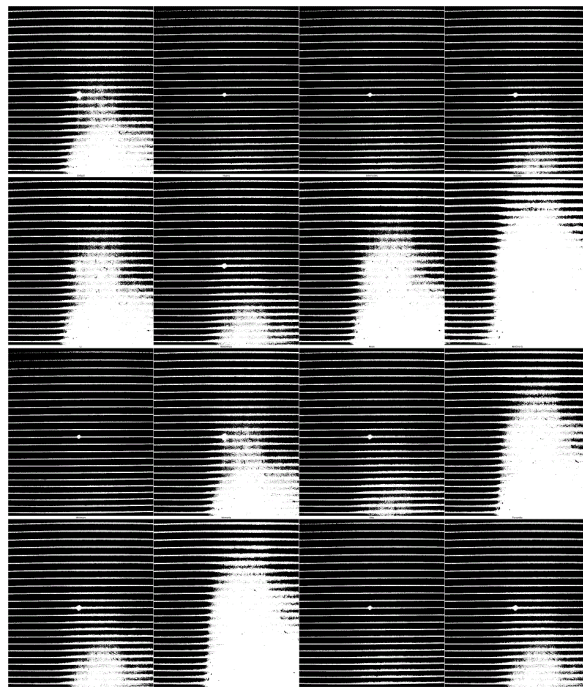
Unfortunately, it is quite difficult to find a suitable threshold value for the whole image.

The reason is that light illumination and surface reflectivity all impact on the signals acquired by a camera. Obviously, in **Figure 4.1**, the top left part is darker than the

bottom right because sun light is projected from the bottom right. In order to have a clear view about the brightness change from top left to bottom right, we plot the perceived brightness from top left to bottom right. **Figure 4.2** presents the result.



*Figure 4.2 Plot chart from raw image data*



(a) IsoData ignore white and black (b) Huang (c) Intermodes (d) IsoData

(e) Li (f) MaxEntropy (g) Mean (h) MinError(i)

(j) Minimum (k) Moments (l) Otsu (m) Percentile

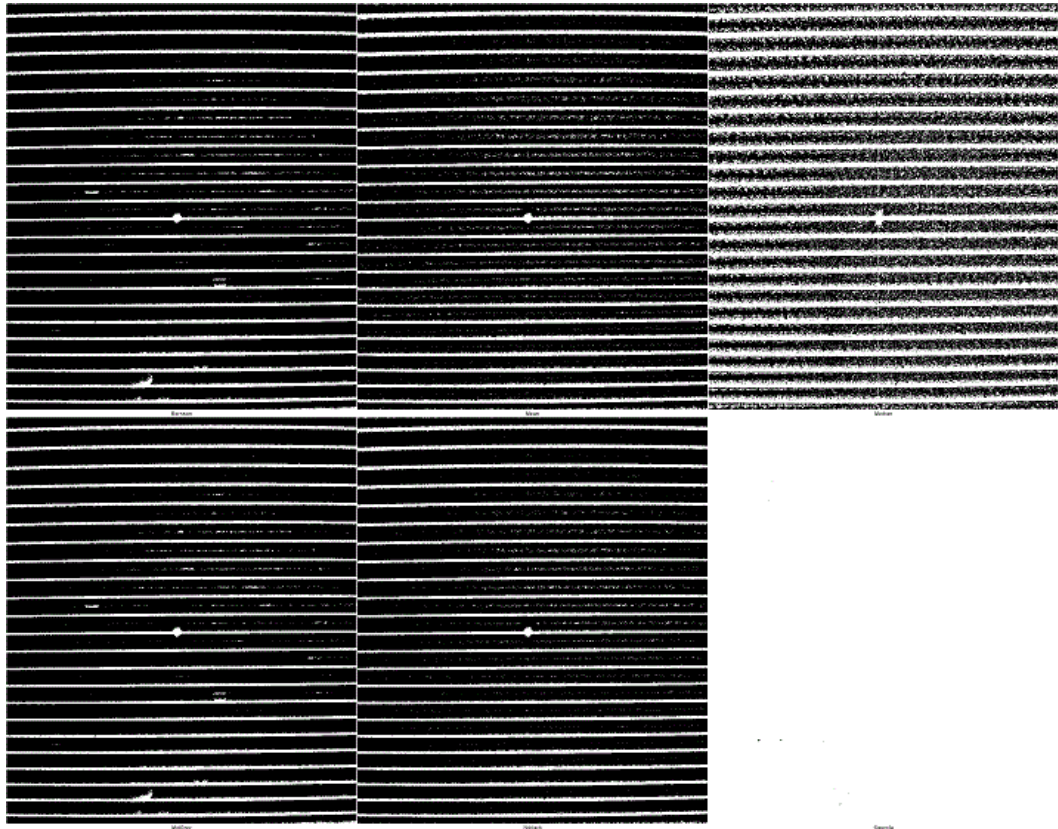
(n) RenyiEntropy (o) Shanbhag (p) Triangle (q) Yen

*Figure 4.3 Global threshold results by using different methods*

From this plot chart, we can find a large grey value disparity in different parts of the image. If a threshold value is selected to be a green line, the left part will not have enough pixels or even missing pixels. Usually, three or four pixels are necessary for produce a line. If there are too few pixels, it will not be clear whether it is a line or just noise. Though it may be a line, it could be an unstable line which cannot be trusted. On the other hand, if the threshold value is selected as the red line, the lines on the left can be identified those on the right are totally missed. That means a global threshold is not applicable for the whole image. **Figure 4.3** shows the segmentation results obtained by using different threshold methods. Details of all these methods can be found at [66]. The results demonstrate the failure of applying global threshold values.

One solution is to use different thresholds in different regions of the image. This method is called the local threshold method. The local threshold method is to find a threshold value in a selected window on the image. The threshold is applied to this window for segmentation. The window is then shifted to its neighbouring area until the whole image is scanned. In [66], local threshold methods were reviewed. **Figure 4.4** gives the results of using local threshold methods. Lines wherever they locate are well detected.

However, local threshold methods may introduce additional noise. The noise is usually found in the dark area between bright objects. In these areas, the threshold value in a selected window is usually very low so that noise may be easily mismarked as an object. These noises can be removed by applying a filter operation before segmentation. **Figure 4.5** gives the result of applying a 5x5 mean filter.



(a) *Bernsen* (b) *Mean* (c) *Median*

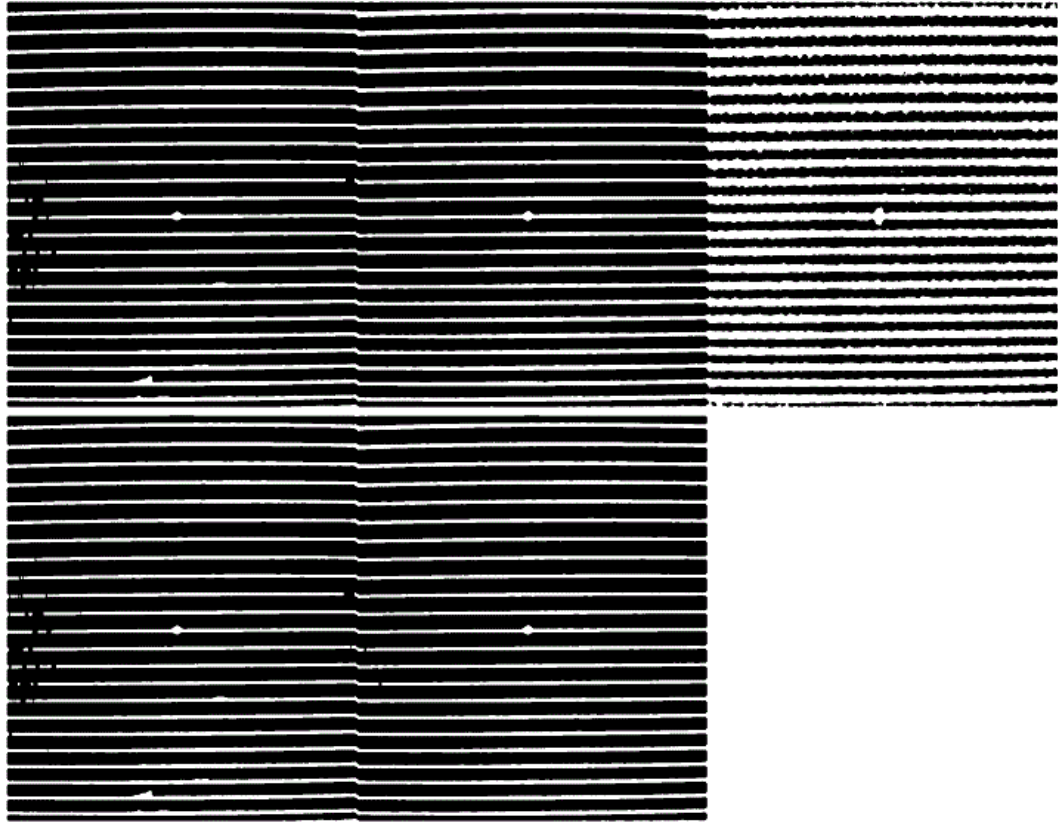
(d) *MidGrey* (e) *Niblack* (f) *Sauvola*

**Figure 4.4** *Local threshold results by using different methods*

After filtering, local threshold methods present a high resolution results compared with global ones. Among the results shown in **Figure 4.5**, the Mean and Niblack methods demonstrate the best performance. A very important index for the selection of threshold methods is the processing time. Unlike global threshold methods, the local threshold methods apply a local threshold operation in each window as it moves over the whole image. The local threshold methods thus take much more time for processing.

Comparing these two methods, see **Equation 4-3, 4-4**.





*Figure 4.5 Local threshold results with mean filter*

Mean:

$$S = \begin{cases} 0 & g_{i,j} < \text{mean} \\ 1 & g_{i,j} \geq \text{mean} \end{cases} \quad (i,j) \in R \quad (4-3)$$

Niblack:

$$S = \begin{cases} 0 & < (\text{mean} + k \cdot \text{sd}) \\ 1 & \geq (\text{mean} + k \cdot \text{sd}) \end{cases} \quad (i,j) \in R \quad (4-4)$$

where  $sd$  is the standard deviation,  $k$  is a factor between 0 and 1.

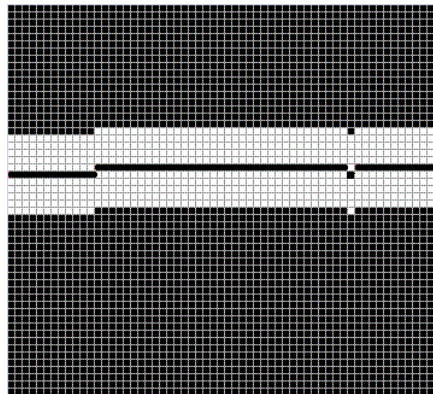
It is clear that the Niblack method needs to calculate the mean first and then the standard deviation which requires more operations for the calculation than the mean threshold method. This calculation needs to be carried out in every small window. This causes a significant increase in the processing time for a large image. And furthermore,

an unknown factor  $k$  needs to be predefined. The local mean threshold method is simple and experiments prove that it is suitable for this application.

From the above comparison and analysis, we decided to use the local mean threshold method with a mean filter as the segmentation method.

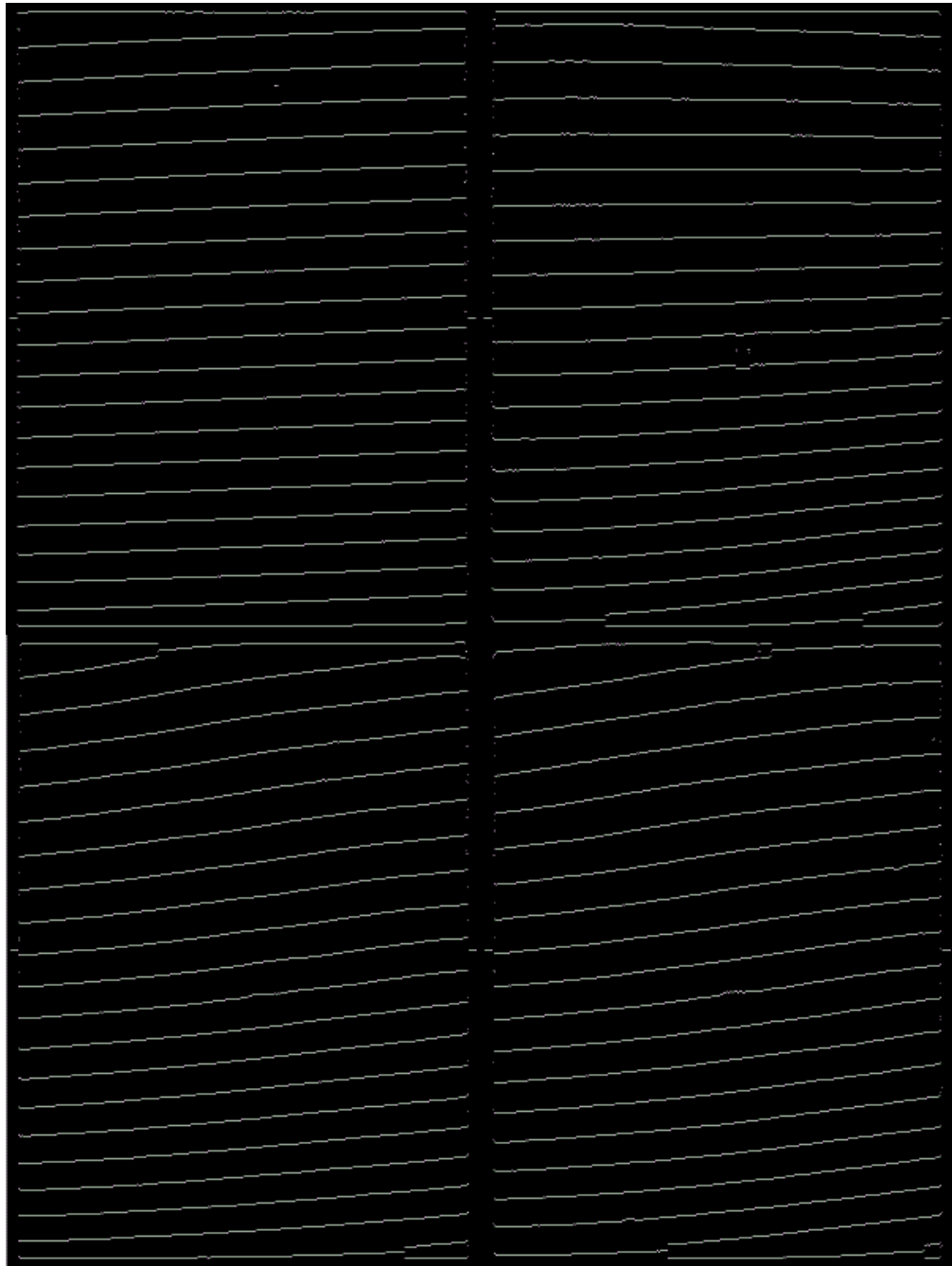
### ***4.3.2 Line Extraction***

After segmentation, clearer lines are presented in the image. But that is not enough for the calculation. We need to extract line geometry from the pre-processed image. That means we need to identify the coordinates of the line points. This thin line is expected to be one pixel wide. We defined the centre pixel along the vertical direction as the extracted position of a line. **Figure 4.6** is an example of the line extraction, where the line position is marked in centre.



***Figure 4.6 Line extraction definition***

**Figure 4.7** shows the extracted line images with different light illumination and different objects. These experiments show the robustness of this line extraction approach.



*(a) on flat plane (b) on an object*

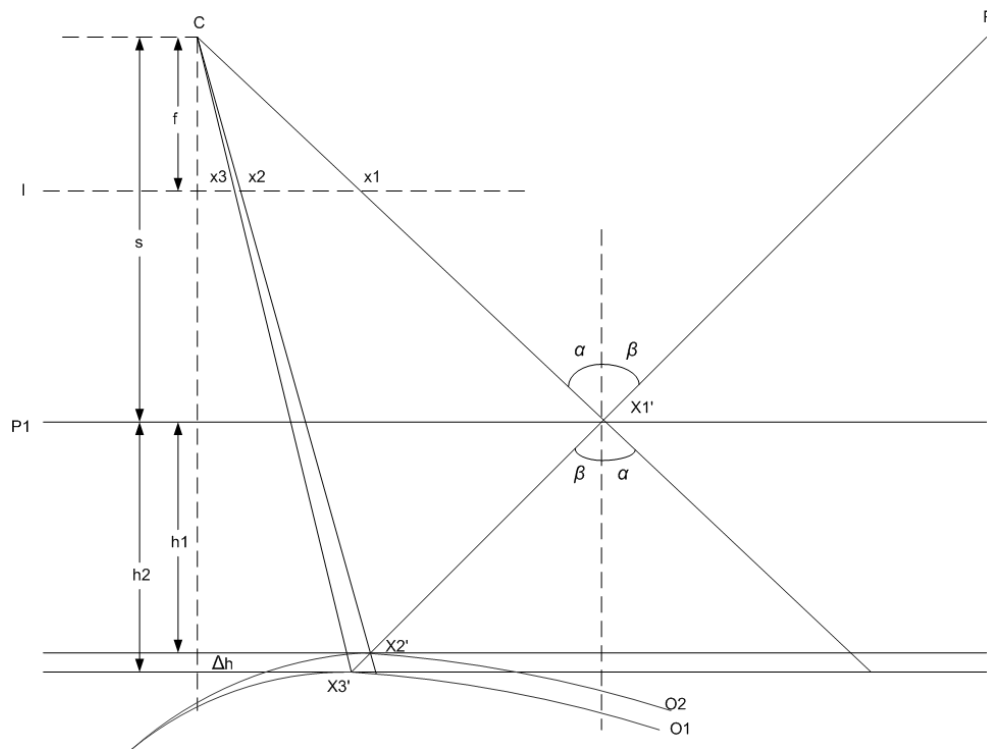
*(c) on an object with different height (d) on the same object with different light  
illumination*

**Figure 4.7** *Extracted line images*

## 4.4 Height Calculation

### 4.4.1 Height Calculation at One Point

After extraction of the lines, the next step is to find the height based on the deformation of the lines. As previously discussed in Section 2.3.1, **Equation 2-3** gives a height calculation model and **Equations 2-4,2-5,2-6,2-7** give a general calibration model. In particular, the height calculation model with line projection used in this application is described below:



**Figure 4.8** Height measurement model

$O1$  is a standard object.  $O2$  is an object to be measured with a slightly changed height.

$P1$  is the reference plane. In this application, we need to measure the height disparity  $\Delta h$ .

Before starting the measurements, calibration must be carried out, which is described in Section 2.3.1. After calibration the distance  $s$  and factor  $K$  can be calculated via

**Equation 2-6, 2-7.** So can rewrite as

$$s + h = \frac{h \cdot f}{K \cdot (x_1 - x_2)} \quad (4-5)$$

By solving **Equation 4-5**, height  $h$  can be calculated by its coordinate  $x$  on the image plane. See the equation below:

$$h = \frac{s \cdot K \cdot (x_1 - x)}{f - K \cdot (x_1 - x)} \quad (4-6)$$

Using **Equation 4-6** in **Figure 4.8**,  $\Delta h$  can be obtained by following equation:

$$\Delta h = \frac{s \cdot K \cdot (x_1 - x_3)}{f - K \cdot (x_1 - x_3)} - \frac{s \cdot K \cdot (x_1 - x_2)}{f - K \cdot (x_1 - x_2)} \quad (4-7)$$

That is the height calculation equation used in this application.

#### 4.4.2 Height Calculation along a Line

Last section described how to get height at one point. The next step is to find all the heights along a line.

**Equation 4-6** is the basic height calculation equation. When we calculate its derivation, we can get the following equation:

$$h = \frac{s \cdot K \cdot (x_1 - x)}{f - K \cdot (x_1 - x)}$$

$$\Rightarrow dh = \frac{[s \cdot K \cdot (x_1 - x)]' \cdot (f - K \cdot (x_1 - x)) - s \cdot K \cdot (x_1 - x) \cdot [f - K \cdot (x_1 - x)]'}{[f - K \cdot (x_1 - x)]^2} \cdot dx$$

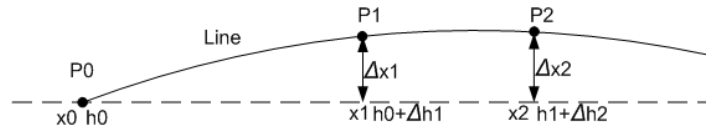
$$\begin{aligned}
&= \frac{[s \cdot K \cdot (x1-x)]' (f \cdot K \cdot (x1-x) - s \cdot K \cdot (x1-x)) [f \cdot K \cdot (x1-x)]'}{[f \cdot K \cdot (x1-x)]^2} \cdot dx \\
&= \frac{-s \cdot K \cdot (f \cdot K \cdot (x1-x) - s \cdot K \cdot (x1-x)) \cdot x}{[f \cdot K \cdot (x1-x)]^2} \cdot dx \\
&= \frac{2 \cdot s \cdot K^2 \cdot x \cdot s \cdot K \cdot f}{[f \cdot K \cdot (x1-x)]^2} dx \tag{4-8}
\end{aligned}$$

In discrete domains, we can rewrite the above equation to

$$\Delta h = \frac{2 \cdot s \cdot K^2 \cdot x \cdot s \cdot K \cdot f}{[f \cdot K \cdot (x1-x)]^2} \Delta x \tag{4-9}$$

For the reason that the projected lines are horizontal parallel lines,  $x1$  which is the reference coordinate on the base plane is a constant along a line. So, height difference  $\Delta h$  depends on the pixel coordinate disparity  $\Delta x$ . That means the related height at one point can be calculated by integration from a reference point. That makes it possible to get all the heights along the line relative to a common reference point.

**Figure 4.9** shows three points on one line as an example. Considering  $P0$  is the start reference point and has known height  $h0$ ,  $x0$ ,  $x1$  and  $x2$  are pixel deviations from the reference line due to variable height. To avoid confusion, the previously used variable  $x1$  in **Equation 4-9** is redefined as  $x'$ .



**Figure 4.9 Height calculation along line**

We get:

$$\Delta h1 = \frac{2 \cdot s \cdot K^2 \cdot x0 \cdot s \cdot K \cdot f}{[f \cdot K \cdot (x' - x0)]^2} \Delta x1$$

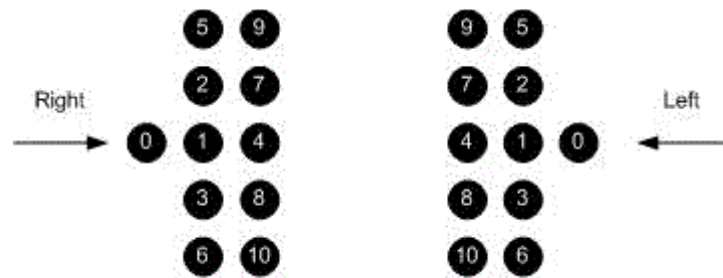
$$\Delta h_2 = \frac{2 \cdot s \cdot K^2 \cdot x_1 \cdot s \cdot K \cdot f}{[f \cdot K \cdot (x' - x_1)]^2} \Delta x_2 = \frac{2 \cdot s \cdot K^2 (x_0 + \Delta x_1) \cdot s \cdot K \cdot f}{[f \cdot K \cdot (x' - x_0 - \Delta x_1)]^2} \Delta x_2$$

$$\Rightarrow \Delta h_n = \frac{2 \cdot s \cdot K^2 (x_0 + \Delta x_{n-1}) \cdot s \cdot K \cdot f}{[f \cdot K \cdot (x' - x_0 - \Delta x_{n-1})]^2} \Delta x_n$$

(4-10)

At any point  $n$ , the height can be calculated by its previous point  $n-1$ . This height calculation can be spread from one starting point to the whole line. This relies on reliable line tracking. A searching method to find neighbouring pixels along a line is necessary because the extracted lines may not be ideal but contain noise. Firstly, they may have breaks somewhere. Secondly, they may have short branches due to noise. Finally, some pixels which are not on the line but very close to the line may be noise pixels.

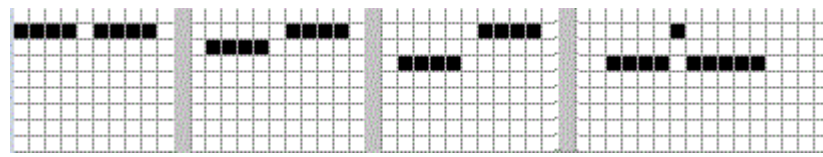
Here we propose a searching method to solve these problems. A searching order is defined as shown in Figure 4.10, where the numbers indicate the sequence to follow.



**Figure 4.10** Line searching sequence

Two searching directions are defined in **Figure 4.10**. If “0” represents the current point, try to find the next point at position “1”. If it cannot find the point at position “1”, then try to find it at position “2”. Do this operation until the next point is found or the number reaches “10”. If no point can be found in these 10 positions, the line has reached the end.

This searching approach can handle the following situations:



(a) Broken lines

(b) Unreliable points

**Figure 4.11 Broken lines and unreliable points**

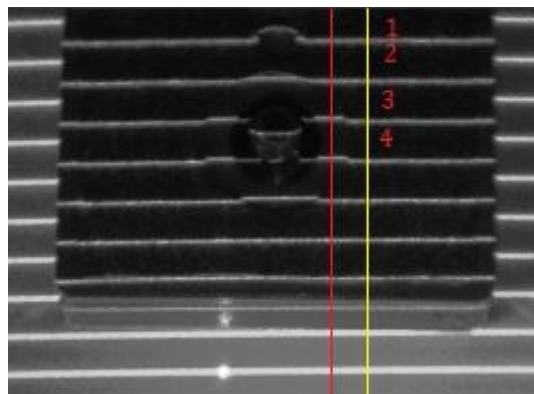
Broken points can be skipped to continuously track the line. Noise points are removed.

To remove the short lines due to noise, we define an acceptable line length. Any line shorter than this length will be removed. Therefore, with the height calculation equations and the searching scheme, a height profile along the line can be obtained.

#### **4.4.3 Height Calculation in the Whole Image**

In the last section, we got the heights along the line. Each line has a reference start point. So those heights along the line are relative heights to its reference point. To have a 2D height map, we need to find out the relative height between lines so that the lines are shifted to fit the surface. Here we present a method for finding the relative heights by calculating the period along the vertical direction.

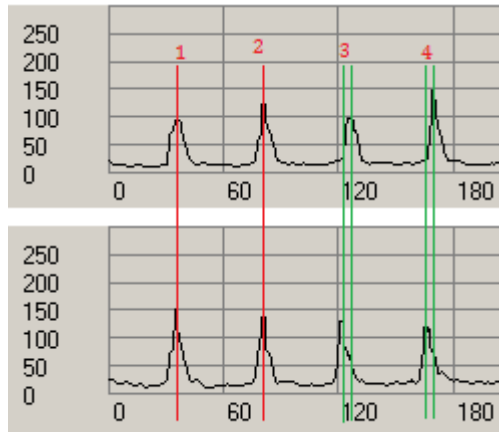
**Figure 4.12** gives a sample image clip when an object is put into the field of view.





**Figure 4.12** Sample image clip with an object

We plot brightness changes along the vertical lines in red and in yellow in **Figure 4.13**. We find that the period of the signals is constant for projection on a parallel plane. The period changes when these two lines are on different height planes. From **Figure 4.13**, we can see the relative heights in **Figure 4.12** where the red line is marked. 1st and 2nd lines are on the same height plane; 2nd and 3rd lines are on a different height plane; 3rd and 4th lines are on the first height plane. A shortened period means the height is increased and a lengthened period means the height is decreased. The height disparity between the 2nd and 3rd lines can be calculated by the variation in the period.



(a) Plot chart along yellow line in figure 4.12

(b) Plot chart along red line in figure 4.12

**Figure 4.13** Plot charts along vertical direction

In order to prove this result, a period calculation model is given in **Figure 4.14**. Two lines are projected onto a flat plane and we get its period on the camera image plane marked as “ $a$ ”. We change the height of the flat plane with a distance marked as “ $h$ ”. Then we get its period “ $b$ ” on the image plane.  $f_c$  and  $f_p$  are the focus lengths of the

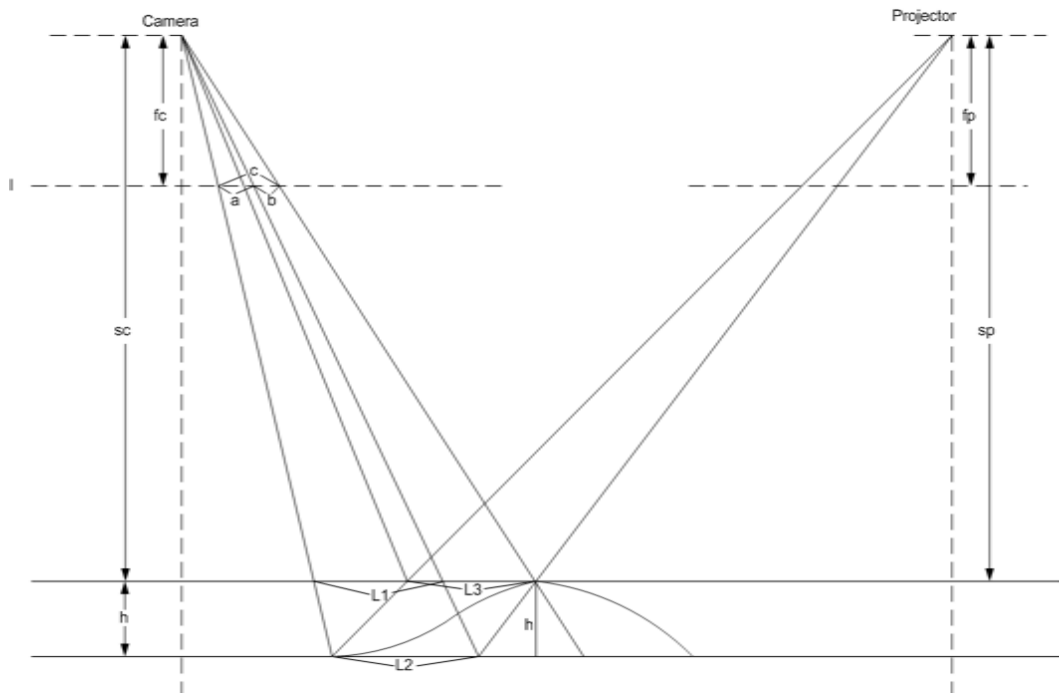
camera and the projector.  $Sc$  and  $Sp$  are the distances from the camera and the projector centres to the reference plane.

Firstly, to prove the periods of the lines on the same height plane are equal. According to triangulation, we get following equations:

$$\frac{L1}{L3} = \frac{a}{b'}, \quad \frac{L1}{L2} = \frac{sc}{sc+h}, \quad \frac{L3}{L2} = \frac{sp}{sp+h} \quad (4-11)$$

$$\Rightarrow \frac{a}{b} = \frac{sc(sp+h)}{sp(sc+h)} \quad (4-12)$$

We assume the camera and projector are located at the same height. So we get  $sc$  is equal to  $sp$ . Hence  $a$  equals to  $b$ . This proves that the two lines are on the same height plane, if their period in image plane is the same.



**Figure 4.14** *Period calculation model*

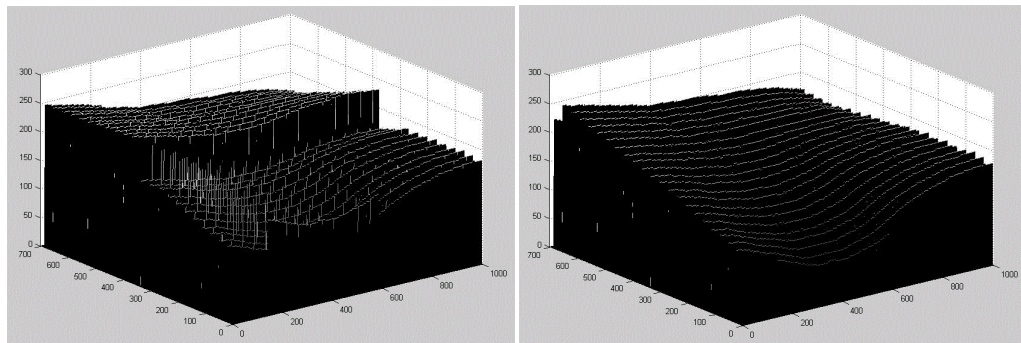
Secondly, we need to find out the height difference related to the period variation. The red line denotes an object. We find the line position is changed. The period in the image

plane becomes “ $c$ ”. Though here we call it period disparity, it is still the pixel coordinate variation. So **Equation 4-6** can also be applied to this model. According to the current definition, height  $h$  can be calculated as below:

$$h = \frac{sc \cdot K \cdot (c-a)}{fc - K \cdot (c-a)} \quad (4-13)$$

This equation proves that we can find the height information from the change in the period.

Through calculating the relative height between lines from the period variation, we can shift all the lines with different reference heights to a surface with a common reference height. **Figure 4.15** shows the reference-unifying process, from independent lines to a continuous surface with a common reference.



*(a)Uncorrected lines (b) corrected lines*

**Figure 4.15** Line correction by period calculation

## 4.5 3D reconstructions

### 4.5.1 Introduction of Kalman Filter

To achieve optimal 3D surface fitting, a spatial Kalman filter is applied as a fusion mechanism taking into account neighbouring measurements. The Kalman filter is a set

of mathematical equations that provides an efficient computational means to estimate the state of a process in a way that minimizes the mean square error [66,67,68]. The Kalman filter uses a system's dynamic model. The Kalman filter has two distinct phases: Predict and Update. The predict phase uses the state estimate from the previous time step to produce an estimate of the state at the current time step. In the update phase, the current prediction is combined with current observation information to refine the state estimate. This corrected estimate is termed as a posterior state estimate. The Kalman filter can be used to obtain an optimal measurement from the fusion of several sensor measurements.

The Kalman filter is described in the equations below:

Predict:

$$\text{State estimate:} \quad \hat{x}_{k|k-1} = F_k \hat{x}_{k-1|k-1} + B_k u_k$$

$$\text{Covariance estimate:} \quad P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + Q_k$$

$$\text{Measurement model:} \quad z_k = H_k x_k + V_k, \quad V_k \sim N(0, R_k)$$

$$\text{Measurement residual:} \quad \tilde{y}_k = z_k - H_k \hat{x}_{k|k-1}$$

$$\text{Residual covariance:} \quad S_k = H_k P_{k|k-1} H_k^T + R_k$$

$$\text{Optimal Kalman gain:} \quad K_k = P_{k|k-1} H_k^T S_k^{-1}$$

(4-14)

Update:

$$\text{Update state estimate:} \quad \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k$$

Update covariance estimate:  $P_{k|k}=(1-K_kH_k)P_{k|k-1}$

(4-15)

Where

$$E[x_k-\hat{x}_{k|k}]=0$$

$$E[x_k-\hat{x}_{k|k-1}]=0$$

$$E[\tilde{y}_k]=0$$

$$P_{k|k}=\text{COV}(x_k-\hat{x}_{k|k})$$

$$P_{k|k}=\text{COV}(x_k-\hat{x}_{k|k-1})$$

$$S_k=\text{COV}(\tilde{y}_k)$$

(4-16)

In the equations,  $x_{n|m}$  represents the estimate of  $x$  at time  $n$  given observations up to, and including at time  $m$ .

After a certain time of iterative calculation using the equations list above, the final state will reach a dynamic balance around the true state within a desired tolerance.

#### **4.5.2 Spatial Kalman Filter for 3D Surface Fitting**

The Kalman filter is usually used as a temporal filter for signal processing. A spatial Kalman filter is proposed in this thesis. In Section 4.4.2, we described how the height is calculated by searching along a line and calculated from the period time. These height calculation methods are based on calculating the line deformation due to height change. No matter what kind of methods we are using, any measurement may be disturbed by

noise. Though the method to generate a 3D surface is described in Section 4.4.3, this generated 3D surface contains a lot noise. So we considered using a spatial Kalman filter to reduce the noise and finally get an optimized surface. This optimized surface can be defined to have a minimal variance.

This spatial Kalman filter, unlike a temporal Kalman filter, is generated in a 2D space by taking into account neighbouring estimates. The height of one point can be adjusted by its neighbours. And on the other hand, it is also the neighbouring point of its neighbours. That means this point can also adjust its neighbours' height. This mutual effect can spread over the whole image, and it results in a dynamic balance. All the heights will tend to the optimal height i.e. the height with a minimal variance.

In order to develop a Kalman filter to produce the optimal estimate of the height, we make the following definition.

$$\textit{System model: } h_0(k_i)=h_0(k) \quad (4-17)$$

Where  $k$  is the present point,  $k_i$  is its neighbouring point. We consider the surface is a continual and smooth surface. The height of one is considered to be the same as its neighbours'. That we have the priori estimate and variance as below:

$$\textit{Priori Estimate: } \hat{h}_0(k)=h_0(k_i)$$

$$\textit{Priori Variance: } P(k)=P(k) \quad (4-18)$$

The a priori estimate should be the same as the present height. And prior variance can have a custom defined initial value. This value will be adjusted automatically as the calculation continues.

According to the previous sections, we have the height measurement model:

$$z_0 = h_0 + \frac{2s \cdot K^2 \cdot x \cdot s \cdot K \cdot f}{[f \cdot K \cdot (x1 - x)]^2} \cdot \delta + n_{0,i} \quad (\text{By searching along the line}) \quad (4-19)$$

$$z_0 = h_0 + \frac{sc \cdot K \cdot \delta}{fc \cdot K \cdot \delta} + n_{0,i} \quad (\text{By period calculation}) \quad (4-20)$$

where  $z_0$  means the measurement value of the present point.  $h_0$  is the height of the present point.  $\delta$  is the measurement value which is the deviation of the pixel coordinates. We define  $n_{0,i}$  as measurement noise with a distribution of  $N(0, R)$ .

Then Kalman gain can thus be calculated as

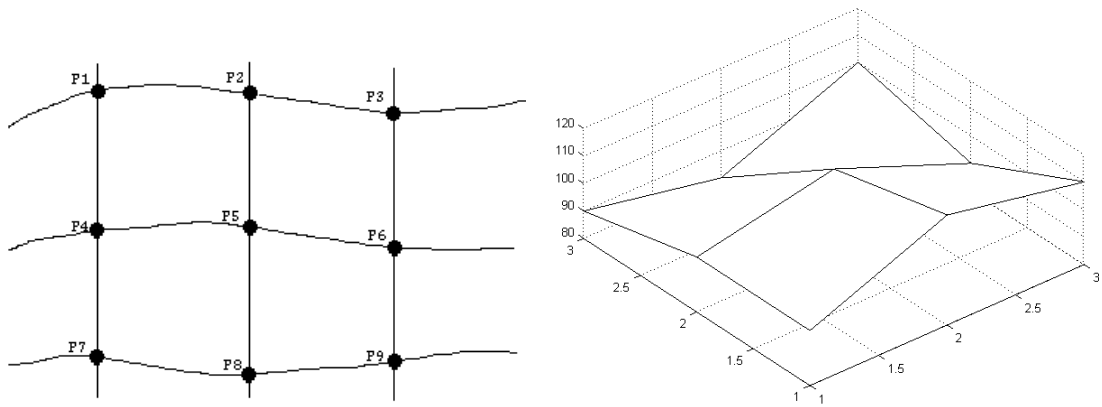
$$K(k) = P(k) (P(k) + R)^{-1} \quad (4-21)$$

The posterior estimate from  $h_0$  can be obtained as

$$\text{Posterior Estimate: } \hat{h}_0(k) = \hat{h}_0(k) + K(k) (z_0 - \hat{h}_0(k)) \quad (4-22)$$

$$\text{Posterior Variance: } P(k) = (1 - K(k)) P(k) \quad (4-23)$$

**Figure 4.16** gives an example for this calculation. In this example, we have a net with 9 points in it. Furthermore, for the clear view we plot these 9 points into a 3D surface. Assume the heights of these points have been calculated by the height calculation method described in Section 4.4. According to the measurement model, the height measurement value of the centre point P5 can be obtained via P4 and P6 by **Equation 4-19**. On the other hand, it also can be obtained via P2 and P8 by **Equation 4-20**.



**Figure 4.16** Points on the lines and the 3D surface

As an example to show how the Kalman filter is used, we consider adjusting the height of  $P5$  with respect to its neighbour  $P2$ . Using **Equations 4-17 to 4-23** we can apply the Kalman equations in the following steps.

*Step 1: Initialize the first prior estimate.  $\hat{h}_0(5) = h_0(2)$ .*

*Step 2: Initialize prior variance.  $P_0^-(5)$ . For the first time calculation, this value can be defined as an initial value.*

*Step 3: Calculate measured value.  $z_0(5) = h_0 + \frac{sc \cdot K(x_5 - x_2)}{fc \cdot K(x_5 - x_2)}$ .  $x_5$  and  $x_2$  are the  $x$  coordinates of  $P5$  and  $P2$ .*

*Step 4: Calculate Kalman gain.  $K(5) = \frac{P_0^-(5)}{(P_0^-(5) + R)}$ .*

*Step 5: Update posterior estimate and variance.*

$$\hat{h}(5) = \hat{h}_0(5) + K(5) \left( z_0(5) - \hat{h}_0(5) \right), P(5) = (1 - K(5)) P_0^-(5).$$

That height of  $P5$  is adjusted by its neighbouring point  $P2$ . We can do the same calculation to adjust the height of  $P6$  by  $P5$ . Obviously,  $P3$  can be adjusted by  $P6$ . And this adjustment can go back to  $P2$  from  $P3$ . That is a recursive calculation method.

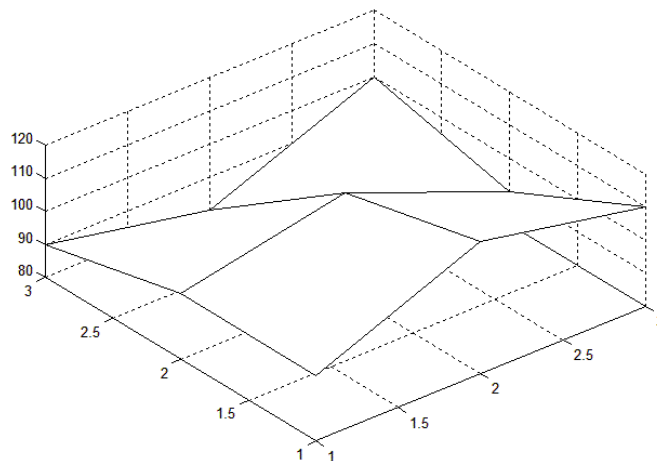


Points affect each other. The Kalman filter ensures the reduction of noise. And the final result is that all the points lead to a dynamic stable status with an acceptable tolerance. This status is the optimized surface. **Table 4.1** shows the original measured heights and optimized heights after applying the Kalman filter.

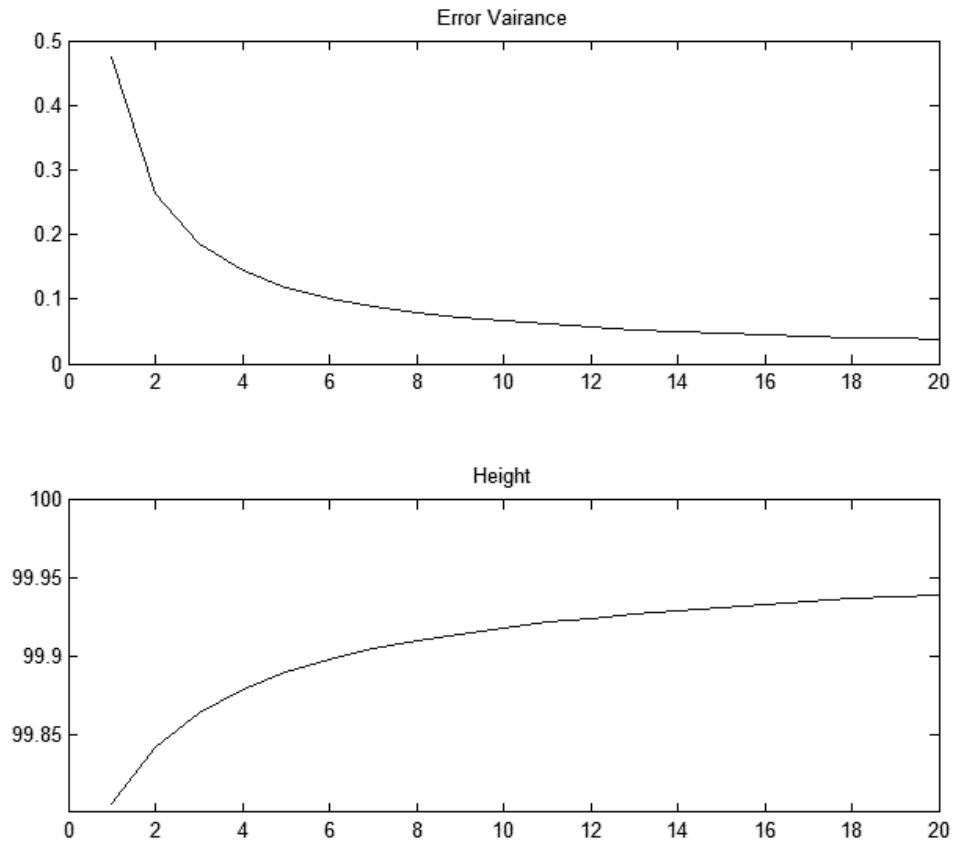
**Table 4.1 Data Result after Kalman filter**

Original			Iterate Once		
100	120	110	99.81	119.77	109.93
100	110	90	100.03	110.05	90.07
90	80	100	90.27	80.26	100.10
Iterate Twice			Iterate Five times		
99.84	119.83	109.91	99.89	119.88	109.92
100.01	110.02	90.03	100.00	110.00	90.01
90.18	80.18	100.10	90.11	80.11	100.08
Iterate 10 times			Iterate 20 times		
99.92	119.91	109.94	99.94	119.94	109.95
100.00	110.00	90.00	100.00	110.00	90.00
90.07	80.08	100.06	90.05	80.06	100.05

**Figure 4.17** gives the 3D surface after applying the Kalman filter. **Figure 4.18** plots the height of point  $P1$  and the error variance of these 9 points at different iteration times. It is clear that the heights tend to a fixed value while the error variance tends to a minimum value. This fixed value of the height is the optimal height at this point and we can regard this as the true height.



**Figure 4.17 3D surface after applying the Kalman filter**



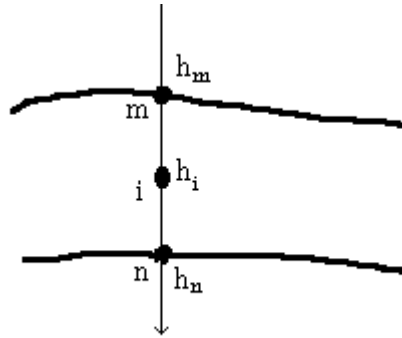
**Figure 4.18 Error variance and height plot chart**

### **4.5.3 Surface Generation**

After Kalman filtering, when we get all the heights along the lines, then we can generate the 3D surface by interpolation. An interpolation method is used to estimate values between known values. Here, heights along the lines have already been obtained. Interpolation should be done between lines. The height along the line is calculated by searching the disparities along the line. These heights can form a continuous curve. Hence, applying a linear interpolation method is enough to generate a smooth surface. This linear interpolation method can be defined as below:

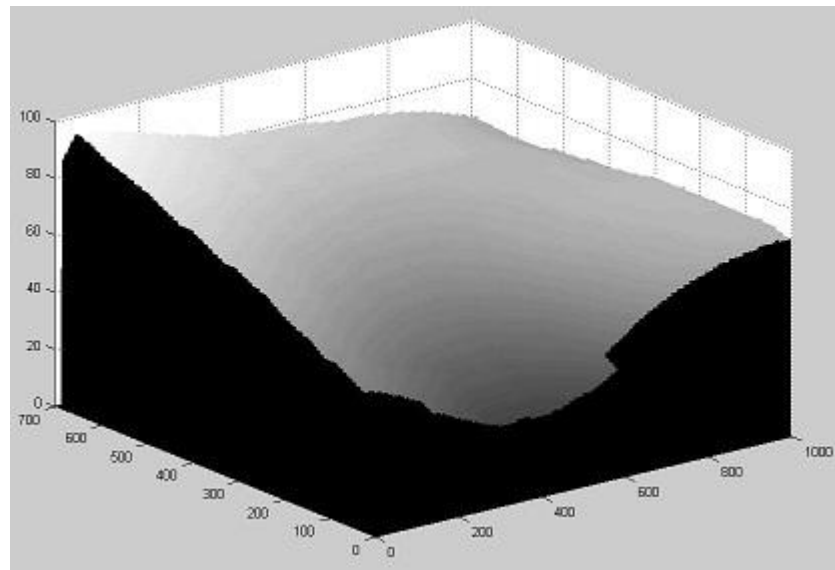
$$h_i = \frac{i-m}{n-m} \cdot (h_n - h_m) + h_m \quad (4-24)$$

$h_m$  is the height on one line with vertical coordinate  $m$ .  $h_n$  is the height on the next line with vertical coordinate  $n$ .  $h_i$  is the height we need to find at vertical coordinate  $i$ . See **Figure 4.17**.



*Figure 4.17 Height interpolation*

By calculating all the heights between the lines, we can generate the 3D surface. **Figure 4.18** shows the surface result after the interpolation is applied on **Figure 4.15 (b)**.



*Figure 4.18 3D surface after interpolation*

## 4.6 Summary

In this chapter, the detailed approach developed in this thesis has been presented. The local mean threshold method gives a robust segmentation that makes line extraction more accurate and less effected by noise. The line extraction method ensures the generation of an image with lines having one pixel width. The line with one pixel width helps to find the vertical coordinates of a point on the line, which indicates the deformation of a straight line and is taken as the measurement of the height.

Height is calculated from a height calculation model, by considering coordinate disparity. Calibration is carried out for defining the model. In brief, the height calculation model is to find the height profile in the pixel domain and the calibration model links the pixel domain with the real world. For this purpose, we propose a line searching method incorporating a spatial Kalman filter to reduce the noise and estimate an optimized height map.

So far, we have got the whole algorithm to generate a 3D surface from a line projection image for a 3D surface measurement. The next chapter will discuss the final prototype system and its experimental evaluation.

# Chapter 5

## Software Implementation and Experiments

## **5.1 Introduction**

The approach and algorithm for 3D measurement have been presented in the last chapter. The software to support the measurement algorithm will be developed in this chapter. An introduction to this software is given in Section 5.2. In Section 5.3, experiments are carried out to evaluate the method in terms of robustness, precision and real time performance. The approach is verified by the practical experiments, which simulate a real application.

## **5.2 Developing the Software for 3D Measurement**

For the quality control of car-bodies produced by stamping machines, a software tool is developed namely “SVR-3D V1.0” (SHRIS Vision Research – 3D, Version 1.0). This software is built up with three levels. The first level is the basic processing level. It is constructed using a proprietary vision library developed in this research programme. This library contains camera IO (inputs and outputs) procedures, low level image processing functions and some basic common controls. Above this vision library, a GUI (graphic user interface) is developed as the second level. This GUI is a friendly interface for end users. At the top level, the product quality control level is developed. It tracks products and records the status of all the measured products. According to the measured status, 3D deformation is monitored. It is proposed that an alarm is triggered when defects and wear in the stamping tool exceed threshold values. This will be the most valuable function for end users. Replacing stamping tools in time ensures a consistent high quality of production.

Besides the software relevant to 3D measurement, there is another component that interfaces the vision system with the control unit. The control unit could be a robot

manipulator or a PLC (Programmable Logic Controller). Two possible methods have been provided. The first is through digital IOs which provide basic operation interfaces with dry contact signals. These digital IOs can be easily connected to present production lines by a few wires. The second is through a field bus that connects the control systems through the communication system. Considering that Siemens PLCs are widely used in the car manufacturing industry, we use Siemens OPC[69] (OLE (Object Linking and Embedding) for Process Control) as the link between a PC and the PLCs or robots. The OPC is a standard industry PC hosting technique. It was developed based on a standard field bus protocol Profibus[70] and Profinet[71]. This standard industry field bus enables other types of PLCs to be accessible by adding Profibus or Profinet interfacing cards [72,73,74,75].

**Table 5.1 IO definitions**

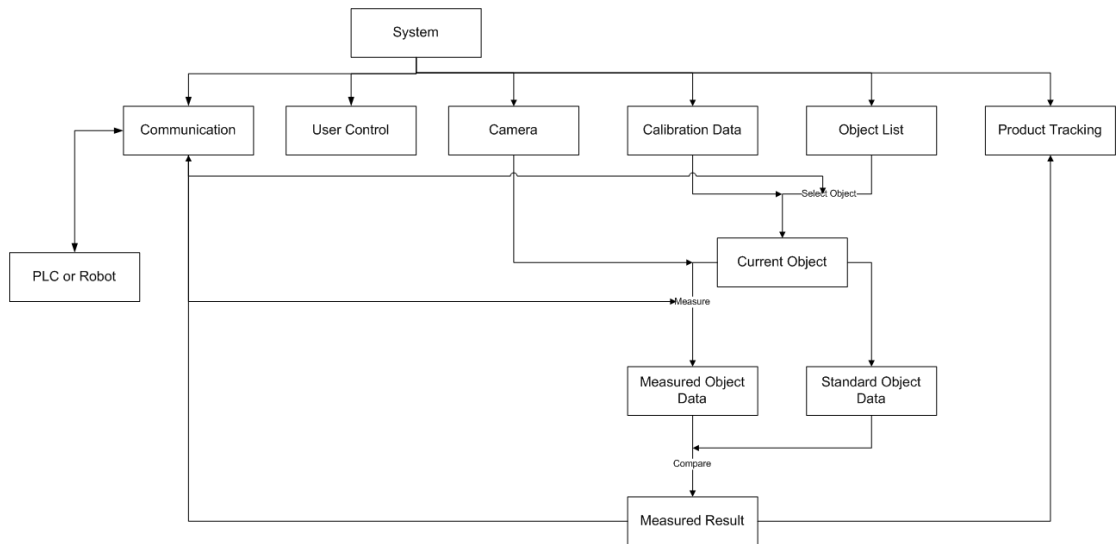
Signals	Digital IO	Profibus/Profinet	Meaning
Start camera	Input1	Input Byte 0 bit 0	1=Camera start capture
Stop camera	Input2	Input Byte 0 bit 1	0=Camera stop capture
Measure	Input3	Input Byte 0 bit 2	1=Start measure
Link Status	Input4	Input Byte 0 bit 3	1=Link Ok
Spare		Input Byte 0 bit 4	Spare
Spare		Input Byte 0 bit 5	Spare
Spare		Input Byte 0 bit 6	Spare
Spare		Input Byte 0 bit 7	Spare
Object Select No		Input Word 2	Current object index
Spare		Input Word 4	Spare
Spare		Input Word 6	Spare
Spare		Input Word 8	Spare

Ready	Output1	Output Byte 10 bit 0	1=Vision system ready
Measure Finished	Output2	Output Byte 10 bit 1	1=Measurement finished
Passed	Output3	Output Byte 10 bit 2	1=Good Product passed
Failed	Output4	Output Byte 10 bit 3	1=Bad Product failed
Current Mode		Output Byte 10 bit 4	1=Auto Mode,0=Manual Mode
Object Changed		Output Byte 10 bit 5	1=Object Changed
Spare		Output Byte 10 bit 6	Spare
Spare		Output Byte 10 bit 7	Spare
Spare		Output Word 12	Spare
Spare		Output Word 14	Spare
Spare		Output Word 16	Spare
Spare		Output Word 18	Spare

**Table 5.1** gives the definition of the IO signals. For the digital IOs, 4 inputs and 4 outputs are used. For the field bus, a communication packet with 10 receiving bytes and 10 sending bytes are defined. Besides the basic measurement functions, an object select signal is added. With those signals, the control unit can switch products by sending a command. That makes it possible to measure different products with one vision system without human involvement. Furthermore, sufficient spare signals are reserved for future uses.

The software processes are depicted by the flow chart in **Figure 5.1**:





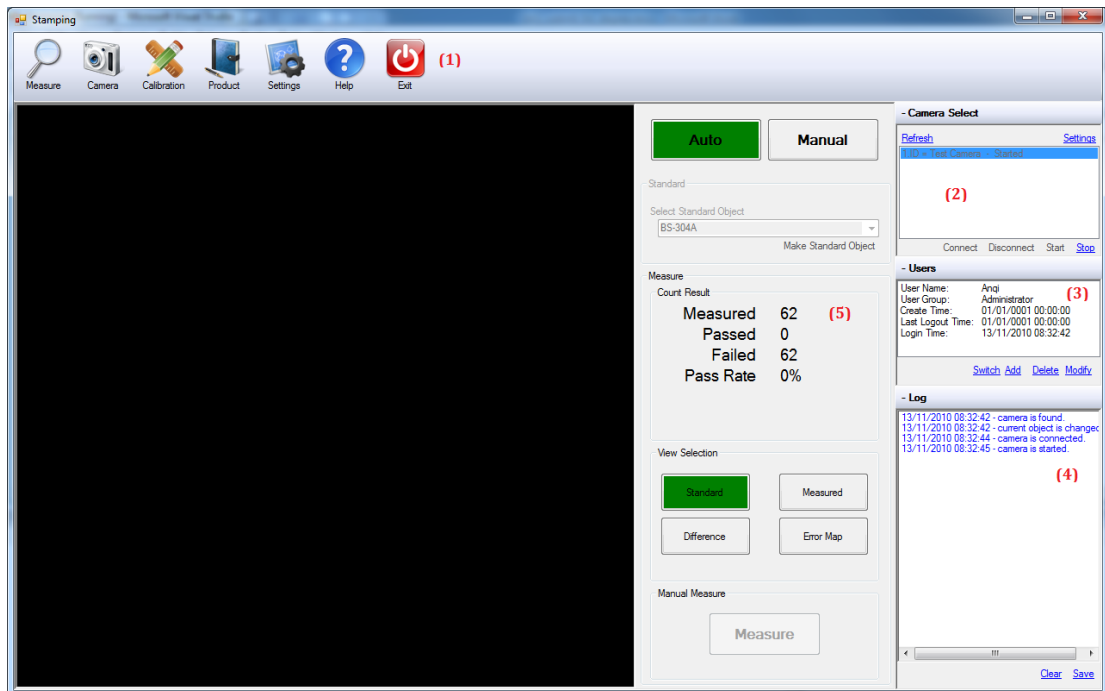
**Figure 5.1 Software running flow chart**

When the control unit (PLC or robot) sends the select object command, the object is selected from the stored object list as the current target object. This current object data contains standard object data, visual calibration data and other necessary data for measurement. When the measure command is received, 3D data will be generated from an image input and the calibrated parameters in the current object data. Then the comparison between the measured object data and the 3D template will be carried out. The result will be recorded in the quality control unit and feedback to the control unit via the field bus communication system.

### **5.2.1 Software overview**

The main screen is divided into five parts, see **Figure 5.2**. The top row, part (1), is the general menu giving access to different functionalities. The right column consists of three general control windows. The camera Control window (2) contains functions for image acquisition from the camera and the camera parameter settings. The user Control window (3) contains functions for switching, adding, deleting and modifying users. The log window (4) records all the events while the software is running. It helps to check the

production status and diagnose faults. The central part (5) is the main screen for functions selected from a general menu.



*Figure 5.2 Software overview*

### **5.2.2 User Rights Control**

This software is designed for industry applications. Usually, on a production line, users can be divided into three groups. The first is operators. They only need to know the basic operations for production. Only measurement and monitoring functions are accessible for this group. The second level is the maintenance level. In this level, users can do all that the operators can. Additionally, they can calibrate the system when the environment changes and sample a standard object when a new product is involved. Also they can change the IO access for the device modification. The third level is for administrators with the highest privilege. Users of this level have full access to all the functions. Compared with the maintenance group, they have extra access to the advanced parameter settings. Such as the camera internal settings, the software

presentation settings and they have the right to backup the production data. The user rights control is shown in **Table 5.2**.

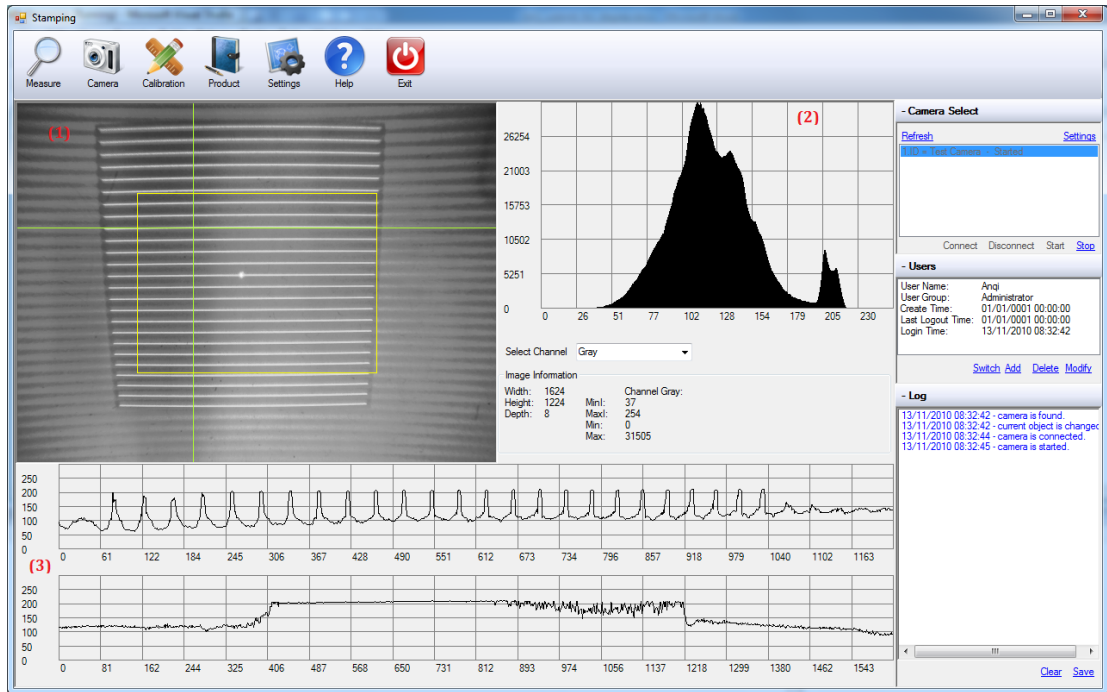
**Table 5.2 User rights control**

Items\User Group	Operators	Maintenance	Administrators
Camera control	View only	Full access	Full access
User control*	No access to others	No access to others	Full access
Log	View only	Full access	Full access
Measure	Full access	Full access	Full access
Calibration	No access	Full access	Full access
Camera view	View only	Full access	Full access
Product view	View only	View only	Full access
Settings	No access	Part of parameters	Full access
Help	Full access	Full access	Full access

\*Before any operation, user should login by entering a password. For the operators and maintenance group, they have no access right to other accounts. Only administrators can add, delete and modify user accounts.

### **5.2.3 Image Capturing**

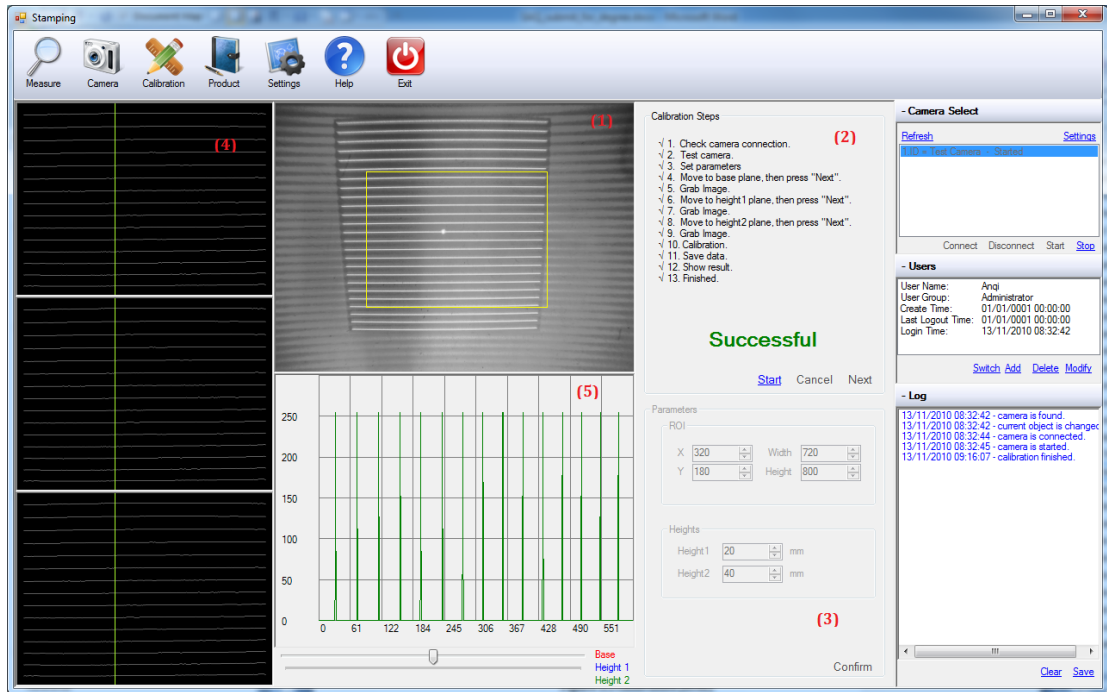
**Figure 5.3** shows the GUI for image capture. A real time view from the camera is displayed in the top left (1). In this view, a yellow rectangle is used to mark the ROI of the `current product. The brightness values along the two green lines are plotted at the bottom (3). These curves give a clear view of the deformation of the projected lines due to variable height. Furthermore, the histogram of a grabbed image is shown on the right beside the camera view (2). The histogram helps to evaluate the environment and the projection illumination. All these functions help the adjustment of the camera and the laser installation and the parameter settings during the system setup. And they also act as an aid for system maintenance.



*Figure 5.3 Camera monitor screen*

#### **5.2.4 Calibration**

Before starting to measure, the calibration must be carried out. **Figure 5.4** depicts the calibration screen. Top centre (1) shows the real time camera view. It helps to observe the changes of the light pattern caused by the presence of the calibration plane. Top right part (2) lists steps and operation instructions for completing the calibration. On the bottom side (3), some parameters can be entered as required during the calibration process. As previously discussed, calibration needs to take three images, which are the reference plane and flat planes with two different fixed heights. After calibration, these three images will be extracted into the line images and shown on the left (4). The user has a clear view of the quality of the extracted lines. Moreover, line data of these three images along the yellow line is displayed at the centre bottom (5). It helps to see the amount of line shifting. After the calibration calculation, all the parameters will be saved in a local file for the further use. Calibration only needs to be done once after the system is set up.

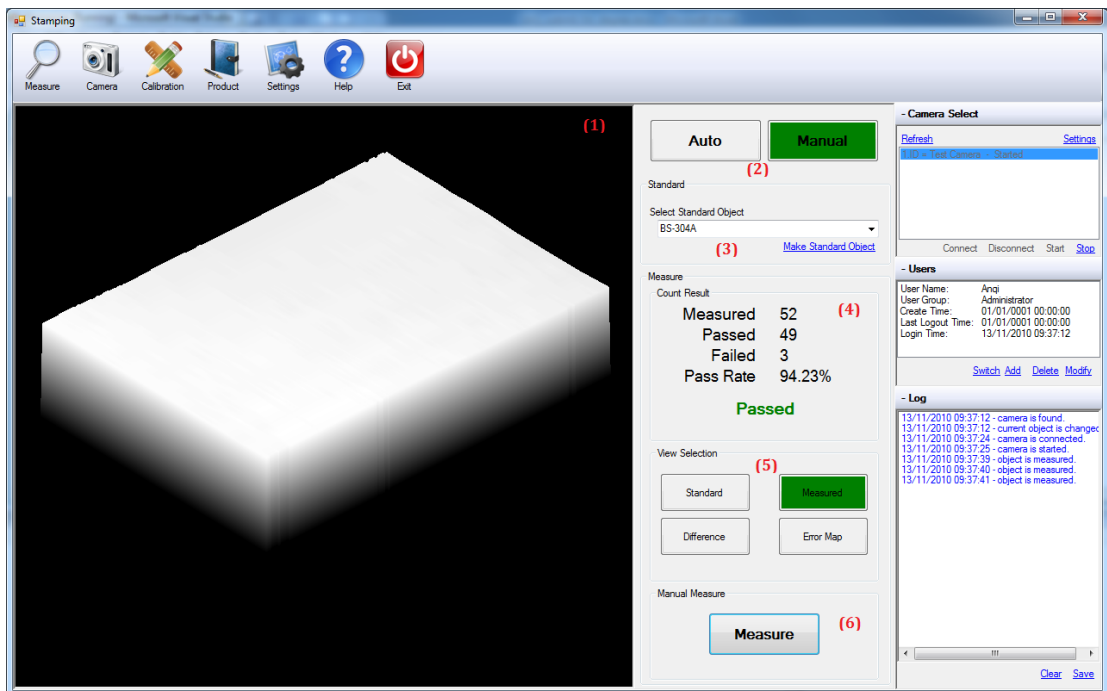


**Figure 5.3 Calibration screen**

### 5.2.5 3D Measurement

**Figure 5.4** is the measurement screen. The main window (1) is the 3D model constructed from the current measurement. For a better 3D view, the 3D model can be rotated and shifted to the desired orientation. The top right part (2) gives access to change the operation mode. In the auto mode, measurement is runs automatically under the control of a robot or a PLC through the digital IOs or a field bus. In the manual mode, the operator can do the measurements by clicking the button (6). A standard object should be measured manually. Below the mode selection buttons there is an object selector (3). An operator can select a standard object in the manual mode or switch between standard objects by receiving an object selection index from the communication system in the auto mode. One more important function in this area is to create a 3D template for a standard object. It defines a standard product. All the measurements are to compare the current measurement result with those of a standard object. In area (4), the measurement result of the current object is presented. The

measured passed and failed products are counted. A pass rate is calculated for monitoring the current production performance. Area (5) can select different 3D pictures displayed in the main window (1). The standard shows the standard object model; the measured shows the current measured model; the difference shows the difference between the measured and standard models; the error map highlights the areas with detected discrepancies.

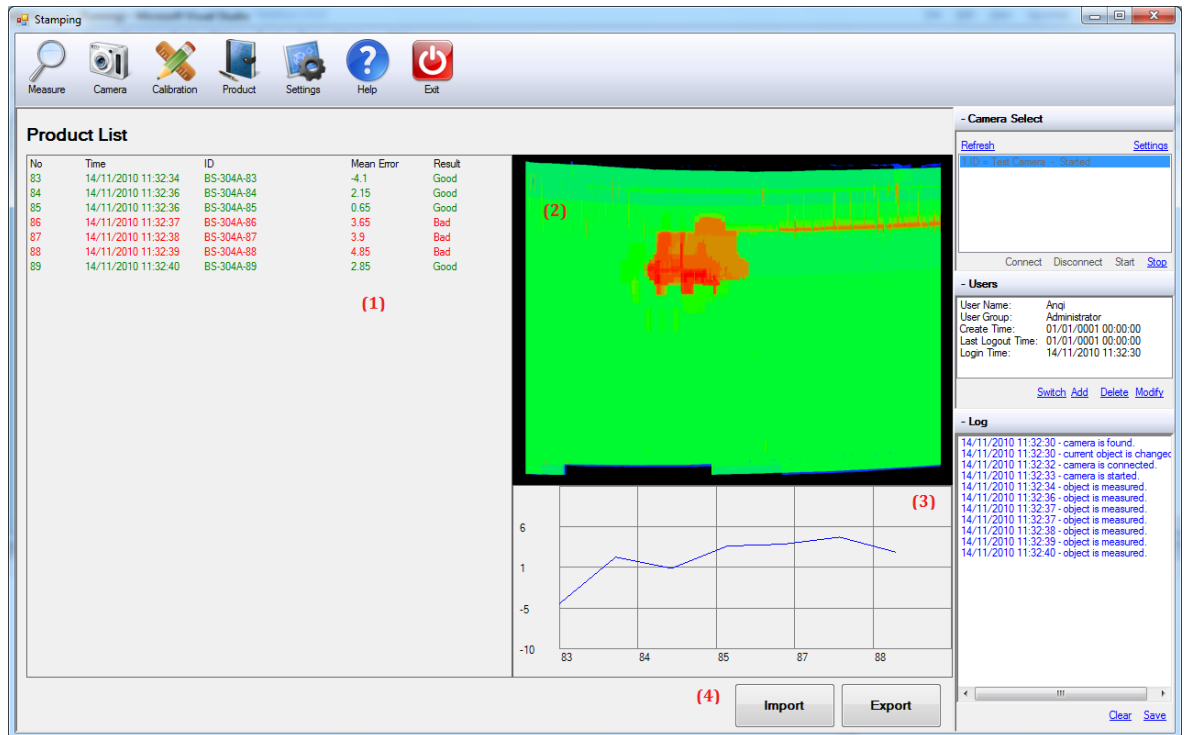


*Figure 5.4 Measurement screen*

### **5.2.5 Product Quality Control**

Product quality control is achieved by tracking the past production performance. **Figure 5.5** is the production screen. Each measurement will be recorded in (1). The quality of every product is traceable by its unique ID. In area (2), the error frequency map is shown. It presents the frequency of errors and where they were found on products. Blue means less error detected in a region. Red means the error was found frequently in a region, which indicates that the region has a consistent deformation that indicates

abrasion of the tool. The stamping tool should be replaced. In area (3), mean error data are plotted. It represents the error trend of evolving abrasion. For the data backup and review, there is an export and import functions in area (4). All the data can be used for production quality control.



*Figure 5.5 Production screen*

### 5.2.6 Parameter Settings

During the system setup all the parameter settings should be inserted. However, these settings may need to be changed after the system is built up. A setting screen **Figure 5.6** provides tools for parameter adjustment. System settings in (1) control how the program is running. For example, the auto save data interval time can be set here. Camera settings in (2) provide support to configure different cameras for the system. 3D visual settings in (3) can select the method of 3D rendering, either as a colour map or a 3D mesh. Visualisation quality can also be adjusted. It is desirable to use standard 3D rendering depending on the CPU and video card performance. That makes it possible to

us a wide range of different PCs even those that have slow CPUs and poor video cards. The last part in (4) is the most important part in the settings screen. This part defines the IO ports and the field bus as the interface of the system to robots or PLCs. Usually, IO usage varies with different host PLC or robot programs. So here we give adjustable IO link definitions for compatibility with existing PLCs or robot programs.

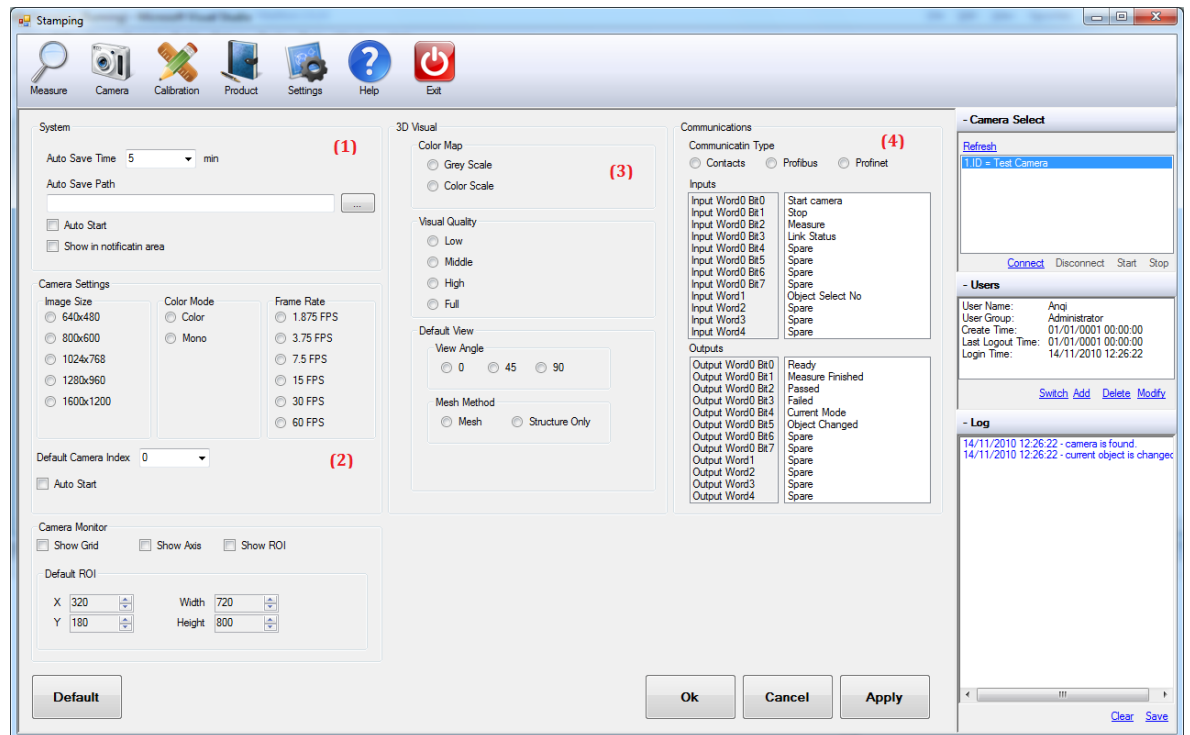


Figure 5.6 Settings screen

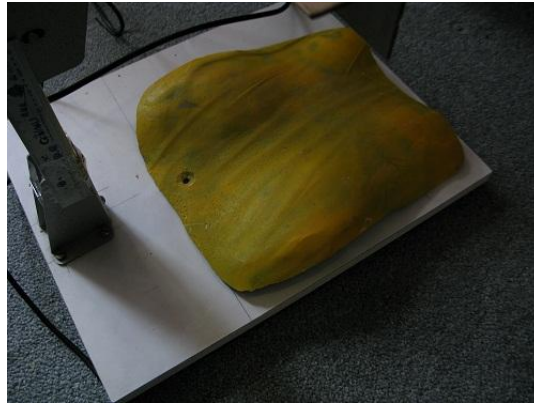
## 5.3 Experiments

### 5.3.1 Approach and Implementation

In order to evaluate its effectiveness, experiments have been carried out to detect tiny shape changes of a sample object, as shown in **Figure 5.7**. The shape change due to abrasion is simulated by attaching a paper flake with about 0.2mm thickness on the object surface. Taking this paper flake as a simulated flaw on an object is sufficient to validate if the precision requirement of 0.5mm has been achieved.



The first step is calibration. We use the stand base as a reference plane. Then lift the camera and laser projector by 3mm and 6mm and to record the lines shifting. Calculate the coordinates of the projected lines at these three heights. According to **Equations 2-6, 2-7**, we can get factor  $K$  and the distance  $s$  at each point on the image.

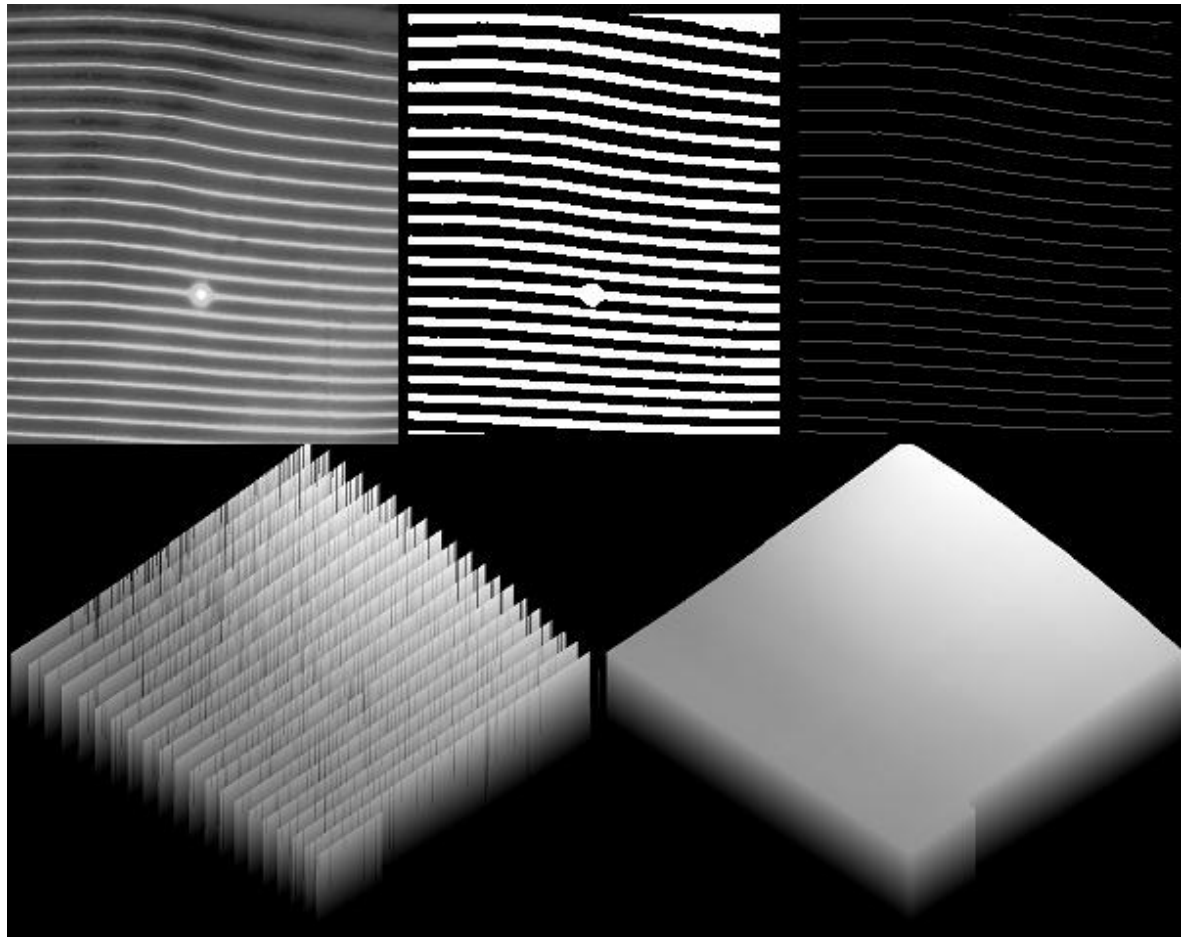


*Figure 5.7 Sample Object*

The next stage is to define a standard object. In this application, what we need to measure is the height difference between the current measured object and the standard object. The height map of the standard object is obtained by the approach described in Section 4.4. **Figure 5.8** show the whole process followed to produce the 3D model of this standard object.

The last stage is to put the paper flake on and off the object to check whether the height difference can be measured. The software should figure out the height difference and where the paper locates. **Figure 5.9 (a) (b)** shows the target object with and without paper. **Figure 5.9 (c)** shows the detected error map. On it the area with this paper flake is clearly marked. **Figure 5.10** shows the measured results while repeating the same measurement 10 times to check whether this paper is successfully marked out every time.

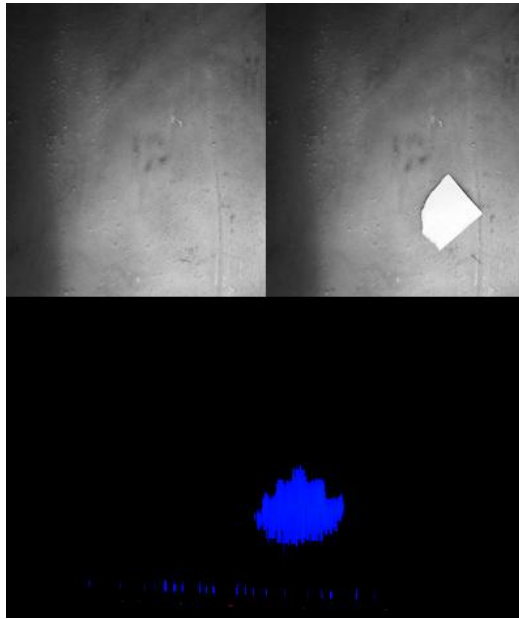
So far, the test on one object has proved that 3D measurement can be achieved and is repeatable using the approach developed in this project Next we need to test its robustness, precision and real time performance.



*(a) Original image (b) Threshold image (c) Extracted lines*

*(d) Height calculated on lines (e) Final surface*

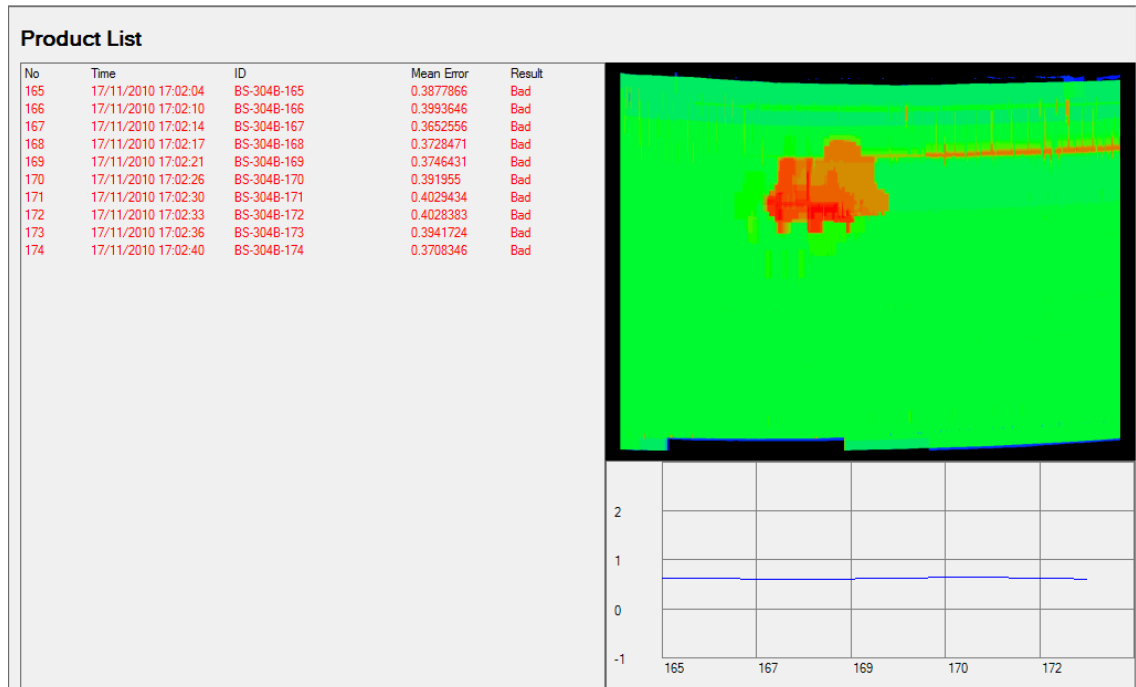
***Figure 5.8 Measurement process***



(a) Standard object (b) Measured object with a paper on it

(c) Measured error map

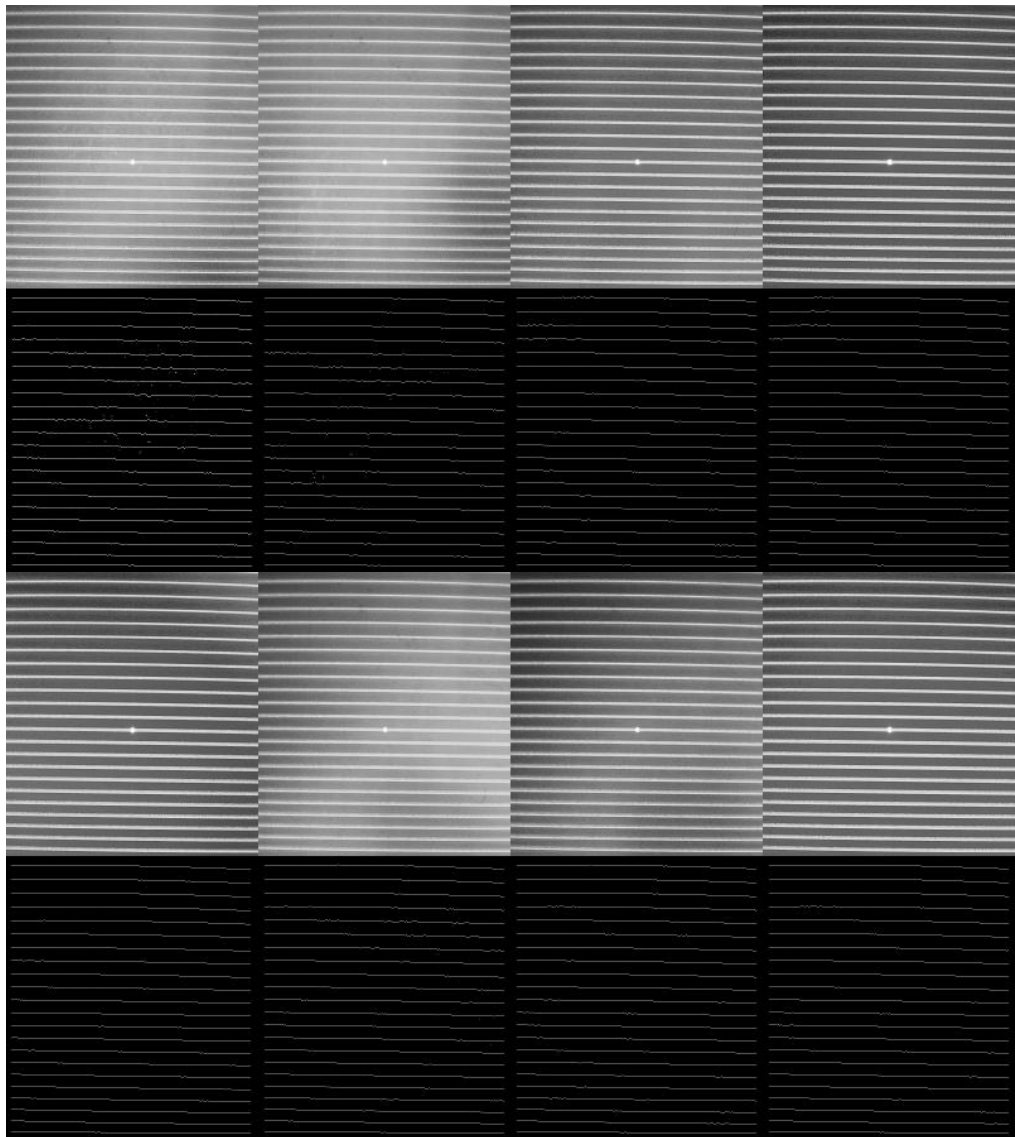
**Figure 5.9 Measured error map**



**Figure 5.10 Measured result with paper on the object**

### 5.3.2 Robustness Test

The proposed measurement method relies on fine line extraction. So its robustness against environment noise is essential for this application. Most of the noise comes from the ambient illumination. We test the line extraction result with different light illumination. **Figure 5.11** gives the results. With a bright background, some noise is found, but can be removed by the line searching method (Section 4.4.2). Hence, no matter whether it is dark or bright, even half dark or half bright, this system can always extract the lines reliably.



*Figure 5.11 Extracted line images with different illumination*

Furthermore, we repeat the measurement on one object with different levels of light illumination and put on different paper flakes at random locations. **Table 5.4** lists results of all the measurements. The measurement should produce a “Bad” result when the paper is put on the object and a “Good” result when the paper is removed. The judgement of “Bad” or “Good” depends on the mean error compared with a threshold. When we set the threshold to be 0.3mm, an object is indicated as “Good” if the measured mean error is less than 0.3mm and indicated as “Bad” otherwise. The “defective” objects were successfully detected in this test.

***Table 5.4 Measure results in different conditions***

Conditions	Result	Error Mean (mm)	Processing Time* (ms)
Object with paper (random paper and position)			
Dark background**	Bad	0.3842	771
Dark background	Bad	0.3144	755
Dark background	Bad	0.4285	748
Dark background	Bad	0.5674	751
Dark background	Bad	0.5599	755
Bright background***	Bad	0.3405	753
Bright background	Bad	0.4702	752
Bright background	Bad	0.5247	746
Bright background	Bad	0.3170	764
Bright background	Bad	0.3538	757
Object without paper			
Dark background	Good	0.2182	748
Dark background	Good	0.1594	770
Dark background	Good	0.1757	753
Dark background	Good	0.2367	777
Dark background	Good	0.1735	748
Bright background	Good	0.1321	750
Bright background	Good	0.1685	754
Bright background	Good	0.2290	745
Bright background	Good	0.2256	754
Bright background	Good	0.2183	755

\*Processing is performed under a 720\*840 pixels ROI and a 5 times iteration calculation of the Kalman filters.

\*\*Dark background: system is put under a desk to avoid sun light

\*\*\*Bright background: system is put in an open room with bright sun light

The above table proves the measurement method can be used on one object with different illumination conditions. That means on a production line we can measure one product with various abrasion levels and in different environments.

### 5.3.3 Precision Test

For checking the measurement precision, we measured a flat plane with known height. Mean and STD (Standard Deviation) are listed in **Table 5.3**. From **Table 5.3**, The STDs for all measurements are less than 0.03mm. In this application, the main task is to measure the height difference between a standard object and a measured object. So the relative height is much more important than the absolute height. Considering a safe tolerance range, we can say that the difference between the two heights is under 0.05mm which means that they have the same height. When the difference measured is greater than 0.05mm, the defect is detected. We can consider 0.05mm as the precision of this measurement system.

**Table 5.3 Height measured after calibration (unit: mm)**

Height	Measure1	Measure2	Measure3	Measure4	Measure5	Mean	STD
0	0.0054	-0.0065	-0.0153	-0.0677	-0.0081	-0.0184	0.028523
2	2.09697	2.09851	2.0894	2.1010	2.0960	2.0964	0.004333
3	2.9597	2.9596	2.9622	2.9518	2.9651	2.9600	0.004945
4	4.1827	4.1382	4.1239	4.1247	4.1219	4.1383	0.025655

### 5.3.4 Real Time Performance

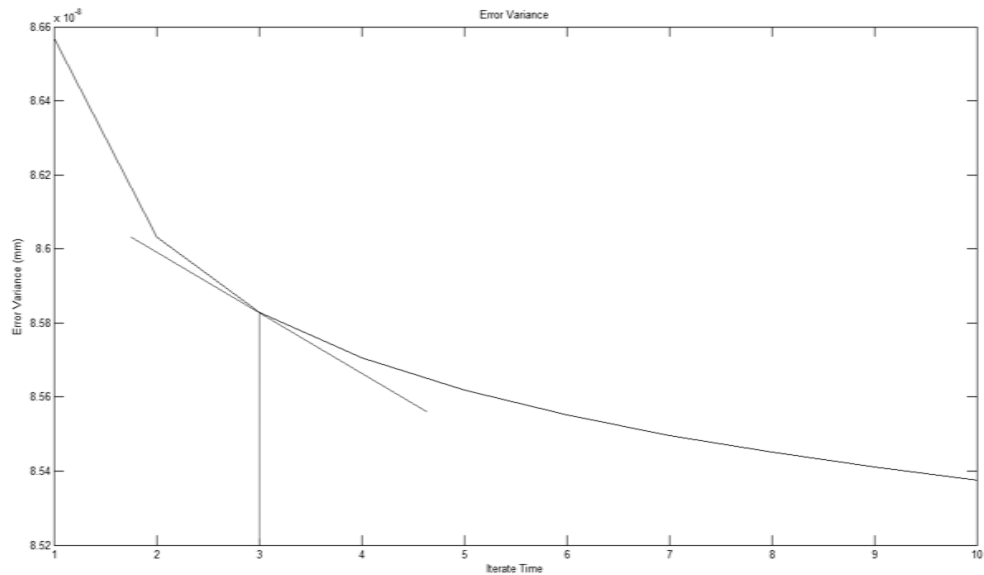
In **Table 5.4**, processing time is also listed. The time is measured from starting the image grab to the completion of the measurement. **Table 5.5** lists the processing times with different ROI settings and iterative calculation times.

**Table 5.5 Processing time with different ROI and iterative calculation times**

Iterative Time	ROI				
	400*300	500*400	800*600	720*840	800*900
1	86	227	358	452	577
2	107	253	421	608	733
3	136	306	592	764	936
4	169	324	624	904	1123
5	192	364	733	1045	1326
10	333	965	1419	1825	2277
15	719	1368	1731	2620	3354

\*Unit: ms

The whole processing time mainly depends on the level of iteration for the Kalman filtering. Figure 5.12 shows the error variance versus the iteration times.



**Figure 5.12 Error variance versus iteration times**

There is a trade-off between error variance and iteration times. The more iteration the greater the reduction in the error variance but it will take more time to complete the calculation. From the curve plotted, we find that the error variance is reduced rapidly at the beginning. Then the reduction ratio is decreasing as the iteration times increase. So the point of compromise is the point with a significant error variance reduction produced in the least acceptable time. Experimentally a three times iteration is the required point.

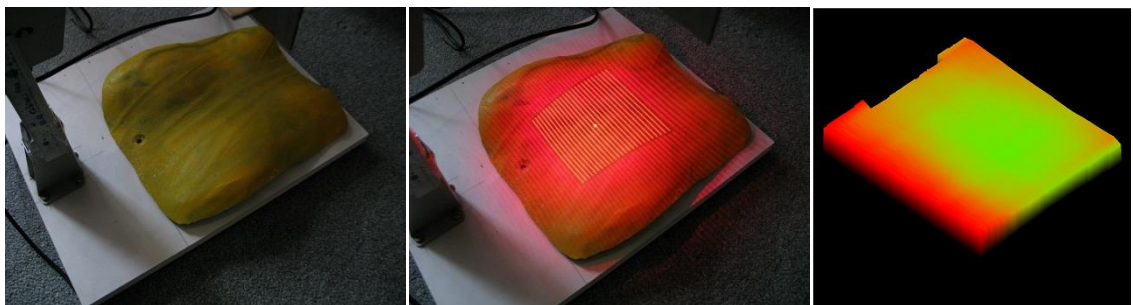
For this system setup, normally the ROI is chosen with a size of 600\*800. The maximum size is 800\*900. With this setting, the whole processing time can be performed within 1s. That is short enough to meet the application requirements.

### 5.3.5 Tests on Different Objects

For simulation of the real application, we conducted experiments on different objects. These objects have different surface shapes and are made of different materials. We used paper flakes and blue tack to simulate height changes. Each object is repeatedly measured 10 times. In these 10 repeats, the first two maintain the original height. The next six have their height slightly changed. The last two are back to the original height.

**Figure 5.13** shows the measured results of these objects.

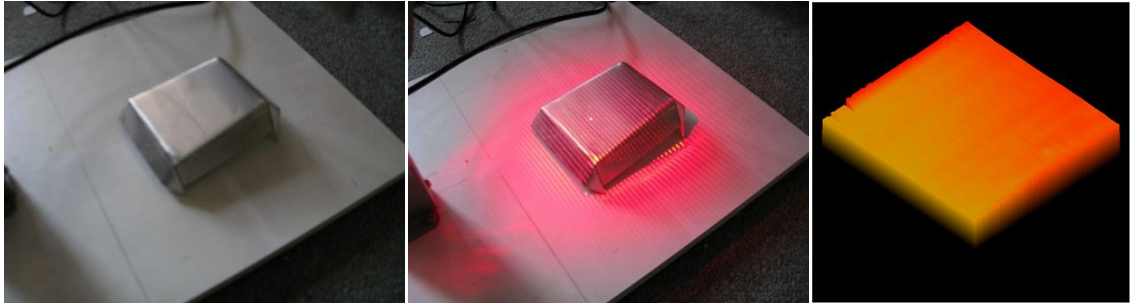
Object 1: Synthetic fibre board



Product List				
No	Time	ID	Mean Error	Result
266	18/11/2010 11:01:28	BS-304B-266	0.2226199	Good
267	18/11/2010 11:01:31	BS-304B-267	0.2194873	Good
268	18/11/2010 11:01:47	BS-304B-268	0.5696023	Bad
269	18/11/2010 11:01:50	BS-304B-269	0.4982646	Bad
270	18/11/2010 11:01:58	BS-304B-270	0.3424048	Bad
271	18/11/2010 11:02:00	BS-304B-271	0.4059537	Bad
272	18/11/2010 11:02:09	BS-304B-272	0.6521823	Bad
273	18/11/2010 11:02:12	BS-304B-273	0.7129788	Bad
274	18/11/2010 11:02:19	BS-304B-274	0.2230754	Good
275	18/11/2010 11:02:20	BS-304B-275	0	Good

Object 2: A stainless steel box





**Product List**

No	Time	ID	Mean Error	Result
367	18/11/2010 11:35:05	BS-304B-367	0.2551985	Good
368	18/11/2010 11:35:07	BS-304B-368	0	Good
369	18/11/2010 11:35:19	BS-304B-369	0.3666983	Bad
370	18/11/2010 11:35:21	BS-304B-370	0.3637559	Bad
371	18/11/2010 11:35:23	BS-304B-371	0.3606012	Bad
372	18/11/2010 11:35:24	BS-304B-372	0.3506216	Bad
373	18/11/2010 11:35:26	BS-304B-373	0.3455502	Bad
374	18/11/2010 11:35:27	BS-304B-374	0.3766695	Bad
375	18/11/2010 11:35:48	BS-304B-375	0.3074613	Bad
376	18/11/2010 11:35:50	BS-304B-376	0.2220216	Good

Object 3: Leather wallet



**Product List**

No	Time	ID	Mean Error	Result
391	18/11/2010 12:01:38	BS-304B-391	0.2448873	Good
392	18/11/2010 12:01:40	BS-304B-392	0.2569401	Good
393	18/11/2010 12:01:53	BS-304B-393	0.9957978	Bad
394	18/11/2010 12:01:55	BS-304B-394	1.046654	Bad
395	18/11/2010 12:01:56	BS-304B-395	1.081024	Bad
396	18/11/2010 12:02:00	BS-304B-396	1.061244	Bad
397	18/11/2010 12:02:04	BS-304B-397	1.015371	Bad
398	18/11/2010 12:02:06	BS-304B-398	1.082307	Bad
399	18/11/2010 12:02:17	BS-304B-399	0.2461172	Good
400	18/11/2010 12:02:19	BS-304B-400	0.233355	Good

*Figure 5.13 Measured results of different objects*

From these results, we found one mistake on the stainless steel shell. The reason is its surface is polished and has a high level of reflection. In this case, a better solution is to use a filter to reduce the light illumination. The tests for the other two objects have achieved the required results. Through the tests, it has been shown that the measurement is effective with different materials. Luckily, on the production sampling line, the target object is raw steel without polish. Its light reflection is close to synthetic fibre board. That makes this approach suitable to real stamping line applications.

## 5.4 Summary

In this chapter, the software developed for the 3D shape measurement has been presented. Experiments demonstrated the effectiveness of the proposed measurement method and verified its suitability for quality control on a stamping line. Moreover, for its use in industry, robustness, precision and real time performance have also been tested.

In comparison with the application requirements listed in Section 3.5, the performance achieved in the tests can be concluded as below:

	Request	Achieved
Precision:	<0.5mm	0.05 mm
Image grabbing time:	<0.5s	<0.2s
Processing time:	<3s	<1s
Robustness:	free of light illumination changes	proved yes

All these requests have been fulfilled. This system is proved to be a robust, precise and real time 3D measurement system.

# Chapter 6

## Conclusions and future works

## 6.1 Conclusions

In this thesis, a 3D vision system has been presented for the measurement and quality control of products on car stamping lines. The objective of this quality control system is to detect the abrasion of the stamping tools. It is not possible to inspect the level of abrasion by eye. Usually, an expensive 3D scanner or coordinate measuring machine is used for checking the product quality; this is less efficient and must be carried out offline. An online and real-time vision system for 3D quality inspection is a highly desirable product for many industries; it will greatly contribute to improvement of product quality. There are a lot of existing approaches for 3D stereo measurement, as discussed in Chapter 2. However, most of them lack of real-time capability for industrial applications. The challenge is to develop a vision system that is robust and can be used for real-time quality inspection of products from a stamping line.

In order to provide a solution for this requirement, a new 3D measurement approach has been developed. As a general industrial application, robustness needs to be taken into account for working in different environments. For this reason, this new approach uses a laser projector to generate a light pattern of parallel lines. Parallel lines can be easily extracted compared to with other complex light patterns, such as those used in the colour coded and phase shift methods. Laser projection ensures that lines are produced with a high illumination competed to the background. They can be easily found and interference effects from noise are greatly reduced. Due to that only one image is needed in the proposed method, a high real-time performance is achieved, especially for dynamic measurement. Through experiment, it has been verified that the proposed algorithm can accomplish a 3D measurement in less than 1s with only 0.2s required for online image acquisition. Furthermore, for increased precision and the reduction of the

effects of noises, a spatial Kalman filter is applied to fuse the measured data in neighbouring areas. With this method, the heights obtained from different directions are adjusted to form an optimized 3D surface with the minimal mean square error.

Additionally, as an industrial solution, a vision product is developed. It contains an industrial camera and a laser projector assembled in a device box with software running on a host PC. Experiments have demonstrated that this new vision system developed as a result of this work has achieved a strong robustness against the uncertainties in the working environments and a high real-time performance. It has fulfilled the requirements for the real application.

There are three main contributions which can be summarised as below:

First, a vision system has been developed for 3D measurement of products on a car stamping line. It provides a solution for the real-time measurement to support car quality control. This real-time measuring system can be totally integrated into an existing product line. It can replace the traditional measuring approaches used to conduct on-line measurement. For its industrial application, this system has been developed by taking into account the following three aspects.

1. The use of Industry standard devices. Unlike lab research projects, in the research and development programme supporting this thesis, all the devices are selected by the considering their suitability for use in industry. In particular, a laser projector is used instead of an office projector. This selection produces many restrictions in the development of the techniques used. A new 3D measurement approach has been developed for this application.

2. Industry installation. For use in industry, the system is designed to be assembled into a device box, see Figure 3.6. This device box can be mounted on a robot arm with

standard cable plug outlets. This design makes it possible to apply this vision system in different production lines.

3. Industrial interfaces. The interfaces contain two parts. One is the application software, named “SVR-3D V1.0” and has been developed. It provides the interface between a user and the vision system. With a friendly GUI, a user can easily operate the system without much background knowledge relevant to measurement and calibration. As an industrial design, all the functions in this software are managed by user rights control. Different users can be provided with different privileges. Also event logs and data backup are supported. The other interface is between the vision system and a control unit. Both the standard industrial field bus and the dry contact IOs are provided by the system, which facilitate its connection with most of the commonly used PLCs and robots.

Secondly, a robust technique for 3D measurement has been developed. It is achieved in two ways.

1. A high signal to noise ratio light pattern. In this thesis, the light pattern is parallel lines which are generated by a laser projector. The parallel lines are easily extracted from the background in comparison with colour coded light patterns. The proposed method needs only a single image for real-time and dynamic measurement, while other methods usually need multiple images for feature extraction. On the other hand, strong illumination from the laser projector enhances the line feature so that the interference effects from the environmental noise are weakened.

2. Information fusion. As described in Chapter 4, the height of a 3D surface can be calculated by searching along the line and also can be obtained through the time measurement. That means height can be calculated in both horizontal and vertical

directions. By fusion along the two directions, a better height map can be constructed. A spatial Kalman filter is applied to carry out the measurement fusion to obtain an optimized height map. With this spatial Kalman filter, the noise is reduced and the 3D surface is smoothed.

Finally, real time 3D measurement has been achieved. The time consumption of the measurement is greatly reduced. The whole time consumption can be divided into two parts. One is for the image acquisition. The reduction in time consumption for image acquisition mainly comes from the reduction in the number of images needed in the developed algorithm. In this newly developed approach, only one image is required. The second part is for information processing. During the processing time, heights are calculated and a 3D surface is generated. However, all these operations can be done offline with no influence on the real-time performance if the time for image acquisition is shorter than the production period between two products. Considering these two parts, we can say the actual online measuring time is only relevant to the first part; the image acquisition time. A single image oriented approach can achieve a higher real-time performance and is more applicable for dynamic measurement.

In summary, an industrially applicable design, the enhancement of system robustness and a high real-time performance are the three main contributions of this thesis to knowledge.

## **6.2 Further Works**

Though the developed vision system has been verified by the experiments, it still needs some further development.

First; the 3D measurement approach mainly focused on height measurement in the smoother areas. Due to the discontinuity in the image, lines could be broken in some regions so that the reconstructed 3D patches cannot be fitted together. Therefore, a technique needs to be developed to consider height measurement for individual patches and effective patch fitting to produce a unique surface, which will rely on the probabilistic estimation of the measurement errors. .

Second; this approach is good at providing relative height measurements where the height is slightly changed from a reference. But for a steep change, the lines tend to be mismatched. In that case, a manual correction function should be provided to the user, which should be developed in the future.

Third; the developed software is still in its testing stage. Significant work is still required for debugging and functional modification before its release.

Finally, this vision system was practically verified in a laboratory environment. In the next stage, we need to carry out a usability study for its applications in a real factory to test it on a real stamping line.



# Reference

- 
- 1 Ishikawa J., Kosuge K. and Furuta K., 1990, *Intelligent control of assembling robot using vision sensor*, *Robotics and Automation, 1990, Proceedings. 1990 IEEE International Conference, 13-18 May 1990, Cincinnati, OH, USA, vol.3, pp.1904-1909*
- 2 Leonard F., Abba G, Ostertag E. and Dossmann Y, 1994, *Real time robotic assembling of moving pieces by visual sensor*, *Control Applications, 1994., Proceedings. The Third IEEE Conference on 24-26 Aug 1994, Glasgow, vol.3, pp.1489-1492*
- 3 Pervez M.S. and Shafait F., 2004, *Real time inspection of assembled valve keys in cylinder heads using CCD cameras*, *ETECH.2004, 31 July 2004, pp.114-120*
- 4 Chen S., Ma Y. and Shi H.B., 2010, *Sausage appearance defect inspection system based on machine vision*, *Control and Decision Conference (CCDC), 2010 Chinese, 26-28 May 2010, XuZhou, China, pp.2485-2489*
- 5 Kim Y.T., Fan S.C., Han S.H. and Go H.S., 2001, *Image-based visual feedback control of a dual-arm robot*, *Industrial Electronics, 2001, Proceedings. ISIE 2001. IEEE International Symposium on 12-16 Jun 2001, Pusan, vol.3, pp.1603-1608*
- 6 Han S.H., Seo W.H., Yoon K.S. and Lee M.H., 1999, *Real-time control of an industrial robot using image-based visual servoing*, *Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on 17-21 Oct 1999, Kyongju, vol.3, pp.1762-1767*

- 
- 7 Chen Q., Georganas N.D. and Petriu E.M., 2007, *Real-time vision-based Hand Gesture Recognition Using Haar-like Features, Instrumentation and Measurement Technology Conference Proceedings. IMTC 2007. IEEE, 1-3 May 2007, Warsaw, pp.1-6*
- 8 Hongmo J., Kim J. and Kim D.J., 2007, *Vision-Based hand Gesture Recognition for Understanding Musical Time Pattern and Tempo, Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE, 5-8 Nov. 2007, Taipei, pp.2371-2376*
- 9 Zhang Q. and Su X., 2002, *An optical measurement of vortex shape at a free surface, Optics & Laser Technology, March 2002, vol.34, pp.107-113*
- 10 Homer J., Longstaf I.D. and Callaghan G., 1996, *High resolution 3-D SAR via multi-baseline interferometry, Geoscience and Remote Sensing Symposium, 1996. IGARSS '96. 'Remote Sensing for a Sustainable Future.', International, 27-31 May 1996, Lincoln, NE, USA, vol.1, pp.796-798*
- 11 Yen H.N., Tsai D.M. and Yang J.Y., 2005, *Full-field 3-D measurement of solder pastes using LCD-based phase shifting techniques, Electronics Packaging Manufacturing, IEEE Transactions on Jan. 2006, vol.29, pp.50-57*
- 12 Cheng F.S. and Chen X.T., 2009, *Integrating 3D vision measurement into industrial robot applications, the Technology Interface Journal/Winter Special Issue 2009 Proceedings of the IAJC-IJME 2008 Conference, vol.10, No.2*
- 13 Spagnolo G. S. and Ambrosini D., 2002, *Diffraction optical element based sensor for roughness measurement, Sensors and Actuators A: Physical Issues 2-3, 1 September 2002, vol.100, pp.180-186*

- 
- 14 Genovese K. and Pappalettere C., 2006, *Whole 3D shape reconstruction of vascular segments under pressure via fringe projection techniques*, *Optics and Lasers in Engineering*, Issue 12, December 2006, vol.44, pp.1311-1323
- 15 Chen L.C. and Huang C.C., 2005, *Miniaturized 3D surface profilometer using digital fringe projection*, *Measurement Science and Technology*, vol.16, No.5
- 16 Shen H., Shi Y.G. and Peng Z.G., 2005, *Applying prior knowledge in the segmentation of 3D complex anatomic structures*, *Computer Vision for Biomedical image applications*, *Lecture Notes in Computer Science*, 2005, Volume 3765/2005, pp.89-199
- 17 Berryman F., Pynsent P., Fairbank J. and Disney S., 2008, *A new system for measuring three-dimensional back shape in scoliosis*, *EUROPEAN SPINE JOURNAL*, vol.17, no.5, pp.663-672
- 18 Evans and Bruce J.W., 2006, *Eye Essentials: Binocular Vision*, ISBN: 0 7506 8850 5
- 19 Daniel Ed. and Marowski G., 1998, *Euclid Introduction. Classical and Medieval Literature Criticism*. vol.25
- 20 Awais, 2010, *History of Vision Science [Internet]. Version 20*. Knol. 2010 Sep 22. Available from: <http://knol.google.com/k/awais/history-of-vision-science/34ovve4otnay1/4>.
- 21 Ackerman and James, 1978, *Leonardo's Eye*. *Journal of the Warburg and Courtauld Institutes*, vol.41
- 22 Kepler J, 1604, *The foundations of modern optics: Paralipomena to Vitellius*

- 
- 23 Lindberg and David C., 1986, *The Genesis of Kepler's Theory of Light: Light Metaphysics from Plotinus to Kepler. Osiris. 2nd Series, vol.2*
- 24 Lindberg and David C., 1977, *Theories of Vision from Al-Kindi to Kepler, 1977 October, Chicago, vol.21, no.4, pp.464*
- 25 Lindberg and David C., 1971, *Alkindi's Critique of Euclid's Theory of Vision. Isis. vol. 62, No. 4, pp. 469-48*
- 26 Wheatstone C., 1838, *Contributions to the Physiology of Vision, Philosophical Transaction of the Royal Society of London, vol.128, pp.371-394*
- 27 Dhond U.R. and Aggarwal J.K., 1989, *Structure from stereo-a review, Systems, Man and Cybernetics, IEEE Transactions on , vol.19, no.6, pp.1489-1510*
- 28 Weng J., Ahuja N. and Huang T.S., 1992, *Matching two perspective views, Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol.14, no.8, pp.806-825*
- 29 Papadimitriou D.V. and Dennis T.J., 1996 *Epipolar line estimation and rectification for stereo image pairs, Image Processing, IEEE Transactions on , vol.5, no.4, pp.672-676*
- 30 Kanade T. and Okutomi M., 1991, *A stereo matching algorithm with an adaptive window: theory and experiments. Proceedings of the 1991 IEEE International Conference on Robotics and Automation (ICRA '91), April, 1991, pp.1088-1095.*
- 31 Belhumeur P. N. and Mumford D., 1992, *A bayesian treatment of the stereo correspondence problem using halfoccluded regions. Computer Vision and Pattern Recognition, 1992. Proceedings CVPR '92., 1992 IEEE Computer Society Conference on 15-18 Jun 1992, Champaign, IL , USA, pp.506-512*

- 
- 32 Geiger D., Ladendorf B. and Yuille A., 1995, *Occlusions and binocular stereo*.  
*International Journal of Computer Vision*, vol.14, pp.211-226
- 33 Horaud R. and Skordas T., 1989, *Stereo Correspondence through Feature Grouping and Maximal Cliques*. *IEEE Journal on Pattern Analysis and Computer Vision*, vol 11, no.11, pp.1168-1180.
- 34 Pritchett P. and Zisserman A., 1998, *Wide Baseline Stereo Matching*. In *Sixth International Conference on Computer Vision, Bombay, India*, pp. 754-760
- 35 Mittal A. and Davis L.S., 2001, *Unified Multi-Camera Detection and Tracking Using Region-Matching*. In *IEEE Workshop on Multi-Object Tracking, Vancouver, Canada*.
- 36 Xu W. and Mulligan J., 2009, *Feature Matching under Region-Based Constraints for Robust Epipolar Geometry Estimation*, *Lecture Notes in Computer Science, Volume 5876/2009*, pp.264-273
- 37 Ponce J. and Genc Y., 1998, *Epipolar Geometry and Linear Subspace Methods: A New Approach to Weak Calibration*, *International Journal of Computer Vision* vol.28. No.3, pp.223-243
- 38 Xie M. and Liu L.Y., 1996, *Colour stereo vision: Use of appearance constraint and epipolar geometry for feature matching*, *Recent Developments in Computer Vision*, vol. 1035, pp.255-264
- 39 Read J.C., 2002, *A Bayesian Approach to the Stereo Correspondence Problem*, *Neural Computation*, vol.14, no.6, pp.1371-1392
- 40 Cernushi-Frias B., Cooper D. B., Hung Y. P. and Belhumeur P., 1989, *Towards a model-based bayesian theory for estimating and recognizing parameterized 3d objects*

---

*using two or more images taken from different positions. IEEE Transactions on Pattern Analysis and Machine Intelligence archive vol.11 no.10*

*41 Roy S., 1999, Stereo Without Epipolar Lines : A Maximum-Flow Formulation, INTERNATIONAL JOURNAL OF COMPUTER VISION, vol.34, no.2-3, pp.147-161*

*42 Salvi J., Pages J. and Batlle J, 2004, Pattern codification strategies in structured light systems, Pattern Recognition vol.37, pp.827-849*

*43 Salvi J. and Mouaddib E., 1997, An Overview of the Advantages and Constraints of Coded Pattern Projection Techniques for Autonomous Navigation, Intelligent Robots and Systems, 1997. IROS '97., Proceedings of the 1997 IEEE/RSJ International Conference on 7-11 Sep 1997, Grenoble , France, vol.3, pp.1264-1271*

*44 Valkenbug R.J., 1996, Accurate 3D measurement using a Structured Light System, Image and Vision Computing, vol.16, no.2, pp.99-110*

*45 Posdamer J. L. and Altschuler M. D., 1982, Surface measurement by space-encoded projected beam systems, Computer Graphics and Image Processing vol.18 no.1, pp.1–17*

*46 Inokuchi S., Sato K. and Matsuda F., 1984, Range imaging system for 3-D object recognition, Proceedings of the International Conference on Pattern Recognition, pp.806–808*

*47 Skocaj D. and Leonardis A., 2000, Range image acquisition of objects with non-uniform albedo using structured light range sensor, Proceedings of the 15th International Conference on Pattern Recognition, vol. 1, pp.778–781*

- 
- 48 Moigne J.L. and Waxman A.M., 1988, *Strutured light pattern for robot mobility*, *IEEE J. Robotics and Automation* vol.4, no.5, pp.541-548
- 49 Yuylsteke P. and Oosterlinck A., 1990, *Range image acquisition with a single binary-encoded light pattern*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.12 no.2, pp.148-164
- 50 Zhang L. and Curless B., 2002, *Rapid Shape Acquisition Using Colour Structured Light and Multi-pass Dynamic Programming*, *Proceedings of the 1st International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, 19-21 June 2002, Padova, Italy, pp.24-36
- 51 Seyedin S. and Shokouhi S.B., 2006, *A New Robust Colour Coding Technique for Solving the Correspondence Problem and 3D Reconstruction in Structured Light Systems*, *Information and Communication Technologies, ICTTA '06. 2nd* , vol.1, pp.1080-1084
- 52 Lavoie P., Ionescu D. and Petriu E., 2007, *Constructing 3D Virtual Reality Objects from 2D Images of Real Objects Using NURBS*, *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2007. VECIMS 2007. IEEE Symposium on* pp.117-124, 25-27
- 53 Caspi D., Kiryati N. and Shamir J., 1998, *Range imaging with adaptive colour structured light*, *Pattern Analysis and Machine Intelligence, IEEE Transactions on* May 1998, vol.20, no.5, pp.470-480
- 54 Bergmann D., 1995, *New approach for automatic surface reconstruction with coded light*, in: *Proceedings of Remote Sensing and Reconstruction for Three-Dimensional Objects and Scenes*, vol.2572, pp.2-9

- 
- 55 Guehring J., 2001, *Dense 3-D surface acquisition by structured light using off-the-shelf components*, *Proceedings of SPIE - The International Society for Optical Engineering* vol.4309 pp 220-231
- 56 Ortuno J.E., Guerra P., Kontaxakis G., Ledesma-Carbayo M.J. and Santos A., 2009, *3D surface acquisition for FMT using high-accuracy fringe projection profilometry*, *Nuclear Science Symposium Conference Record (NSS/MIC), IEEE* , pp.4009-4012
- 57 Karout S., 2007, *Two-Dimensional Phase Unwrapping*, *General Engineering Research Institute (GERI)*, *Liverpool John Moores University*, Available from: <http://www.ljmu.ac.uk/GERI/90207.htm>
- 58 Yen H.N. and Tsai D.M., 2004, *A fast full-field 3D measurement system for BGA coplanarity inspection*. *The International Journal of Advanced Manufacturing Technology*, 2004, vol.24, pp.1-2
- 59 Yu X.Y, Wu H.B (2008) *Structured-light-based 3D measurement technique using Gray code and line-shift fringe*, *Chinese Journal of Scientific Instrument* Vol.29 No.4
- 60 Wu H.B. and Chen Y., 2006, *3D Measurement technology by structured light using strip-edge-based Gray code*, *Journal of Physics, Conference Series* 48, pp.537-541
- 61 Hall-Holt O. and Rusinkiewicz S., 2001, *Strip boundary codes for real-time structured-light range scanning of moving objects*, *In Eighth IEEE International Conference on Computer Vision*, vol.2, pp.359-366
- 62 Anderson B.L., 1994, *The role of partial occlusion in stereopsis*. *Nature*, vol.367, pp.365–368.



---

63 Geiger D., Ladendorf B. and Yuille A., 1995, *Occlusions and binocular stereo.*

*International Journal of Computer Vision*, vol.14, pp.211-226

64 Malik J., 1996, *On Binocularly viewed occlusion Junctions*, *COMPUTER VISION —*

*ECCV '96 Lecture Notes in Computer Science*, 1996, vol.1064, pp.167-174

65 *Auto Threshold and Auto Local Threshold*, Available from:

<http://www.dentistry.bham.ac.uk/landinig/software/autothreshold/autothreshold.html>

66 Welch G. and Bishop G., *An Introduction to the Kalman Filter*, 2006, TR 95-041

*Department of Computer Science University of North Carolina at Chapel Hill Chapel*

*Hill, NC 27599-3175*

67 Basar T., 2000, *A New Approach to Linear Filtering and Prediction Problems* ,

*Control Theory: Twenty-Five Seminal Papers (1932-1980)*, pp.167–179

68 Brown R.G. and Hwang P.Y.C., 1996, *Introduction to Random Signals and Applied*

*Kalman Filtering. 3rd Edition. John Wiley and Sons, Inc. New York.*

69 Siemens, *OLE for process control, Standardized access to process data*, Available

from: <http://www.sea.siemens.com/us/internet->

[dms/ia/AutomationComm/Automation/Docs/6ZB5310-0EW02-0BA5.pdf](http://www.sea.siemens.com/us/internet-dms/ia/AutomationComm/Automation/Docs/6ZB5310-0EW02-0BA5.pdf)

70 Siemens, *PROFIBUS, the unique fieldbus system: on the road to success*, Available

from: <http://www.industry.siemens.co.uk/automation/uk/en/industrial->

[communications/profibus/Pages/Default.aspx](http://www.industry.siemens.co.uk/automation/uk/en/industrial-communications/profibus/Pages/Default.aspx)

71 Siemens, *PROFINET-the leading industrial Ethernet standard*, Available from:

<http://www.industry.siemens.co.uk/mcms/automation/en/industrial->

[communications/profinet/Pages/Default.aspx](http://www.industry.siemens.co.uk/mcms/automation/en/industrial-communications/profinet/Pages/Default.aspx)

---

72 Brad Communications, *PROFIBUS DP Scanner For the Allen-Bradley PLC-5 Controller*, Available from: [http://www.applicom-int.com/files/pdf/en/SST\\_PFB-PLC5.pdf](http://www.applicom-int.com/files/pdf/en/SST_PFB-PLC5.pdf)

73 Omron, *Profibus-DP Product Family* , Available from:  
[http://www.omron247.com/marcom/pdfcatalog.nsf/PDFLookupByUniqueID/F038BA64F9086FE086256D73006CDE6E/\\$File/POV\\_ProfiBus.pdf?OpenElement](http://www.omron247.com/marcom/pdfcatalog.nsf/PDFLookupByUniqueID/F038BA64F9086FE086256D73006CDE6E/$File/POV_ProfiBus.pdf?OpenElement)

74 Mitsubishi, *USER'S MANUAL, FX3U-64DP-M PROFIBUS-DP Master Block* ,  
Available from:  
[http://www.usedplcs.co.uk/usedplcs/manuals/mitsubishi/PLC/FX3U\\_64DP\\_M\\_Manual.pdf](http://www.usedplcs.co.uk/usedplcs/manuals/mitsubishi/PLC/FX3U_64DP_M_Manual.pdf)

75 Siemens, *Profinet component editor*, Available from:  
[https://a248.e.akamai.net/cache.automation.siemens.com/dnl/zI/zIwMDk3AAAA\\_22557412\\_Tools/22557412\\_CBA\\_Editor\\_V11\\_e.pdf](https://a248.e.akamai.net/cache.automation.siemens.com/dnl/zI/zIwMDk3AAAA_22557412_Tools/22557412_CBA_Editor_V11_e.pdf)