

This is to certify that the
dissertation entitled

DEVELOP RAPID 3D SURFACE MEASUREMENT SYSTEMS
FOR QUALITY INSPECTION IN THE MANUFACTURING
INDUSTRY

presented by

Quan Shi

has been accepted towards fulfillment
of the requirements for the

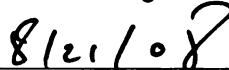
Ph.D.

degree in

Electrical and Computer
Engineering



Major Professor's Signature



Date

PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.
MAY BE RECALLED with earlier due date if requested.

DATE DUE	DATE DUE	DATE DUE

**DEVELOP RAPID 3D SURFACE MEASUREMENT
SYSTEMS FOR QUALITY INSPECTION IN THE
MANUFACTURING INDUSTRY**

By

Quan Shi

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Electrical and Computer Engineering

2008

ABSTRACT

DEVELOP RAPID 3D SURFACE MEASUREMENT SYSTEMS FOR QUALITY INSPECTION IN THE MANUFACTURING INDUSTRY

By

Quan Shi

A general framework of automated dimensional inspection systems for the manufacturing industry is introduced in this dissertation. 3D surface inspection is a process to evaluate the quality of a part's 3D shape. A traditional inspection system usually uses a contact-based Coordinate Measurement Machine(CMM) to acquire the 3D coordinates of a single point by touching a part surface one at a time. A point cloud, used to represent a part's 3D shape, usually contains thousands of points measured by sequentially touching the part surface. The deviation between the designed shape and the measured point cloud forms an error map, which uses a color schematic representation to visualize the distribution of manufacturing errors over the entire part surface. Surface dimensional measurement using a CMM is time consuming. Besides, the contact-based measurement strategy limits applications to only rigid part surfaces. A soft part surface usually may not be inspected by a CMM. In another aspect, when surfaces are coated with special material, the contact-based measurement method may not be allowed because an unpredictable movement or force from a CMM's probe may scratch the surface. According to the above limitations of a CMM, a rapid non-contact surface measurement system is necessary to the manufacturing industry. Vision-based automated dimensional inspection systems, with advanced computing technologies, can dramatically improve the inspection efficiency.

In this dissertation, two novel optical area sensing systems are developed for 3D shape inspection of lambertian surfaces and reflective surfaces. According to surface

reflection property, a triangle-based area sensor is developed for measuring lambertian surfaces, and a back-imaging system is developed for measuring reflective surfaces. A feedback scheme is developed to improve the measurement quality. A CAD-guided robot area sensor planner is developed to estimate the viewing pose of a 3D sensing device relative to the part surface. By controlling the viewing position and orientation, the quality of the measured point clouds can be optimized for error map generation. The feedback approach is also applied in the back-imaging system for measurement optimization. The developed back-imaging system is an innovative technology for automated dimensional inspection of automotive glasses.

TABLE OF CONTENTS

LIST OF TABLES	vi
LIST OF FIGURES	vii
CHAPTER 1	
INTRODUCTION	1
1.1 Motivations	1
1.2 Background and literature review	2
1.2.1 Automated Visual Inspection Systems	2
1.2.2 3D Imaging on Lambertian and Reflective surfaces	3
1.2.3 Planning and control of an Automated Visual Inspection System	7
1.2.4 Data analysis for quality evaluation	12
1.3 Objectives and challenges	16
1.4 Organization of the dissertation	17
CHAPTER 2	
DEVELOPING A BACK-IMAGING SYSTEM FOR DIMENSIONAL INSPECTION OF REFLECTIVE SURFACES.	19
2.1 3D surface measurement using Structured Lighting Method.	20
2.1.1 Measurement principle	20
2.1.2 Error analysis	22
2.1.3 Accuracy, precision, and point density.	24
2.1.4 Gray coding and line shifting method	25
2.1.5 Obtain sub-pixel accuracy using the GCLS method.	27
2.1.6 Image contrast and measurement uncertainty	28
2.2 Calibrate a 3D area sensor using a pixel-to-pixel strategy	30
2.3 Developing a back-imaging system for reflective surface measurement.	35
2.3.1 Projection screen	36
2.3.2 Unknowns variables of a back-imaging system.	37
2.3.3 Measurement principle	38
2.3.4 Measurement using the Golden Section method	40
2.4 A Multi-projection based Back-imaging System	42
2.5 Bottom layer reflection problem	43
2.6 Vector-based system calibration	46
2.7 Chapter summary	50
CHAPTER 3	
PROCESS PLANNING AND CONTROL OF AN AUTOMATED DIMENSIONAL INSPECTION SYSTEM	51
3.1 CAD-guided area sensor planning.	52
3.1.1 Task constraints for CAD-guided area sensor planning	53

3.2 Feedback Control of the Automated Dimensional Inspection System	58
3.2.1 Model of the closed-loop dynamic inspection process	59
3.2.2 Model of system components	60
3.2.3 Model of the dynamic measurement process	65
3.2.4 Stability analysis of the dynamic inspection process	65
3.3 Applying the dynamic inspection process to 3D surface measurement.	66
3.3.1 Applying the general framework to solve the hole problem.	66
3.3.2 Stability analysis of the specific model in solving hole problem.	70
3.4 Feedback design of the back-imaging system.	71
3.5 Chapter Summary	73
 CHAPTER 4	
DATA ANALYSIS AND VERIFICATION FOR QUALITY INSPECTION.	75
4.1 Point clouds registration	76
4.1.1 Registration using robot kinematics.	76
4.1.2 Transformation using the quaternion algorithm	79
4.1.3 Robot hand-eye calibration using ICP method	81
4.2 Integration of geometric features for point cloud registration	85
4.3 Integration of user-inputs to a weighted point cloud registration method	86
4.4 A link clustering algorithm for point cloud filtering	88
4.5 Surface-oriented error map generation	89
4.6 Chapter Summary	93
 CHAPTER 5	
EXPERIMENTAL RESULTS	95
5.1 Lambertian surface measurement using the Automated Dimensional Inspection System.	95
5.1.1 Calibration setup and implementation steps	95
5.1.2 Measurement performance of developed area sensor	98
5.1.3 Inspection on an automotive pillar	100
5.2 Reflective surface measurement using the Back Imaging System	104
5.3 Chapter Summary	106
 CHAPTER 6	
CONCLUSIONS.	117
 BIBLIOGRAPHY.	120

LIST OF TABLES

2.1 One-Pix Error Analysis, $\text{res}=0.151$, $S=393.7$, unit: <i>mm</i>	23
2.2 One-Pix Error Analysis, $d=429.3$, $S=393.7$, unit: <i>mm</i>	23
2.3 One-Pix Error Analysis, $d=429.3$, $\text{res}=0.078$, unit: <i>mm</i>	24
4.1 Weights for Point Cloud Registration	87
5.1 Measurement accuracy after calibration	99

LIST OF FIGURES

2.1 Measurement configuration of area sensor using structured light method	21
2.2 An area sensor prototype	21
2.3 Digitized sinusoid wave projection pattern	25
2.4 3D shape measurement using GCLS method	26
2.5 Image quantization error in edge detection	27
2.6 Edge detection using interpolation strategy	28
2.7 An image of stripes in high/low contrast area	29
2.8 Intensity profile in high/low contrast area	30
2.9 Area sensor calibration using pixel-to-pixel strategy	31
2.10 Calibration of sensor parameter d1 S1 using a linear least square method	32
2.11 Calibrating the offset angle	33
2.12 Reflection of a point light source on a reflective surface and a diffuse surface. (a)Reflection on a diffuse surface, (b) Reflection on a reflective surface.	36
2.13 Measuring a reflective surface using diffuse-based projection method	37
2.14 3D shape measurement of reflective surface using recursive searching	38
2.15 3D shape measurement of reflective surface using recursive searching	39
2.16 An accelerated searching using the golden section method	41
2.17 The flow chart of the golden section searching method	42
2.18 3D shape measurement of reflective surface using multi-project technique . . .	43
2.19 Bottom layer reflection	44
2.20 Bottom layer reflection of a glass	45

2.21 An intensity chart around a white dot	45
2.22 Distinguish the bottom layer reflection using a black dot pattern	46
2.23 An intensity chart around a black dot	47
2.24 Calibrate the view vector of an image point	48
2.25 Estimate the world coordinates of an image point, a theoretical model	48
2.26 Estimate the world coordinates of an image point, a perspective model	49
3.1 Automated dimensional inspection system	52
3.2 Deriving visibility constraints of an area sensor	54
3.3 Clustering the triangles of a CAD model according to the visibility constraint	55
3.4 Integration of planning constraints using a bounding box method	56
3.5 A recursive viewpoint searching algorithm.	57
3.6 An simplified open-loop robot sensor planning scheme	58
3.7 A close-loop robot sensor planning scheme developed for a 3D dimensional inspection process	59
3.8 Diagram of a sensor planning system using dynamic methods for 3D shape inspection of automotive parts	60
3.9 Structure diagram of a viewpoint evaluator	64
3.10 Unknown shadow space formed by geometry occlusion	67
3.11 Estimate viewpoints by using bounding box method	68
3.12 An example of reflection spot	69
3.13 An analytical model of light reflection	69
3.14 Rotation of view direction to conquer reflection problem. (a) Sensor setup with light reflection problem occurring (b) and (c) Avoid a detected brightness spot by relocating the area sensor	70

3.15 Improve the measurement accuracy by adding more cameras	71
3.16 A networked-based back-imaging system	73
3.17 The flow chart of the designed feedback system	74
4.1 Coordinate transformations to the robot based frame in the automated dimensional inspection system	77
4.2 Coordinate transformations to the world frame of the automated dimensional inspection system	77
4.3 Solve the ambiguity of the closest point for the ICP based calibration algorithm. (a) point-to-point distance (b) point-to-plane distance	83
4.4 The flow chart of an area-sensor-based robot hand-eye calibration using an ICP algorithm	84
4.5 Flow chart of a link clustering algorithm for outlier filtering	90
4.6 Closest distance of a point to a triangle. (a) Projection of a measured point falls inside of a triangle (b) Projection of a measured point falls outside of a triangle	91
4.7 Calculate error distance of a point to its corresponding triangle	92
5.1 An automated dimensional inspection system	96
5.2 Calibration setup on a DYNA 2400C NC machine. (a) calibration for acquiring depth information (b) calibration for obtaining a 3D point cloud	96
5.3 Calibration by moving the calibration board with patterns of control points	98
5.4 Calibration results of exploding vector $K(x,y)$	99
5.5 Calibration of transformation matrix from a 2.5D height map to a point cloud.	100
5.6 Testing sensor precision using a gauge.	101
5.7 3D Shape of an MSU Logo stamp	102
5.8 3D shape of a computer mouse	102

5.9 Dimensional measurement of a door panel part using the automated area sensing system	103
5.10 3D shape measurement of a surface with a small depth range	103
5.11 3D shape measurement of a surface with a big depth range	103
5.12 An automotive pillar for dimensional inspection	104
5.13 A tessellated CAD model of the pillar	104
5.14 Estimated viewpoints	105
5.15 3D scan on different viewpoints	105
5.16 Point clouds measured on each viewpoint	106
5.17 The measured point cloud of part pillar-m32510	107
5.18 An interpolated point cloud of part pillar-m32510, top view	107
5.19 An interpolated point cloud of part pillar-m32510, side view	108
5.20 An interpolated point cloud of part pillar-m32510	108
5.21 An error map of part pillar-m32510	109
5.22 A back-imaging system for automotive glasses inspection	109
5.23 Detect marker points from the back image. (a) A recorded image, (b) Identify pixels of top layer reflection, (c) Detected points, (d) An expanded view of the detected pixels on the image plane	110
5.24 Detect marker points from the back image. (a) Image of the calibration board at position A, (b) Image of the calibration board at position B, H is the height difference in the world frame, (c) and (d) Detected view vectors.	111
5.25 Improve the accuracy of the surface norm measurement using the feed-back design	112
5.26 The measured point cloud using the iteration-searching method, side view . . .	113
5.27 The measured point cloud using the iteration-searching method, top view . . .	114

5.28 Measurement of part ch-3201	115
5.29 The measured point cloud using the iteration-searching method, top view . . .	116

CHAPTER 1

Introduction

This chapter presents an overview of this thesis as well as its motivations, objectives, and challenges. The organization of this thesis is provided at the end of this chapter.

1.1 Motivations

Since the 21st century, vision-based 3D sensing technology had become a revolution to manufacturing industries. The 3D shape of a part, usually modeled using geometry primitives such as points, lines, curves, and surfaces, can now be modeled by a set of points, named a point cloud, which makes two research areas: reverse engineering and rapid dimensional inspection. A variety of commercialized 3D sensors had been developed and utilized in many applications. However, two problems are recently discussed that limited further applications of this technology: first, an automated or semiautomated 3D measurement system is necessary in certain manufacturing industries such as automotive manufacturing. Second, as still the major challenge of this new technology, there lacks of a vision-based 3D method to measure a reflective surface, for example, an automotive glass. The current methodologies and 3D sensors can only measure lambertian surfaces, which diffuse lights to all directions. Methods and devices for measuring lambertian surfaces has been proved cannot be used for

reflective surface measurement. Aiming to this industrial requirement, this thesis is trying to develop effective methods, practicable devices, and reliable systems for quality inspection of both lambertian surfaces and reflective surfaces.

1.2 Background and Literature Review

1.2.1 Automated Visual Inspection Systems

Inspection is the process of determining if a product deviates from a given set of specifications [1]. In the manufacturing industry, inspection needs the measurement of part features such as assembly integrity, surface finish, and more general, geometric dimensions.

In industrial plants, inspection is usually performed in two different ways: 100% inspection and batch inspection. 100% inspection means every product need to be inspected, whereas batch inspection means only a set of samples will be inspected. In both type of inspection systems, the human inspector is sometime not reliable because of the inconsistency or fatigue.

A CMM is a traditional system commonly used for dimensional inspection. A probe is used to touch part surface and obtain the coordinates of points. Though the accuracy of a CMM can reach 0.0001", the measurement capability is limited by its contact-based measurement principle. Besides, the point-by-point measurement process makes a CMM not feasible for 100% inspection of many manufacturing parts.

Machine vision has been widely used for quality inspection in today's manufacturing industry. Machine vision is concerned with the theory and technology for building artificial systems that obtain information from images or multi-dimensional data. The acquisition of an image consists of a bi-dimensional digitalization of the observed scene. The amount of light recorded by the sensor from a certain scene point depends on the type of lighting, the reflection characteristics and orientation of the

surface being imaged, and the location and spectral sensitivity of the sensor. With the growing capability of computers, machine vision brings more and more intelligence and automation to manufacturing industry.

Automated visual inspection (AVI) systems provide significant advantages over human inspection from fatigue, throughput, time, and accuracy. The advances in cost, performance, and design methodologies have resulted in an explosion of application areas, where AVI systems have become an integral component of quality control schemes in product inspection and certification. Generally, AVI systems represent a quantitative feedback node to identify and eliminate problems at various stages in the production process.

Though many 2D image-based machine vision system have been successfully implemented in various applications, 3D surface dimensional inspection is still a problem opening for a proper solution. 3D imaging technology, since the 1970s, has received much concern in developing a vision-based automated dimensional inspection system. In this thesis research, methods for 3D freeform surface inspection are discussed and two novel measurement systems are presented for measuring both lambertian and reflective surfaces.

1.2.2 3D Imaging on Lambertian and Reflective Surfaces

A. Review of 3D imaging technology

The problem of retrieving 3D information from an object by vision has a long history. Basically, existing techniques can be separated into two groups, *contact* and *non-contact* techniques. The use of contact techniques has been limited due to the problems involved in touching the surface of an object. Objects can be deformed during this operation, producing dimensional errors on the estimation of the shape. This group of techniques can be divided into destructive techniques, like slicing, or non-destructive techniques, such as the articulated arms of industrial robots and

CMMs.

Non-contact techniques have certain advantages. Shape can be measured in the presence of delicate objects, hot environments, deformable objects, big scenes, etc. This group of techniques usually has two categories: transmissive and reflective techniques.

Transmissive techniques are based on computer tomography. This method, based in X-rays, is widely used in medical imaging. Other transmissive scanners are based on time-of-flight lasers, which compute the distance to a surface by the round-trip time of a pulse of light. A laser is used to emit a pulse of light and the time that passes before the reflected light is seen by a detector is counted. Since the speed of light is already known, the round-trip time determines the travel distance of the light, which is twice of the distance between the scanner and the surface. This kind of range finder is used in large object measurement such as airborne dimension acquisition.

Reflective techniques can be classified into optical and non-optical. Non-optical techniques such as sonar or microwave radars are mainly used in robotics. In addition, optical techniques can be divided into two types, active sensors and passive sensors. Active sensors emit some kind of radiation and detect its reflection to probe the presence of an object or environment. The possible types of radiation used include Digital Light Projectors (DLP), laser triangulation and interferometry.

Passive scanners do not emit any kind of radiation themselves, but instead rely on detecting the radiation reflected by objects. Most scanners of this type detect visible light because it is readily available ambient radiation. Other types of radiation, such as infrared, could also be used. Passive methods can be very cheap because in most cases they do not need special hardware. Most common examples are shape-from-X, where X represents the method used to determine the shape, that is motion, stereo, shading, silhouette and texture, among others. All these techniques are based on the use of several images of the object/scene, and are known as multiple-view geometry.

Although the term *3D imaging* in machine vision is first appeared in 1970s, the origin of this modern technique is attributed to the introduction of epi-polar geometry by Longuet-Higgins, 1981[2]. Epi-polar geometry has been studied in the last decades, producing a wealth of knowledge in this field, and has been extensively used in camera calibration, 3D acquisition and correspondence problem simplification.

Despite multi-view geometry being widely used in computer vision, it presents some drawbacks when it is used in 3D imaging. The first problem is the correspondence problem. In other words, determining relationships from pixels of different views is not a trivial step. The second important problem is the resolution of the acquisition. These techniques usually work with a small number of points, so that dense reconstructions are difficult to obtain.

In order to overcome both drawbacks, active sensors are commonly used when dense reconstructions are required. Based on laser or coded structured light, several commercial sensors are available nowadays. This kind of sensor is basically used to get 3D models of objects or scenes. However, modeling is not only a three-dimensional acquisition of the object/scene but a complex problem composed of several steps: First, some techniques can be applied to determine the best position of the camera with the aim of reducing the number of views of a given object/scene. Sometimes, it is also used to acquire images from incomplete objects or scenes, especially in map building [3][4]. Second, 3D acquisition involves obtaining the object/scene structure or depth, which can basically be delivered into two different representations known as range images or point clouds: a range image is a bi-dimensional representation of the object/scene, where the intensity of each pixel of the image is directly related to the depth. A point cloud is simply a set of 3D points digitized from the measured surface.

B. Review of 3D Scanners

According to projector light source, 3D scanners can be divided into two cat-

egories: laser line scanners and white light area scanners. A laser line scanner is usually more robust to intensity noises. A laser line scanners can have small size and less weight to be able to mount on a multi-joint equipment, such as a robot arm or a CMM. 3D laser line scanners have been widely used in reverse engineering. However, for quality inspection, accuracy of a laser scanner is often limited by the field of view, point density, and sensor resolution etc. White light based area sensors have shown better performance than laser line scanners for quality inspections. However, most white light area sensors is usually bulky and heavy to be mounted on a motion platform. A proper white light area scanner is essential to an automated dimensional inspection system.

A white light area sensor usually contains two parts, a projector and a camera: The projector is used to put a set of encoded patterns on part surfaces such that the camera can decode those patterns for acquisition of 3D part shape. The encoded pattern affects all of the measurement performance such accuracy, precision, and point density etc. Many different codification strategies have been developed [5, 6, 7, 8, 9, 10], which can be mainly categorized as time-multiplexing, spatial neighborhood coding, and De Bruijn sequence. At present, Gray Code and Phase Shifting (GCPS) method is widely used for white light area sensors [11]. However, the sinusoid wave used in the GCPS method is not robust to intensity noises, which causes measurement errors in point clouds. Gray Code and Line Shifting (GCLS) method, developed in this thesis, using a square wave to form the projection pattern, shows better measurement performance.

Calibration of a 3D sensor is necessary to obtain satisfied measurement quality. Trobina [12] developed an error model of an area sensor, in which the sensor model was developed using perspective geometry and a maximum likelihood estimation method. To calibrate lens distortion of an area sensor, a multiple-plane method was proposed in [13]. Because an area sensor is usually made by a camera and a projector, it is considered to be calibrated simultaneously. A relative error model

[14] was developed for calibration of an area sensor. In [15], a self-calibration method is developed to calibrated an equivalent viewing model for acquiring the 3D surface shape. According to published documents, a camera *Pinhole Model* is widely used for area sensor calibration. Pinhole model is a theoretical lens model that assumes all incident light is emitted from a single point, the pupil of the projector lens and all recorded light passes through a single point of the camera lens. Those two pupils then become two end points of a line, with another point on surface, a triangle can be generated that is usually called an active triangle. Therefore, the triangulation model of an area sensor relies on the well calibrated lens so that the pupils of the camera and the projector are located accurately. Camera calibration is only used to determine the intrinsic and extrinsic parameters of the camera. For an area sensor, except parameters of the camera, parameters of the projector and geometry relationship between the camera and projector all need to be calibrated. In this thesis, we develop a pixel-to-pixel strategy to calibrate our area sensor prototype. Instead of calibrating a camera and a projector using the pinhole model, we calibrate the active triangulation model directly based on each pair of corresponding pixels between the projector and the camera. The pixel-to-pixel strategy significantly simplifies the calibration tasks, and more important, improves the measurement accuracy.

1.2.3 Planning and Control of an Automated Visual Inspection System

To many applications in the manufacturing industry, inspection of a large part requires 3D sensors to be configured on different viewpoints for high-accuracy, high-precision, and high-density reconstruction of a part surface. The process typically involves planning a set of views, physically altering the relative object-sensor pose, taking scans, registering the acquired geometric data in a common coordinate frame of reference, and finally integrating range images into a non-redundant model. Ef-

iciencies could be achieved by automating or semi-automating this process. While challenges remain, there are adequate solutions to the scan-register-integrate tasks. On the other hand, view planning remains an open problem, i.e., the task of finding a suitably set of sensor poses and configurations for specified reconstruction or inspection goals. Besides, if we want to use these 3D data in applications that require data with a high degree of accuracy like inspection tasks, it is mandatory that the 3D points be acquired under the best conditions of accuracy. Thus we must find and model the most important parameters affecting the accuracy of the range sensor. An error model, along with the CAD model of the part, is required to produce a sensing plan to completely and accurately acquire the geometry of the part. The sensing plan is comprised of the set of viewpoints that defines the exact position and orientation of the area sensor relative to the part.

High-quality model building requires view planning for performance-oriented reconstruction, which is defined as model acquisition based on a set of explicit quality requirements expressed in a model specification. High-quality inspection tasks are also specification driven. In addition to all-aspect coverage, measurement quality may be specified in terms of accuracy, precision, sampling density, and other factors such as processing time. Specified measurement quality may be fixed or variable over the object surface. Performance-oriented view planning requires suitable models of both sensor and positioning system performance. These can be combined in an imaging environment specification. Specifically, it requires the following:

- a 3D sensor model with a description of the sensing geometry and a characterization of measurement performance within the calibrated region;
- a positioning system model describing the degrees of freedom, range of motion, and positioning performance within the movement envelope. There has been relatively little work on performance-oriented view planning. Cowan and Kovcsi [16] used a constraint satisfaction approach for sensor location subject

to task requirements which included resolution, focus, field of view, visibility, view angle, and prohibited regions.

- Often, the knowledge of environment or the model of the object to be inspected. For manufacturing inspection, a CAD model is always provided for quality evaluation. Hence, a CAD-guided area sensor planning system is a general solution to this type of applications.

A. CAD-guided Process Planning

An industrial robot/CMM can be used to automatically move an area sensor to multiple viewpoints for measuring a large-scale part surface. For a non-contact measurement, it is difficult for a human operator to teach robot viewpoints properly only by observation. A sensor planner can enhance the measurement quality and efficiency [17, 18, 19]. In 2003, Scott et al. [20] summarized the automatic methods of planning a 3D sensor for reverse engineering and vision inspection. Sensor planning systems usually can be categorized into two groups: model-based and non-model-based. Non-model-based planning strategies are mainly developed for mobile robots in object searching and optimized trajectory planning. Model-based sensor planning methods are more reliable than non-model-based methods because the prior CAD model embeds fidelity into the planning strategy. In this thesis, the vision-based 3D shape measurement system is developed for quality inspection. CAD model of the inspected part is always acquirable. Hence, model based sensor planning methods are focused in this chapter.

In [16], Cowan and Kovesi first introduced an automatic vision sensor planning system, in which task constraints are developed as searching functions such that the planning problem can be solved as constraint-satisfaction problem. Tarabanis et al. [21] first introduced the term: Machine Vision Planner (MVP) system, which integrates the robot vision task constraints with a CAD model and a sensor model. The MVP task specification includes visibility, field of view, resolution, focus, and image

contrast. Sheng et al. [22] implemented a CAD-Based robot motion planning system by combining the generate-and-test and the constraint-satisfaction approaches. One problem in those system is that sensors usually are considered as a single camera, or equivalent to a single camera model.

For planning an area sensor, the triangulation model of the area sensor model is very different to a single camera model. Therefore, planning an area sensor becomes a different problem from planning a camera. To estimate viewpoints of an area sensor, task constraints for both camera and projector have to be satisfied. Meanwhile, different planning constraints are applied according to the different types of area sensors. As the 3D sensing technologies, devices, and systems kept growing in recent decades, methods of planning of a 3D sensor kept the same pace. An early work was developed by Pito [23], in which the Next Best View (NBV) strategy does viewpoint searching for a 3D sensor. Prieto et al. developed a 3D laser camera and a NURBS (Non-Uniform Rational B-Splines)-based sensor planner [24], in which viewpoints are projected on a 2D bitmap such that the occlusion and collision problem can be solved using a 3D voxel model. For planning a white-light-based area sensor, Chen and Li [25] developed a model-based sensor planning system by mounting two video cameras on a PUMA robot. An equivalent sensing model couples constraints of both cameras for the sensor planning issue.

B. Feedback-based Process Control

A *qualified* point cloud is a group of 3D points that represent the actual 3D shape of a part surface. This representation has to be accurate, smooth, and complete. “Smooth” means the distance between sampled points (point density) need to be confined to a small value, and “complete” means that each region of a part surface needs to have a representation in a point cloud. A qualified point cloud is essential to an automated dimensional inspection system for surface quality control. Improving the design of a sensor itself, such as increasing the resolution of a cam-

era/projector, can reduce measurement uncertainties. In another aspect, a sensor's viewing pose, including location and orientation, is also related to measurement accuracy [17]. Though a CAD-guided area sensor planner provides optimized viewpoints according to *a priori* geometry information of the part, the surface characteristics, such as color, texture, especially reflection property, makes the measurement performance unpredictable by only CAD data. The obtained point cloud, though not qualified yet for further data processing, can provide useful geometry shape for estimating another set of viewpoints. Measurement on the updated viewpoints can be used to improve the measurement accuracy.

In a real measurement, the accuracy of 3D shape measurement is usually not evenly distributed over a point cloud [17, 26]. It is also unavoidable to have “holes” [27] appear in measured point clouds, which makes the measured surface incomplete to use for representing the 3D shape of a part surface. To overcome this problem, online measurement data can be fed back for estimating new viewpoints. A positional space (PS) algorithm [28] gave a solution to the NBV problem without any *a priori* CAD information. This PS algorithm mainly focuses on measuring all visible surfaces of an unknown object. In [29], a ray-casting method is developed to incrementally build a meshed model of a part with unknown shape. Data redundancy was minimized by reducing the number of viewpoints in algorithm. However, for quality control purpose, the first criteria of a sensor planner is measurement quality but not the measurement speed. Although data redundancy increases the computation time, it can also be used to improve the measurement accuracy. Therefore, the optimization task of a sensor planner should be measurement quality. This criteria was also claimed in [27], in which, a method of clustering unseen directions was introduced to optimizes the “completeness” (percentage of surface sampled) of scanned surfaces.

Except holes, measurement errors are inevitable in an *Automatic Dimensional Inspection (ADI)* system. The sources of uncertainties include:

1. Intensity noise
2. Image quantization error
3. Poor surface property of light reflection.

Intensity noise is often a random noise in the process of counting photon energy for each pixel. This type of noise mainly depends on the image grabbing system and can be reduced by filtering techniques. The quantization error has been well studied [30, 31]. Quantization error unavoidably exists in almost all vision systems and is bounded by the size of pixels. A high resolution camera usually can provide better measurement accuracy. With a same camera, a proper viewpoint can also be used to reduce this quantization error [17]. Surface property of light reflection is a problem particularly when a projector is involved in a vision system. Because various materials have different reflection properties, the projected patterns may not be clearly recorded. For example, in a low contrast region of an image, error is often increased because the edge of projected fringes easily becomes noisy.

Many view planning systems [32, 20] emphasize minimizing the number of viewpoints and the length of paths. For shape inspection, the priority of the accuracy needs to be higher than the topology of the sensor path. Hence, a view planner should be designed to optimize the measurement accuracy as well.

1.2.4 Data Analysis for Quality Evaluation

Data analysis includes several important issues to develop a correct dimension deviation report for quality evaluation. Data analysis includes two major topics: registration and error map generation. Registration of point clouds is to find Euclidean motions among a set of point clouds or, between a point cloud and the model set of a given part, respect to a common reference frame. Error map is usually a color-coded map that shows the deviation of the measured point cloud and its CAD model. In

manufacturing, error map is needed to evaluate the part quality. And furthermore, adjust the manufacturing process such that a correct part can be made. Obviously, a wrong error map will cost expensive to manufacturers.

Three factors need to be concerned for error map generation: measurement accuracy, alignment between point clouds and the CAD model, and a method to calculate the shape difference. Measurement accuracy is related to 3D scanner design. The alignment between point clouds and CAD model, usually defined as a registration problem, includes two categories: 1. registration of multiple point clouds into one common coordinate frame and, 2. registration of the entire point cloud to the CAD model. The first registration task is to develop a correct point cloud for representing the true shape of the measured part, whereas the second registration task is to find a correct transformation for matching the point cloud and CAD together for comparison. Both of them are critical for quality evaluation. After the entire point cloud is aligned/registered to its CAD model, shape differences will be quantized by the distance of each measured point to the surface of the CAD model. A color-encoded picture, called error map, is used to visualize the distribution of these manufacturing errors. Methods to calculate the error map can be different: a point-to-plane distance is often used to generate an error map for 3D dimensional inspection. However, for the inspection of membrane vibration displacement, a distance-along-vector is calculated, which the shape variation is calculated in a specified direction.

The registration problem can be categorized into coarse registration and fine registration: coarse registration determines an initial alignment and fine registration estimates a best solution to be close to the truth as much as possible. Without coarse registration, fine registration is time consuming and often fails to find a solution. Methods for coarse registration include principal component analysis (PCA) [33], point signature [34], and image transformation [35], etc. For an automated inspection system, coarse registration can be done by a mechanical platform such as

an industrial robot or a CMM. For fine registration, statistical approaches are often applied to match the point cloud to the CAD model. The Iterative closest point (ICP) algorithm [36] is a widely used method for solving the fine registration problem. Given two data sets that represent an identical geometric shape in two coordinate systems, the ICP algorithm iteratively transforms a data set toward another one until a closest match is obtained. In other words, the transformation between two data sets is considered a “best-fit”, such that the average of the shape differences on between the two data sets reaches a minimum. Except for processing time, finding a local minimum is the greatest concern in ICP algorithms. Many extensions of the ICP algorithm have been developed to solve the local minimum problem and improve the algorithm efficiency [37][38]. ICP-based algorithms have been applied successfully in many applications such as pattern recognition and reverse engineering.

However, an ICP-based algorithm is not suitable in solving fine registration problem for inspection purposes. As mentioned above, an error map is a visualization of absolute shape differences, whereas a transformation using a best-fit criterion is to minimize overall shape differences. A limitation of ICP-based algorithms is that the shapes of two data sets to be registered need to be considered identical. Since finding the difference between the measured point cloud and the CAD model becomes the goal, an ICP-based algorithm is not feasible for this application.

For a 3D shape inspection system, an ordinary registration approach in industry is to integrate a non-contact 3D scanner with a contact mechanical probe: several feature points measured using the probe are used as the “skeleton” and points calculated by the 3D scanner are used as the “skin”. The “skeleton” is then used to register the entire point cloud to the CAD model. However, the drawback of this type of systems is that it changes a non-contact based 3D scanning system into a contact based scanning system, which significantly limits the application scope.

Direct coordinates transformation is the simplest and most efficient way to reg-

ister a point cloud to its CAD model. However, if the kinematics of the motion platform do not provide enough accuracy [39], the system will utilize the motion repeatability for point cloud registration, if the repeatability of the sensor viewpoints is satisfied. In most cases, the repeatability of a motion platform could be much better than the motion accuracy for point cloud registration. Therefore, registration can be done automatically on those calibrated viewpoints. Second, to improve registration performance, geometry features on a part surface, including corners, edges, and holes, can be used for matching two data sets. Moreover, the analytical geometry features such as peaks, valleys, ridges, determined by surface gradients, can also be integrated to improve the registration quality. Third, when a specific portion of a part surface is available, the ICP algorithm can be applied to that area to find the registration directly. Finally, very often in many applications, certain user-defined features, such as points or lines, are set as “anchor” references for point cloud registration. That user-input information becomes the most important for point cloud registration. Therefore, instead of only using a single statistical value, the weighted registration technique is to find the best “fitting” solution according to all available information from the measurement system, part surface, and user-inputs.

It needs to be pointed out before the discussion of the proposed scheme: the absolute correct registration matrix may not be able to be found. Registration quality should not depend on a minimized number between the two data sets. A registration is “qualified” only when the derived error map can successfully report the error between the manufacturing part and its designed shape. Testing using a CMM verified error map maybe reasonable in applications.

1.3 Objectives and Challenges

The objective of this thesis is to develop a general framework for automated dimensional inspection systems, which include topics not only in the development of theoretical methodology but also application implementations. Though certain 3D sensors are available in industry, this thesis aims at the development of automated 3D sensing systems, and especially, a system to inspect reflective surfaces, which does not have a solution at present.

The main challenges involved in this thesis include:

- 3D sensing methods that satisfies all measurement requirements including accuracy, precision, point density, scanning field, and processing time.
- An automated inspection system. To gain the best quality of measurement, proper viewing poses need to be configured. Especially for a large auto part, a 3D sensor has to be set to different viewpoints for acquiring the entire part surface. An automated inspection system needs to be developed to meet this purpose.
- A valid error map for quality evaluation. An error map is the final result of the developed 3D surface inspection system. A valid error map tells the distributions of the manufacturing errors. Failure to gain a correct error map will prove expensive to the manufacturer. A standard of evaluation based on the measured point clouds needs to be developed.
- An innovative 3D shape measurement approach for specular surface measurement. The current 3D sensors can only measure Lambertian surfaces, which diffuse light in all directions. To measure a windshield's surface, the current 3D imaging methods cannot be applied because of the surface properties. New methods need to be developed for this application.

1.4 Organization of the Dissertation

This dissertation proposal describes a general framework for development of an automated dimensional inspection system. It includes the following chapters:

- Chapter 2 describes a general framework for active vision based 3D shape measurement, which includes encoding algorithms, image processing methods, 3D sensor design, area sensor model, error analysis, and calibration strategy. A digital area sensor is developed based on the proposed methodologies. Compared to commercial 3D digitizers, our digital area sensor has certain advantages for an automated dimensional inspection system. Meanwhile, the general framework also includes a novel 3D shape inspection method, which is called back-imaging method for measuring a reflective surface.
- Chapter 3 introduces a general frame work of a control system for an automated dimensional inspection system. The general framework includes a CAD-guided view planning method and a feedback design for 3D shape measurement control. The contribution is that, quality control is considered in a system point of view. In comparison, in the field of dimensional inspection, performance of quality evaluation is only determined by the capability of a 3D sensor. In fact, a proper inspection process is also critical to inspection results. With the developed feedback controller, quality of a measured point cloud is supervised to meet the specified requirements. In this thesis, a control system is designed for a robot view planner for measurement of Lambertian surfaces. For the measurement of a reflective surface using the back-imaging system, the feedback control system is designed to select proper sensors for measurement optimization.
- Chapter 4 discusses data processing techniques for decision making of an automated dimensional inspection system. Our strategies for point cloud registration and error map generation are presented in this chapter. Point cloud

registration is about matching multiple point clouds together and aligning the entire point cloud to a desired data set, i.e., a CAD model or a pre-verified point cloud. Point cloud registration is critical to error map generation. Though many researchers use the best-fit approach, for dimensional inspection, we develop a combination solution that integrates robot kinematics, photogrammetry, surface features, local surface fitting, and user datum points for a proper alignment estimation of transformations for error map generation.

- Chapter 5 reports the preliminary results of our automated dimensional inspection systems, which demonstrates a testing example on an automotive pillar, m32501, from Ford Motor Company. The back-imaging method is also implemented and tested on an automotive side door glass, provided by PPG Industries.
- Chapter 6 summarizes the developed methodologies and our contributions.

CHAPTER 2

Developing a Back-Imaging System for Dimensional Inspection of Reflective Surfaces

Dimension inspection using a CMM is time consuming because the probe of a CMM must touch the surface point-by-point. The automotive industry needs a rapid 3D dimensional inspection system. A vision-based non-contact 3D scanner can acquire a part's 3D shape patch-by-patch, which reduce the processing time significantly.

Because of the physical characteristic of a part surface, an incident light ray will be diffused in all directions by a lambertian surface, whereas it is reflected in a single direction by a reflective surface. This property makes optical measurement of lambertian surfaces much easier than reflective surface because a camera can be generally set to any location above the inspected part. For a reflective surface, the location of sensing device has to be in the line of the reflected ray. The difference between these optical phenomena determines that: for a lambertian surface, images recorded *on* the part surface are real images; while for a reflective surface, since it works as a mirror, the recorded images appear to be "*behind*" the part. Therefore, the

camera records a mirrored image that does not exist on the part surface and we call this way of 3D shape measurement on reflective surfaces the *back-imaging* method.

Although the approaches of measuring a lambertian surface and a reflective surface are different, there is a general framework of 3D surface measurement that contains methods required by both. This chapter describes the general framework by developing a 3D area sensor for lambertian surface measurement. Then, an innovative back-imaging system is introduced that is developed for reflective surface measurement. The application of this back-imaging system is for a rapid windshield dimensional inspection.

2.1 3D Surface Measurement using Structured Lighting Method

2.1.1 Measurement principle

If a part has lambertian surfaces to reflect incident light in all directions, the relative height of any surface point to a reference plane can be solved in a pair of similar triangles $\triangle ADC$ and $\triangle BDE$ as shown in Figure 2.1: h_D represents the distance from one surface point D to the reference plane. S is the stand-off distance from camera frame to the reference plane and d represents the baseline distance from the camera to the projector.

This method is usually called triangulation-based structured lighting method. An area sensor prototype is developed based on this scenario as shown in Figure 2.2. This prototype include a Sony XCD710 digital camera, a Plus V-1100 Digital Light Processing (DLP) projector, and a plano convex lens to adjust the size of projection field.

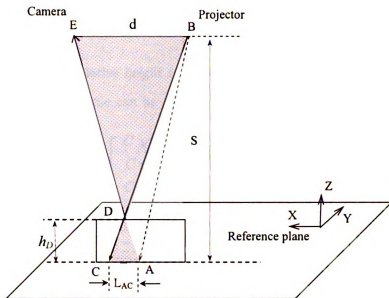


Figure 2.1. Measurement configuration of area sensor using structured light method

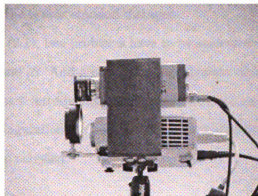


Figure 2.2. An area sensor prototype

$$h_D = \frac{L_{AC} \times S}{d + L_{AC}} \quad (2.1)$$

Eqn.(2.1) shows the way to calculate the height h_D . S and d are a pair of sensor parameters that need to be acquired from sensor calibration. Unknown variable L_{AC} is the distance between two corresponding point A and C, which is measured by

image analysis. In the rest of this chapter, we use h and L to represent h_D and L_{AC} respectively.

After the surface relative height is obtained, the 3D coordinates of the surface points in the camera frame can be calculated using the perspective transformation model of the camera:

$$\begin{bmatrix} {}^C X_D \\ {}^C Y_D \\ {}^C Z_D \\ 1 \end{bmatrix} = {}^C T_I \begin{bmatrix} u \\ v \\ 1 \\ 1 \end{bmatrix} \quad (2.2)$$

$${}^C T_I = \begin{bmatrix} \frac{-(S-h)}{F} & 0 & 0 & 0 \\ 0 & \frac{-(S-h)}{F} & 0 & 0 \\ 0 & 0 & (S-h) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.3)$$

where u and v are the image coordinates of the surface point D . F is the camera focal length and S is the sensor standoff distance.

According to Eqn.(2.1), two problems have to be answered: how to determine the sensor parameter S and d ? And second, how to calculate distance L precisely from images? Before we start introducing our methods, it is necessary to show the error estimation of this triangulation method, because it reveals what requirements for the area sensor have to be satisfied.

2.1.2 Error analysis

Eqn.(2.1) can be rewritten as Eqn.(2.4), where res represents the camera resolution and m represents the pixel distance counted from point A to C :

$$h = \frac{res \times m \times S}{d + res \times m} \quad (2.4)$$

Assume there is an one-pixel error of a measurement L , Eqn.(2.5) gives an esti-

mation model of height error Δh :

$$\Delta h = \left| \frac{res \times S}{d + (2m - 1) \times res + \frac{m \times (m - 1) \times res^2}{d}} \right| \quad (2.5)$$

where res represents camera resolution, which is another parameter of an area sensor. From Eqn.(2.5), it can be seen that all sensor parameters S, d, res play roles in this one-pixel error model. However, the baseline distance d is found to have more impact to accuracy. As shown in next three tables, the measurement uncertainties Δh are vary according to sensor parameters and pixel distance (represented by the number of pixels m). Each table only sets one sensor parameter as a variable while maintain the other two as constants:

Table 2.1. One-Pix Error Analysis, $res=0.151$, $S=393.7$, unit: mm

	$d = 114.7$	$d = 213.4$	$d = 315.6$	$d = 429.3$
$m=53pix$	0.4535	0.2590	0.1792	0.1335
$m=64pix$	0.4414	0.2551	0.1774	0.1325
$m=87pix$	0.4177	0.2446	0.1737	0.1304

Table 2.2. One-Pix Error Analysis, $d=429.3$, $S=393.7$, unit: mm

	$res=0.192$	$res=0.174$	$res=0.151$	$res=0.078$
$m=53pix$	0.1681	0.1530	0.1335	0.0702
$m=64pix$	0.1665	0.1517	0.1325	0.0699
$m=87pix$	0.1632	0.1489	0.1304	0.0693

The one-pixel error estimation shows that a larger baseline distance d , a shorter stand off S , or a higher camera resolution res will bring better measurement accuracy. Notice this error analysis is a simulation result under the assumption that there is

Table 2.3. One-Pix Error Analysis, $d=429.3$, $res=0.078$, unit: mm

	$S = 451.6$	$S = 434.7$	$S = 420.5$	$S = 393.7$
$m=53pix$	0.0805	0.0775	0.0750	0.0702
$m=64pix$	0.0802	0.0772	0.0747	0.0699
$m=87pix$	0.0795	0.0766	0.0741	0.0693

exactly one-pixel error. Normally, the measurement error of L is measured in sub-pixel accuracy such that the real error is bounded by this one-pixel error.

2.1.3 Accuracy, precision, and point density

Measurement accuracy is usually a specification of an uncertainty to the true value. Because of intensity noises, measurement of L between point A and C will contain error, which will cause error in a point cloud.

Measurement precision is another sensor specification for evaluating the 3D scanner's capability of sensing the minimum depth change, which is useful in identifying small surface features such as dints and dents. From Eqn.(2.1), it can be seen that precision is only determined by the measurement resolution of L , which gives a critical issue in 3D shape inspection: how to precisely allocate point A and C on the camera image plane.

For any 3D scanner used for quality control purposes, it is often required to specify the above three numbers. They might vary according to the applications and costs, but, once the task is given, designing a proper 3D scanner and an automatic dimensional inspection system should focus on satisfying those three basic requirements.

Point density specifies the average distance between two measured points. It is also called the resolution of a 3D scanner, which mainly determined by the number of pixels on camera image plane and the size of the field of view.

2.1.4 Gray coding and line shifting method

The Phase Shifting method is a common solution for obtaining the profile of a freeform surface. To measure a discontinuous surface, the Gray Code method can be applied for unwrapping the calculated phase map, the so-called GCPS method. However, for a sinusoid curve, at the non-linear portion of peak and valley, the magnitude (intensity) does not have significant variation with phase value. Therefore, random noise may easily increase the error of a derived phase map.

Considering intensity noise, a square wave is much more robust than a sinusoid wave. Such a method, named GCLS (Gray Code and Line Shifting), is used for 3D shape measurement. The drawback of GCLS method is that it needs more images than GCPS method to obtain required point density, which slows the measurement process. However, for inspection, the measurement accuracy and measurement precision are more emphasized. Hence the GCLS method is adopted in this thesis for the developed area sensor.

If camera resolution res_x , res_y are pre-calibrated, and image coordinates (x_A, y_A) , (x_C, y_C) of points A and C are known, L can be calculated by Eqn.(2.6):

$$L = \sqrt{(res_x \times (x_A - x_C))^2 + (res_y \times (y_A - y_C))^2} \quad (2.6)$$

For a square pixel camera, res_x and res_y may equal to each other and are often represented by res , which is the length of a square area on reference plane projected to a single image pixel, so the unit of res is mm/pixel.

Figure 2.3 shows a 3-bit Gray coding process; three patterns were used to generate eight stripes. In practice, for a 1024×768 camera, six patterns are actually used to generate 64 stripes, and each stripe has 12 pixels. Shifting each stripe 12 times can generate $64 \times 12 = 768$ codes. The number of codes is also determined by the requirement of point density. For example, if 0.4mm spacing between two measured points is required, for a 1024×768 camera on 200×150 viewing field, the number of

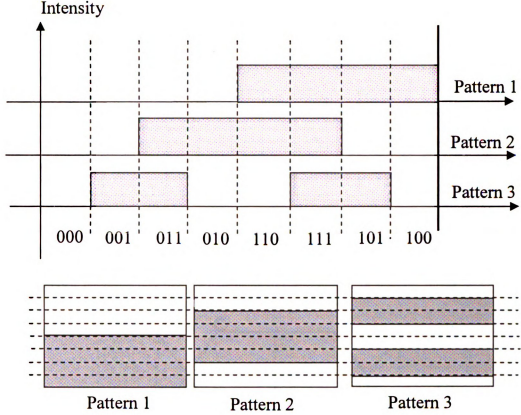


Figure 2.3. 3D shape measurement using GCLS method

pixels to be shifted each time will be $1024 * 0.4/200 \approx 2$. The total number of codes is then $64 \times 6 = 384$.

2.1.5 Obtain sub-pixel accuracy using the GCLS method

For the stripes of the area sensor, precisely determining the boundaries of strips is a critical issue for a successful 3D shape measurement. The gray coded line shifting (GCLS) method is proposed to our digital light processor (DLP)-based range sensor because it is more robust to intensity noise. Furthermore, sub-pixel measurement accuracy can also be obtain using the square wave. instead of using the sinusoid wave in the GCPS method. If not, quantization error will be added into measurement

results. As shown in Figure 2.4, the edge of a stripe can arbitrarily fall in anywhere between an image pixel n and its next pixel $n + 1$.

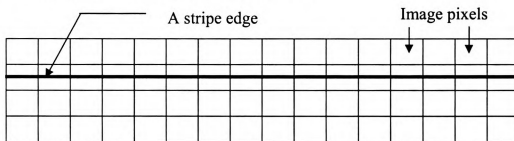


Figure 2.4. Image quantization error in edge detection

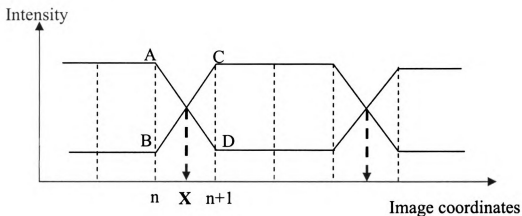


Figure 2.5. Edge detection using interpolation strategy

An edge detection method is illustrated in Figure 2.5: using another image with opposite intensity stripes, A, B, C, and D represents four intensity values that will be used to calculate X, the location of a stripe boundary. Sub-pixel accuracy can be obtained with this interpolation calculation.

Even though, the estimation of X may have error because of image noises. But this error can be bounded in one image pixel. On a view surface that has a norm parallel to the viewing direction, resolution is often defined by the size of a piece surface projected onto one pixel. This resolution is a constraint widely used in many

view planning system. For a freeform surface, it may contain many small patches, norms of each small patch will be vary so that the resolution of each patch is quite different. Average norm of all small patches, can be used to determines a viewpoint for those patches. This method only satisfies the global resolution constraint. However, for a small patch under that viewpoint, the angel between the surface norm and the viewing direction may exceed the threshold such that the resolution constraint cannot be satisfied. Hence, the one pixel error bound is increased and the measurement uncertainty will be out of tolerance. For surface shape inspection, this small area needs a new viewpoint to satisfy the measurement constraints.

2.1.6 Image contrast and measurement uncertainty

Quantization error exists in all over the image. However, measurement uncertainty can be increased in a low contrast region in the image. Considering intensity noises, stripes in a high contrast area will be more robust than stripes in a low contrast area. Figure 2.6 illustrates an image with high/low contrast regions. The intensity profiles of two lines in Figure 2.6 are displayed in Figure 2.7: the top curve shows the intensity profile of the line through a low contrast area and the bottom curve shows the intensity profile of another line through a high contrast area. Image noises can be seen at the peak and valley of each wave, which affects the accuracy of the edge detection algorithm.

Since the part surface property is unpredictable, viewpoint estimation from the part's CAD cannot solve this problem. To record an image with the proper image contrast, the feedback design need to be incorporated to modify the position and orientation of the planned viewpoints.

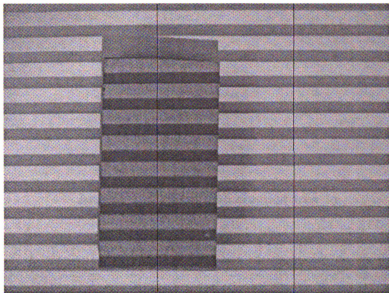


Figure 2.6. An image of stripes in high/low contrast area

2.2 Calibrate a 3D area sensor using a pixel-to-pixel strategy

Calibration of the developed area sensor is a task to determine the sensor parameter set S, d, res , in which res is relatively easy to determine: it could be done by counting pixels of a known rectangle in an image. However, it is not easy to estimate the sensor stand-off S and baseline distance d . The camera pin-hole model is widely used for 3D sensor calibration to obtain a pair of S and d , according to the principal points of the camera and projector lens. To gain better measurement accuracy, a pixel-to-pixel method is developed to calibrate (S, d) for only a pair of corresponding points.

As shown in Figure 2.8, points B_1 and E_1 are corresponding with each other. And light from B_1 to point D_1 is reflected to E_1 . Notice that E_1 may not be on the camera plane; it is a virtual point which can be determined by two lines: the reflected light beam D_1E_1 , and a base line B_1E_1 that is parallel to the reference plane.

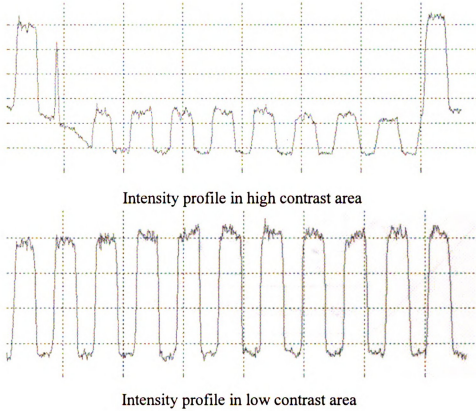


Figure 2.7. Intensity profile in high/low contrast area

plane. Therefore, to measure the height of point $D1$ using Eqn.(2.1), $(d1, S1)$ need to be calibrated. Similarly, $(d2, S2)$ are required for measuring $D2$. For each encoded points on projection plane $(B(i, j), i = 1...m, j = 1...n)$, a set of sensor parameter matrices $(d(i, j), S(i, j), i = 1, 2...m, j = 1, 2...n)$ need to be calculated. $m \times n$ is the dimension of a point array, which is specified by the number of encoded points projected in camera viewing field.

A linear least square method can then be used to calibrate sensor parameter $d(i, j), S(i, j)$. Illustrated in Figure 2.9, by controlling movements $(h1(i, j)...hn(i, j))$ of reference plane n times, a set of $(L1(i, j)...Ln(i, j))$ can be measured using GCLS

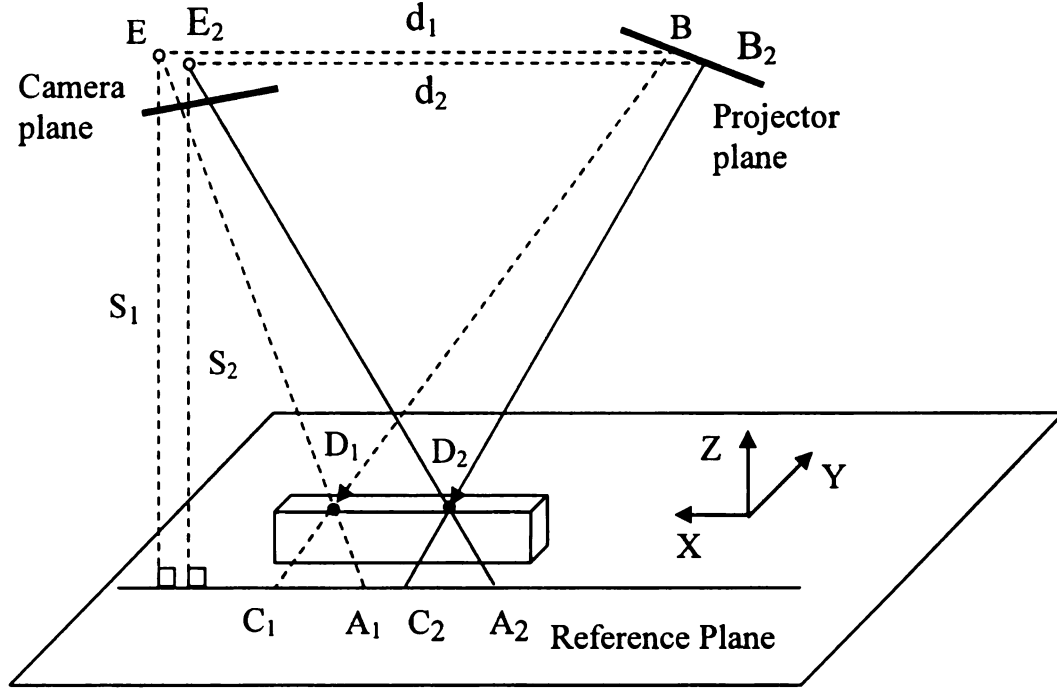


Figure 2.8. Area sensor calibration using pixel-to-pixel strategy

method, then $d(i, j)$, $S(i, j)$ are calculated by Eqn.(2.7).

$$\begin{bmatrix} h_{1(i,j)} & -L_{1(i,j)} \\ \vdots & \vdots \\ h_{n(i,j)} & -L_{n(i,j)} \end{bmatrix} \cdot \begin{bmatrix} d(i, j) \\ S(i, j) \end{bmatrix} = \begin{bmatrix} -h_{1(i,j)} \times L_{1(i,j)} \\ \vdots \\ -h_{n(i,j)} \times L_{n(i,j)} \end{bmatrix} \quad (2.7)$$

Therefore, the surface height map h can be calculated using Eqn.(2.8): where i and j represent the image coordinates, row and column.

$$h(i, j) = \frac{L(i, j) \times S(i, j)}{d(i, j) + L(i, j)} \quad (2.8)$$

Eqn.(2.7) only gives the solution to calibrate the baseline distance d and standoff S . It is assumed that the distance L is measured accurately. Error in the measurement of L will cause calibration uncertainties. The previous section focuses on image analysis for better measurement of L . However, inaccurately positioning the camera and the projector will cause a system offset angle α , perpendicular to stripes, which

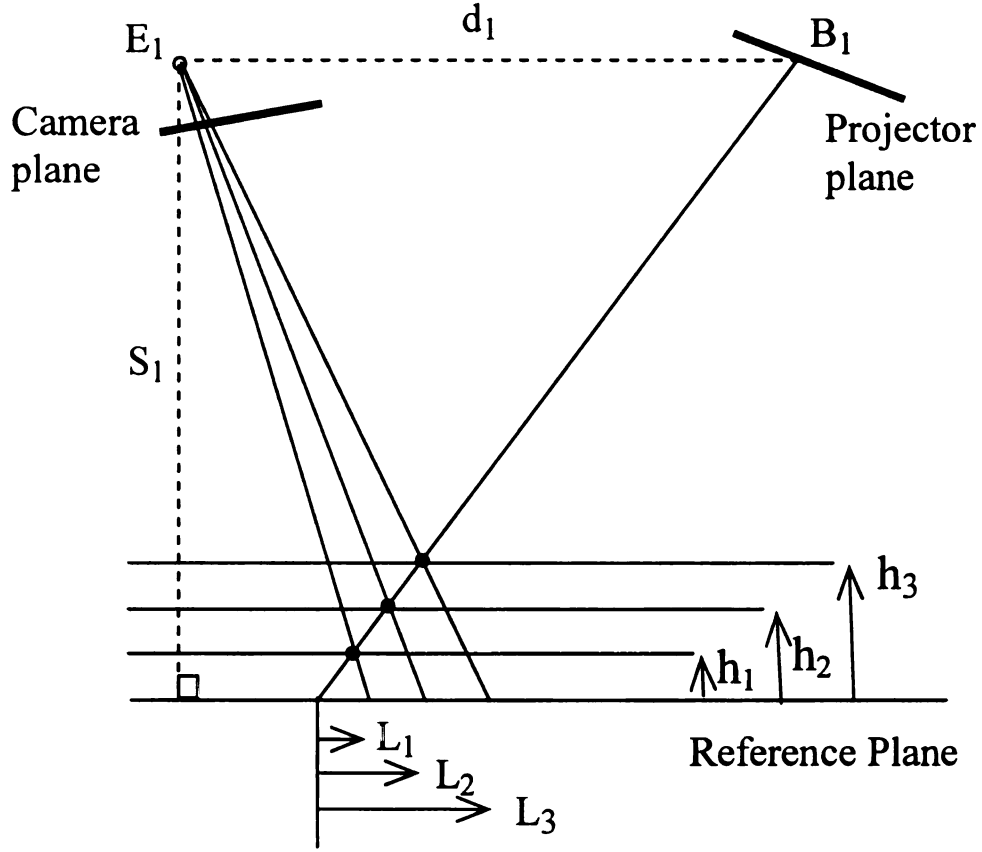


Figure 2.9. Calibration of sensor parameter d_1, S_1 using a linear least square method

creates error to L . Shown in Figure 2.10: point D has height H_D to reference plane that is going to be measured. AC and $A'C'$ are on the projected lines of MP and $M'P$ on reference plane, respectively. Line AC is perpendicular to the coded stripes and line $A'C'$ has an angle to line AC which is the offset angle α . Using a pixel-to-pixel strategy, M is a guess point corresponding to P . But actually M' is the real point corresponding to P . $\triangle M'DP$ and $\triangle A'DC'$ are similar triangles and hence distance L' between points A' and C' should be used to calculate H_D . But, because patterns are usually encoded along either image rows or image columns, codes are identical in one stripe. For example, point A' and A have same codes because they are in a same stripe. If α is not calibrated, instead of L' , L will then be used in calculating H_D .

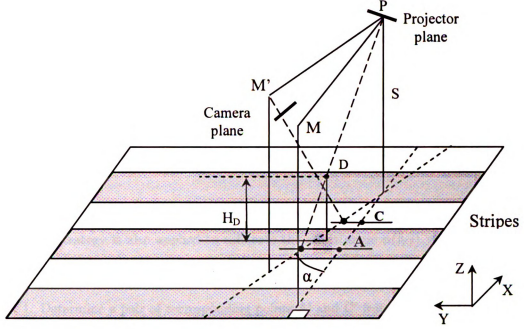


Figure 2.10. Calibrating the offset angle α

Therefore, the length error ΔL between L' and L causes the error ΔH_D in measuring H_D . This system error will bring error into calibration of sensor parameter (S, d) as well. Eqn (2.10) shows the calculation of this error term ΔL and Eqn (2.12) shows the calculation of ΔH_D .

$$L'_{(i,j)} = L_{(i,j)} \sec(\alpha) \quad (2.9)$$

$$\Delta L_{(i,j)} = L'_{(i,j)} - L_{(i,j)} = (1 - \sec(\alpha)) \times L_{(i,j)} \quad (2.10)$$

$$\Delta h_{(i,j)} = h'_{(i,j)} - h_{(i,j)} \quad (2.11)$$

$$\Delta h_{(i,j)} = \frac{L'_{(i,j)} \times S_{(i,j)}}{d_{(i,j)} + L'_{(i,j)}} - \frac{L_{(i,j)} \times S_{(i,j)}}{d_{(i,j)} + L_{(i,j)}} \quad (2.12)$$

A regular way used by several commercial 3D scanners is to encode points using both horizontal and vertical stripes, which needs much more time for image processing. The method proposed in this thesis calibrates the offset angle α in off-line mode, which saves time for real-time inspection process. Therefore, as shown in Eqn.(2.10), L' can be derived by the measured L and the calibrated offset angle α . This corrected L' will then be used in Eqn.(2.8) to calculate the relative height of a surface point to the reference.

Similar to sensor baseline and standoff distance calibration, the developed pixel-to-pixel strategy is also applied to calibrate the offset angle $\alpha(i, j)$, which basically has two steps:

1. Determine a pair of corresponding points A' and C' from two sets of images.
2. Calculate the offset angle $\alpha(i, j)$.

As described above, area sensor calibration cannot be done in one step. Sensor parameters need to be calibrated one-by-one by using the proposed pixel-to-pixel strategy. Generally, calibration can be conducted by following sequence:

1. Setup calibration reference with respect to the area sensor.
2. Determine camera resolution res , this can be done easily by counting the size of view field and the number of camera pixels.
3. Record images, to each coded point, calculate the installation offset angle $\alpha(i, j)$
4. To each coded point, calculate the baseline distance $d(i, j)$ and $S(i, j)$ using the linear least square method.

2.3 Developing a back-imaging system for reflective surface measurement

In automotive glass manufacturing industries, there is an increasing requirement to quickly determine the 3D shapes of glasses for manufacturing inspection using vision based 3D sensing techniques. However, physical properties of the specular surfaces such as reflectivity and transparency do not allow us to apply the triangulation-based techniques for measuring a reflective surface. This is because the projected light is mainly reflected to a single direction and little light reaches the sensing devices. Figure 2.11 illustrates the reflection phenomena of a point light source to a reflective surface and diffuse surface.

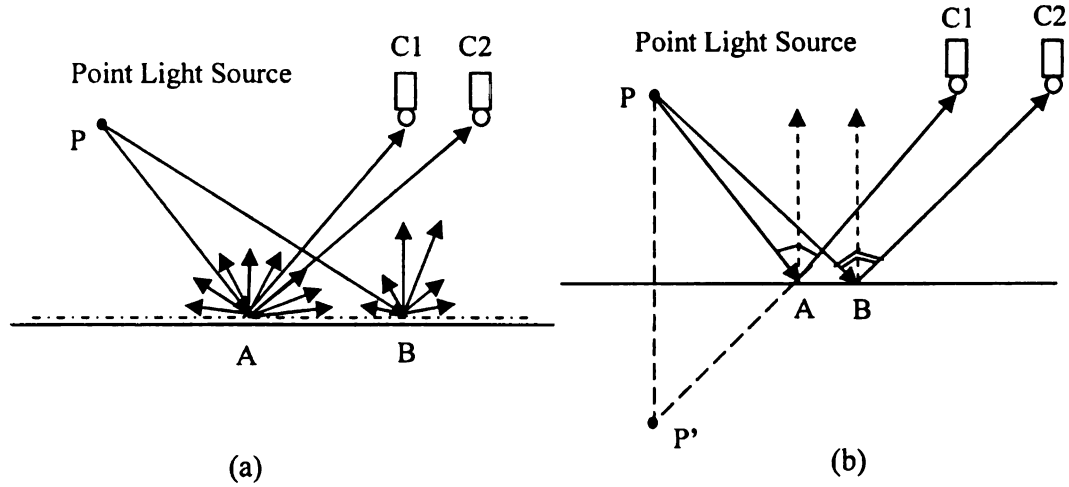


Figure 2.11. Reflection of a point light source on a reflective surface and a diffuse surface. (a) Reflection on a diffuse surface, (b) Reflection on a reflective surface.

For a point light source P , if it emits light to a point A or B on a diffuse surface as shown in Figure 2.11(a), the incident light will be scattered in all directions such that both cameras $C1$ and $C2$ will be able to record an image of point A or B on the surface. However, on a reflective surface, the reflection will go a single direction so that reflection of point A can only be recorded by camera $C1$ whereas reflection of

point B can only be recorded by $C2$.

Specular reflections of an object are determined by the surface normal and can therefore lead to the effect that for a same projected pattern, various images can be perceived at different sensing positions. However, this reflection property can also be used to implement a new measurement system using similar ideas from existing 3D imaging techniques. In other words, the reflective surface can be used as a mirror to reflect encoded patterns from a screen. Hence, a novel 3D shape measurement system is proposed in this thesis, which is going to be developed for automotive glass inspection.

2.3.1 Projection Screen

Considering the reflection property of reflective surfaces, it is obviously that a point light source such as a DLP projector is not proper for measuring a reflective surface. Instead of, a diffuse-based light source is necessary for constructing a projection pattern. As shown in Figure 2.12, a projection screen is used, each point will emit lights to the part surface. Under this circumstance, the measured part surface functions as a mirror. And an virtual image, so called “back image” is generated accordingly. Therefore, for a fixed camera $C1$, there is at least one incident light of a point in the sensing region BC to be detectable. The method, for measuring reflective surfaces like an automotive glass is then called a “Back-imaging” method.

2.3.2 Unknowns variables of a back-imaging system

The triangle-based 3D sensing method has only one unknown variable to solve, relative height h . In contrast, there are two unknown variables, position and orientation of a surface point, need to be determined for measuring a reflective surface. The triangle-based model cannot solve two unknown variables simultaneously.

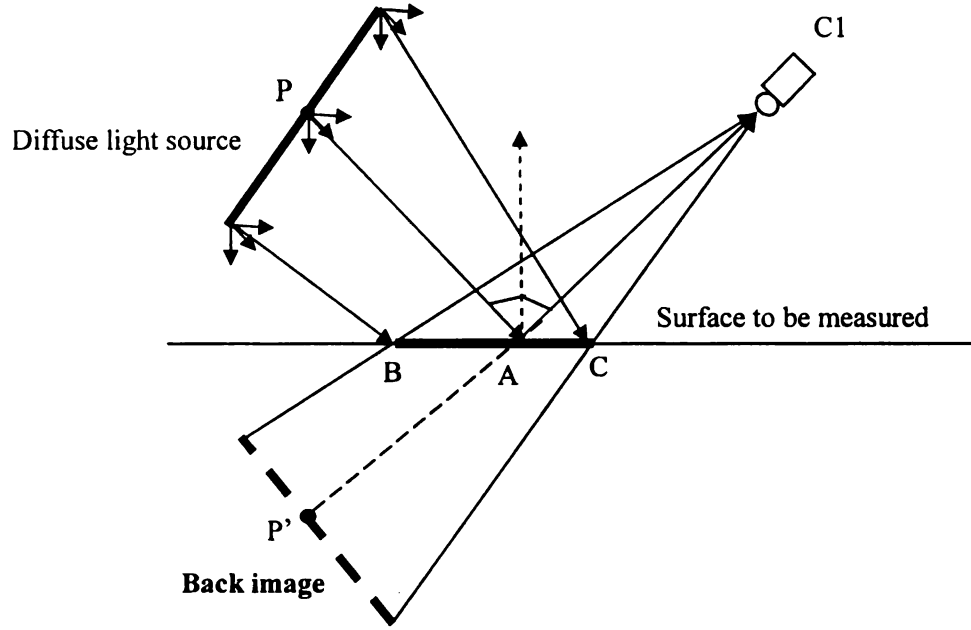


Figure 2.12. Measuring a reflective surface using diffuse-based projection method

2.3.3 Measurement principle

For those two unknown variables, a solution exists as long as any of the two unknowns is acquirable. By utilizing the CAD information, the normal vector V can be initially estimated. Therefore, with knowing vector U and V , we can determine W and hence the coordinates of point P . While a certain uncertainty is presented in calculating vector V , we still can search a proper location on vector U around the initial point P to find a best estimation of vector W and then determine the point P accordingly.

As shown in Figure 2.13, coordinates of projection point $P1$ is predetermined in the world frame. Location and orientation of the camera is also calibrated. Therefore, the viewing vector U becomes known to each pixel on the image plane. With knowing the image point of $P1$, vector U is specified and then the position of surface point P , is related to its surface normal Vs : Vs evenly divides the angle between vector U and vector W , according to the reflection law.

In the proposed iteration-searching algorithm, unknown variables position P and

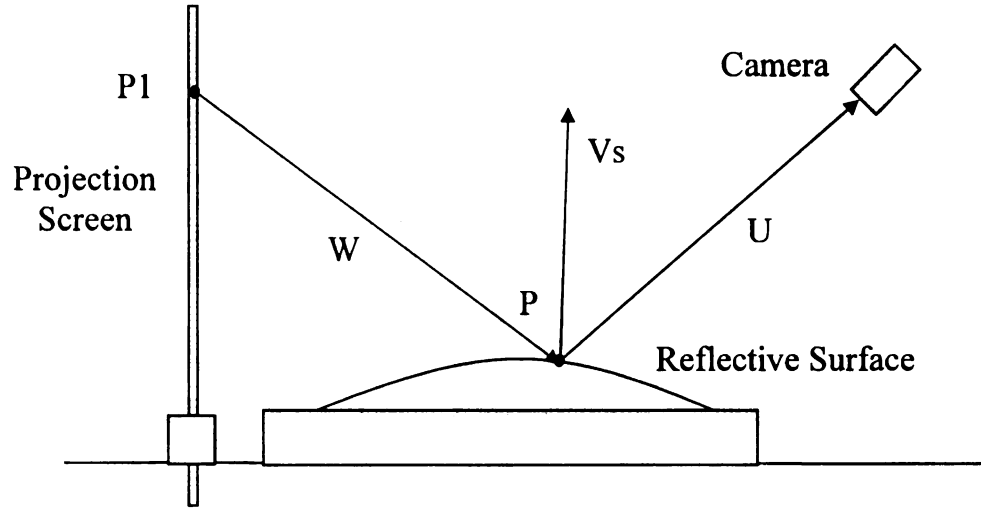


Figure 2.13. 3D shape measurement of reflective surface using recursive searching

normal V_s can be initially calculated by the intersection of vector U and the tessellated CAD model. However, on real surface, the reflection point will stay in the designed surface, and therefore a manufacturing error distribution needs to be detected. Actually, finding the error between the real surface and the designed surface is exactly the purpose of this inspection system. However, by starting with this initial point, the iteration-searching process can be speed up significantly because the real surface point usually exists closed to this initial point.

Figure ?? shows the developed iteration-based searching method. Point $P1$ is a marked point on the diffused-based projection screen, from the mirrored back image, the pixel location of $P1$ can be determined. And therefore, vector $U1$ can be determined through camera calibration. Details of image analysis for calculating point $P1$ and $U1$ will be introduced in next section.

An initial point $P2$ is first calculated by the intersection of vector $U1$ with one of triangles tessellated from the CAD model. Then by testing point $P3$ on the designed surface, a new position $M3$ and $V3'$ will be obtained as an update of point $P2$ and its vector $V2$. Meanwhile, the projection vector $W3$ is calculated to replace the

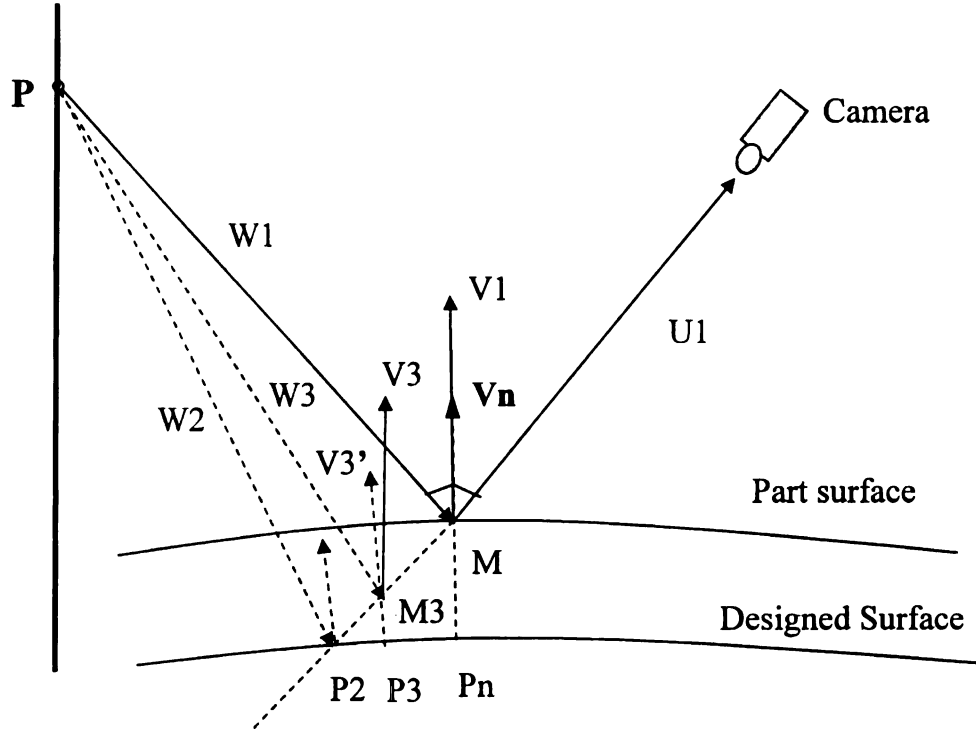


Figure 2.14. 3D shape measurement of reflective surface using recursive searching

initial vector $W2$. By checking the normal of $U1$ and $W3$, vector $V3$ is determined as the “correct” surface normal. Evaluating the angle between vector $V3'$ and $V3$, a new point and new vector will then be estimated in the next step. This process recursively executed until a point Pn is found such that on position M , the surface normal $V1$ separate the angle between projection vector $W1$ and view vector U , i.e. angle $a1 = a2$.

2.3.4 Measurement using the Golden Section method

For the developed searching method, point on CAD surface has to be tested one-by-one. It will cost much time to reach an optimized solution. An alternative way uses golden section method in a quick searching of the solution. As shown in Figure ??, after initial position $P2$ and vector $V2$ are calculated, position $P3$ can be calculated

according to the estimated surface error Δ using Eqn.(2.13).

$$L_{P2 \rightarrow P3} = 2 \times \Delta \times \cotan(\beta) \quad (2.13)$$

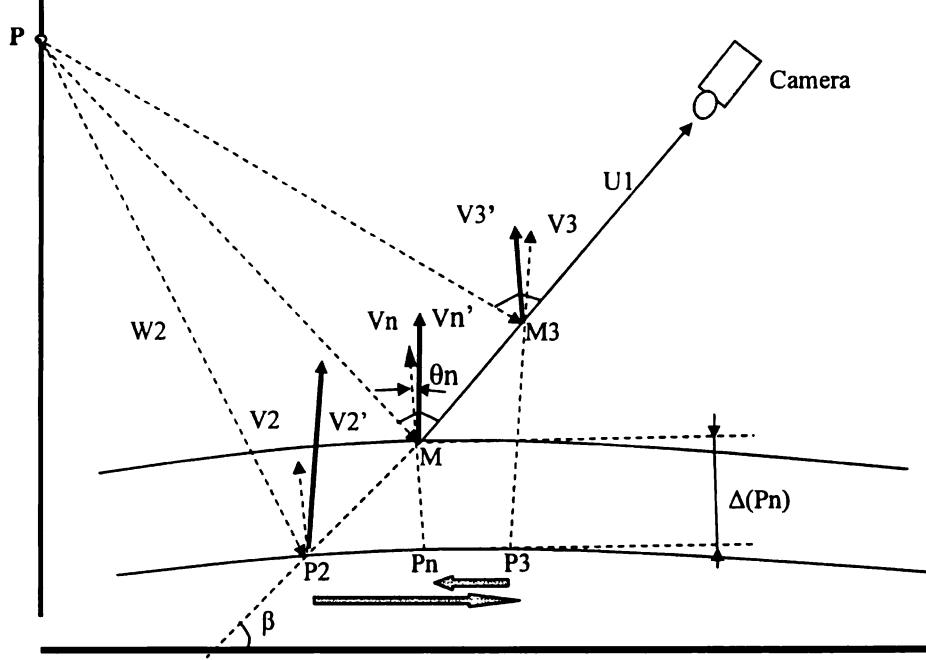


Figure 2.15. An accelerated searching using the golden section method

where the estimated surface error Δ can be a predetermined largest error that has been reported in a manufacturing process. Vector $V3$ is CAD normal on point $P3$, point $M3$ can then be determined on the view vector $U1$. By checking the offset angle $\theta3$, the iteration process then use golden section method to determine point $P4$ on CAD surface, and hence $V4$, offset angle $\theta4$ etc. The process stops after n steps when the offset angle $\theta(n)$ is less than a threshold value. Point M is then decided as the surface point of reflection corresponding to view vector $U1$ and marker point P . This process is also described in a flow chart in Figure 2.16.

According to the above discussions, two issues are critical to the measurement accuracy of the designed back-imaging system:

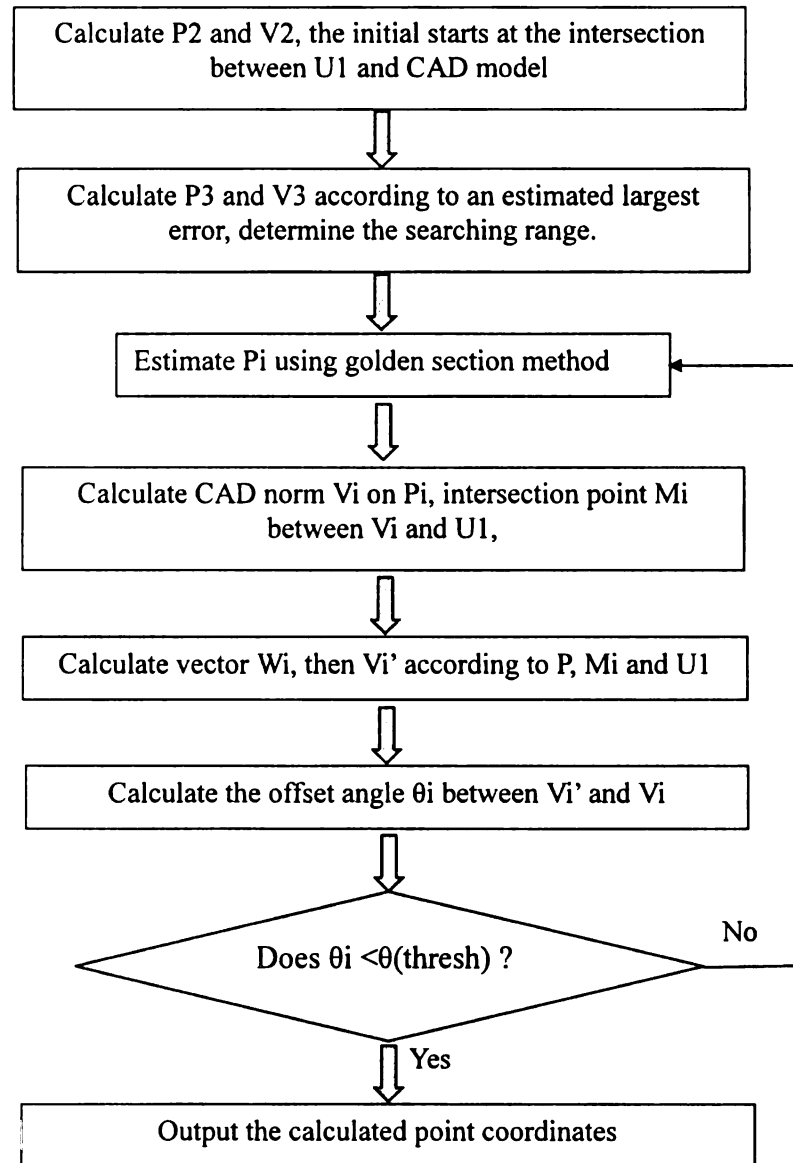


Figure 2.16. The flow chart of the golden section searching method

1. Does U1 point to an image point $I(i, j)$ corresponding to marker point P?
2. Is U1 the correct vector for the image point $I(i, j)$?

The first issue is related to image processing method in calculating the image coordinates of marker point P. Since a glass is usually laminated with multiple layers,

a incident light will have a second reflection from the bottom layer of glass which may noise the image for position identification. The second issue is related to the calibration of viewing vectors. The next two sections will introduce methods in solving these two problems.

2.4 A Multi-Projection based Back-imaging System

Another innovative multi-projection method is also developed to solve those unknown variables simultaneously. As shown in Figure 2.17, the reflection vector U , the projection beam vector W , and the surface normal vector V have an intersection right on the part surface. Since vector U can be determined by camera, then as long as the projection vector W is known, the coordinates and surface normal at point P can be easily calculated by vector U and W .

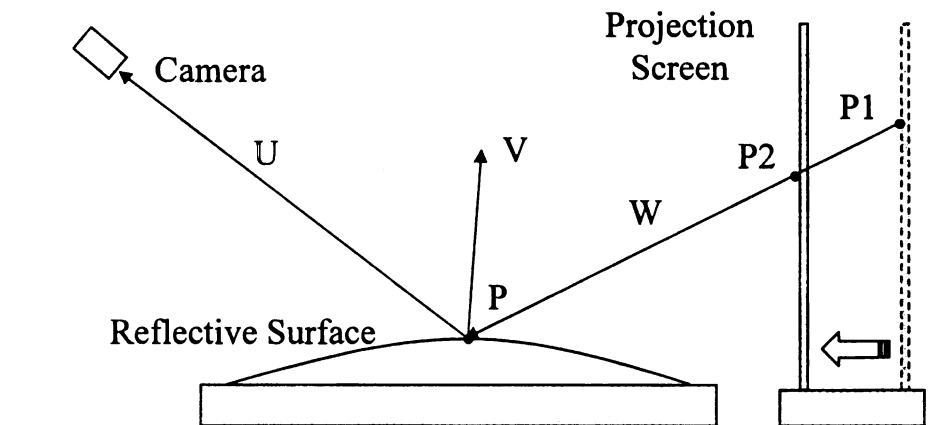


Figure 2.17. 3D shape measurement of reflective surface using multi-project technique

To obtain this projection vector W , the encoded projection screen can be moved toward the camera several times, the correspondent point can then be identified that forms the projection vector W . Though the algorithm is fairly much easier than

the iterative-searching method, physically implement this method is much complex because the mechanic motion needs to be precisely controlled. The cost of such system will be much higher than the iterative searching based method. In the other aspect, the entire projection screen has to be encoded for measurement. The algorithm for image coding and decoding costs more time than the iterative searching method, which makes it not a practical way for an industrial application.

2.5 Bottom layer reflection problem

When an electromagnetic wave, a visible light in this research, passes a layer of another media from air, reflection and refraction will both happen simultaneously. The refracted light, will be reflected again at the other side of media as can be seen in Figure 2.18. Notice that the bottom layer reflection is parallel to the top

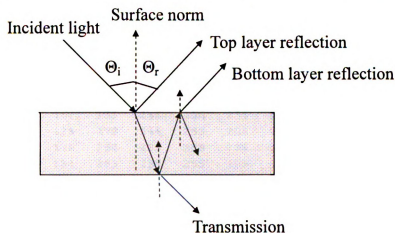


Figure 2.18. Bottom layer reflection

layer reflection but intensity is dramatically reduced because most of light energy is transmitted through the glass. This phenomena is illustrated in Figure 2.19.

For image processing, if a white dot is used to mark a point, the target circle will be blurred at bottom. Then, the real location of the target point cannot be distinguished

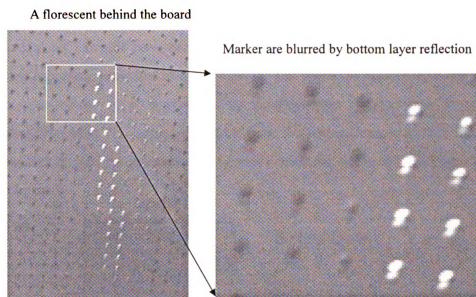


Figure 2.19. Bottom layer reflection of a glass

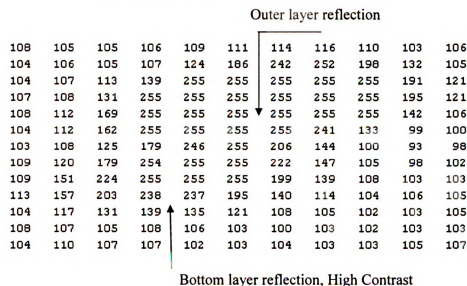


Figure 2.20. An intensity chart around a white dot

because the CCD sensor is saturated. Such an intensity map around a white dot can be seen in Figure 2.20. However, since the energy of the bottom reflection will be

reflection will be smaller than the top layer reflection, a black dot can be used as a mark point. In this reversed situation, the real location of a target point and blurred region can be separated in image analysis, and hence the bottom layer reflection problem is solved. As a consequence, the image position $I(i, j)$ of point P can be precisely identified. The principle of distinguish the bottom layer reflection is shown in Figure 2.21, and the intensity map around a black dot can be seen in Figure 2.22.

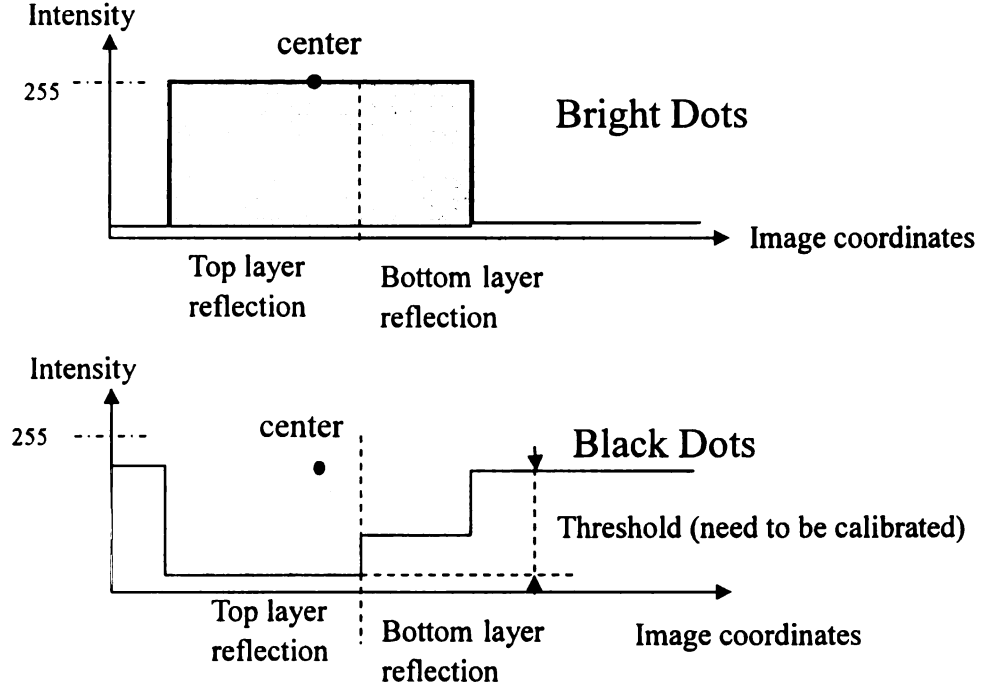


Figure 2.21. Distinguish the bottom layer reflection using a black dot pattern

Considering certain image noise may exist, an intensity weighted estimation method, described by Eqn.(2.14) is used to calculate the center of the marker.

$$x_C = \frac{\sum_{i=1}^k ((I_m - I_i) \cdot X_i)}{k \cdot I_m - \sum_{i=1}^k I_i} \quad (2.14)$$

With this identified image point $I_{(x_C, y_C)}$ of the mark point P, the corresponding viewing vector $U_{(x_C, y_C)}$ can be obtained by a vector-based camera calibration method that is introduced as following.

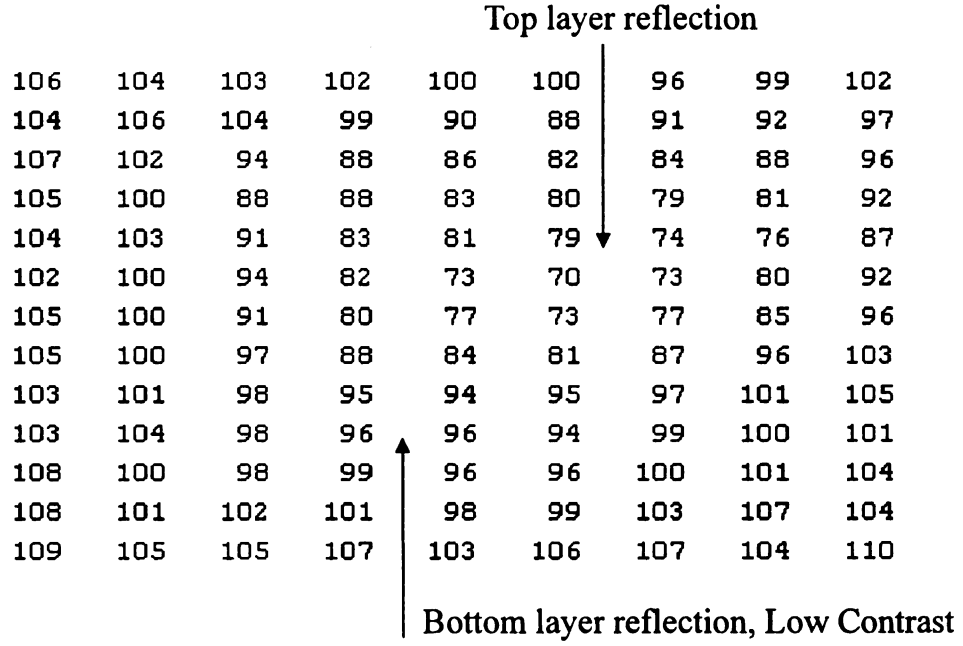


Figure 2.22. An intensity chart around a black dot

2.6 Vector-based system calibration

Camera calibration is required in most vision metrology systems. As discussed in Chapter one, a pin-hole lens model is often used to calibrate the intrinsic and extrinsic parameters of the camera. However, for advanced lens used in a metrology system. Every sensing pixel needs to be calibrated one-by-one to satisfy the requirement of measurement accuracy. Described in previous section of this chapter, a pixel-to-pixel strategy is developed to calibrate an area sensor. This strategy can also be applied to the back-imaging system in deriving the viewing vector for each pixel. This calibration technique, focuses on derives a set of vectors for image pixels, is called vector-based camera calibration.

Figure 2.23 illustrates the method to calculate the view vector $U(i, j)$ of an image pixel $I(i, j)$. By raising up a predefined reference board, the view vector $U(i, j)$ can be determined by point $P1(i, j)$ and $P2(i, j)$. Though the algorithm is not difficult,

challenge to develop a system to calibrate all pixel simultaneously.

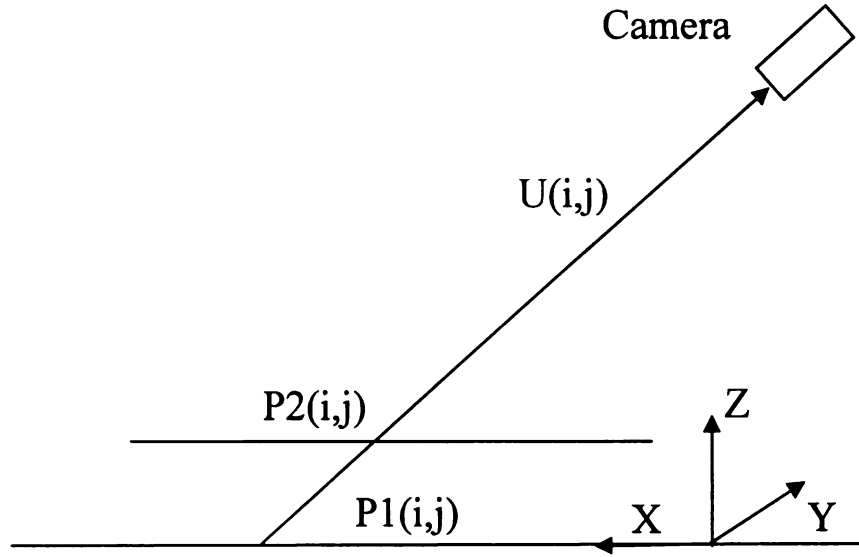


Figure 2.23. Calibrate the view vector of an image point

Figure 2.24 shows an automatic calibration method proposed for the calculation of point $P1_{(i,j)}$ and $P2_{(i,j)}$ from image analysis. By setting up a grid pattern of dots in the world frame, one image is recorded and each intercross dot in Figure 2.24 can be estimated using image analysis technique. Those marker points then function as reference points for calculation of 3D coordinates of any other projected image point.

Theoretically, in image analysis, point C1 and C2 are calculated first in the image coordinate frame. The ratio of R1C1 to C1R2, $\lambda(1)$, and R1C2 to C2R3, $\lambda(2)$, is then derived. With knowing the ratio $\lambda(1)$ and $\lambda(2)$, the location of point $P_{(i,j)}$ can then be estimated. However, in reality, the prospective model of the camera lens distorts the square pattern on the image, which forms a pattern as shown in Figure 2.25:

Therefore, the coordinates of point C2 need to be calculated within this perspective image model. Position of $U_{(i,j)}$ is then calculated using the reference markers and coordinates of C1 and C2.

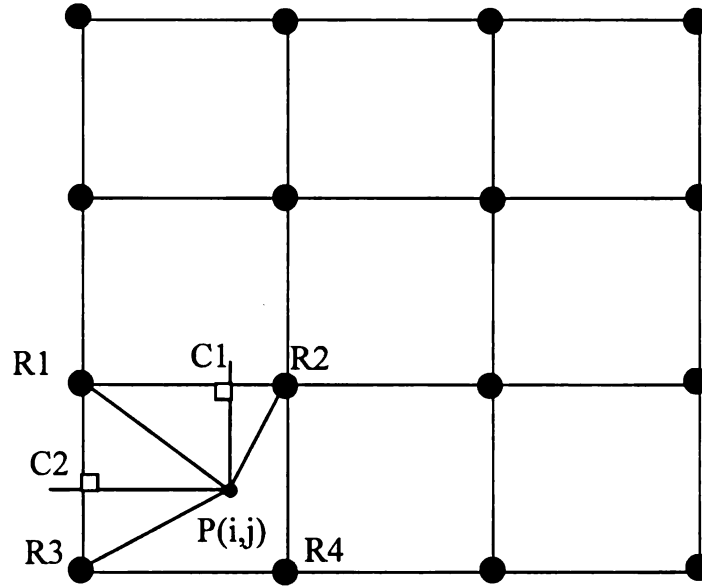


Figure 2.24. Estimate the world coordinates of an image point, a theoretical model

2.7 Chapter Summary

This chapter introduces two measurement systems, for the inspection of lambertian surfaces and reflective surfaces. For a part with a lambertian surface, a robot-integrated automated dimensional inspection system is developed. In contrast with the common strategy the using phase shifting method, a line shifting method is developed in this thesis, which shows better measurement precision. A pixel-to-pixel calibration strategy is proposed in this thesis that calibrates the area sensor. Though a general approach calibrates the camera and the projector individually in the simplified pin-hole lens model, the pixel-to-pixel strategy is developed based on the real lens system. It is therefore much more reliable because each pair of encoded point is calibrated within the triangle model of an area sensor.

For a part with a reflective surface, such as automotive glass, an innovative back-imaging system is developed in this thesis. Using a diffuse-based projection screen, an iteration-based searching algorithm is developed to extract 3D shape of the re-

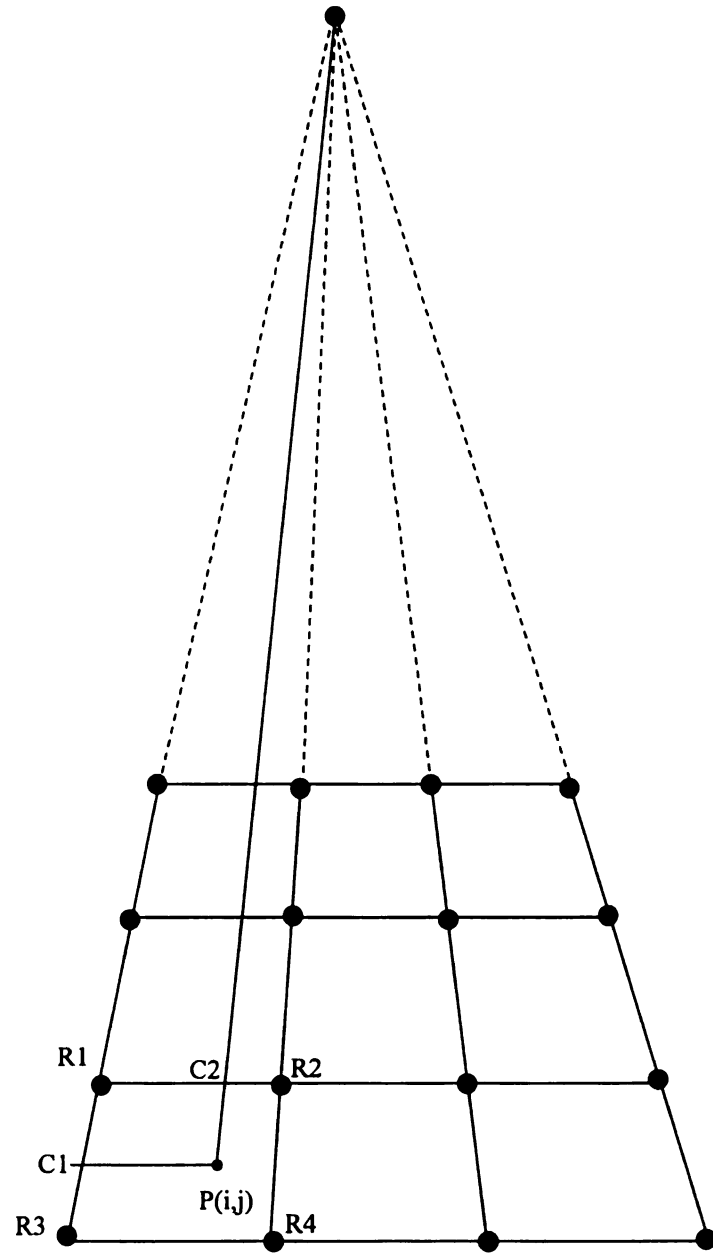


Figure 2.25. Estimate the world coordinates of an image point, a perspective model

flective surface from the mirrored images. The CAD model of the part is necessary in this iteration-based searching method. The golden section method is applied to expedite the searching process. The problems of bottom layer reflection and view

vector calibration are critical to the measurement accuracy. Methods to solve these two problems are introduced in this chapter.

CHAPTER 3

Process Planning and Control of an Automated Dimensional Inspection System

For a large part, multi-view measurement is necessary to obtain a point cloud of the entire part surface. This can be done by moving either the area sensor or the part itself. Moving the sensor or the part is a similar problem considering the geometric relationship between the sensing device and part surface. The problem is how to properly plan and control this motion process to improve the measurement performance.

For measurement on a lambertian surface, the area sensor needs to be moved to different viewpoints to cover the entire surfaces and to improve measurement quality. A robotic integrated dimensional inspection system is necessary for this type of application because:

- The robot can maintain the area sensor steady during the measurement process.
- The robot can automatically move the area sensor, which is especially required for a Flexible Manufacturing System (FMS).

- The inspection process can be totally automated by planning and control of the robot.

3.1 CAD-guided Area Sensor Planning

The structure of an automated dimensional inspection system is shown in Figure 3.1. There are three main steps in this system. The first is sensor planning: task constraints of the area sensor's viewpoints are generated from the model of the area sensor and the CAD model. An area sensor planner estimates viewpoints that satisfy all task constraints. The calculated viewpoints will then be sent to the robot controller.

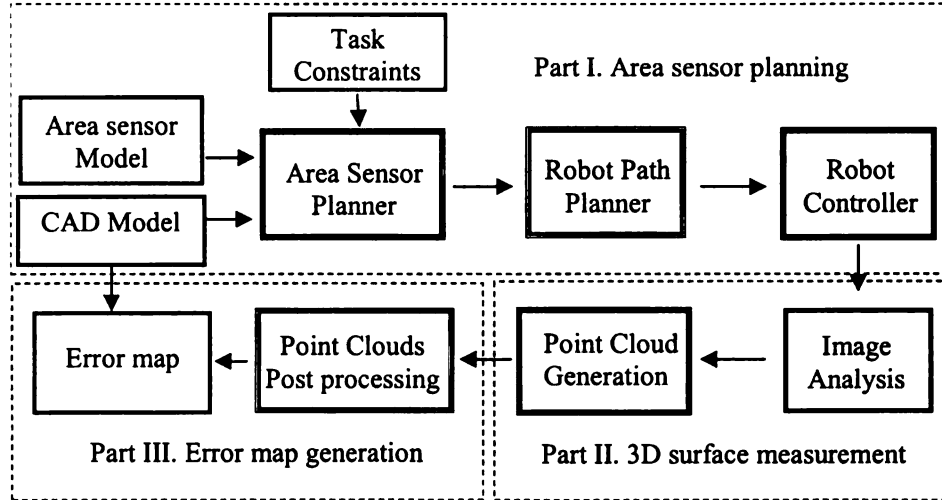


Figure 3.1. Automated dimensional inspection system

Second, the 3D shape of the part surface is scanned. A group of encoded stripes is projected to the part surface. The deformation of the stripes is then used to calculate the 3D shape, as described in the first chapter.

Finally, the measured point cloud is matched and compared to its CAD model. An error map can then be generated for quality evaluation of inspected parts.

A CAD-guided area sensor planning is only a part of the automated dimensional inspection system. Since the geometric shape of the part, the designed CAD model, is usually acquirable in manufacturing industry, it could be very convenient to obtain an initial set of viewpoints that can cover the majority of the part surface. This section describes methods for the CAD-guided area sensor planning.

3.1.1 Task Constraints for CAD-guided area sensor planning

There are five constraints considered in the developed automated dimensional inspection system: visibility, field of view, resolution, point density, and depth of focus. In contrast with previous planning systems, constraints of both the camera and projector have to be satisfied: the visibility constraint demands that less occlusion exist between the part and the area sensor. Two types of occlusions exist: one is along the incident beam of projection and the other is along the reflection beam for camera recording. Shown in Figure 3.2, V is the vector of sensor viewing norm, $V(avg)$ is defined by the average norm of a patch of surface. C_1, C_2 are vectors of camera viewing normal constraints, and P_1, P_2 are vectors of projection normal constraints. S_1, S_2 are vectors of surface normal constraints that are limited by C_1, P_1 and C_2, P_2 . This constraint insures that this patch surface can be illuminated by the projector and images of this patch surface can be taken by the camera. Threshold angles of θ_1 and θ_2 are used to define visibility constraints and angle θ_{tri} represents the angle between any single triangle and the surface average norm, as shown in Eqn.(3.2).

$$\theta_{tri} = \arccos\left(\frac{V_{tri} \cdot V_{avg}}{\|V_{tri}\| \|V_{avg}\|}\right) \quad (3.1)$$

$$|\theta_{tri}| < \begin{cases} |\theta_1|, & \text{sign}(\theta_{tri}) = \text{sign}(\theta_1) \\ |\theta_2|, & \text{sign}(\theta_{tri}) = \text{sign}(\theta_2) \end{cases} \quad (3.2)$$

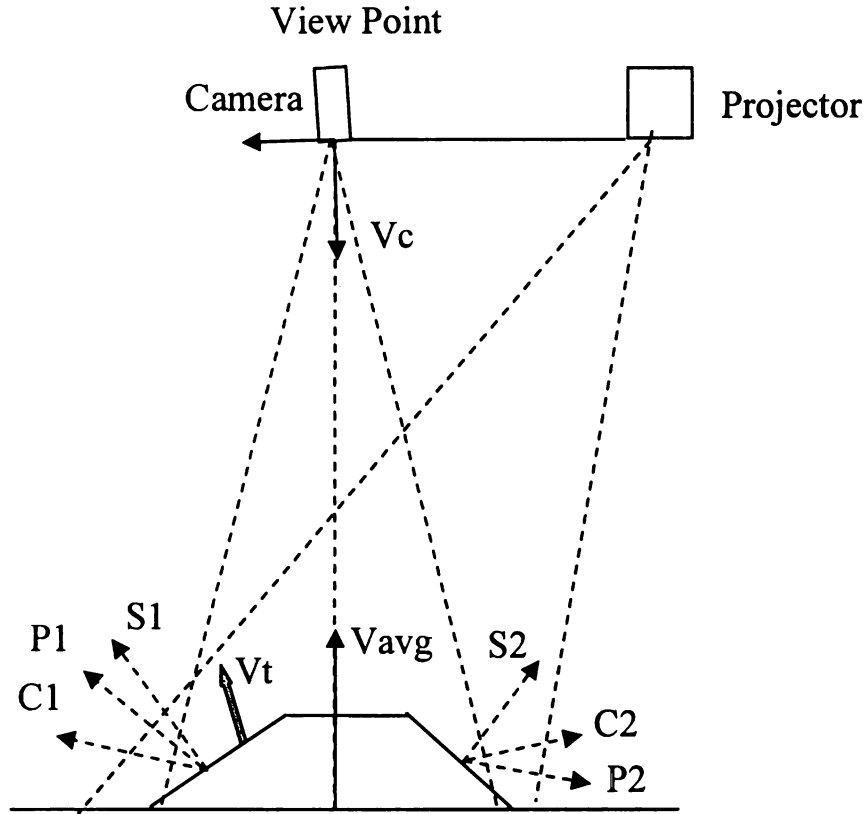


Figure 3.2. Deriving visibility constraints of an area sensor

The given CAD model is first tessellated into triangles. Then, based on the normals and areas of triangles, a clustering algorithm can be used to separate all triangles into a set of smooth patches. The whole process for generating a candidate patch is illustrated in Figure 3.3.

The other four constraints are described as follows:

1. Field of view determines the size of maximum inspection area. It is usually a rectangular field. In the developed area sensor planner, it was determined by the standoff and camera viewing angles.
2. Resolution defines the entity's minimal dimension to be mapped onto one camera pixel.

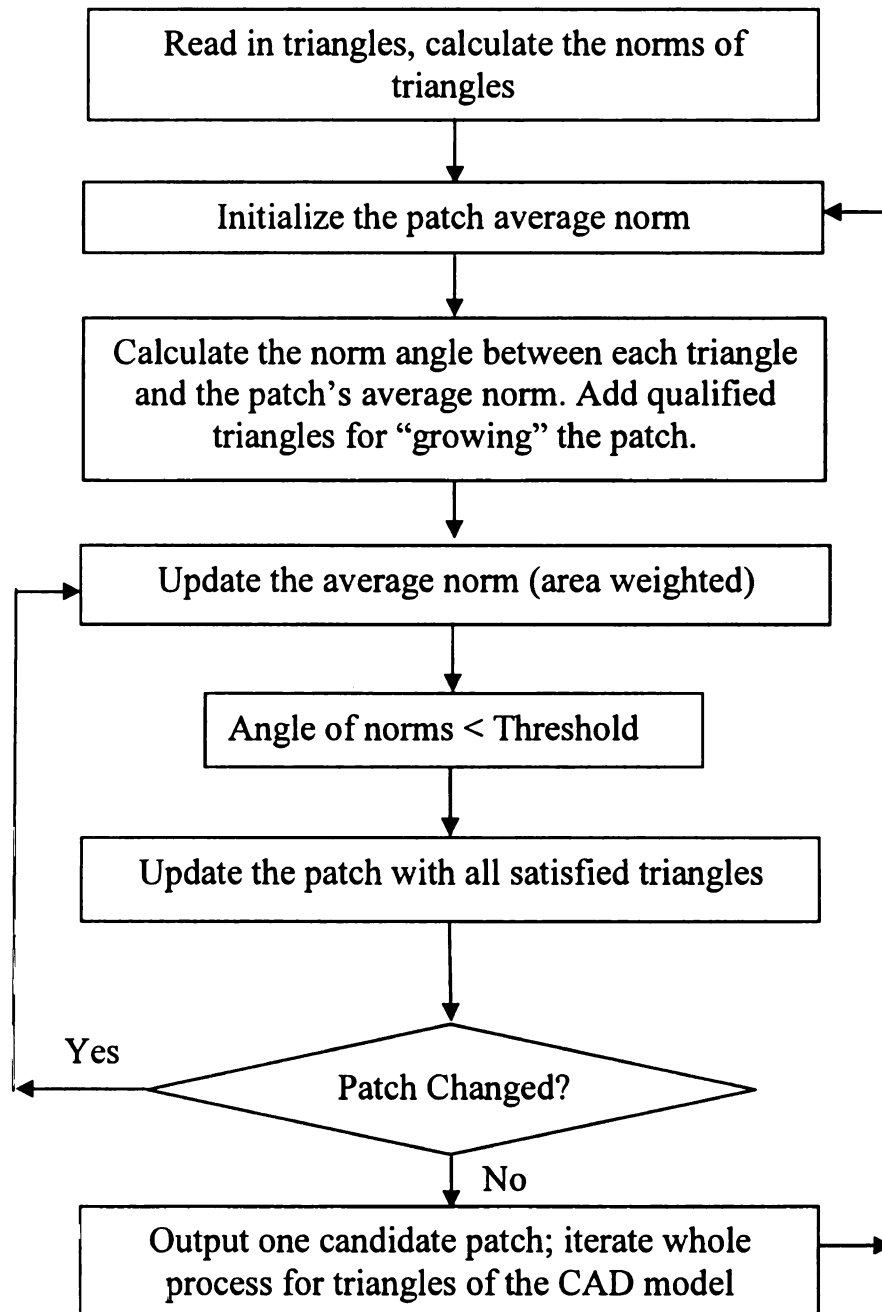


Figure 3.3. Clustering the triangles of a CAD model according to the visibility constraint

3. Point density is a constraint determined by the field of view and the resolution of the projector, which is a new constraint developed for the automated area

sensor planning system. Point density constraint ensures that enough points can be measured for certain area of surfaces.

4. Focus constraint defines the farthest measurement distance and the nearest measurement distance from a viewpoint.

A bounding box method is developed to integrate all constraints of both a camera and a projector for searching viewpoints. Illustrated in Figure 3.4, a candidate patch is concluded in a bounding box, the width and the length of the box specify the field of view and the height of the box specifies the farthest and nearest focus distances. A viewpoint can be estimated on the center line that passes through the bounding box.

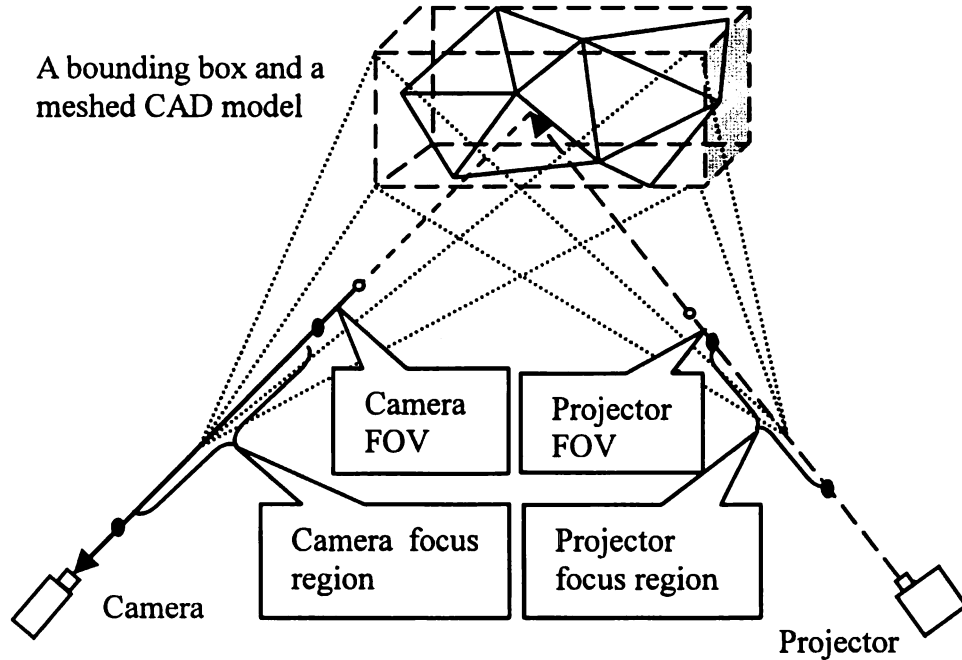


Figure 3.4. Integration of planning constraints using a bounding box method

$$\min_{\vec{P}} \sum_{i=1}^N F(C_{i1}, C_{i2}, \dots, C_{i5} \in R_C \mid P_{i1}, P_{i2}, \dots, P_{i5} \in R_P) \quad (3.3)$$

The searching algorithm can be described by Eqn.3.3, where \vec{P} represents the potential viewpoints, N represents the number of meshed triangles in the inspected field, $C_{i1}, C_{i2}, \dots, C_{i5}$ and $P_{i1}, P_{i2}, \dots, P_{i5}$ are camera and projector constraints respectively. Projector constraints have to be satisfied first, and then camera constraints are considered. Because the viewpoint may exist in a region, a pose with the minimized measurement error can then be estimated.

Figure 3.5 illustrates this searching algorithm for viewpoint generation: for a candidate patch, a bounding box will be calculated first, then if all measurement constraints cannot be satisfied, the patch in this bounding box will be split into two patches, and each patch has a new bounding box generated for another test. The whole process is iteratively executed until a qualified viewpoint is reported.

3.2 Feedback Control of the Automated Dimensional Inspection System

CAD-guided area sensor planning scheme is only an open-loop system, which can be simplified as shown in Figure 3.6. Viewpoints of an area sensor are generated from the off-line robot sensor planner through a CAD model, a sensor model, and required task constraints. The quality of the measured point clouds totally relies on the off-line programmed viewpoints. In a real measurement, location and area of “holes” are usually unpredictable or hard to compute. For example, for “holes” generated by light reflection, although the center of a brightness spot can be estimated using Fresnel’s law, estimation of the size of a spot is extremely hard because it depends

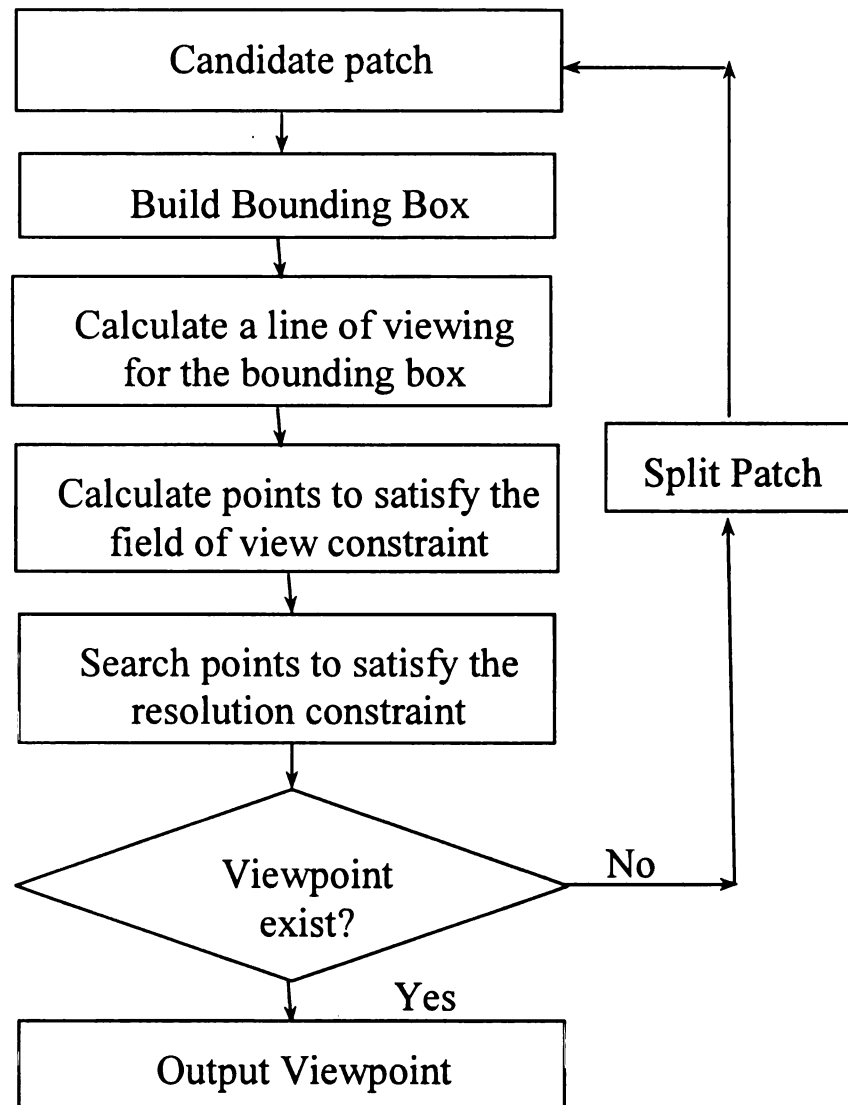


Figure 3.5. A recursive viewpoint searching algorithm.

on the material and smoothness of part surfaces.

For 3D dimensional inspection purpose, measurement quality is the optimization criteria in designing a sensor planner. Completeness is one of the requirements of a “qualified” point cloud. Accuracy can also be improved by using redundant data. Both of them can be optimized by relocating an area sensor.

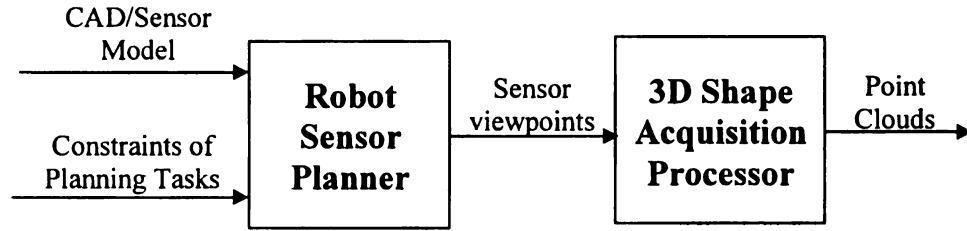


Figure 3.6. An simplified open-loop robot sensor planning scheme

This section describes a closed-loop robot area sensor planning system for automated dimensional inspection. Continuing from previous research, dynamic inspection methods are introduced in a general framework, which integrates model-based scheme and real-time sensing feedback for covering the unpredictable “holes” and also improving measurement accuracy. The major advantages of this dynamic system are:

1. A model-based sensor planner can initialize a set of viewpoints which covers the majority of part surface in a short time.
2. The point cloud is output only when it is “qualified”, which is predetermined as a system set point. This criterion provides a standard evaluation of a point cloud so that it can be used for comparing to the CAD model to generate an inspection error map. An “unqualified” point cloud may cause loss in discarding good parts and keeping parts with wrong shapes.
3. Defects in real measurement, a light reflection spot or a piece of shadow region in an image, can be recovered by predicting another viewpoint automatically.

This feedback system can be illustrated in Figure 3.7. After a set of viewpoints is initialized from a given CAD model, an online sensor planner can add new sets of viewpoints to improve the quality of measured point clouds.

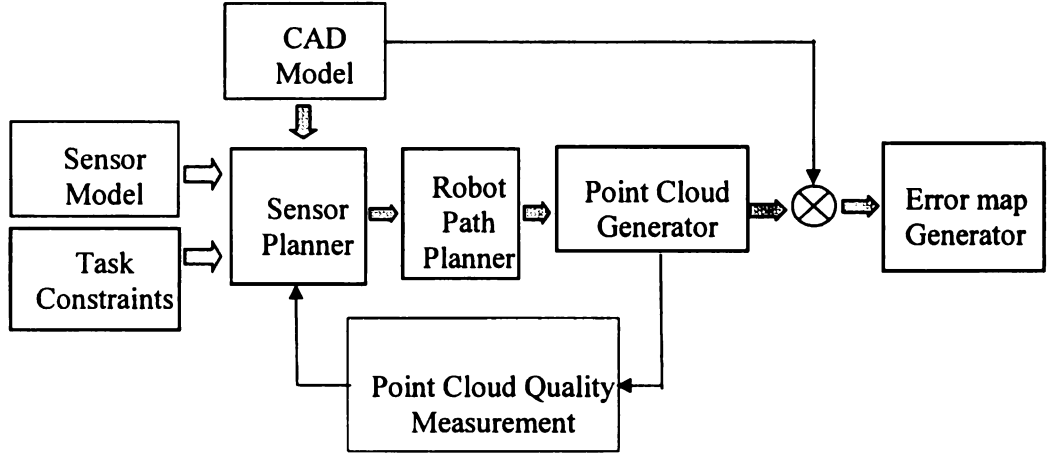


Figure 3.7. A close-loop robot sensor planning scheme developed for a 3D dimensional inspection process

3.2.1 Model of the closed-loop dynamic inspection process

To acquire the qualified 3D point clouds, the developed closed-loop system includes a group of methods in different areas: a model based sensor planner using a bounding box method, image segmentation and recognition techniques, a 3D point filtering method, a point cloud registration strategy, and a pixel-to-pixel calibration method. Figure 3.8 shows the diagram of the proposed feedback system, which integrates all methods together for automatical dimensional inspection:

3.2.2 Model of system components

As shown in Figure 3.8, the proposed system contains four functions: dynamic sensor planner, point cloud generator, viewpoint evaluator, and error map generator.

1. Dynamic Sensor Planner: The dynamic sensor planner has four inputs: CAD model, area sensor model, task constraints, and feedback information. Viewpoints are output of the planner.

- 1.1. The part's CAD model is tessellated into K triangles. Triangle norm can be

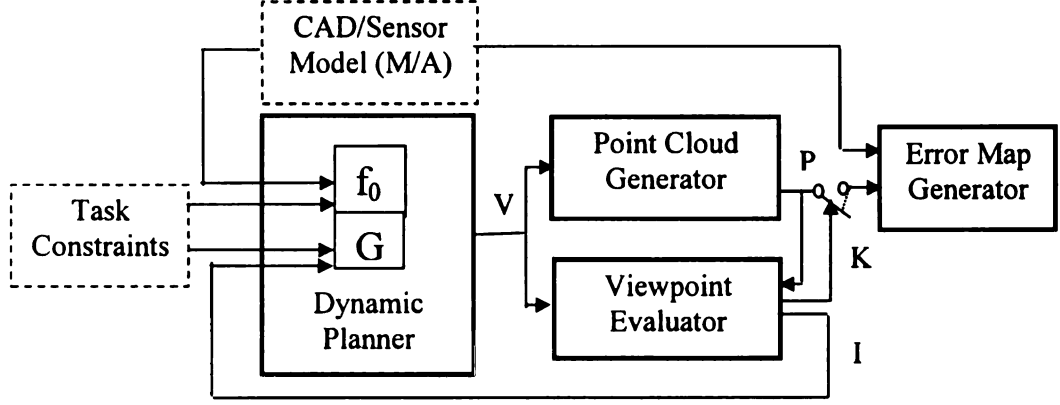


Figure 3.8. Diagram of a sensor planning system using dynamic methods for 3D shape inspection of automotive parts

determined by the order of its three vertexes: $p1 \rightarrow p2 \rightarrow p3$.

1.2. TC is a set of task constraints of the area sensor:

$$TC = \{fov, S, \rho, f_d, \eta\} \quad (3.4)$$

where the field of view fov is defined by the length L and width W of a rectangle area; S is the standoff distance of an area sensor; ρ represents the image resolution; f_d represents the focus distance which contains two values: nearest focus distance, and farthest focus distance; η represents visibility of the area sensor, determined by three vectors: projection vectors \overrightarrow{PV} , camera viewing vector \overrightarrow{CV} , and surface norm vector \overrightarrow{SV} . A piece of surface is visible if the following equation is satisfied:

$$\begin{cases} \arccos(\frac{\overrightarrow{PV} \cdot \overrightarrow{SV}}{\|\overrightarrow{PV}\| \|\overrightarrow{SV}\|}) < \theta_{th1} \\ \arccos(\frac{\overrightarrow{CV} \cdot \overrightarrow{SV}}{\|\overrightarrow{CV}\| \|\overrightarrow{SV}\|}) < \theta_{th2} \end{cases} \quad (3.5)$$

where θ_{th1} ensures that the encoded patterns can be projected onto the surface and θ_{th2} ensures that this piece of surface can be “seen” by camera.

1.3. An area sensor model can be described by Equation (3.6), which is used in

many triangle-based area sensors:

$$h = \frac{L \cdot S}{d + L \cdot S} \quad (3.6)$$

1.4. Defect map I is a group of 3D points which are the output of the view-point evaluator. Usually, three types of defects are considered: a shadow map IS , a reflection map IR , and an inaccuracy map IA .

1. Shadow map IS : 3D Measurement within a shadow is not possible because no projected patterns can be detected by the camera. A shadow map is a group of points extracted from the point cloud at the boundary of a shadow region.

$$IS = \{p_i(x_{si}, y_{si}, z_{si}) | i = 1, 2 \dots ks\}, \quad (3.7)$$

where ks represents the number of points extracted from a point cloud.

2. Reflection map IR : Similar to a shadow map, a reflection map is a set of points which is extracted from the boundary of a glossy spot. Measurement in bright spots is not available because the projected stripes cannot be seen by the camera as well:

$$IR = \{p_i(x_{ri}, y_{ri}, z_{ri}) | i = 1, 2 \dots kr\} \quad (3.8)$$

where kr represents the number of points extracted from a point cloud.

3. Accuracy map IA : According to measurement error analysis in Chapter 2 of this thesis, accuracy is not evenly distributed over a point cloud. For a single surface point, the measurement accuracy is related to the sensor viewpoint. Differences of measurements on two viewpoints generate an accuracy map IA :

$$IA = \{p_i(x_{ai}, y_{ai}, z_{ai}) | i = 1, 2 \dots ka\} \quad (3.9)$$

where ka represents the number of 3D points in a region where the measured 3D shape is different with the previous measurement.

1.5 A viewpoint includes a location p and a viewing vector v . V_0 represents the initial set of viewpoints generated from CAD model, and V_k represents the sets of viewpoints estimated from feedback information, k is used to represent the iteration times.

$$V_0 = \{\Psi_{0i} = (p_i, v_i), p_i \in R^3, v_i \in R^3\} \quad (3.10)$$

$$V_k = \{\Psi_{ki} = (p_{ki}, v_{ki}), p_{ki} \in R^3, v_{ki} \in R^3\} \quad (3.11)$$

$$V = (\bigcup_{k=1}^n V_k) \bigcup V_0 \quad (3.12)$$

Besides the above four inputs and one output, this designed dynamic sensor planner has two functions:

1. Initial viewpoint configuration f_0 : An initial viewpoint configuration is a process to estimate viewpoints based on the given CAD model of a part. Function f_0 represents a bounding box algorithm developed to find a viewpoint set V_0 from CAD model M .

$$f_0 : M \mapsto V_0 \quad (3.13)$$

2. Feedback viewpoint configuration g_k : g_k is a projection from the defect map I to new viewpoints:

$$g_k : I \mapsto V_k \quad (3.14)$$

2. Point Cloud Generator Γ : Given a set of viewpoints V , Γ represents a process that obtains point clouds from each viewpoint:

$$\Gamma : V \mapsto Pc \quad (3.15)$$

Pc represents a point cloud, which is used to represent a 3D free-form surface of the part. Because it contains millions of 3D points, algorithms to reduce the point density are often required. Equation (4.13) describes how Pc is defined:

$$Pc = \{p_i(x_i, y_i, z_i) | i = 1, 2 \dots m\} \quad (3.16)$$

3. Viewpoint Evaluator Λ : Λ is a function to make a judgment about whether or not the quality of a point cloud Pc satisfies the predetermined conditions. If not, defect map I will be fed back to the dynamic sensor planner to update set V . This function can be represented by Equation (3.17) as follows.

$$\Lambda : V \mapsto I \quad (3.17)$$

where viewpoint set V is the input to Λ . Figure 3.9 illustrates the detail structure of this function. Two point clouds will be measured sequentially. Differences between those 2 point clouds will be input to the error evaluator. Meanwhile, an image processor is designed to detect shadow and light reflection regions in a point cloud. Three-dimensional points at the boundaries around holes are extracted using a point cloud identifier. Symbol u_k and v_k are used for system stability analysis and will be described next. A logic switch signal K is one output signal of Λ . Q is a cost function. If Q is less than a threshold, the switch k will be closed such that the iteration process will stop and the current point cloud can then be output for comparison with a CAD model.

4. Error Map Generator Δ : Function Δ can be described by Equation (3.18):

$$\Delta : (M, Pc) \mapsto E \quad (3.18)$$

where M represents the CAD model and Pc represents a point cloud. Error map E is the final output of the feedback system. An error map E includes a set of 3D points

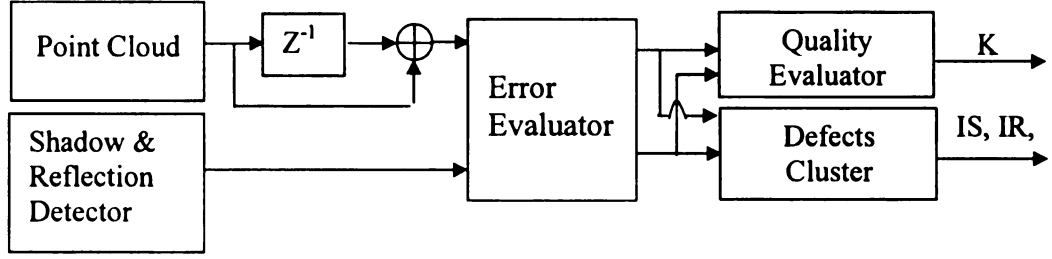


Figure 3.9. Structure diagram of a viewpoint evaluator Λ

and their distances D in a metric space to the closest triangle in CAD model M .

$$D(p_i, M) = \min(d(p_i, T_j), i = 1, 2 \dots m, j = 1, 2 \dots n) \quad (3.19)$$

$$E = \{ \langle p_i, D(p_i, M) \rangle \mid i = 1, 2 \dots m \} \quad (3.20)$$

3.2.3 Model of the dynamic measurement process

Based upon the above definitions, a state space model of the dynamic sensor planning process is developed to describe the process of this proposed closed-loop planning system, shown by Equation (3.21): where V and Pc are state variables, M and TC are inputs, and error map E is the output of the system:

$$\begin{cases} V \doteq f_0(M, TC) \cup g_k(I) \\ Pc \doteq \Gamma(V) \\ E = \Delta(M, Pc) \end{cases} \quad (3.21)$$

where \doteq indicates that V and Pc are accumulated results from iterations. As shown in Figure 3.8, given a CAD model M and a set of task constraints TC , a group of viewpoints V_0 will be initialized first, a point cloud Pc is then generated according to V_0 . Initially, V is equal to V_0 . Two functions Γ and Λ are then going to be executed based on this viewpoint set V . As described previously, Γ is an execution which obtains point clouds Pc from set V . Meanwhile, function Λ evaluates the current viewpoint by detecting holes. If necessary, a group of new viewpoints will then be

generated through function g_k . The measured point cloud will be kept updating until the shadow/reflection detector reports a stopping signal. Based on this automatic control scheme, an error map will be generated only when the measured point cloud is considered to be qualified. which then can be used to compare with its CAD model M for surface quality inspection.

3.2.4 Stability analysis of the dynamic inspection process

Stability of this dynamic inspection process is analyzed to ensure the iteration process will converge. A cost function Q is defined in Equation (3.22). In that equation, u represents the total area of holes, which is called a hole cost. And v represents the total area of shape difference between two point clouds, called a distance cost. Then, Q can be optimized by adding more viewpoints such that both u and v are minimized. The cost function Q can be represented on a complex plane such that u and v are the real and imaginary part respectively. A weight number w is defined as a ratio between the hole cost and the distance cost.

$$Q = \|w \cdot u_k + jv_k\| \quad (3.22)$$

This recursive planning process stops when Q is reduced to a tolerable value. Ideally, Q would be 0, indicating that the present point cloud has no holes and is exactly matches the point cloud measured in the previous step. Because new viewpoints will always bring more 3D points to update the point cloud, holes can be filled by continuously adding viewpoints. In another respect, accuracy of the point clouds can also be improved by adding redundant points. Hence, the system will be stable after a finite number of iterations, k .

3.3 Applying the dynamic inspection process to 3D surface measurement

In this section, the proposed general frame work is applied to solve the hole problem. Both shadow and light reflection can generate holes in a point cloud. holes are often filled with a predefined constant. As described in previous section, the proposed closed-loop scheme can fill holes by estimating new viewpoints. A bounding box algorithm is developed to estimate viewpoints for covering holes.

3.3.1 Applying the general framework to solve the hole problem

As shown in Figure 3.9, a viewpoint evaluator is implemented as a shadow/reflection controller. The dynamic sensor planner then has 3 functions f_0 , g_{kS} and g_{kR} , which represent a model-based sensor planner, a shadow-based sensor planner, and a reflection-based sensor planner. The model-based sensor planner has been discussed in Chapter 2.

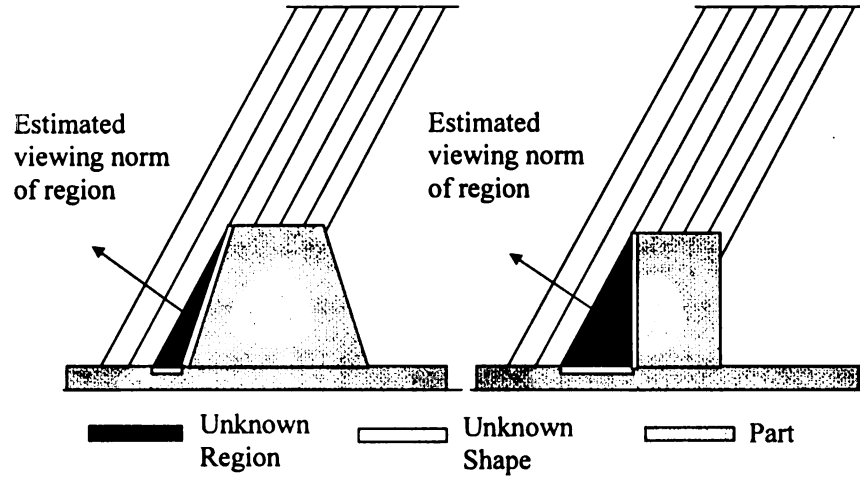


Figure 3.10. Unknown shadow space formed by geometry occlusion

1. Development of g_{kS} for covering shadow region:

Figure 3.10 illustrates a shadow problem in 3D shape measurement: the part with grey color represents a 3D part, the part with white color represents the unknown shape that need to be measured. The arrow represents a possible viewing vector that is going to be estimated to measure this unknown shape.

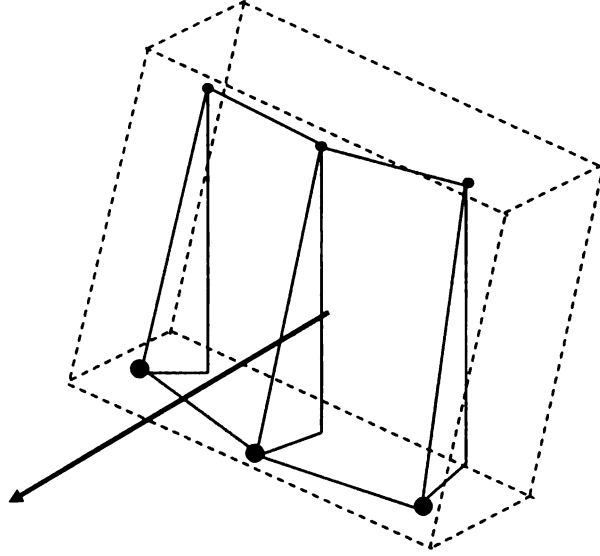


Figure 3.11. Estimate viewpoints by using bounding box method

Figure 3.11 illustrates the strategy for generating a bounding box by collecting 3D points around the detected shadow area. Considering measurement constraints and the geometry complexity, one viewpoint may not be able to cover the entire region. Therefore, the bounding box may be separated into two or more bounding boxes. The complete planning process can be described as following steps: (1) detect a shadow region, (2) extract 3D points around boundary of the shadow region, (3) cluster points into a “smooth” patch (Hierarchical clustering algorithm), (4) generate a bounding box, determine the center location and the viewing vector of the box, (5) find a viewpoint that satisfies all task constraints.

2. Development of g_{kR} for covering reflection region:

Light reflection often causes a bright spot on images such that the information of scene is not available. The reflection problem is especially sensitive on a glossy surfaces, such as an automotive part made of sheet metal. Figure 3.12 illustrates the reflection spot obtained on an aluminum board.

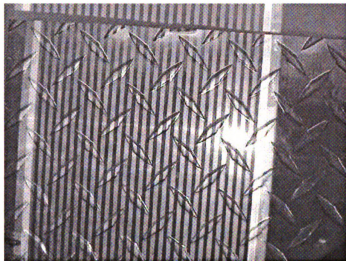


Figure 3.12. An example of reflection spot

Powder spray can effectively eliminate the reflection, but it introduces dimension errors. Besides, many parts are not allowed to be painted. Solving the reflection problem without powder spray is required.

An analytical model of light reflection is described in Figure 3.13: light reflection is categorized into three parts: uniform diffuse reflection, directionally diffuse reflection, and specular reflection [40]. Analysis using this analytic model will be very time consuming. A simulation model, the Phong model [41], is widely used because of the simplicity in computation. Similar to the analytical model, the Phong model has three parts: uniform diffuse reflection I_A , directionally diffuse reflection I_D , and specular reflection I_S , which is formulated in Equation (3.23).

$$I_R = K_A I_A + \frac{1}{D^2} (K_D (V \cdot N) I_D + K_S I_S \cos^n(V \cdot R)) \quad (3.23)$$

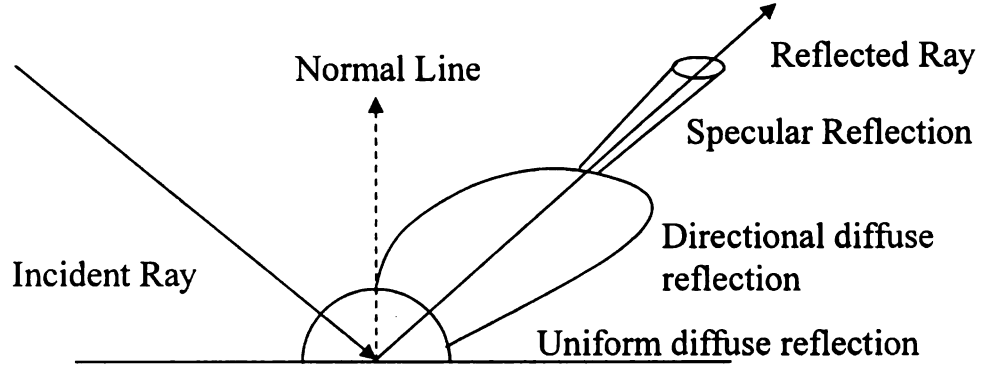


Figure 3.13. An analytical model of light reflection

According to the Fresnel law, the direction of reflection will principally depend on the incident ray. Therefore, modifying projection and view angle can avoid light reflection. Figure 3.14 illustrates the strategy to estimate new viewpoints: V is the camera viewing vector, and N is the incident vector. θ is defined as an angle between the incident direction and camera view direction. θ can be calculated by Equation (3.24). Angle 2θ specifies the size of the specular reflection cone.

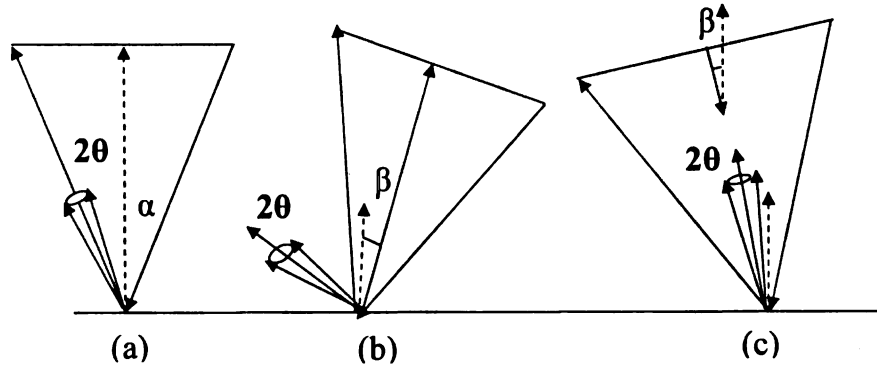


Figure 3.14. Rotation of view direction to conquer reflection problem. (a) Sensor setup with light reflection problem occurring (b) and (c) Avoid a detected brightness spot by relocating the area sensor

$$\theta = \arccos(V \cdot N) \quad (3.24)$$

For a bright spot, both of backward rotation and forward rotation can avoid specular reflection, as shown in Figure 3.14(b) and 3.14(c) respectively. α is the angle between the projection beam and the surface norm, and β is the angle between the effective view direction and the surface norm. The following Equation states that the rotation of area sensor β has to be greater than θ so that 3D surfaces in a reflection region can be recovered by this new viewpoint.

$$\alpha + \beta - \theta > \alpha - \beta \quad (3.25)$$

$$\beta > \frac{1}{2}\theta \quad (3.26)$$

3.3.2 Stability analysis of the specific model in solving hole problem

When apply the general framework to solve the hole problem, the error cost is assumed to be zero. Therefore, the dynamic inspection process will be stable only when the area of the hole is less than a threshold. And the cost function Q becomes:

$$Q = u_k \quad (3.27)$$

For a sheet surface with fewer geometric features, iteration times k could be small and the feedback process can quickly reach the set point. As a result, holes are filled by measurement on updated viewpoints.

3.4 Feedback design of the back-imaging system

The measurement using a single camera is an open-loop system, that means, the measurement quality depends on the correctness of each step in image processing, camera calibration etc. This open-loop system, again, cannot guarantee the quality

of the measured point cloud is satisfied. Feedback design of the inspection system provides a possibility to measure the part from different viewing poses, results from different views can be used to evaluate and also improve the measurement quality by properly designing the point cloud based controller.

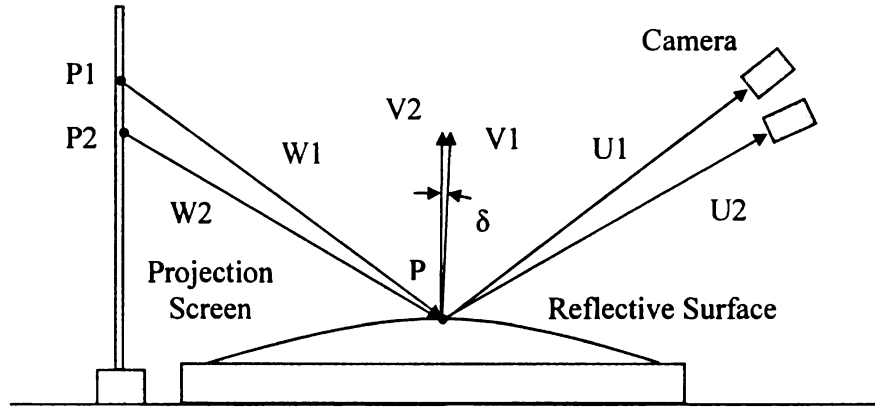


Figure 3.15. Improve the measurement accuracy by adding more cameras

Figure 3.15 shows an example that how a new camera is used to improve the measurement performance. Once the view vector U1 and the project vector W1 are determined in the iterative searching, position P and surface normal V1 are reported as the surface information. Based on the coordinates of this point P, a view vector U2 can be determined for another camera installed in the back-imaging system. From image analysis, the corresponding projection point P2 can be accordingly. Vector V2 can then be determined based on vector U2 and W2 following the reflection law. Therefore, if the position P is the real surface point, theoretically, the angle between the vector V2 and V1 will equals to 0. In real measurement, position P is a correct position on the real part surface if the angle δ between vector V1 and V2 is smaller than a threshold. If δ is larger than the threshold value, then the real surface normal V equals to the average of vector V1 and V2. With the surface normal V determined, position P can be updated on the view vector U1. By adding more cameras, the surface normal V will be further close to the true value. Hence, the measurement

quality can be improved.

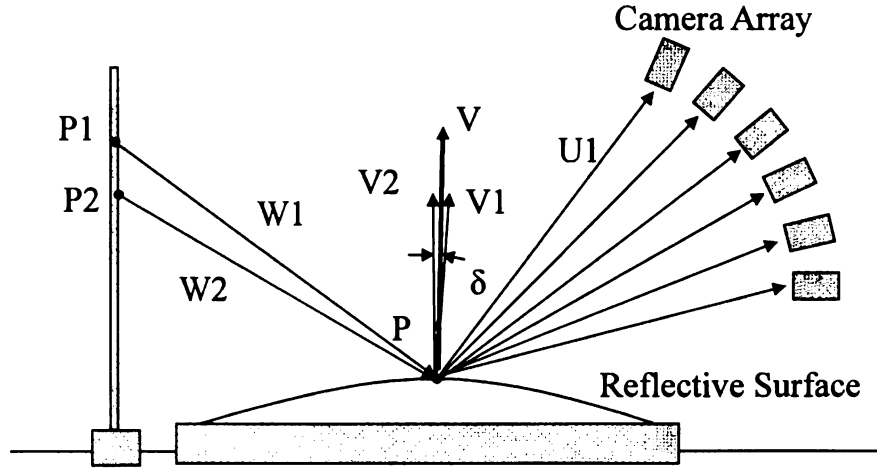


Figure 3.16. A networked-based back-imaging system

Adding new viewpoints by moving the camera with a robot arm is easy to be implemented. However, unlike the robot-aided area sensing system discussed before for measuring a lambertian surface, the developed back-imaging system requires certain stability of the projection screen and the calibrated camera. Also, since the mirrored image stays behind the glass surface not on the surface, images on different viewpoints will be different with each other. Therefore, instead of using a robot to move the camera, it is more reliable to setup a group of cameras for the enhancement of measurement accuracy.

A networked sensing system is then designed to improve the measurement accuracy for the back-imaging system. As shown in Figure 3.16, multiple cameras are installed for a certain piece of glass. The number of cameras that will be used for surface measurement depends on how the averaged surface normal V is updated in the feedback system. Since by adding more cameras, the measured surface normal will be close to the real value, therefore the difference between the vector V_n and the vector in the last step V_{n-1} will become smaller and smaller. Once the variation between V_n and V_{n-1} reaches a threshold, the feedback system will then stop and

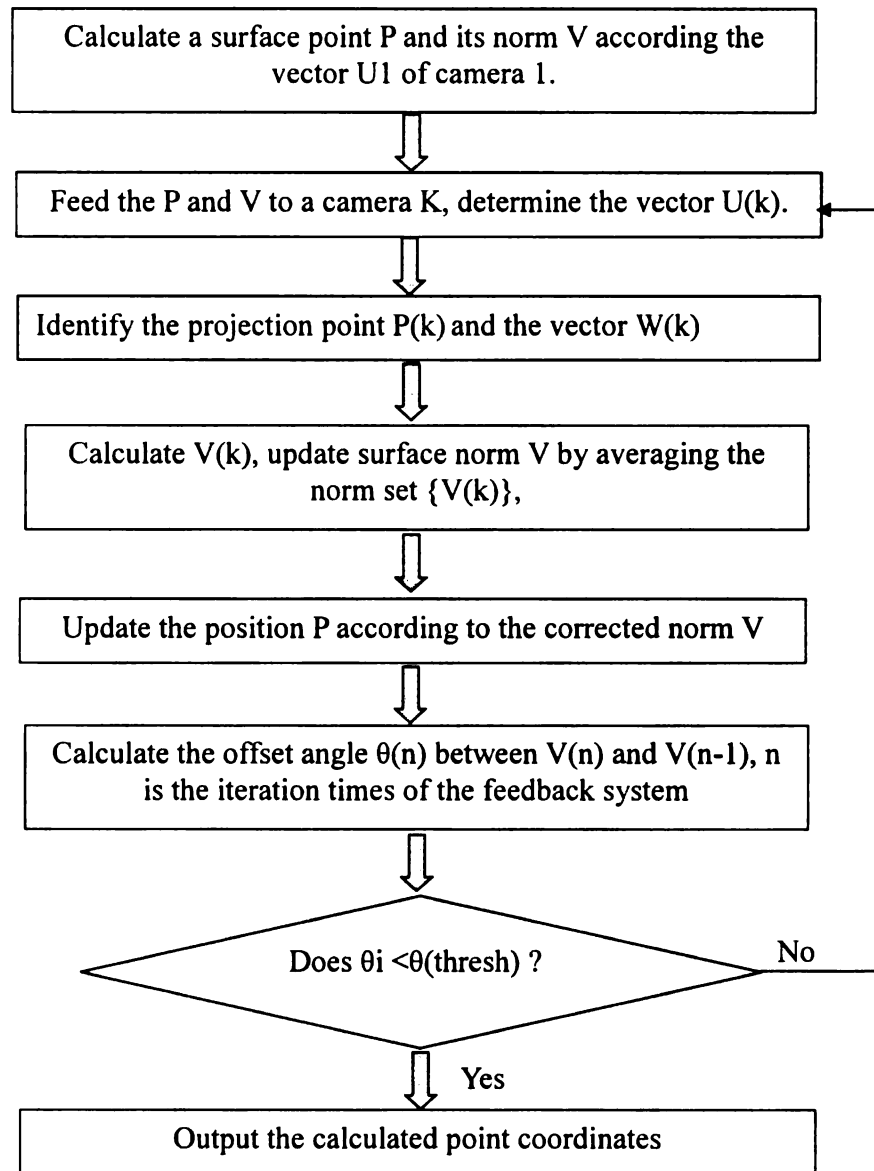


Figure 3.17. The flow chart of the designed feedback system

the vector V_n is used to estimate the position P. The entire process of the feedback system can be seen in the flow chart in Figure 3.17

3.5 Chapter Summary

In this chapter, a CAD-guided area sensor planning method is developed for the automated dimensional inspection system. Shadow and reflection are two problems that generate holes in a measured point cloud, which need to be compensated in real time measurement. A visual feedback control is necessary to guarantee that the quality of measured point clouds is satisfactory for dimensional inspection. A theoretical model of such a feedback system is established in this chapter. The set of viewpoints, can be updated recursively until it meets the measurement requirements.

For the back imaging system, since the camera needs to be fixed steadily for the iteration-based searching algorithm, instead of moving the camera using a robot, a group of cameras can be installed in the back-imaging system. The feedback system then automatically add more images from different viewing poses to verify and improve the measurement accuracy. Though more cameras are required in this case, the cost of the entire inspection system is reduced because the robot is not necessary.

CHAPTER 4

Data Analysis and Verification for Quality Inspection

The previous chapters describe methodology to control the quality of point clouds. Two problems are still remaining for quality verification: point cloud registration and error map generation. The measured point clouds need to be stitched together for representing the part surface, which needs to be compared to its CAD model for quality inspection.

Registration of point clouds for quality inspection includes two tasks: match point cloud to another point cloud and match a point cloud to a CAD model. Registration of point clouds is to calculate a transformation between two data sets: either from a point cloud to another point cloud, or from a point cloud to a CAD model. Many fitting algorithms, include the well-known Iterative Closest Point (ICP) algorithm, are developed for point cloud registration. However, the best fitting strategy is not proper for quality inspection because the algorithm itself assumes the two data sets need to have identical geometric shape, such that the algorithm finally converges. For quality inspection, best-fit algorithms cannot satisfy the requirement of detecting shape errors. Therefore, many fitting-based registration methods cannot be directly

applied to quality inspection. In this thesis, a weight-based registration method is developed for the automated 3D shape inspection systems. Robot viewpoints, surface features, gradients, and user-defined datum points are all accounted for to obtain a transformation matrix for registration. In this thesis, the correctness of the registration from a point cloud to the CAD model is determined by an error map, instead of a minimized distance calculated by the best-fit of two shapes.

4.1 Point Clouds Registration

4.1.1 Registration using Robot Kinematics

Figure 4.1 illustrates the transformations that are required for point cloud registration. In Figure 4.1: $\{B\}$, $\{T\}$, $\{C\}$, $\{W\}$, and $\{P\}$ represent the frames of the robot base, the robot end-effector, the area sensor, the world, and the part, respectively. T_B^T , T_T^C , T_C^W , T_P^W , T_B^W , and T_C^P represent the transformation matrices from the robot base to the robot end-effector, the robot end-effector to the area sensor, the area sensor to the world, the part frame to the world, the robot base to the world, and the transformation from the area sensor to the part frame, respectively.

Of these matrices, T_B^T can be obtained using robot calibration techniques; T_P^W and T_B^W can be determined by the calibration of a robot workcell. Sometimes, the calibration of T_P^W and T_B^W are combined as one problem of a robot part calibration, which determines the transformation matrix from the robot base to the part frame.

The remaining problems, T_T^C , T_C^P and T_C^W , are about the calibration of a vision sensor in a robotic system. Among of them, calibration of T_T^C is often called a robot hand-eye calibration [42].

If the platform's motion accuracy is satisfied for 3D shape inspection, the robot hand-eye relationship T_T^C can be directly used to register point clouds into the robot base frame. For example, if a CMM is used to move a 3D scanner, the measured data

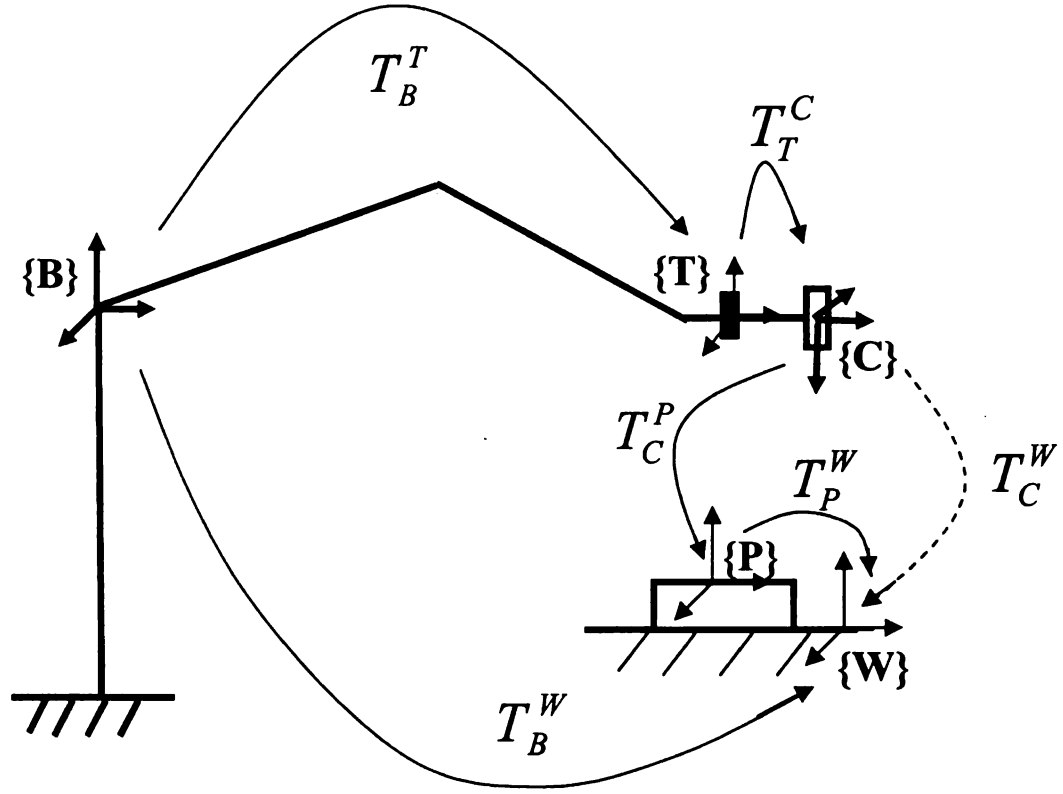


Figure 4.1. Coordinate transformations to the robot based frame in the automated dimensional inspection system

on each viewpoint can be transformed into the CMM base frame. The advantage of this type of system is that once T_T^C is pre-calibrated, on any viewpoint, point clouds can be automatically registered together for error map generation. However, such a motion platform becomes costly to many applications.

If a 3D scanning system is implemented on an industrial robot and the motion accuracy is not satisfactory for 3D shape inspection, then the alternative way for point cloud registration is to calibrate T_C^W , the transformation from viewpoints to a world frame, as long as the robot motion repeatability is good enough. As shown in Figure 4.2, W is the world coordinate frame, which is predefined in the measurement workstation. The robot will move the area sensor to predetermined viewpoints, rep-

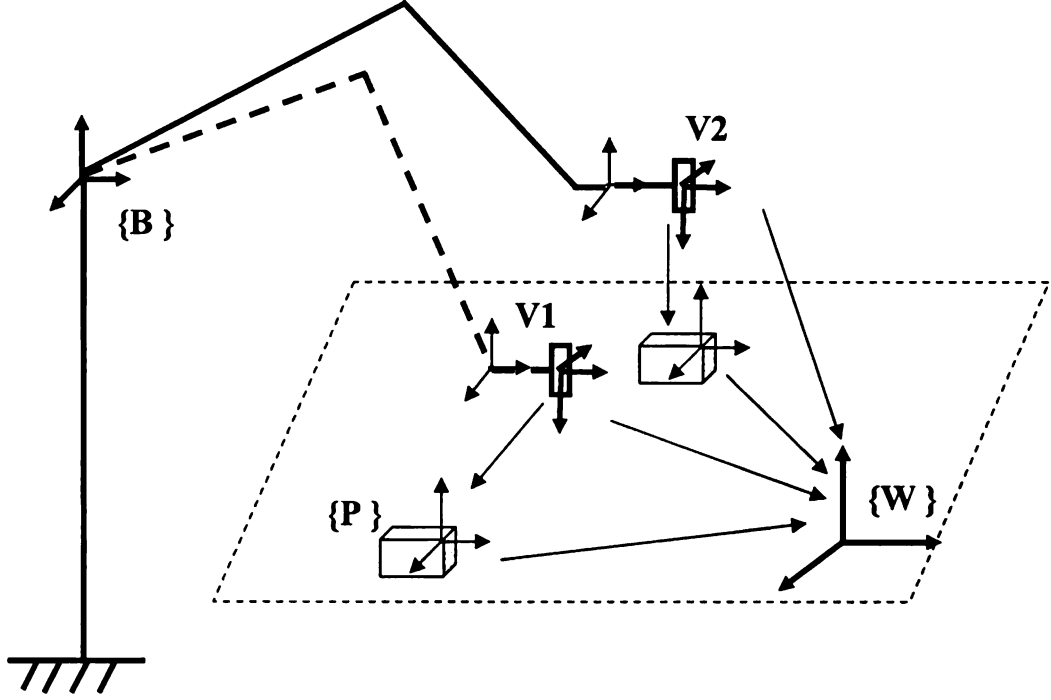


Figure 4.2. Coordinate transformations to the world frame of the automated dimensional inspection system

represented by $V1$ and $V2$. $C1$ and $C2$ are the frames of the area sensor, in which a point cloud will be measured. T_{V1}^W and T_{V2}^W are transformations from sensor frame to the world frame, which are required for point cloud registration. To determine T_V^W for viewpoint i , T_C^P and T_P^W are required:

$$T_V^W = T_C^P \cdot T_P^W, i = 1, 2, 3... \quad (4.1)$$

A calibration gauge is used to determine T_C^P and T_P^W . T_P^W can be easily determined by setting up gauge at pre-calibrated locations in the inspection workstation. Coordinates of control points on the surface of gauges are pre-determined. Since a calibration gage is used, an ICP-based best-fit method will be valid to calculate matrix T_C^P in iterations.

In this section, a ICP-based calibration method for obtaining T_T^C or T_C^W will be

introduced.

1. Calibrate hand-eye relationship T_T^C , if the accuracy of the robot is satisfactory:

Usually, robot hand-eye calibration requires the calibration of the camera itself first. Camera calibration has to deal with the problem of the camera characteristics [43, 44, 42]. However, for an automated dimensional system, an optical 3D scanner is mounted on the robot end-effector which is able to acquire 3D coordinates directly, and hence, T_T^C can be obtained by carrying out T_C^P . We begin with the following relationships:

$$T_B^T \cdot T_T^C \cdot T_C^P \cdot T_P^W = T_B^W \quad (4.2)$$

$$T_T^C = (T_B^T)^{-1} \cdot T_B^W \cdot (T_P^W)^{-1} \cdot (T_C^P)^{-1} \quad (4.3)$$

Equation (4.2) establishes the transformation relationship of the automated dimensional inspection system. Therefore, T_T^C can be calculated from the matrix $(T_C^P)^{-1}$, assuming that T_B^T , T_P^W , and T_B^W are known, as shown in (4.3). Thus, the only problem left is to calibrate the matrix T_C^P .

2. Calibrate each viewpoint T_C^W , if the robot accuracy is not within tolerance, but the robot repeatability is satisfactory:

Equation (4.4) shows the method to calibrate the transformation from one viewpoint to the world frame: knowing a calibration part frame T_P^W , matrix T_C^P is the only required information need to be derived.

$$T_C^W = T_C^P \cdot T_P^W \quad (4.4)$$

Therefore, in both situations, the transformation from the 3D scanner frame to the part frame is required. The following subsections describe our method to derive

T_C^P in details: we first introduce a quaternion mathematical model; and then briefly describe the developed 3D shape measurement method using an area sensor; and finally discuss an ICP-based transformation calibration.

4.1.2 Transformation using the quaternion algorithm

A quaternion is an extension of a complex number, which is defined in the form,

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \quad (4.5)$$

where \mathbf{i} , \mathbf{j} , and \mathbf{k} are orthogonal 3D vectors. A quaternion can be treated as a 4D vector. A unit quaternion q has norm of one and can be expressed by

$$q = \cos(\theta/2) + \sin(\theta/2)V \quad (4.6)$$

where θ is a rotation angle and V is a unit vector that represents a rotational axis. Then, a unit quaternion q represents an arbitrary rotation with parameters (V, θ) in a 3D space. A quaternion based rotation matrix, therefore, becomes:

$$R_q = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}$$

Therefore, the coordinate transformation (a rotation matrix R and a translation vector T) from the area sensor frame to the part frame can be solved in two steps using the quaternion algebra:

$$\begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} = R \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} + T = q \circ \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \circ q^* \quad (4.7)$$

where \circ denotes a quaternion multiplication and q^* represents the conjugate of the quaternion q . Let P_p denote the coordinates in part frame ($P = [X, Y, Z]^T$) and $P(C)$ denote the coordinates in sensor frame ($P(C) = [X(C), Y(C), Z(C)]^T$). We obtain:

$$\Omega(P_p + P_c)x - \Omega(T)x - T = P_C - P_P \quad (4.8)$$

where Ω represents a skew symmetric matrix. For m points employed for calibration, rotation parameter x can be calculated by solving the following system of linear equations:

$$\Omega(M_i + N_i)x = M_i - N_i, \quad \text{where, } i = 1, 2, \dots, m-1 \quad (4.9)$$

After x is solved, V and θ can be obtained by

$$V = x/\|x\| \quad (4.10)$$

$$\theta = 2\text{atan}(x_{\max}/V_{\max}) \quad (4.11)$$

Therefore, the rotation matrix R and the translation vector T are found by:

$$R = \begin{bmatrix} V_x\lambda_\theta + C_\theta & V_xV_y\lambda_\theta - V_zS_\theta & V_xV_z\lambda_\theta + V_yS_\theta \\ V_xV_y\lambda_\theta + V_zS_\theta & V_y\lambda_\theta + C_\theta & V_yV_z\lambda_\theta - V_xS_\theta \\ V_xV_z\lambda_\theta - V_yS_\theta & V_yV_z\lambda_\theta + V_xS_\theta & V_z\lambda_\theta + C_\theta \end{bmatrix}$$

$$T = P_p - R \cdot P_c \quad (4.12)$$

Where $\lambda_\theta = 1 - \cos(\theta)$, $C_\theta = \cos(\theta)$, and $S_\theta = \sin(\theta)$. To calibrate the matrix T_C^P , the 3D coordinates of points in both area sensor frame C and part frame P need to be determined. Coordinates of points in part frame P can be predetermined by making a calibration gage, while the corresponding points in area sensor frame C can be easily calculated from a 3D shape measurement as described in Chapter 2.

4.1.3 Robot Hand-eye Calibration using ICP Method

In this section, an ICP-based method is implemented for the area-sensor-based robot hand-eye calibration. The ICP-based algorithm works properly for robot hand-eye calibration because the calibration part is well fabricated such that the shape is close

enough to its CAD model, which satisfies the requirements to apply an ICP-based algorithm.

$$P_C = \{p_i(x_i, y_i, z_i) | i = 1, 2 \dots m\} \quad (4.13)$$

Mathematically, a point cloud P_C is a set of 3D points and the CAD model of a calibration gauge can be tessellated into a set of triangles, shown in (4.13). The developed ICP-based calibration algorithm is used to determine the transformation matrix T_C^P between PC and SM . Symbol m and n are used to represents the total number of points in the point cloud and the total number of triangles in the CAD model. Usually, m is much larger than n , because m is related to the resolution of the area sensor, and n is the number of triangles tessellated from the CAD model of the calibration gauge.

Like many existing ICP-based algorithms, our calibration method is a recursive process; each loop has two steps:

1. For a point in the set P_C , find a closest point $N(K)$ in the CAD model. When using the triangle representation, a point set PM can be extracted from the CAD model.
2. Compute a transformation matrix T such that for a series of points $P(j)$, the moved points $(T \cdot P(j))$ are close to their corresponding points $N(j)$ in set $P(M)$, with a cost function:

$$\Delta_i = \sum_{j=1}^M \|T_i \cdot (p_j) - n_j\|^2 \quad (4.14)$$

Equation (4.14) can be minimized in the least squared sense. Once $P(j)$ and $N(j)$ are determined, the quaternion-based method can be applied to calculate the matrix T . Subsequentness, a new series of points $N(j)$ can be found. This process is iteratively executed until the function Δ converges to a predefined threshold.

The closest point is often calculated using a point-to-point distance. An alternative way is to use a point-to-plane distance, which was introduced in [45]. The reason is explained in the following.

One problem to register a point cloud to a CAD model is that the point densities of two sets can be very different. A point cloud often contains more points than a CAD model. As shown in Figure 4.3(a), $T1$ and $T2$ are two triangles in a tessellated CAD model. Points $n1, n2$ and $n3$ are extracted from the CAD model. And $p1, p2$ and $p3$ belong to a point cloud. When the point-to-point distance is applied, both $p1$ and $p2$ have $n1$ as their closest point, which raises ambiguity in determining the closest point. This can be solved using the point-to-plane distance. As shown in Figure 4.3(b), according to the surface gradient of each point, a projection from the set of the point cloud to the set of the CAD model can be established, therefore each point in the point cloud would have a unique point in the CAD model. Point-to-plane distance is also applied in developing the error map.

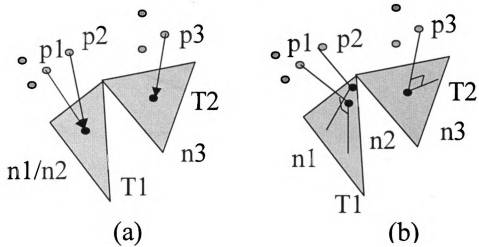


Figure 4.3. Solve the ambiguity of the closest point for the ICP based calibration algorithm. (a) point-to-point distance (b) point-to-plane distance

The entire process of the ICP-based algorithm for the area-sensor-based robot

hand-eye calibration can be summarized in Figure 4.4. More often, since the threshold of Δi is not easy to predefine, the system converges when the change of Δi , δ , is close to 1:

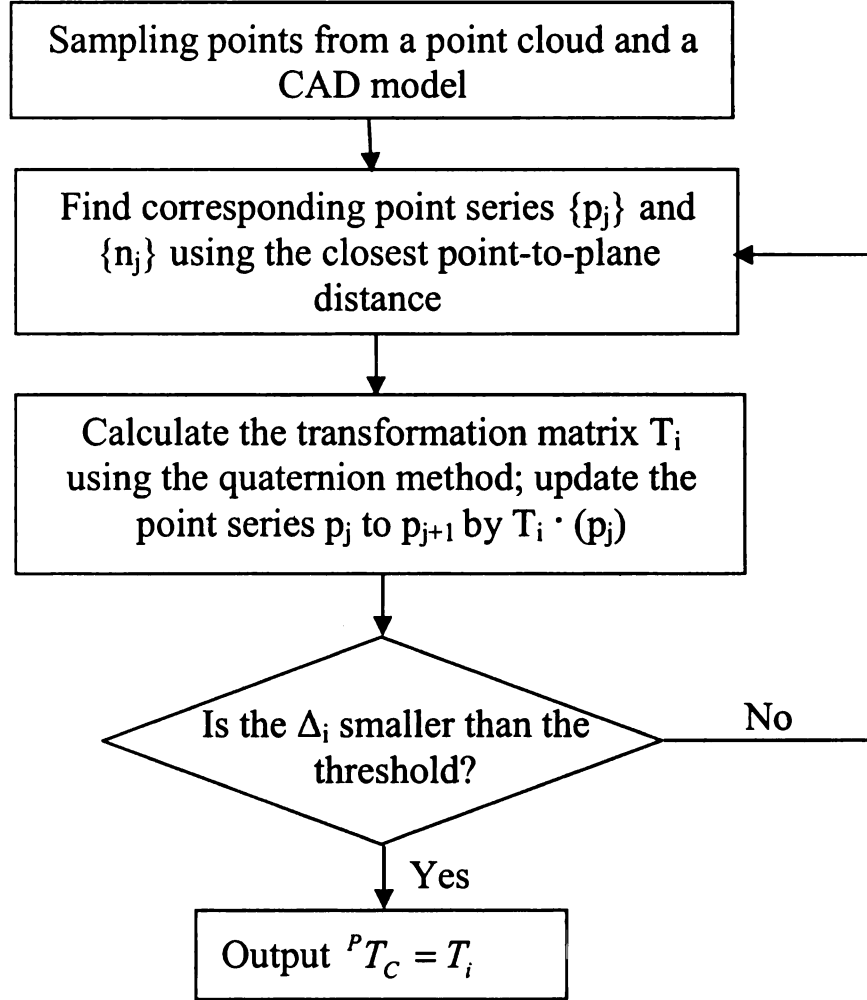


Figure 4.4. The flow chart of an area-sensor-based robot hand-eye calibration using an ICP algorithm

$$\delta = \frac{\Delta_i - \Delta_{i-1}}{\Delta_{i-1} - \Delta_{i-2}} \times 100\%. \quad (4.15)$$

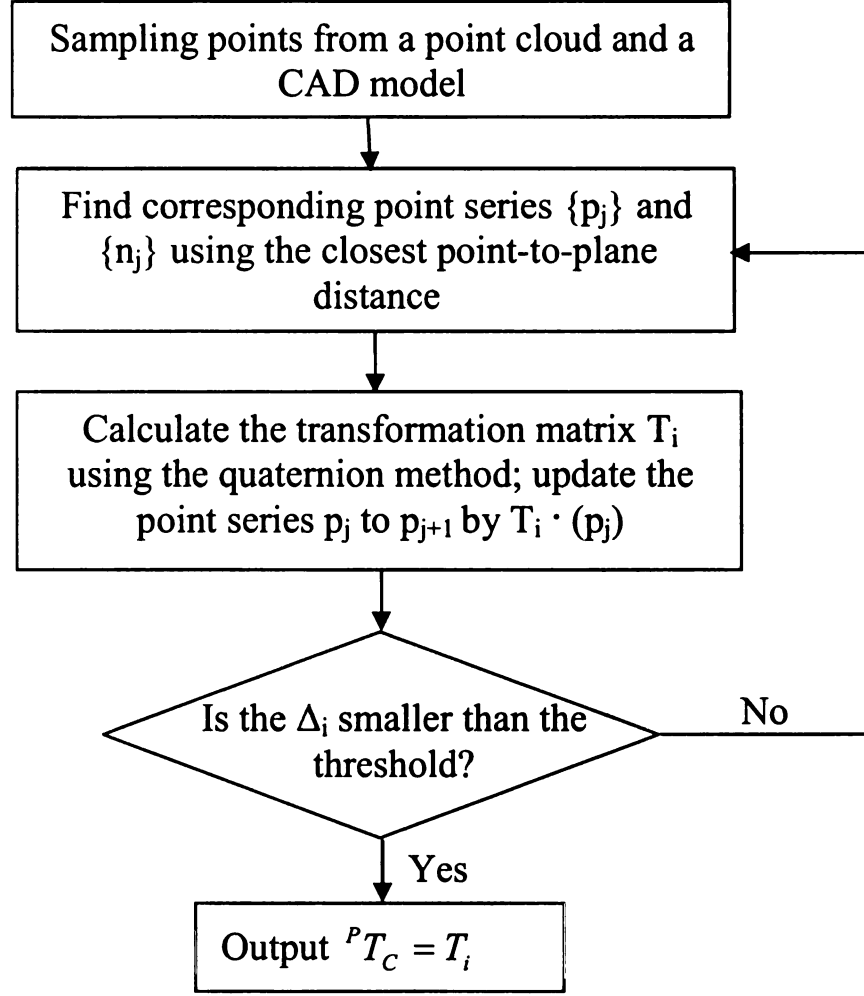


Figure 4.4. The flow chart of an area-sensor-based robot hand-eye calibration using an ICP algorithm

curvature K and mean curvature H :

$$K = \frac{f_{uu}f_{vv} - f_{uv}^2}{(1 + f_u^2 + f_v^2)^2} \quad (4.16)$$

$$H = \frac{f_{uu} + f_{vv} + f_{uu}f_v^2 - 2f_u f_v f_{uv} + f_{vv}f_u^2}{2(1 + f_u^2 + f_v^2)^{\frac{3}{2}}} \quad (4.17)$$

$$f(u, v) = k_0 + k_1u + k_2v + K_3u^2 + K_4u^2 + K_5v^2 \quad (4.18)$$

$$f_u = \frac{\partial f}{\partial u}, f_v = \frac{\partial f}{\partial v} \quad (4.19)$$

$$f_{uu} = \frac{\partial^2 f}{\partial u^2}, f_{uv} = \frac{\partial^2 f}{\partial u \partial v}, f_{vv} = \frac{\partial^2 f}{\partial v^2} \quad (4.20)$$

where f is the surface function described by (4.18). The first- and second-order surface gradients are given by (4.20). The surface features can then be categorized by K and H with a "neighborhood voting" technique: for each point in a point cloud and a CAD model, a label such as peak, flat, and valley etc. are defined according to the gradient pattern of its neighbor vertices. Therefore, a point cloud can be registered to its CAD model by matching up the pattern of the labels. The entire process can be summarized by the following sequence:

1. Determine the surface gradient of each point in the point cloud and the CAD model.
2. Calculate the Gaussian curvature K and mean curvature H according to the surface gradients.
3. Label features according to the value of K and H .
4. Register the point cloud to the CAD model by matching up the labels of the geometric features.

4.3 Integration of user-inputs to a weighted point cloud registration method

Usually, the measured part need to be assembled with other parts. Therefore, several user specified locations are used to connect as links to the other parts. Those points, once defined, will significantly determine the transformation of a point cloud in its

CAD frame. Meanwhile, to align point clouds and CAD model together, user may also need choose certain points, lines, curves, circles, plans as reference to anchor a point cloud into CAD model, all of those user-inputs will becomes the most important factors in solving point cloud registration problem.

$$V = \lambda_1 V_\alpha + \lambda_2 V_\beta + \lambda_3 V_\gamma \quad (4.21)$$

$$\theta = \min(\angle(N_M, F_{N_P}(V, \theta))) \quad (4.22)$$

To generate an error map, all acquirable information needs to be used to calculate a transformation matrix for properly matching measured point clouds to the CAD model. In 3D Cartesian space, representing a transformation by a rotation vector and rotation angle, Eqn. (4.21) and (4.22) show how to calculate the rotation axis and rotation angle: If T_α represents the transformation matrix derived from kinematics of motion platform, T_β represents the transformation matrix derived from geometric features, and T_γ represents the transformation matrix from the user-inputs. Each transformation matrix can be decomposed into a rotation R and a translation t using the quaternion algorithm. For those three rotations, i.e, R_α , R_β and R_γ , they can also be represented by three rotation vectors V_α , V_β , V_γ and three rotation angles θ_α , θ_β , θ_γ .

The finalized rotation vector V is a weighted result of those three rotations. And the weights λ_1 , λ_2 , and λ_3 can be predefined according to the prior information of the part and the inspection system:

As shown in the above table I, user-inputs receive the most consideration of the weights: when user-specified surface points are more than six, the registration transformation can almost be determined uniquely without knowing other information. However, if user inputs are less than required, registration will more depend on the calibration of the viewpoints and the available surface features; if the part surface is a sheet metal part with few features, registration will mainly be determined by the motion platform.

As shown in the above table I, user-inputs receive the most consideration of the weights: when user-specified surface points are more than six, the registration transformation can almost be determined uniquely without knowing other information. However, if user inputs are less than required, registration will more depend on the calibration of the viewpoints and the available surface features; if the part surface is a sheet metal part with few features, registration will mainly be determined by the motion platform.

Once the rotation axis is determined, the rotation angle θ can be calculated such that the average surface normals of the point cloud and the CAD model are equal to each other. As shown in (4.22), N_M and N_P represent the average surface normals of a part and a point cloud, FN_P is a function of N_P with respect to transformation parameters V and θ . The rotation angle θ is determined by minimizing the angle between the surface norms of the CAD model and the rotated point cloud.

4.4 A link clustering algorithm for point cloud filtering

Outliers is usually generated by intensity error. When the area sensor loses its focus for camera/projector lens, it can also cause outliers in a point cloud. There are often two types of outliers: one type of outliers exist randomly because of intensity noise or dark spots on part surface. The other type of outliers often stay at the boundary of a shadow region, which usually caused because the limited boundary illumination condition.

The first type of outliers are similar as a pulse function in signal processing. It can be easily removed by many filters using the 2D image processing techniques. The second type of outliers, which often form a stripe of points, is not easily to be identified and removed.

The second type of outliers, which often form a stripe of points, is not easily to be identified and removed.

Since a point cloud is developed from a 2.5D height map, in which the unit of X/Y axis is usually in *pixel* and the unit of Z axis is in *mm*. Therefore, filtering techniques of 2D image processing may also be applied for 3D point cloud filtering, as long as the it removes noise without changing other data points. It has to be noticed that many smoothing algorithms cannot be applied because the data points are modified such that the small defects like dints/dents cannot be distinguished any more.

In this paper, we use a link clustering algorithm for identifying and removing outliers. As shown in Figure 4.5, the clustering algorithm iteratively labels data point in the 2.5D height map till each point is assigned to a class. After the labelling process, all points are classified into groups to form several patches that are “linked” to each other. The link distance threshold between two neighbor points can be adjusted according to the resolution of area sensors. This distance is usually set to $0.1-0.5mm$, according to the continuity of surfaces. For a smooth surface, a class of data patch can contains thousands of points. On the contrary, multiple small data patches will be generated on a terrain surface with many steps and corners.

The calculation cost of this link clustering algorithm depends on the surface property, it is usually fast for a point cloud measured on a smooth surface. But even on a terrain surface, the entire process can be done with in 1 minute for a height map with 1024×768 points.

4.5 Surface-oriented error map generation

Error map is a color-coded map that is often developed from its point cloud and CAD model. This color-coded error map is actually a modification of Z coordinates: the X and Y coordinates of an error map are same as the point cloud, whereas the Z coordi-

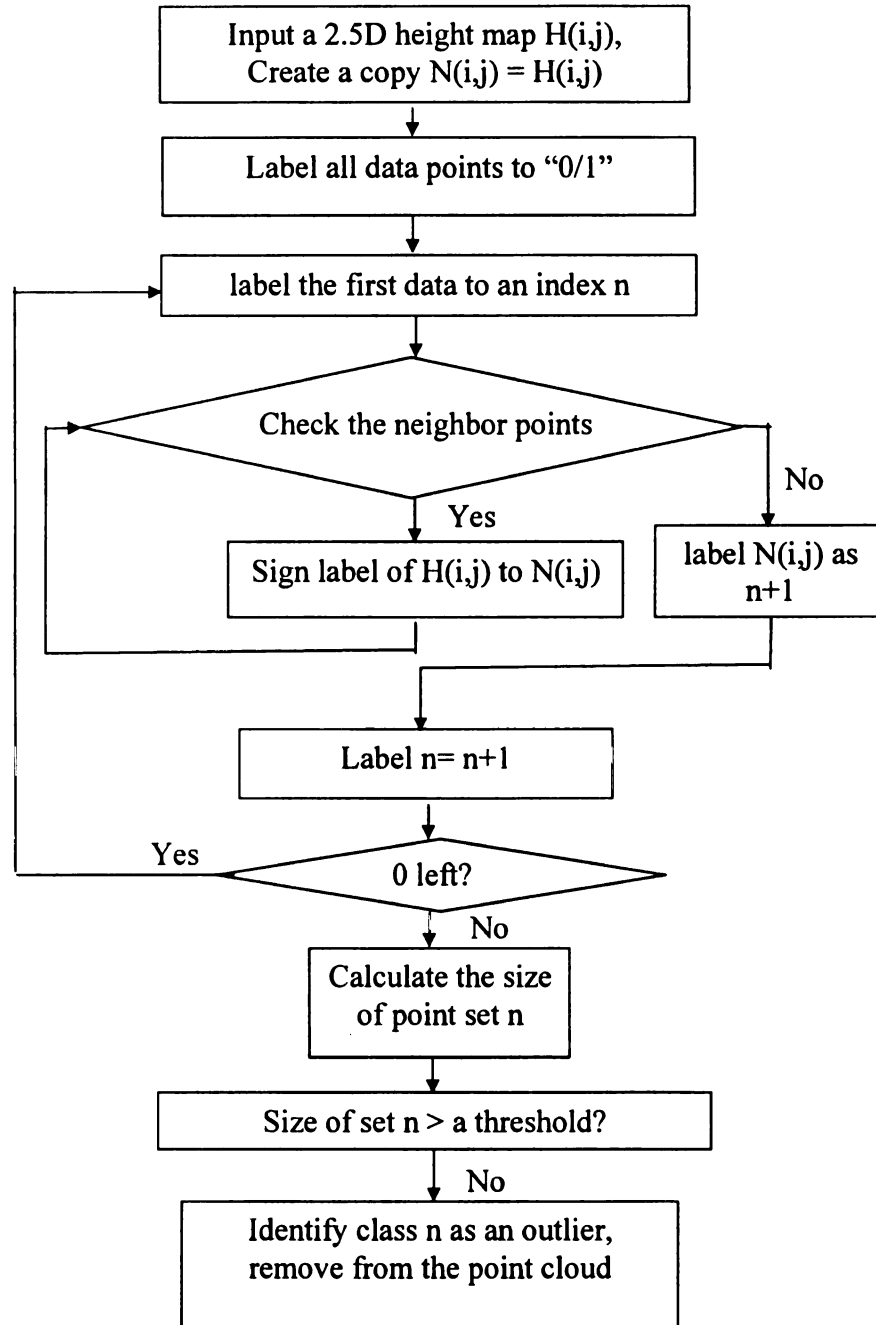


Figure 4.5. Flow chart of a link clustering algorithm for outlier filtering

nates of an error map is the distance calculated from a point to its closest triangle. It is necessary to point out that, this color-coded map is used to visualize the dimension

differences between the manufacturing surfaces and the design surfaces. It is more emphasize on the relative surface manufacturing error than the absolute shape error. In another word, the color distribution all over the map is more important because it reveals problems in the manufacturing process. Correction of increase/decrease force, temperature, time, voltage/current will be designed accordingly based on this color distribution, or generally called, an error map.

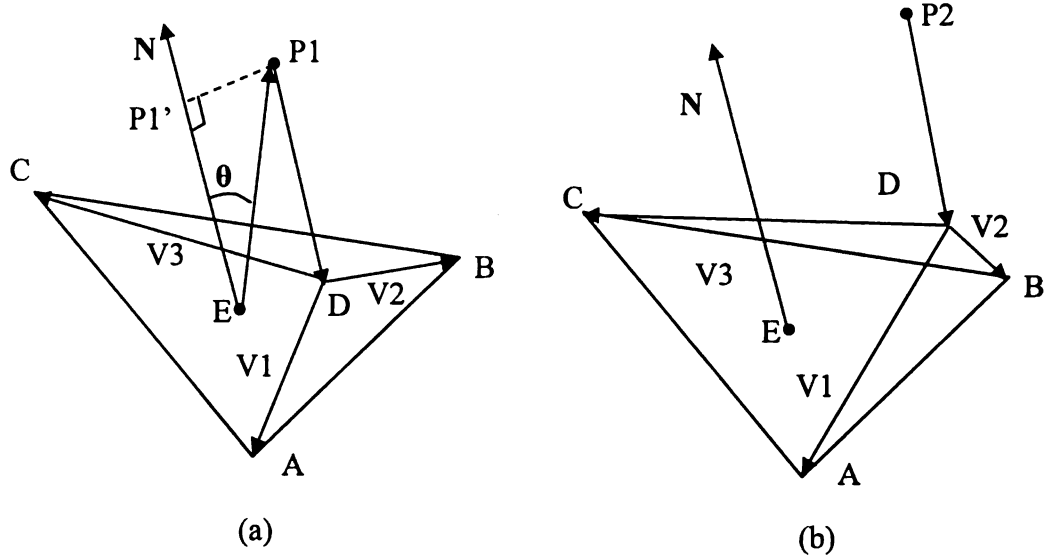


Figure 4.6. Closest distance of a point to a triangle. (a) Projection of a measured point falls inside of a triangle (b) Projection of a measured point falls outside of a triangle

In general, the error of a point is calculated as its distance to its closest triangle, which is determined according to two conditions:

1. The shortest distance of a measured point to the center of a triangle.
2. The projection of the measured point has to be in the triangle.

As shown in Figure 4.6(a): N is the norm and point E is the center of a triangle $\triangle ABC$, for a measured point $P1$, θ is the angle between two norms and the value

Eqn. (4.23):

$$L_{EP1'} = L_{EP1} \cdot \frac{N \cdot V_{EP1}}{\|N\| \cdot \|V_{EP1}\|}. \quad (4.23)$$

Therefore, if point D is in the triangle $\triangle ABC$, the sum of angles among three vectors $V1, V2, V3$ equals to 2π . If not, point D will be outside of this triangle, as illustrated in Figure 4.6(b).

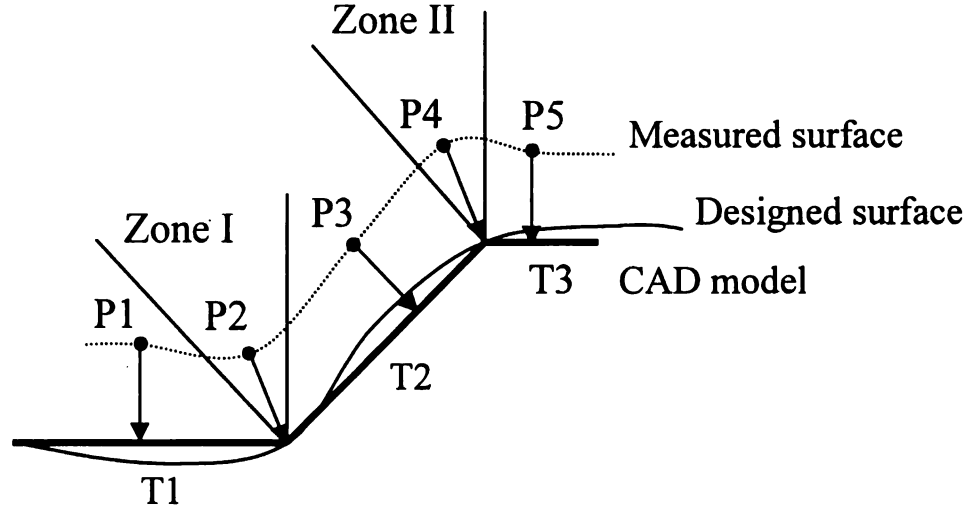


Figure 4.7. Calculate error distance of a point to its corresponding triangle

Figure 4.7 illustrates the strategy to calculate surface error. In Figure 4.7, there are three types of representations of a part surface, (1) a designed surface from CAD/CAM modelling software, (2) a tessellated CAD model that contains triangles for representing the free-form surface, (3) a cloud of points used to represent the measured surface on the manufactured part. The designed surface, though it should be used by definition to derive an error map according to a measured point cloud, is often replaced by a triangulation model in calculating an error map. The reason is that the calculation of the distance from a point to a free-form surface cost more time, considering a large auto part that has millions of points measured for dimension inspection. The approximate error between the free-form surface model and the triangulation model

lation of the distance from a point to a free-form surface cost more time, considering a large auto part that has millions of points measured for dimension inspection. The approximate error between the free-form surface model and the triangulation model is usually controlled by the dimension of a triangle, the maximum approximation error, the largest distance of a point from a free-form surface to its triangle, can be controlled under as small as to several micrometers. Many triangulation software had implemented this functionality. Therefore, in this paper, the CAD model tessellation error is not included in the error map. The error map is derived according to the measured point cloud and the triangulation model of a part surface.

The process to calculate an error map is:

1. Calculate the norm of each triangle of a CAD model.
2. Find the closest triangle of a measured point in the CAD model.
3. Calculate the distance of the measured point to the its triangle.
4. Check the neighbor triangles of the closest triangle: if a measured point falls into multiple triangles, the surface error will then be calculated from the shared triangle boundary to the measured point. For example, point $P2$ and $P4$ of zone I and zone II in Figure 4.7.

4.6 Chapter Summary

For an automated dimensional inspection system, point cloud registration cannot be executed based on best-fit algorithms. The common ICP-based methods, although they have been adopted in many applications, are not feasible for dimensional inspection. The absolute shape deviations can be determined by coordinate transformation techniques. A novel robot hand-eye calibration method is developed in this chapter to calculate the transformation matrix from the 3D area sensor to a common world

frame, which is used as the coordinate system of point clouds. To guarantee the point cloud match to the CAD model for error map generation, user defined datum points and part surface features are also integrated in the calculation of registration matrices. A weighted registration approach is then developed in this chapter for matching point clouds.

Besides point cloud registration, a link clustering method is proposed in this chapter which is designed to remove the outliers of a 3D point cloud. Point-to-plane distance and a surface-oriented error calculation method is also developed in this chapter. After all of the above techniques are applied to the measured point clouds, a color-encoded error map can then be developed to visualize manufacturing dimensional errors.

CHAPTER 5

Experimental Results

This chapter shows the experimental results of the developed 3D area sensor, the automated dimensional inspection system, sensor and system calibration and the results of the developed back-imaging system.

5.1 Lambertian Surface measurement using the Automated Dimensional Inspection System

Figure 5.1 illustrates the developed automated dimensional inspection system. The area sensor is made by a Sony XCD710 camera and a Plus V-1100 digital projector. One Matrox image grabber is used for image collection. The software of sensor planning and point cloud calculation are developed using C++.

5.1.1 Calibration Setup and Implementation Steps

The calibration was implemented on a DYNA 2400C NC machine, a reference board was fixed on the platform of the NC machine and can be moved along 3 directions. Figure 5.2(a) shows the calibration setup of area sensor parameters $d(x, y)$, $S(x, y)$ and XY plane offset angle α . Figure 5.2(b) shows the calibration setup for calibrating

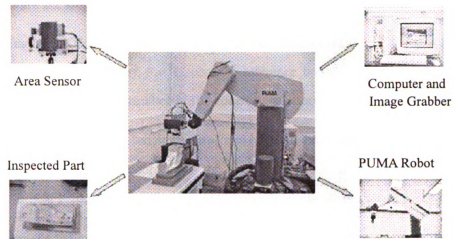


Figure 5.1. An automated dimensional inspection system

exploding vector K to calculate correct X and Y coordinates.

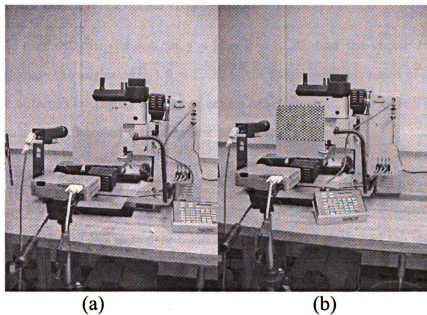


Figure 5.2. Calibration setup on a DYNA 2400C NC machine. (a) calibration for acquiring depth information (b) calibration for obtaining a 3D point cloud

There are six steps to get $d(x, y)$, $S(x, y)$, and $\alpha(x, y)$:

1. Using the GCLS method, generate and project horizontal stripes ($hi, i = 1...m$) and vertical stripes ($vj, j = 1...n$) onto reference board sequentially, m, n are used to represent the number of the coded lines. Record all images for analysis.
2. For each pair of detected horizontal and vertical lines (hi, vj), $i = 1...m, j = 1...n$, calculate the coordinates of the intersection point $P0(hi, vj)$, $i = 1...m, j = 1...n$. Points $P0(hi, vj)$ are encoded points on reference plane.
3. Move the calibration board forward one step, repeat the step 1 and 2, find the corresponding point $P1(hi, vj)$, $i = 1...m, j = 1...n$.
4. Repeat step (3) 30 times, find all groups of corresponding points $Pk(hi, vj)$, $i = 1...m, j = 1...n, k = 1...30$. Points $Pk(hi, vj)$ represent encoded surface points.
5. For each $Pk(hi, vj)$, $i = 1...m, j = 1...n, k = 1...30$, find the vector $P0Pk(ij)$, calculate the angles $\alpha k(ij)$, $k = 1...30$ to the stripe gradient, result of $\alpha(ij)$ can be obtained by taking average of angles $\alpha k(ij)$, $k = 1...30$. Figure ?? showed some of the detected corresponding point pairs.
6. Calculating (dij, Sij) , $i = 1...m, j = 1...n$ using Equation (2.7).

Figure 5.3 illustrates the calibration process of collecting data of control points. The reference board was moved 30 times with 1mm each step toward the area sensor viewing direction. The control points are calculated in sub-pixel resolution by using a line fitting method.

A measurement on a known height flat surface was used to evaluate the calibration result. A statistical result is shown in Table III, in which when applying more calibration points, the mean of the measurement error is closed to 0mm with a standard deviation of 0.0162mm. It has been analyzed in Chapter 2 that the baseline distance

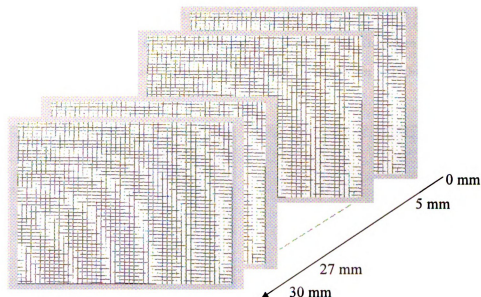


Figure 5.3. Calibration by moving the calibration board with patterns of control points

d has a great impact on the measurement accuracy. Increasing the baseline distance d can dramatically reduce the measurement error.

Table 5.1. Measurement accuracy after calibration

No. of calibration points	Mean: $\mu(mm)$	Std: $\sigma(mm)$
4096	0.0018	0.0261
16384	0.0011	0.0185
65536	0.0006	0.0162

5.1.2 Measurement performance of developed area sensor

A well fabricated gauge is used to evaluate a sensor measurement precision. As shown in Figure 5.4 there are four slots machined with different depths, from left to right: $32 \pm 1\mu m$, $24 \pm 1\mu m$, $14 \pm 2\mu m$, and $6 \pm 2\mu m$. The right side of Figure 5.4 shows the

measured point cloud of the gauge, in which three of these slots has been detected, only the slot with height $6 \pm 2\mu\text{m}$ may not be seen clearly. Therefore, the precision of the developed sensor is specified to $14\mu\text{m}$, within a $200\text{mm} \times 150\text{mm}$ measuring field.

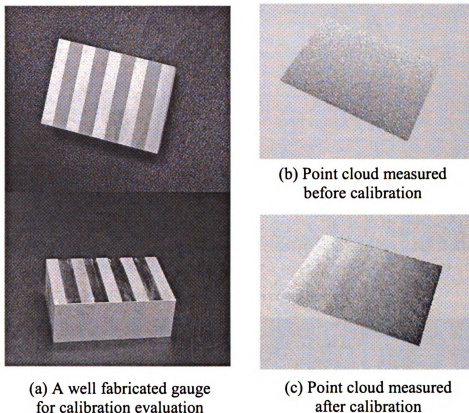


Figure 5.4. Testing sensor precision using a gauge (Four slots were made on the gauge, from left to right, the depth are: $32 \pm 1\mu\text{m}$, $24 \pm 1\mu\text{m}$, $14 \pm 2\mu\text{m}$, and $6 \pm 2\mu\text{m}$)

Figure 5.5 visualizes 3D shape of a computer mouse. The grey color represents the surface depth to the bottom of the mouse. This point cloud can be used for humanoid product design, which is extremely necessary in the mouse manufacturing industry.

Figure 5.6 shows an implementation of a robotic area sensing system for optical dimensional measurement of a door panel in the automotive industry. Figure 5.7 and

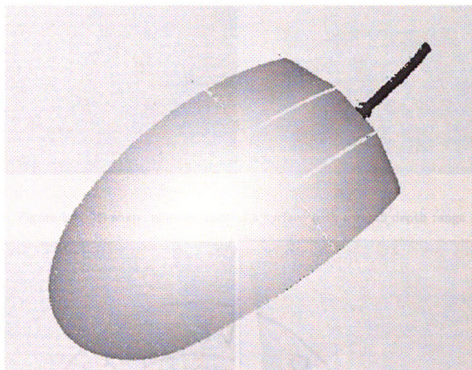


Figure 5.5. 3D shape of a computer mouse

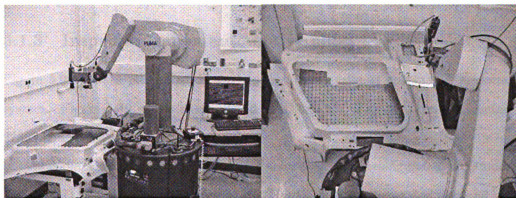


Figure 5.6. Dimensional measurement of a door panel part using the automated area sensing system

Figure 5.8 display the measured point clouds.

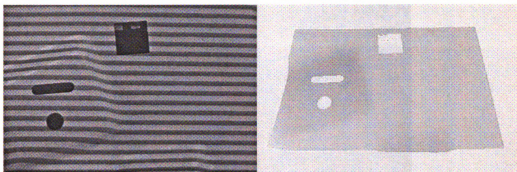


Figure 5.7. 3D shape measurement of a surface with a small depth range

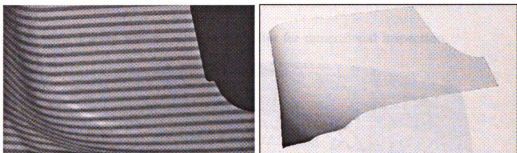


Figure 5.8. 3D shape measurement of a surface with a big depth range

5.1.3 Inspection on an automotive pillar

Figure 5.9 shows a pillar that is used to test the developed automated inspection system. Figure 5.10 shows a tessellated CAD model. This CAD model contains 22803 triangles.

Viewpoints are automatically generated according to the triangulated CAD model. Figure 5.11 shows a viewpoint simulation. Height maps are generated first and then converted to point clouds. Figure 5.12 illustrates two robot viewpoints during the 3D shape measurement.

Figure 5.13 16 shows individual point clouds measured on each viewpoint. This set of point clouds are then matched together to represent the geometry shape of the real part surface, as shown in Figure 5.14. The number of points in the overlap areas

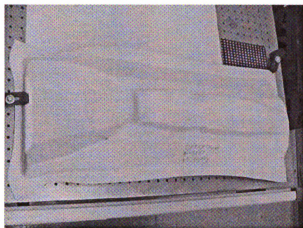


Figure 5.9. An automotive pillar for dimensional inspection

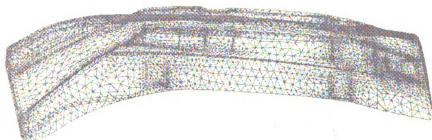


Figure 5.10. A tessellated CAD model of the pillar

of point clouds can be interpolated for better surface representation. Figure 5.15 shows the final point cloud. This point cloud can then be used for quality evaluation.

Figure 5.16 shows the side view of the measured point cloud of this pillar. By comparing the shape deviations using point-to-plane distance, an error map is generated as shown in Figure 5.17.

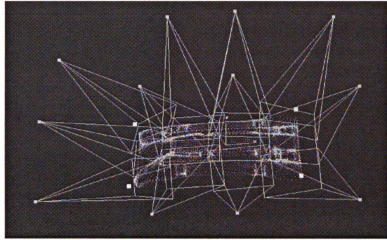


Figure 5.11. Estimated viewpoints

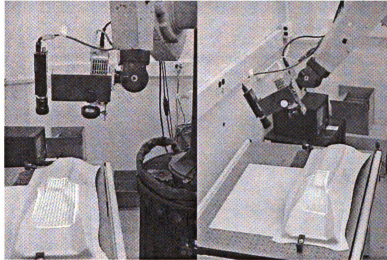


Figure 5.12. 3D scan on different viewpoints

5.2 Reflective Surface Measurement using the Back-Imaging System

Figure 5.18 illustrates the setup of the developed back imaging system. There are currently two camera with reduced field of view, a glass can be measured separately

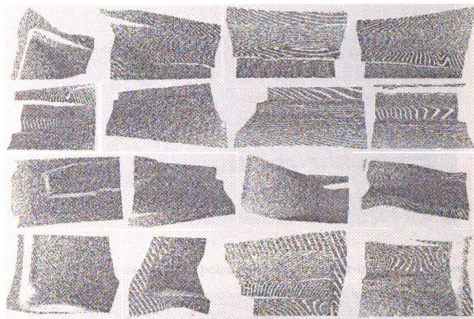


Figure 5.13. Point clouds measured on each viewpoint

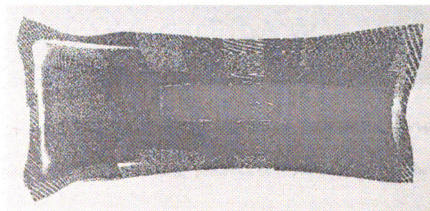


Figure 5.14. The measured point cloud of part pillar-m32510

and then registered together to the world coordinate frame of this back imaging system. Figure 5.19 shows the process to identify the mark points from a back image. The mark points is evenly spaced on the projection screen. Figure 5.20 shows results of the vector-based camera calibration. By moving the calibration board, the 3D

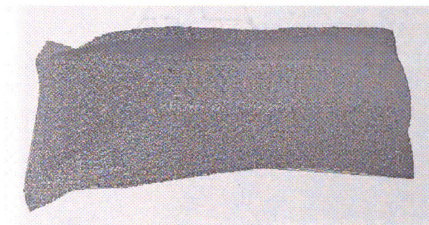


Figure 5.15. An interpolated point cloud of part pillar-m32510, top view

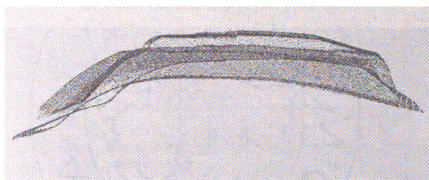


Figure 5.16. An interpolated point cloud of part pillar-m32510, side view

coordinates of any pixel on the image plane can be identified in deriving the viewing vector.

For each marker point, the back-imaging system use image from another camera to verify and update the calculated results in feedback. Figure 5.21 shows the vectors calculated in the feedback process.

Figure 5.22 and Figure 5.23 show current result using the iterative searching method for the automotive glass measurement. Figure 5.24 shows a measurement on part ch3201, which is a car's side window. Matching measurements from another

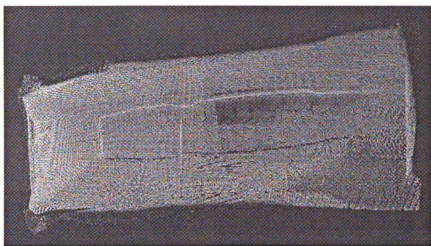


Figure 5.17. An error map of part pillar-m32510

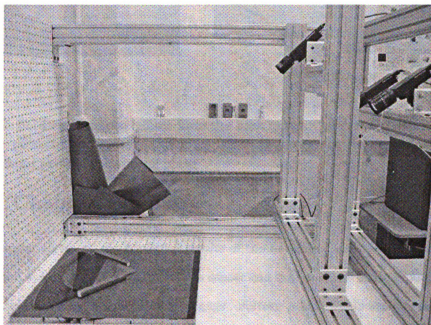


Figure 5.18. A back-imaging system for automotive glasses inspection
camera, the entire point cloud can be seen in Figure 5.25

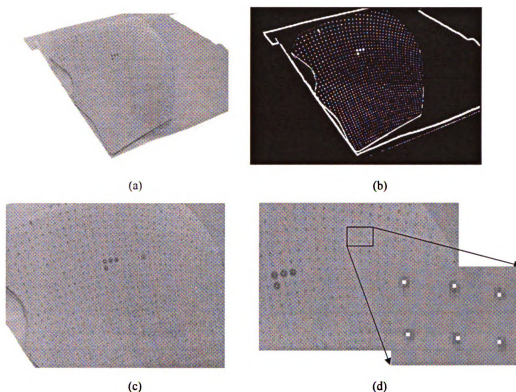


Figure 5.19. Detect marker points from the back image. (a) A recorded image, (b) Identify pixels of top layer reflection, (c) Detected points, (d) An expanded view of the detected pixels on the image plane

5.3 Chapter Summary

This chapter shows all experimental results of the developed automated dimensional inspection systems. The developed area sensor can detect small depth change about $14\mu\text{m}$ s, evaluated on a well-fabricated gage. Sixteen point clouds are measured and registered together to form an entire point cloud using the proposed methods. For the back-imaging system, top layer and bottom reflection are separated. Sensor view vector searching methods are both displayed. The measurement results shows that both methods are effective in obtaining the 3D shape of an automotive glass.

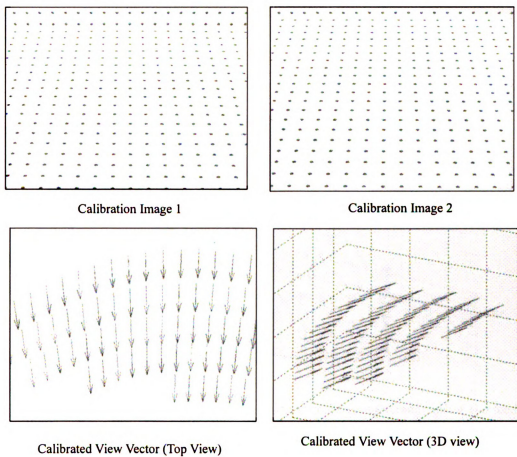


Figure 5.20. Detect marker points from the back image. (a) Image of the calibration board at position A, (b) Image of the calibration board at position B, H is the height difference in the world frame, (c) and (d) Detected view vectors

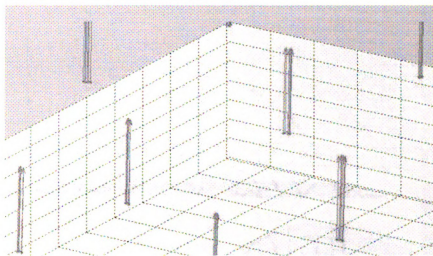


Figure 5.21. Improve the accuracy of the surface norm measurement using the feed-back design

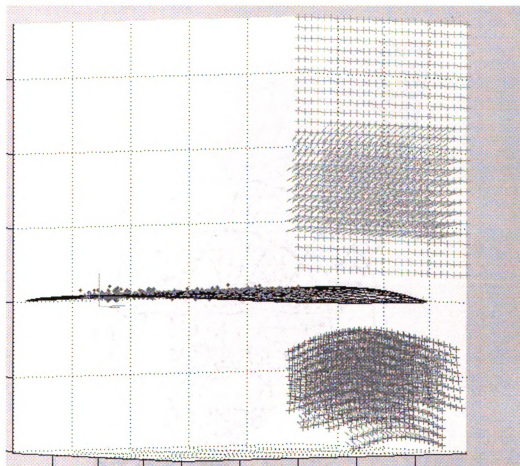


Figure 5.22. The measured point cloud using the iteration-searching method, side view

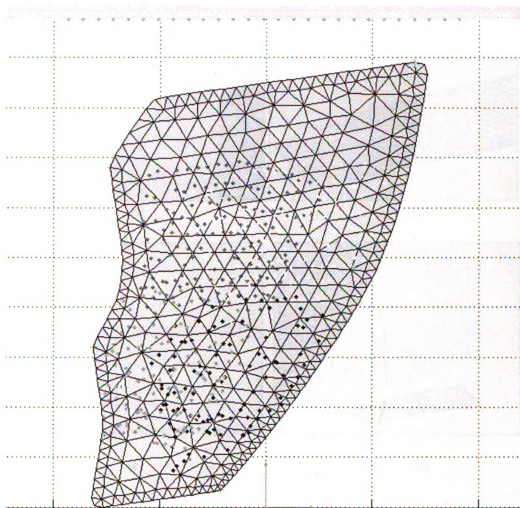


Figure 5.23. The measured point cloud using the iteration-searching method, top view

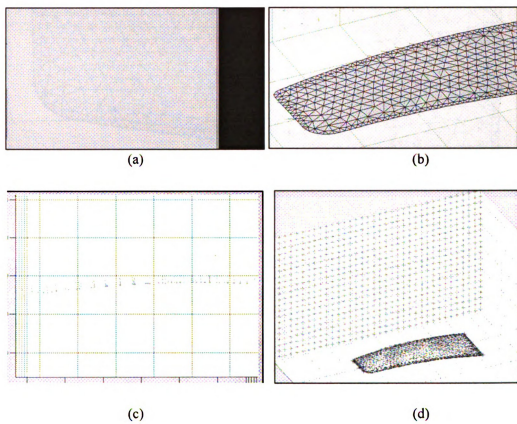


Figure 5.24. Measurement of part ch3201, (a) A recorded back image, (b)

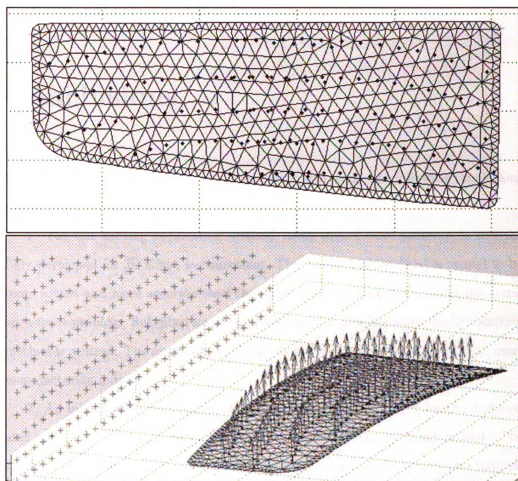


Figure 5.25. The measured point cloud using the iteration-searching method, top view

CHAPTER 6

Conclusions

Based on the results and discussion presented in this study, the following conclusions can be made:

1. The line shifting codification method, using square wave patterns, is successfully developed for 3D shape measurement. The line shifting method is proved to be more robust and more accurate than the phase shifting method in solving the correspondence problem. Sub-pixel image analysis becomes possible when the line shifting method is utilized. The sensitivity of the developed area sensor is satisfactory to most manufacturing industries.

2. An innovative pixel-to-pixel calibration strategy is developed for the designed area sensor prototype. Quality inspection requires an accuracy measurement. The traditional sensor calibration method calibrates the camera and the projector individually using a lens pin-hole model. However, a calibration residual always exists when a real lens system is simplified as a pin-hole model. The pixel-to-pixel calibration strategy does not assume all light projection and imaging beams pass through a single point. Instead of that, the pixel-to-pixel strategy calibrates sensing parameters to each corresponding pair, which significantly reduces calibration errors. Also, it needs to be pointed out that the developed calibration method calculates the area sensor pa-

rameters, whereas the traditional methods calibrate camera and projector separately. Consequently, the developed calibration system is also much more efficient.

3. The CAD-guided sensor planning method is proved to be a practicable approach for this automated dimensional inspection system. An initial set of viewpoints can be generated that covers the entire part surface.

4. The 3D visual feedback control improves the measurement quality by estimating viewpoints from the measured point clouds. Feedback control is especially necessary for a shiny metal surface because of unpredictable intensity noise and to set appropriate illumination conditions. The developed feedback control can successfully compensate sensor viewpoints so that a qualified point cloud can be obtained.

5. For a robot-integrated automated dimensional inspection system, point clouds can be registered together to the CAD model using robot kinematics, part features, user-specified datum points, and the ICP method. To obtain a satisfactory registration quality, all those factors need to be integrated for estimating the transformation matrix.

6. An innovative back-imaging system is developed in this thesis for reflective surface measurement. For this back-imaging system, two methods are developed for measuring the 3D shape of an automotive glass. If the CAD model of a part is given, a recursive algorithm can be applied for searching the coordinates of the surface points. The feedback design is also proposed for the back-imaging system. By adding more cameras at different viewpoints, measurement performance can be improved by averaging the obtained results.

In summary, non-contact optical 3D surface measurement is more effective approach than a contact based measurement strategy using CMMs. To apply the state-of-the-art 3D sensing technology for manufacturing inspection, an automated dimensional inspection system is essential because the quality of measured point clouds needs to be controlled. This thesis has developed a general framework for the 3D

area sensor and a feedback-based automated dimensional inspection system. Moreover, as another main contribution in non-contact-based 3D shape measurement, the searching-based back-imaging system is proved to be effective for reflective surface measurement for the glass manufacturing industries.

BIBLIOGRAPHY

- [1] Timothy S. Newman and Anil K. Jain. A survey of automated visual inspection. *Computer Vision and Image Understanding*, 61(2):231–262, 1995.
- [2] H. Longuet. A computer algorithm for reconstructing a scene from two projections. 293:133–135, 1981.
- [3] C. Dumont L. Wong and M. Abidi. Next best view system in a 3-d object modeling task. In *International Symposium on Computational Intelligence in Robotics and Automation*, pages 306–311, 1999.
- [4] S. L. Stoev and W. Strasser. A case study on automatic camera placement and motion for visualizing historical data. In *IEEE Visualization*, pages 545–548, Boston, Massachusetts, 2002.
- [5] J. L. Posdamer and M. D. Altschuler. Surface measurement by space-encoded projected beam systems. *Computer Graphics and Image Processing*, 18(1):1–7, 1982.
- [6] G. Hu and G. Stockman. 3-d surface solution using structured light and constraint propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4):390–402, 1989.
- [7] N. Kiryati D. Caspi and J. Shamir. Range imaging with adaptive color structured light. *Pattern analysis and machine intelligence*, 20(5):470–480, 1998.
- [8] D. Bergmann. New approach for automatic surface reconstruction with coded light. In *Remote Sensing and Reconstruction for Three-Dimensional Objects and Scenes, SPIE*, volume 2572, pages 2–9, 1995.
- [9] B. Curless L. Zhang and S. M. Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *1st International Symposium on 3D Data Processing, Visualization, and Transmission*, pages 24–36, 2002.

- [10] Kosuke Sato. Range imaging based on moving pattern light and spatio-temporal matched filter. In *IEEE International Conference on Image Processing*, volume 1, pages 33–36, 1996.
- [11] J. Guhring. Dense 3-d surface acquisition by structured light using off-the-shelf components. In *Videometrics and Optical Methods for 3D Shape Measurement*, volume 4309, pages 220–231, 2001.
- [12] M. Trobina. Error model of a coded-light range sensor. *BIWI-TR-164 Report*, 1995.
- [13] C. H. Meng T. S. Shen. Digital projector calibration for 3-d active vision system. *Transaction of the ASME*, 124:126–134, 2002.
- [14] M. Carocci G. Sansoni and T. Rodella. Calibration and performance evaluation of a 3-d imaging sensor based on the projection of structured light. *IEEE Transactions on Instrumentation and Measurement*, 49(3):628–636, 2000.
- [15] Y. F. Li and S. Y. Chen. Automatic recalibration of an active structured light vision system. *IEEE Transactions on Robotics and Automation*, 19(2):259–268, 2003.
- [16] C. K. Cowan and P. D. Kovesi. Automatic sensor placement from vision task requirements. *IEEE transaction on pattern analysis and machine intelligence*, 10(3):407–416, 1988.
- [17] M. Marcfat C. C. Yang and F. W. Ciarallo. Error analysis and planning accuracy for dimensional measurement in active vision inspection. *IEEE Transactions on Robotics and Automation*, 14(3):476–487, 1998.
- [18] H. D. Park and O. R. Mitchell. Cad based planning and execution of inspection. In *1988 Computer Society Conference on Computer Vision and Pattern Recognition*, pages 858–863, 1998.
- [19] M. Ishii S. Sakane and M. Kakikura. Occusion avoidance of visual sensors based on a hand eye action simulator system: Heaven. *Advanced Robotics*, 2(2):149–165, 1987.
- [20] J. F. Rivest W. R. Scott, G. Roth. View planning for automated three-dimensional object reconstruction and inspection. *ACM Computing Surveys*, 35(1):64–96, 2003.
- [21] R. Tsai K. Tarabanis and P. Allen. The mvp sensor planning system for robotic vision tasks. *IEEE Transactions on Robot and Automation*, 11(1):72–85, 1995.

- [22] M. Song W. Sheng, N. Xi and Y. Chen. Cad-guided robot motion planning. *International Journal of Industrial Robot*, pages 143–151, 2001.
- [23] R. Pito. Automated surface acquisition using range cameras. *Pito, R., Automated surface acquisition using range cameras 1996. PhD Thesis, University of Pennsylvania.*, 1996.
- [24] P. Boulanger F. Prieto, T. Redarce and R. Lepage. Cad-based range sensor placement for optimum 3d data acquisition. In *2nd International Conference on 3-D Digital Imaging and Modeling*, pages 128–137, 1999.
- [25] S. Y. Chen and Y. F. Li. Automatic sensor placement for model-based robot vision. *IEEE Transactions on System Man and Cybernetics, Part B*, 34(1):393–408, 2004.
- [26] W. Sheng Y. Chen Q. Shi, N. Xi. Development of dynamic inspection methods for dimensional measurement of automotive body parts. In *IEEE International Conference on Robotics and Automation*, 2006.
- [27] P.Cignoni G. Impoco and R.Scopogno. Closing gaps by clustering unseen directions. pages 307–316, 2004.
- [28] R. Pito. A solution to the next best view problem for automated surface acquisition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):1016–1030, 1999.
- [29] P. K. Allen M. K. Reed and I. Stamos. Automated model acquisition from range images with view planning. *IEEE Computer Society Conference Proceedings on Computer Vision and Pattern Recognition*, 9:72–77, 1997.
- [30] S. D. Blostein and T. S. Huang. Error analysis in stereo determination of 3-d point positions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9, 1987.
- [31] B. Kamgar-Parsi. Evaluation of quantization error in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11, 1989.
- [32] A. Domingo Sappa M. Angel Garcia, S. Velazquez and L. Basanez. Autonomous sensor planning for 3d reconstruction of complex objects from range images. pages 3085–3090, 1998.
- [33] Y. D. S. Lee D. Chung. Registration of multiple-range views using the reverse-calibration technique. *Pattern Recognition*, 31(4):457–464, 1998.
- [34] C. J. R. Chua. Point signatures: a new representation for 3d object recognition. *International Journal of Computer Vision*, 25(1):63–75, 1997.

- [35] A. Johnson. *Spin-images: a representation for 3-d surface matching*. Ph.D. thesis, Carnegie Mellon University, Philadelphia, USA, 1997.
- [36] P. Besl and N. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
- [37] R. J. Campbell and P. J. Flynn. A survey of free-form object representation and recognition techniques. *Computer Vision and Image Understanding*, 81:166–210, 2001.
- [38] T. Jost. Fast icp algorithms for shape registration. In *Pattern Recognition: 24th DAGM Symposium Proceedings*, pages 91–99, 2002.
- [39] E. Klaas B. Breuckmann, F. Halbauer and M. Kube. 3d-metrologies for industrial applications. In *Proceedings of SPIE, EUROPTO Series, Rapid Prototyping and Flexible Manufacturing*, pages 20–30, 1997.
- [40] F. X. Sillion X. D. He, K. E. Torrance and D. P. Greenberg. A comprehensive physical model for light reflection. *Computer Graphics*, 25(4):175–186, 1991.
- [41] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.
- [42] Z. S. Roth H. Zhuang. *Camera-Aided Robot Calibration*. CRC Press Inc., 1996.
- [43] R. Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 4:323–344, 1987.
- [44] P. Cohen J. Weng and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(10):965–980, 1992.
- [45] M. Potmesil. Generating models of solid objects by matching 3d surface segments. In *International Joint Conference on Artificial Intelligence*, pages 1089–1093, 1983.
- [46] A. Sarti and S. Tubaro. Detection and characterisation of planar fractures using a 3d hough transform. *Signal Processing*, 82(9):1269–1282, 2002.
- [47] M. A. Sutton N. Li, P. Cheng and S. R. McNeill. Three-dimensional point cloud registration by matching surface features with relaxation labeling method. *Experimental Mechanics*, 45:71–82, 2005.

MICHIGAN STATE UNIVERSITY LIBRARIES



3 1293 03062 4658