

University of Windsor

Scholarship at UWindor

Electronic Theses and Dissertations

Theses, Dissertations, and Major Papers

2008

Feature-based hybrid inspection planning for complex mechanical parts

Ahmed Mohamed Nabil Mohib
University of Windsor

Follow this and additional works at: <https://scholar.uwindsor.ca/etd>

Recommended Citation

Mohib, Ahmed Mohamed Nabil, "Feature-based hybrid inspection planning for complex mechanical parts" (2008). *Electronic Theses and Dissertations*. 7919.
<https://scholar.uwindsor.ca/etd/7919>

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.

FEATURE-BASED HYBRID INSPECTION PLANNING FOR COMPLEX MECHANICAL PARTS

by

Ahmed Mohamed Nabil Mohib

A Dissertation

Submitted to the Faculty of Graduate Studies through
Industrial and Manufacturing Systems Engineering
in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy at the
University of Windsor

Windsor, Ontario, Canada

2008

© 2008 Ahmed M. N. Mohib



Library and Archives
Canada

Published Heritage
Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et
Archives Canada

Direction du
Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file *Votre référence*
ISBN: 978-0-494-57648-9
Our file *Notre référence*
ISBN: 978-0-494-57648-9

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

AUTHOR'S DECLARATION OF ORIGINALITY

I hereby certify that I am the author of this thesis, and that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis and have included copies of such copyright clearances in my appendix.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Globalization and emerging new powers in the manufacturing world are among many challenges, major manufacturing enterprises are facing. This resulted in increased alternatives to satisfy customers' growing needs regarding products' aesthetic and functional requirements. Complexity of part design and engineering specifications to satisfy such needs often require a better use of advanced and more accurate tools to achieve good quality. Inspection is a crucial manufacturing function that should be further improved to cope with such challenges. Intelligent planning for inspection of parts with complex geometric shapes and free form surfaces using contact or non-contact devices is still a major challenge. Research in segmentation and localization techniques should also enable inspection systems to utilize modern measurement technologies capable of collecting huge number of measured points.

Advanced digitization tools can be classified as contact or non-contact sensors. The purpose of this thesis is to develop a hybrid inspection planning system that benefits from the advantages of both techniques. Moreover, the minimization of deviation of measured part from the original CAD model is not the only characteristic that should be considered when implementing the localization process in order to accept or reject the part; geometric tolerances must also be considered. A segmentation technique that deals directly with the individual points is a necessary step in the developed inspection system, where the output is the actual measured points, not a tessellated model as commonly implemented by current segmentation tools.

The contribution of this work is three folds. First, a knowledge-based system was developed for selecting the most suitable sensor using an inspection-specific features taxonomy in form of a 3D Matrix where each cell includes the corresponding knowledge rules and generate inspection tasks. A Travel Salesperson Problem (TSP) has been applied for sequencing these hybrid inspection tasks. A novel region-based segmentation algorithm was developed which deals directly with the measured point cloud and generates sub-point clouds, each of which represents a feature to be inspected and

includes the original measured points. Finally, a new tolerance-based localization algorithm was developed to verify the functional requirements and was applied and tested using form tolerance specifications.

This research enhances the existing inspection planning systems for complex mechanical parts with a hybrid inspection planning model. The main benefits of the developed segmentation and tolerance-based localization algorithms are the improvement of inspection decisions in order not to reject good parts that would have otherwise been rejected due to misleading results from currently available localization techniques. The better and more accurate inspection decisions achieved will lead to less scrap, which, in turn, will reduce the product cost and improve the company potential in the market.

DEDICATION

*To my whole family, friends and colleagues who stood by me and gave me support
without which I would never have accomplished this achievement.*

ACKNOWLEDGEMENTS

It is difficult to express my appreciation and gratitude to my Ph.D. supervisor, Dr. Hoda A. ElMaraghy for her contributions, guidance and continuous support throughout the course my doctorate studies at the University of Windsor. With her enthusiasm and dedication at work, my ambitions to perform state-of-the-art research for my doctoral thesis grew. I was inspired and motivated by her valuable guidance, sound advice and insightful comments. Being her student was a continuous learning process through which I gained a lot of experience in the research field. Dr. H. ElMaraghy provided us with background on manufacturing system, which we were lacking; this was fundamental to our research at the IMS centre. My committee members, Dr. Waguih ElMaraghy, Dr. Guoqing Zhang and Dr. Jonathan Wu, are to be highly recognized as well for their time and valuable suggestions that added to this work. I have been engaged with Dr. Waguih in many fruitful discussions regarding my research work; his input in the IMS centre meetings was also beneficial to me on so many levels.

I am grateful to all faculty of both Industrial and Mechanical Engineering at the University of Windsor. Many thanks to those who helped me create a healthy research environment on campus including my colleagues in the Intelligent Manufacturing Systems (IMS) Centre, Ms. Zaina Batal, the research administrative assistant of the IMS Centre and Ms. Jacquie Mummery and Ms. Brenda Schreiber, the secretary of the Industrial and Manufacturing Systems Engineering Department. Ms. Batal was very generous, providing all the help and assistance that I needed; also the two administrative staff of IMSE, Ms. Mummery and Ms. Schreiber were always considerate and nice to me and did their best to finish for us all the administrative and paper work we needed.

My utmost gratitude to my dear parents for their endless sacrifices and care throughout my life. I would like also to thank my sister and her family for their love and support. A very sincere appreciation to my beloved wife who has provided me with every kind of support that I needed while sharing with me this life time experience with unmatched patience in addition to my lovely daughters who have added an inspirational dimension to my life.

I would like to recognize my professors at Cairo University in shaping me as a scholar; I believe I had an excellent undergraduate education in one of the most prestigious engineering schools all over the Middle East. Many thanks to my Masters' advisors Dr. Adel Shalaby and Dr. Attia Gomaa who first educated me about Industrial Engineering and always stressed the importance of linking research to practice. I especially recognize Dr. Ashraf Nassef who introduced me to the field of academic research and was a source of encouragement for me to continue in this career and relate it to my mission; and above all, he was always and still is my friend.

I would like to extend my gratitude to all my friends who have always been a source of encouragement and support in my academic and social life.

TABLE OF CONTENTS

AUTHOR’S DECLARATION OF ORIGINALITY	III
ABSTRACT.....	IV
DEDICATION	VI
ACKNOWLEDGEMENTS	VII
LIST OF TABLES	XIII
LIST OF FIGURES	XIV
LIST OF ABBREVIATIONS	XVI
NOMENCLATURE.....	XVII
1. INTRODUCTION	1
1.1 BACKGROUND.....	1
1.2 MOTIVATIONS	6
1.3 PROBLEM STATEMENT	6
1.4 RESEARCH HYPOTHESES	8
1.5 OBJECTIVES AND APPROACH.....	8
1.6 OVERVIEW OF THE DISSERTATION	10
2. INSPECTION PLANNING CHALLENGES	12
2.1 INTRODUCTION.....	12
2.2 ON-LINE / OFF-LINE / ADAPTIVE CMM PLANNING	14
2.3 AUTOMATION.....	15
2.3.1 <i>Part and Probe Orientations</i>	15
2.3.2 <i>Accessibility</i>	17
2.3.3 <i>Collision-free Path Generation</i>	18
2.4 SAMPLING	19
2.5 TOLERANCE VERIFICATION TECHNIQUES.....	21
2.6 REGISTRATIONS AND LOCALIZATION	24
2.6.1 <i>Point Approach Localization</i>	27
2.6.2 <i>Feature Approach Localization</i>	29

2.6.3 Comparison between Measured Geometry and CAD model	31
2.7 SUMMARY AND CONCLUSIONS	32
3. HYBRID INSPECTION PLANNING	35
3.1 INSPECTION PLANNING – AN OVERVIEW	35
3.2 RELATED WORK	37
3.2.1 Inspection planning techniques	37
3.2.2 Multi-sensor inspection	38
3.3 PROPOSED HYBRID INSPECTION PLANNING MODEL	40
3.4 KNOWLEDGE-BASED SENSOR SELECTION MODELING	41
3.4.1 Model Parameters and Assumptions	42
3.4.2 Sensor Selection Decision Variables	52
3.4.3 Knowledge Rules Formulation (Constraints and Limitations)	53
3.4.4 Sampling	66
3.4.5 Knowledge-based Sensor Selection Output	67
3.5 ORDERING OF INSPECTION TASKS - MODELING AND OPTIMIZATION	68
3.5.1 Problem Formulation	68
3.5.2 Mathematical Model	71
3.6 APPLICATION – A WATER PUMP HOUSING INSPECTION CASE STUDY	74
3.7 SUMMARY AND CONCLUSIONS	79
4. CAD-BASED SEGMENTATION OF UN-ORGANIZED POINT CLOUD FOR INSPECTION	82
4.1 MANAGING UN-ORGANIZED POINT CLOUDS	82
4.2 RELATED WORK	84
4.2.1 Input Data Format	85
4.2.2 Propagation Techniques	87
4.2.3 Separation Techniques	89
4.3 PROPOSED SEGMENTATION ALGORITHM	92
4.3.1 Point Cloud Preparation and Initialization	94
4.3.2 Sub-point Cloud Initiation (New Feature)	97
4.3.3 Region Growing Mechanism	99
4.3.4 Loop Conditions	102
4.3.5 Feature Rectification	103
4.4 FEATURE RECOGNITION IN A POINT CLOUD	104
4.4.1 Planarity Test and Skeleton Computation for Non-Plane Surfaces	104
4.4.2 Geometric Primitive Recognition (GPR)	105

4.5 SUMMARY AND CONCLUSIONS.....	109
5. TOLERANCE-BASED LOCALIZATION.....	111
5.1 LOCALIZATION AND TOLERANCE VERIFICATION – AN OVERVIEW	111
5.2 RELATED WORK	113
5.2.1 Correspondence Search.....	113
5.2.2 Rigid Body Transformation	114
5.2.3 CMM-based Tolerance Verification techniques	116
5.3 MATHEMATICAL FORMULATION FOR CORRESPONDENCE.....	117
5.3.1 Closest point on a line to a point in space.....	118
5.3.2 Closest point on a plane to a point in space.....	119
5.3.3 Closest point on a cylinder to a point in space.....	120
5.4 QUATERNION	121
5.5 MINIMUM TOLERANCE-ZONE MODELS FORMULATIONS APPLIED TO FORM TOLERANCE	122
5.5.1 Straightness of a median line.....	122
5.5.2 Straightness of a surface line.....	123
5.5.3 Flatness	125
5.5.4 Cylindricity.....	125
5.6 PROPOSED ITERATIVE MINIMUM ZONE LOCALIZATION ALGORITHM.....	126
5.6.1 Algorithms Formulation.....	129
5.6.2 Iteration Loops	130
5.7 EXPERIMENTAL NUMERICAL VERIFICATION	131
5.8 DISCUSSIONS.....	134
5.9 SUMMARY	136
6. CONCLUSIONS.....	137
6.1 CONCLUSIONS	137
6.2 SIGNIFICANCE	142
6.3 FUTURE WORK.....	144
REFERENCES.....	146
APPENDIX A.....	157
TOLERANCE ZONE DEFINITIONS	157
A.1 STRAIGHTNESS OF A DERIVED MEDIAN LINE: CYLINDRICAL VOLUME.....	157
A.2 STRAIGHTNESS OF A SURFACE LINE: AREA BETWEEN TWO PARALLEL LINES	157
A.3 FLATNESS: VOLUME BETWEEN TWO PARALLEL PLANES	158

A.4 CYLINDRICITY: VOLUME BETWEEN TWO COAXIAL CYLINDERS	158
APPENDIX B	160
DATA SETS FOR LOCALIZATION ALGORITHM.....	160
VITA AUCTORIS	169

LIST OF TABLES

<i>Table (2.1) Geometric and Dimensional Tolerances.....</i>	<i>22</i>
<i>Table (3.1) Comparison of hybrid inspection approaches</i>	<i>39</i>
<i>Table (3.2) Advantages and Disadvantages of Laser sensors and touch probes.....</i>	<i>47</i>
<i>Table (3.3) Number of points considered for tactile probe (Beg and Shunmugam, 2002)</i>	<i>67</i>
<i>Table (3.4) Knowledge-based Sensor Selection Output</i>	<i>68</i>
<i>Table (3.5) Part features information.....</i>	<i>75</i>
<i>Table (3.6) Probe and probe orientations selected to inspect the water pump housing.....</i>	<i>77</i>
<i>Table (3.7) Cost Matrix.....</i>	<i>77</i>
<i>Table (4.1) Comparison of different segmentation techniques</i>	<i>91</i>
<i>Table (5.1) CAD data for straightness of median line.....</i>	<i>131</i>
<i>Table (5.2) CAD data for straightness of surface line.....</i>	<i>131</i>
<i>Table (5.3) CAD data for flatness of a plane.....</i>	<i>132</i>
<i>Table (5.4) CAD data for cylindricity.....</i>	<i>132</i>
<i>Table (5.5) IMZ algorithm and benchmark results for straightness of median line</i>	<i>132</i>
<i>Table (5.6) IMZ algorithm and benchmark results for straightness of surface line</i>	<i>133</i>
<i>Table (5.7) IMZ algorithm and benchmark results for flatness of a plane.....</i>	<i>133</i>
<i>Table (5.8) IMZ algorithm and benchmark results for cylindricity.....</i>	<i>133</i>

LIST OF FIGURES

<i>Figure (1.1) Measured data (Clark, 2000)</i>	3
<i>Figure (1.2) Tolerance zone</i>	4
<i>Figure (1.3) Localization process and deviation verification using FOCUS</i>	5
<i>Figure (1.4) Selecting and fitting a cylinder with METRIS</i>	6
<i>Figure (1.5) IDEF₀ of the proposed inspection system</i>	10
<i>Figure (1.6) Three Stages of the proposed inspection system</i>	10
<i>Figure (2.1) Challenges in inspection systems</i>	13
<i>Figure (2.2) 3-2-1 Approach to locate DCS (Menq et al., 1992)</i>	26
<i>Figure (3.1) Water pump housing containing both prismatic and free form features</i>	36
<i>Figure (3.2) Proposed Hybrid Inspection Planning approach flowchart</i>	40
<i>Figure (3.3) CMM Structure</i>	42
<i>Figure (3.4) Probe Head Angles</i>	43
<i>Figure (3.5) METRIS Laser Scanner LC50</i>	44
<i>Figure (3.6) METRIS Laser Scanner LC50 Specifications</i>	44
<i>Figure (3.7) Laser Scanner Triangulation Method</i>	45
<i>Figure (3.8) Laser Strip Digitization Technology</i>	45
<i>Figure (3.9) Typical Geometric Features</i>	48
<i>Figure (3.10) Sixteen types of STEP AP224 defined Manufacturing Features</i>	49
<i>Figure (3.11) Examples of Manufacturing Features</i>	49
<i>Figure (3.12) Geometric and Dimensional Tolerances Classification</i>	50
<i>Figure (3.13) Knowledge rules 3D decision Matrix</i>	51
<i>Figure (3.14) Angle α between the probe head orientation and surface normal direction</i>	53
<i>Figure (3.15) Knowledge rules – cell allocation</i>	54
<i>Figure (3.16) Knowledge rules – Tolerance Value Verification</i>	55
<i>Figure (3.17) Knowledge rules – Datum Verification</i>	56
<i>Figure (3.18) Possible Form of Straightness</i>	60
<i>Figure (3.19) Examples of difference between axis and surface element straightness</i>	60
<i>Figure (3.20) Laser Scanner – Visibility</i>	63
<i>Figure (3.21) Knowledge rules – Occlusion</i>	64
<i>Figure (3.22) Cylindrical hole occlusion calculation</i>	65
<i>Figure (3.23) Knowledge rules – Feature Dimensions</i>	66
<i>Figure (3.24) (a) Laser Scanner Parameters; (b) Scan Path showing distance parameters</i>	67
<i>Figure (3.25) Inspection Operations Modeled as Nodes in a Traveling Salesperson Problem (TSP) Model</i>	70
<i>Figure (3.26) Obtaining the required sequence from the solution TSP tour</i>	73
<i>Figure (3.27) Inspection Features in Water Pump Housing</i>	76

<i>Figure (4.1) Proposed Inspection Specific Segmentation flowchart.</i>	93
<i>Figure (4.2) Example Part.</i>	93
<i>Figure (4.3) Point cloud preparation a) point cloud b) two spheres and c) normal vector estimation.</i>	94
<i>Figure (4.4) d_1 and d_2 from laser scanner parameters.</i>	95
<i>Figure (4.5) Normal vector estimation at point P_i.</i>	96
<i>Figure (4.6) Point Data Structure (PDS).</i>	97
<i>Figure (4.7) Seed Point Selection Characteristics.</i>	98
<i>Figure (4.8) (a) good seed point and (b) bad seed point.</i>	98
<i>Figure (4.9) angle α_{ij} estimation.</i>	101
<i>Figure (4.10) angle α_{ij} in all directions.</i>	102
<i>Figure (4.11) Medial Axis Transform Example using Powercrust.</i>	105
<i>Figure (4.12) Signatures of some geometric primitives.</i>	106
<i>Figure (4.13) Geometric Primitives Recognition flowchart.</i>	107
<i>Figure (4.14) Cone slope calculation.</i>	108
<i>Figure (4.15) Axis of the round slot half-cylinder.</i>	109
<i>Figure (5.1) A simple prismatic shape.</i>	118
<i>Figure (5.2) Sample information in STEP file.</i>	118
<i>Figure (5.3) Closest point projected on a line to a point in space.</i>	119
<i>Figure (5.4) Closest point projected on a plane to a point in space.</i>	120
<i>Figure (5.5) Closest point projected on a cylinder to a point in space.</i>	121
<i>Figure (5.6) Minimum Zone for Straightness of Median Line.</i>	123
<i>Figure (5.7) Minimum Zone for Straightness of Surface Line.</i>	124
<i>Figure (5.8) Minimum Zone for Flatness Tolerance.</i>	125
<i>Figure (5.9) Minimum Zone for Cylindricity Tolerance.</i>	126
<i>Figure (5.10) Iterative Minimum Zone Localization flowchart.</i>	128
<i>Figure (A.1) Straightness Tolerance Zone of a Median Line.</i>	157
<i>Figure (A.2) Straightness Tolerance Zone of a Surface Line.</i>	158
<i>Figure (A.3) Flatness Tolerance Zone.</i>	158
<i>Figure (A.4) Cylindricity Tolerance Zone.</i>	159

LIST OF ABBREVIATIONS

ASME	American Society of Mechanical Engineers
CAD	Computer Aided Design
CMM	Coordinate Measuring Machine
DCS	Design Coordinate System
GA	Genetic Algorithm
GD&T	Geometric Dimensioning & Tolerancing
GAMS	General Algebraic Modeling System
GPR	Geometric Primitive Recognition
ICP	Iterative Closest Point
IMZ	Iterative Minimum Zone
MAT	Medial Axis Transform
NRC	National Research Council
PDS	Point Data Structure
RD	Range Data
RI	Range Image
STEP	STandard for the Exchange of Product data
SVD	Single Value Decomposition
TSP	Traveling Salesperson Problem

NOMENCLATURE

Inspection Planning Model:

Part Related Parameters:

- NF is the number of features in the inspected part.
- i is the index that represents the feature ID.
- n_i is the number of repetition of feature i .
- IF _{i} is the inspection feature (tolerance requirement) assigned to feature i .
- DF _{i} is the datum Feature ID of feature i if it exists.
- MF _{i} is the manufacturing feature type of feature i .
- GF _{i} is the geometric shape of feature i .
- GP _{i} is the geometric parameter such as the Length/Width ratio for a plan, the Diameter/Depth ratio for a cylinder, or the Larger/Smaller-Diameter/Depth ratio in a cone corresponding to the feature i .
- FO _{i} is the orientation of feature i
- O _{i} is the occlusion of feature i .
- Cell(IF, MF, GF) is a cell in the 3D decision matrix where each cell is populated with a set of rules for sensor selection relevant to the conditions represented by the cell.

Sensor Related Parameters:

- j is the index to the sensor type. is 1 or 2
- A Angle A is the probe head PH10 angle that can incline from vertically down to horizontal 105 degrees around the perpendicular axis. Its range is [0,105]. It has a limitation in steps of 7.5 degrees.
- B Angle B is the probe head PH10 angle that can rotate 360 degrees around its axis. Its range is [-180,180]. It has a limitation in steps of 7.5 degrees.
- D_S Stand off distance of the laser scanner.
- L_S Distance between Emitter and Receptor in the laser scanner
- θ_E is the incident angle of the laser beam from the Emitter
- θ_R is the incident angle of the laser beam to the Receptor
- Res _{j} is the resolution of the sensor

Acc_j is the accuracy of the sensor
 Rep_j is the repeatability of the sensor

Sensor Selection Decision Variables:

Op $Op = \{Op_k\}$ is the $1 \times K$ list of inspection task operations, which is a vector of the inspection task operations Op_k .

j_i is The sensor j used to inspect feature i .

A_{ij} is the probe head angle A used to inspect feature i using sensor j .

B_{ij} is the probe head angle B used to inspect feature i using sensor j .

α_{ij} is the average angle between the probe head orientation and the normal direction to the surface inspected from feature i using sensor j

P_{ij} is a number that represents the part orientation P_{ij} to inspect feature i using sensor j that define the inspection operation Op_k .

$(x,y,z)_{ij}$ is a key point $(x,y,z)_{ij}$ is a point to start the inspection operation Op_k .

Ordering of Inspection Tasks:

m, n are indices of two successive inspection task operations that runs from 1 to K (number of inspection tasks).

X_{mn} 0-1 integer decision variable, where m and n runs from 1 to K . The value of the decision variable is 1 if the route between digitization operation nodes n and m is taken in the obtained solution tour; otherwise it is zero.

C_1 is the non-digitization effort taken to change part orientation between two successive operations and expressed time units.

C_2 is the non-digitization effort taken to change sensor between two successive operations and expressed time units.

C_3 is the non-digitization effort taken to change probe head orientation between two successive operations and expressed time units.

C_4 is the time taken by the probe head to travel between two successive operations.

C_{mn} is the total non-digitization effort to switch between operations m and n .

V is the rapid traverse speed of the coordinate measurement machine head.

$d((x,y,z)_m, (x,y,z)_n)$ is the distance between the two key points of the two successive operations m and n

Segmentation Algorithm:

- N_F is the number of features in the CAD model
- SP_{SF} is a seed point for each feature.
- K_{SF} is a threshold value as a continuity measure for the feature.
- $P = \{P_i\}$ is the point cloud set of measured points P_i
- (S_1, S_2) are two-levels sphere neighborhood functions to determine the neighbor of the point P_i .
- (R_1, R_2) are the two radius corresponding to the two-levels sphere neighborhood functions.
- d_1 is the distance between two points in the laser strip. This value is obtained from the laser scanner settings.
- d_2 is the distance between two laser strips. This value is obtained from the laser scanner settings.
- d is the distance between two consecutive points in the point cloud.
- D_{ij} is the distance between two neighbor points P_i and P_j .
- n_i is the normal direction of the plane surface that can be fitted to the point P_i and its neighboring points in the sphere S_1
- α_{ij} is the angle between the two normal vectors for two neighbor points i and j
- W_{ij} is a weight factor between two neighbor points P_i and P_j .
- W_i is the cumulative weight for point P_i .

Tolerance-based Localization Algorithm:

- N_p is the number of corresponding points.
- \vec{A} is a point on the CAD model
- \vec{B} is a point on the CAD model
- \hat{t} is a unit vector (that represents the inspected shape on the CAD model)
- \vec{P}_i is a measured point from the point cloud set
- \vec{P}'_i is the closest point on the matching feature to the point \vec{P}_i
- \vec{P}'_{il} is a point on the center line of a cylinder, that is the closest to the point \vec{P}_i from the point cloud.
- \vec{P}_S is a point on the CAD model
- l' is a distance projected on a line
- $\vec{r}_{pp'}$ is a vector perpendicular to a plane joining between a point \vec{P}_i

- in the point cloud and the corresponding point \vec{P}_i' on the plane
- \vec{q} is a unit quaternion
- \hat{u} is a unit vector
- θ is the rotation angle around the unit vector \hat{u}
- R is a the rotation matrix to apply 3D rigid body transformation.
- MZ_t is the calculated minimum tolerance zone such as MZ_{St} for straightness, MZ_{fl} for flatness and MZ_{Cyl} for cylindricity.
- t is the value of the tolerance size
- r is the radius of a cylinder
- r_i is the value of the distance between the point \vec{P}_i and the axis \hat{T} (unit vector)
- r_{max} is the maximum radial distance between all the point \vec{P}_i and a median line.
- \hat{C}_p is the unit normal vector to a cutting plane.
- e is the amount of error generated due to un-satisfied constraint.
- D is the set of distances d_i
- x_i is a point on the CAD model corresponding to \vec{P}_i
- F is the type of form tolerance to be verified
- \vec{P}_k is the rotated and translated point \vec{P}_i in iteration k .
- $\Sigma_{pp'}$ is the cross covariant matrix between the measured point set and the corresponding projected points on the CAD model.
- $Q(\Sigma_{pp'})$ is a 4x4 matrix whose components generated from the cross covariance matrix between two pairs of point sets.
- $\vec{\mu}_p$ is the mean of the measured point set.
- $\vec{\mu}_x$ is the mean of the projected point set on the CAD model.

1. INTRODUCTION

In today's markets, the manufacturing industry is under tremendous pressure to respond to customers' requests quickly and effectively. Offering good quality products has always been a goal for competitive manufacturing enterprises. Studying quality in manufacturing systems includes, in a general sense, product design, process design and control and finally inspection with its various levels. In this chapter, the motivation behind the current work, the proposed approaches, and an overview of the dissertation is presented. Different elements of the inspection process are briefly overviewed. Finally, the components of the proposed inspection system are outlined.

1.1 Background

Inspection has been defined as a process of examining attributes of a part and determining if it does or does not conform to design specifications. Design functional requirements or assembly conditions on a manufactured part are normally translated into geometric constraints to which the part must conform. These constraints are expressed in terms of the American Society of Mechanical Engineers' standards (ASME Y14.5M-1994) for Dimensioning and Tolerancing (GD&T). The examined attributes may be quantitative (e.g. dimensions) or qualitative (e.g. appearance). The process of inspecting parts also involves identifying the kind of defect (deviation or lack of conformity from design specifications) or defects a part may have. Inspection is generally carried out to ensure desired product quality, minimize waste of materials (scrap), unnecessary stoppages of production equipment, interruptions in production flow, return of goods sold, avoid dissatisfaction among customers, and, in general, to safeguard business reputation. Today's market, which seeks high variety, high volume and good quality products, urged the inspection systems to include tools with high technology. Those tools could be contact or non-contact such as Coordinate Measuring Machines (CMM), laser and white light technology. They differ not only in the way they measure the part and capture the data but also in the speed and accuracy.

Traditionally (and in some industries up till now) the inspection process was based on visual examination or some basic tools such as Go/No-Go Gauges. The development of new technologies and the growing trend of automation in manufacturing gave the inspection process new dimensions. There are generally two types of modern measurement data acquiring methods: contact measurement and non-contact measurement. The Coordinate Measuring Machine (CMM) is one of the most effective inspection facilities used in the manufacturing industry. It is built with highly accurate and stable machine structure, simple yet precision transmission mechanisms, a highly reliable sensing system, data processing software and a computer system (Merat *et al.* 1991). The CMM consists of a machine tool like structure, with precision sideways and scales and some form of sensing to determine the point of contact. A variety of sensor technologies are in common use; touch trigger probes, contact scanning probes and a variety of non-contact probes such as laser sensors. Different probes and accessories such as the extension poles, which are changeable and adjustable, can be installed. The probe can incline in an angle from vertically up to horizontal and rotate 360 degrees around its axis and 105 degrees around the perpendicular axis to achieve most orientations required to acquire the data. Those data are transferred to the computer through a controller.

For product with regular geometric primitive features, such as planes and cylinders in the form of holes, pockets, slots and keyways, the inspection techniques and equipment are already mature in the current industrial applications. Also, a touch probe mounted on a CMM can be used effectively to assess the coordinates of these features. For parts with sculptured surfaces, the inspection process becomes complex. It is difficult to use hard gauges and very time consuming using traditional contact-type measuring devices on a sampling basis. Also, these parts are usually expensive and any undetected defects can cause costly errors at final assembly of products or in the performance of an entire machine system. For such products, 100% inspection may be required. Non-contact methods such as laser scanning can quickly provide a large amount of digitized point from the surface. In recent years, extensive research has been carried out to tackle both fundamental and application issues concerning sculptured surface inspection. Blais (2004) reviewed 20 years of development in the field of 3-D laser imaging at the National

Research Council (NRC). In the market, there exist two major types of laser-scanning devices: a) Laser-scanning probe mounted on a CMM, and b) A robotic arm equipped with a laser probe as its end effector.

The accuracy of three-dimensional laser scanning devices has been improved greatly and can provide a viable solution for this purpose. However, it has not reached the accuracy of contact devices such as CMMs. Figure 1.1 shows the difference between measurement results of a water pump housing scanned on a sampling basis using a touch probe and the same part scanned using a laser scanner (Clark, 2000). The points in Figure 1.1(a) are more accurate than the points in Figure 1.1(b). However, the surface fitted to a smaller number of points would be less accurate than the one fitted to a larger number of points.

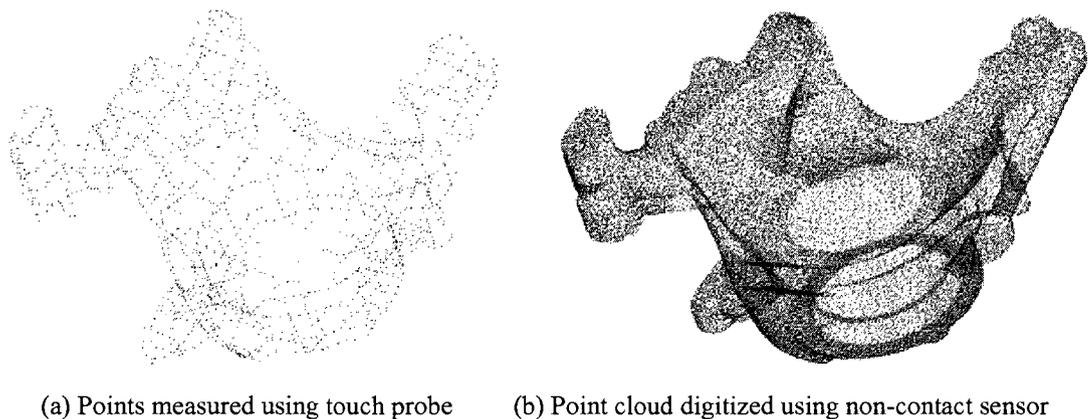


Figure (1.1) Measured data (Clark, 2000).

Surface inspection involves not only data acquisition but also variation quantification and location on the measured surface. In designing manufactured parts, surfaces are often assigned a dimensional tolerance to control the variations of size and a geometrical tolerance to relate this surface with the rest of the product. To verify the acceptance of a manufactured surface, one needs to check if the measured values fall within the designed tolerance zone. The tolerance zone can be regarded as a space between the offset boundaries of a nominal design part, which describes the permissible variation range of geometric characteristics as shown in Figure 1.2.

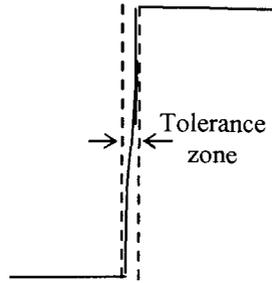


Figure (1.2) Tolerance zone.

In case of sculptured surface, the inspection techniques differ from those used for regular shaped features. To compare the measurement surface with the design model, it is essential to arrange these two surfaces in a common coordinate system. The digitized data of the product in the Measurement Coordinate System (MCS) is then compared with the design model in the Design Coordinate System (DCS). The first step in conducting such inspection is to align these two coordinate systems together. This process is called *localization*. In localization research, all the techniques in the literature were trying to align the digitized part coordinate system to the design coordinate system through iterative processes and optimization techniques. All the approaches addressed in the literature were based on *point-to-point* or *point-to-plane* correspondence. In both cases, tolerance requirements are restricted to verify the coordinate deviation after the localization process.

The state of the art now in inspection systems that use large amount of measured point is to perform the localization process to align the Measurement Coordinate System (MCS) of a part to the Design Coordinate System (DCS) as shown in Figure 1.3. The methods for alignment between design model and measurement data acquired by these systems normally include a traditional 3-2-1 approach, semi-automatic (human-computer intervention) and automatic processes such as *best fit* and *feature-based alignment*. With the semi-automatic processes, users need to make the initial alignment by manually arranging the design model and measurement data sufficiently close. The inspection systems, then, carry out the remainder of the localization operations by minimizing the difference between all the points of the point cloud and the corresponding point on the CAD model. Then the deviation between the point cloud and the CAD model is

expressed in the form of colored error maps which display the difference between the coordinates of corresponding points on the CAD and fitted (substitute) models, however, this difference is not related to all important tolerance specifications.

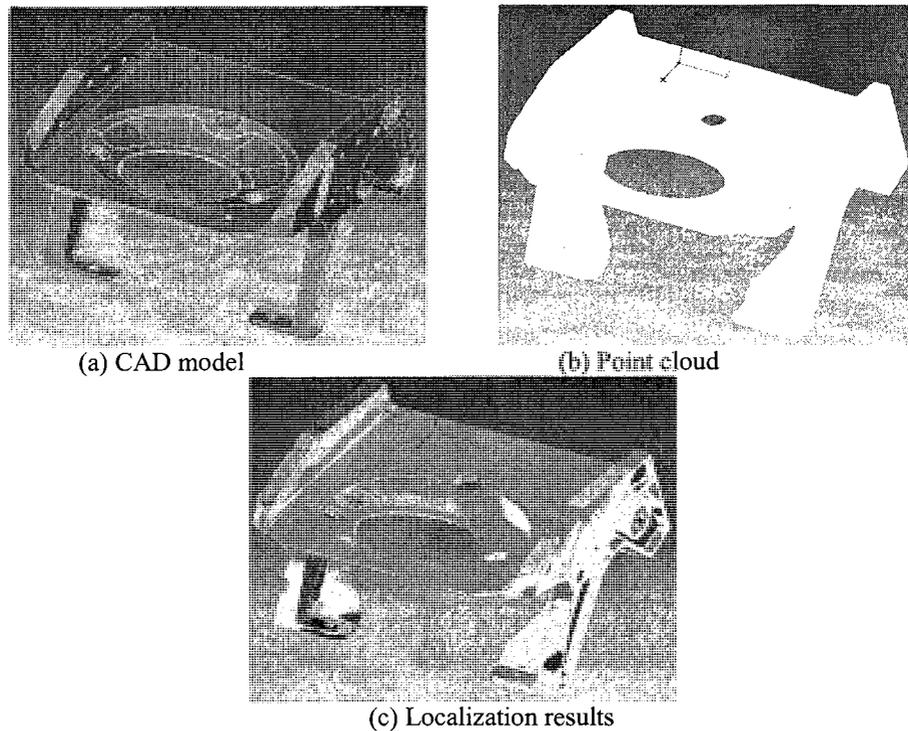


Figure (1.3) Localization process and deviation verification using FOCUS.

Sometimes features such as planes, circles, lines, spheres and/or some other quadratic surfaces, are used to start the automatic alignment. The process of selecting, separating and fitting the geometric feature to the points manually in available commercial inspection data analysis systems is illustrated in Figure 1.4. Such process is prone to errors and totally depends on the experience of the operator. Such error is compensated later with successive iteration of computation for the closest point. However, this alignment process tries to minimize the least square errors but neglects the geometric tolerance requirements specified by the designer.

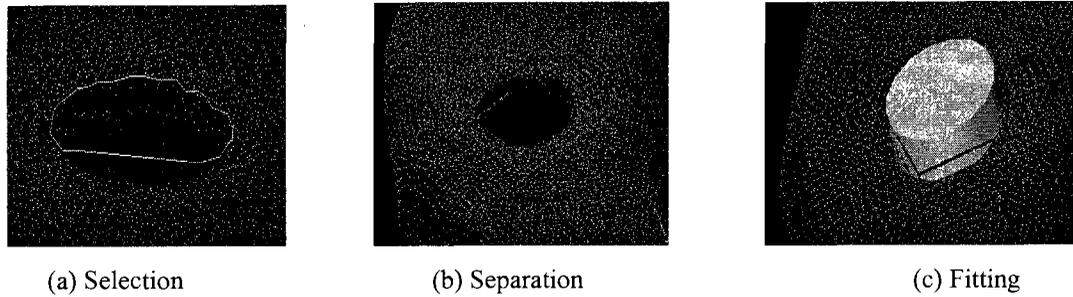


Figure (1.4) Selecting and fitting a cylinder with METRIS.

1.2 Motivations

Products quality including their geometric and dimensional accuracy and conformance with the design intent expressed by the specified tolerances is one of the deciding factors in today's manufacturing competition. In the new market, the aesthetic aspect of the products is becoming an important concern for the customers. The complexity of a product surface together with the customer's expectations concerning the accuracy level places a high demand on the efficiency of the inspection process involved in the manufacturing systems. In addition, the availability of new materials and production tools make possible the fabrication of very complex shapes, thus providing a greater freedom to the designer's creativity. Hence, Inspection became more and more a complex process. In addition, reducing manufacturing cost through decreasing the rejection rate places additional demands on the in-process inspection to achieve a high performance level. The previous inspection challenges were addressed through different approaches by quality engineers and by using different inspection tools. The motivation of this research is to address the inspection challenges identified earlier and to overcome the shortcomings of contact and non-contact inspection by proposing a hybrid approach for inspection planning, digitization, and data interpretation that capitalizes on their strengths.

1.3 Problem Statement

Contact measurement is more accurate than non-contact one. However, it suffers from low speed and added errors due to surface fitting using a relatively small number of points. Non-contact measurement is recognized as being able to capture huge number of points leading to a better fitted surface but at the same time the accuracy of each

measured point is less than that of the contact measurements. Moreover, the large number of points, which causes computation burden, is usually reduced by filtering the point cloud. This causes another source of inaccuracy due to the dislocation of the original point in the point cloud to an averaged point in the resulting filtered point cloud. In addition, non-contact measurement is unable to digitize internal occluded features. Therefore, a need for a hybrid inspection planning approach that utilizes the benefits of both measurements techniques and overcome their shortcoming is highly recognized. The generated plan would efficiently combine the use of the two types of sensors, select the appropriate sensor for the inspection tasks and optimize the sequence of these tasks to improve the quality of inspection decisions.

Another shortcoming in current inspection planning practices is that most of the localization techniques are limited to the calculation and minimization of the absolute deviation of the measured part's dimension from the original CAD model. Tolerances requirements are then verified. This would produce misleading conclusions regarding final inspection decisions such as accepting bad parts and rejecting good parts. A further step that includes a comprehensive approach where form and geometric tolerances are considered in the localization process in a single step is required to overcome this shortcoming.

A segmentation process that divides the measured point cloud into meaningful segments (sub-point clouds) corresponding to the features to be inspected is needed to perform such tolerance-based localization process. Current segmentation algorithms deal with mesh representations and associated loss of accuracy compared to the one of the original measured points. A segmentation algorithm that deals directly with the point cloud and produce the same point cloud but divided based on the inspected features from the CAD model is needed to accomplish the previously mentioned goals.

1.4 Research Hypotheses

The main hypotheses of the current research are:

1. Tactile sensors with their low digitization speed are not best suited for the current manufacturing environment with its increasingly complex part designs. Non-contact sensors on the other hand are not as accurate as contact scanners and fall short to reach occluded or shadowed areas in the measured part. Therefore, a hybrid (contact/non-contact) digitization technique would best match the current challenging digitization requirements. Mathematical modeling and programming is a crucial solution method to address such hybrid inspection planning problem.
2. Available localization techniques, with their rigid definition of minimization of the deviation of all the measured points from the CAD model, are not best suited for the current manufacturing environment with its complex parts and associated tolerances. Tolerance verification techniques on the other hand are limited to simple parts with single feature. Therefore, a localization technique that is also able to verify the tolerance requirements in earlier stages would improve the inspection decisions.
3. To obtain sound inspection decisions from the proposed tolerance-based localization algorithm for independent features, an automatic segmentation of the obtained un-organized point cloud from different types of sensors with different orientation, is a necessary step, where the output is the original measured points for each feature and not a substitute, which would lead to more accurate inspection decision.

1.5 Objectives and Approach

The objective of this research is to develop a hybrid inspection planning system (Contact/Non-contact) for complex geometric surfaces (Prismatic and/or Free-Form) that is capable of automatically determining the best method of measurement for given features, planning the hybrid inspection tasks and analyzing and manipulating the different sets of data obtained from both types of measurement. The objective is to optimize the speed of measurement and accuracy of contact measurement and at the same

time capture a huge number of points including the internal features of the part. The system would have the potential of minimizing the input from human inspector, reducing inspection error and decreasing time and cost. In addition, the objective behind the proposed system is to overcome the shortcomings of localization process by including geometric tolerances leading to better inspection results.

Inspection planning can be considered as a process that transfers design data to the inspection system and the entire inspection operation is carried out within a minimum time and with reduced uncertainty. The overall inspection planning process consists in generating all possible inspection plan tasks containing specific information about how toleranced geometries are to be inspected. This process starts with identifying features in the CAD model of the part to be inspected and selecting the methods of measurement (contact/non-contact) required for each feature type; then, the inspection tasks are ordered to minimize the effort to switch between sensors, sensors' orientations and part orientations. Once the part is digitized, the measurement points are analyzed. This analysis includes segmentation, localization and finally tolerance verification. The previous overall inspection planning system is summarized in the following IDEF₀ model shown in Figure 1.5. The proposed inspection system is composed of three stages; each stage is detailed in a separate chapter. The three stages of the proposed approach are illustrated in Figure 1.6. The first stage is the generation of the point cloud in terms of selection, planning and digitization. The second stage addresses the point cloud preparation in terms of segmentation of the point cloud into sub-point clouds. The third stage is the point cloud verification process by applying the developed tolerance-based localization algorithm.

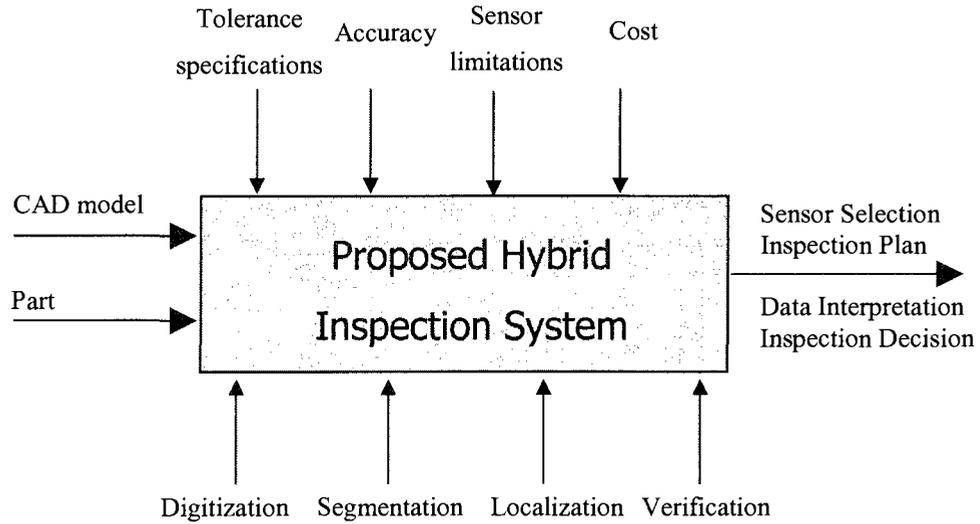


Figure (1.5) IDEF₀ of the proposed inspection system.

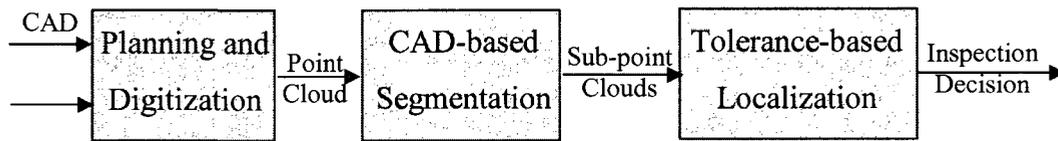


Figure (1.6) Three Stages of the proposed inspection system.

1.6 Overview of the Dissertation

The following is an outline of the dissertation:

- Chapter Two presents the different challenges in the inspection planning field of research and the different approach from researchers to address. Thorough critiques are provided in the subsequent chapters.
- Chapter Three sketches the planning methodology, in which a knowledge-based system has been developed for selecting the most suitable sensor for the inspection task using a proposed inspection-specific features taxonomy, followed by a new Travel Salesperson Problem (TSP) formulation, which has been developed for sequencing of hybrid inspection tasks, where a novel sub-tour elimination constraint has been formulated. Details of the proposed mathematical model are provided. A water pump housing case study was used to illustrate the need for using two different types of sensors to obtain a complete and accurate point cloud.

- Chapter Four presents the developed segmentation algorithm to divide the obtained point cloud into sub-point clouds, based on information from the CAD model. Each sub-point cloud includes the original measured point and represents a feature to be inspected.
- Chapter Five describes the third stage of the proposed inspection system. An iterative tolerance-based localization algorithm has been developed and demonstrated, where the minimum tolerance zone is estimated in each iteration. Experimental data for primitive basic geometric parts that were used as benchmark problem by most of the literature was used to illustrate and validate the method.
- Chapter Six concludes the dissertation with a brief discussion and a list of the research findings and conclusions.

2. INSPECTION PLANNING CHALLENGES

This chapter provides a review of the literature with the most relevance to the problem of inspection process planning and its challenges. Since the thesis address different challenges in various steps of the inspection process, this chapter presents only a general review with background about the different challenges and the different approaches by researchers to overcome these challenges. More focused and detailed critiques are presented in each chapter to highlight the corresponding challenges and position this work relative to the various schools of thought. A chronological order was generally followed.

2.1 Introduction

Several comprehensive reviews about inspection techniques of objects including 3D mechanical parts were conducted. Among them, Newman and Jain (1995) surveyed the automated visual inspection systems and techniques covering the literature before 1993. They presented taxonomy of the inspection systems problems based on their sensory input and the type of inspection decisions to be made. Limaiem and ElMaraghy (2000) summarized the main characteristics of some of the most important works in tactile inspection planning using CMM. They based the classification on the accessibility analysis and the operations sequencing. Malamas *et al.* (2003) focused in their survey on industrial vision systems. Concentrating on more recent developments, Li and Gu (2004) provided a literature review about inspection and comparison techniques for parts with freeform surfaces, which covers both contact and non-contact measurements. They classified the inspection planning based on the tool used to digitize the part.

The inspection planning can be classified based on the tool used for measuring the part's dimensions. "Hard gauges" are the traditional tools for inspecting geometric features. The "envelope principle" used in tolerance specifications evolved from gauging technology. For instance, a go-gauge provides an envelope for checking the maximum material condition, whereas a no-go-gauge provides an envelope for checking the

minimum material condition. The process of collecting and interpreting CMM or laser sensors data to inspect geometric features is sometimes called “Soft Gauging”. Neither the ASME nor the ISO standards specify a method for establishing the (minimum zone), and several different algorithms have been developed for various typical features.

Planning for inspection of surfaces (Regular form / free form surfaces) and the number of parts to be inspected can be a challenge in inspection researches as the main goal. In details several challenges, such as sampling, accessibility analysis and part and probe orientation selection, appear to achieve this goal. A classification of the inspection systems challenges and areas of research in the last decades based on sensory input and the type of inspection decisions to be made is presented in Figure (2.1). Some of the challenges in both contact and non-contact types can be applied to the other, such as automation and collision free path planning can be applied to the non-contact type of measurement, but this classification is a basis for the literature review.

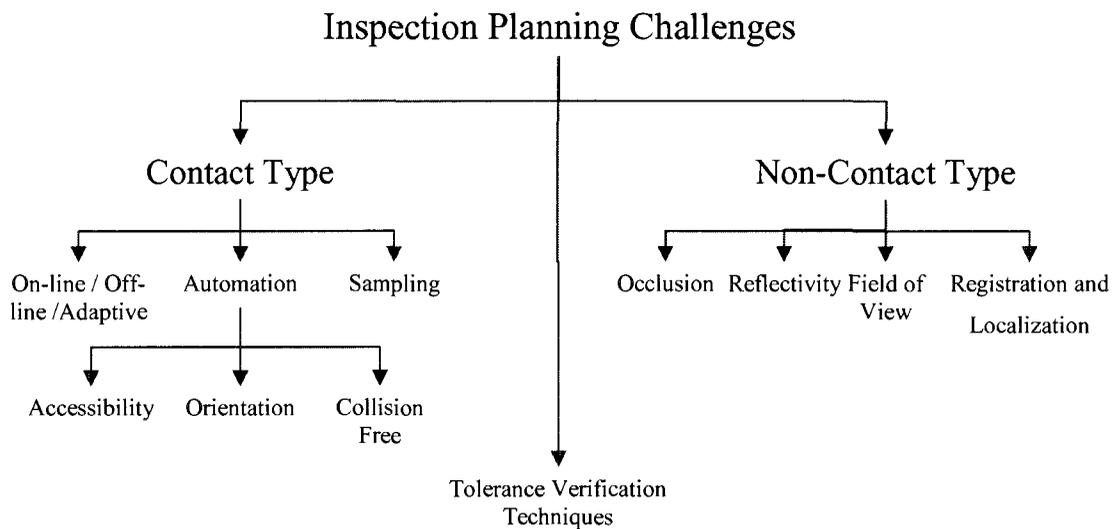


Figure (2.1) Challenges in inspection systems.

The following sections provide an in depth review of the approaches that dealt with the basic element of inspection planning using CMM and Laser scanners, which includes collision free path generation, accessibility analysis, orientation selection and sampling. They also review the different techniques for tolerance verification. The main problems

for laser scanning such as occlusion, out of field of view and localization are also addressed; in addition to other research work that might be adopted to solve this problem.

2.2 On-line / Off-line / Adaptive CMM Planning

CMM used to be programmed manually by moving the measuring probe through a sequence of moves, which are then repeated during the subsequent measurement. This is called the teaching method, which cannot be done before the part is manufactured. It is tedious, time consuming, and ties up expensive equipment. Off-line programming systems avoid some of these drawbacks because they work with a computer model of the part, rather than the physical part itself. However, they still involve considerable manual work and require powerful simulation and visualization capabilities. Some industries use the laser scanner for on-line inspection for its digitization speed while CMM are used for off-line inspection on a sampling basis. Recent researches extend the idea of integrating the CMM into the manufacturing process to address the automation of off-line programming techniques with a trend towards a more intelligent and adaptive inspection environment (Ziemian and Medeiros, 1997). This leads to a decrease in the total time dedicated to the inspection process, an increase in the program accuracy and an increase in the productivity of the machine.

Chen *et al.* (2004) and Yang and Chen (2005) proposed a new environment for CMM inspection path planning named Haptic Virtual CMM (HVCMM). HVCMM is a simulation model of the CMM's operation and its measurement process in a virtual environment with haptic perception as if an operator were in front of a real CMM and moving a real CMM probe. By pointing a probe at the 3D computer-aided design (CAD) model of the part, a haptic device is used to generate the collision-free inspection path of a part using teach pendant programming. Surface Voxels are used for quick collision detection.

Lin and Lin (2001) used, for on-line inspection, the grey prediction in grey theory to plan the number of measuring points of the next work piece and to predict the geometry tolerance dimension of the next work piece. A grey system is a system in which part of

the messages is known and the other is unknown. The grey system theory is a theory dedicated to resolving analysis modeling, prediction, decision-making and control in a grey system. Its main functions are the effective processing of the uncertainty, multi-input, discrete data and data incompleteness.

Liu *et al.* (2003) addressed the high speed CNC-CMM integrated machining center. They presented a framework for CMM part programming that differs from traditional approaches based on the CAD data, by analyzing NC machining codes. Hence, the machined features are inspected after the completion of any machining operation.

2.3 Automation

The need to automate the inspection process appeared with the advent of programmable CMM and more accurate and flexible tools. Three inter-related challenges face the automation of the inspection processes; 1) feature accessibility, 2) work piece/probe orientation and 3) a path free of collision.

2.3.1 Part and Probe Orientations

The work piece orientation, probe selection and probe orientation are usually determined through clustering all possible set of faces accessible with common geometrical constraints and minimizing the changes of probes and part orientation.

ElMaraghy and Gu (1987) developed the first expert system for inspection planning. In this system, inspection features were grouped according to their measurement features and prioritized based on the importance of their functional requirements. The expertise of human inspection planners has been transcribed into expert rules, and used for clustering features to be inspected. The task planner was developed based on a feature oriented computer-aided modeling approach using PROLOG. Inspection features were grouped according to their dimensional reference datum and the GD&T requirements, and then assigned inspection priorities based on the nature and magnitude of the related tolerance. Feature accessibility by the CMM probe in a given part orientation is also checked, and measurement points are clustered and planned accordingly.

Ziemian *et al.* (1997) tackled the orientation of the work piece for a CMM equipped with an indexable probe. The objective was to automate the probe selection decision and part setups. A heuristic technique is implemented to analyze accessibility results and probe selections in defining the set of work piece orientations.

Limaiem and ElMaraghy (1997a, 1998, and 1999) achieved a high degree of CMM inspection automation by formalizing the different tasks and the knowledge related to each step in the inspection process. The formulation was optimized by minimizing set-up changes, probe changes and probe orientation changes. They integrated the path planning with the accessibility analysis, clustering and sequencing to make it easier to generate alternative path plans when changing the probe orientation or changing the probe itself. They discretized the working space, then approximated the part and the objects in the environment by a set of Cartesian boxes using Octree decomposition and approximated the probe using a set of spheres. An algorithm for simplified interference checking was developed.

Beg and Shunmugam (2002) developed an object-oriented planner for the inspection of prismatic parts. Two types of protrusion have been incorporated: rectangular bosses and cylindrical bosses. The depressions included are: step, slot, hole, prismatic hole, counter-bore, slot with round ends, and an open type T-slot. The problem of selection of part orientation was formulated as one involving ranking of the base surfaces, i.e. parent faces of the prismatic part, based on the following criteria given in decreasing order of importance to ensure stability and maximum number of features are inspected without any changes in part orientation. They applied Fuzzy logic for decision making of the selection of part orientation and sequencing of probe orientation. The sampling is based on a fixed number for each feature and the allocation is based on the aspect ratio according to the tolerance specified. Concerning the accessibility analysis, the actual probe unit is approximated to a rectangular block and two cylinders of different radii. This approach is to select a probe orientation and determine whether it is feasible.

Hwang *et al.* (2004) proposed a CMM inspection planning method that minimize the number of part setups and probe orientations using greedy heuristic method and a continuous Hopfield Neural Network to minimize the inspection feature sequence. The proposed method was limited to prismatic parts and the probe orientations are limited to five orientations along the axes of the CMM coordinate system.

Cho *et al.* (2005) developed a series of heuristic rules by analyzing the features information such as the nested relationships and the possible probe approach directions to inspect work pieces having many primitives.

2.3.2 Accessibility

One of the major issues is to ensure that a suitable set of points on each of the part's surfaces to be measured can be reached by the probe without collisions. This is called accessibility analysis. A quantitative characterization of the accessibility of a surface feature is computed as the bounds (set of probe orientation) of the associated feasible probe orientations. Based on this concept, two types of accessibility analysis are known. The first one is known as Local Accessibility Analysis, where these bounds are specifically defined as the Local Accessibility Cone of the feature, considering only the feature itself and only the characteristic of face. The second one is known as the Global Accessibility Analysis where these bounds are specifically defined as the Global Accessibility Cone of the feature, considering the entire work piece and potential intersections with all features of the part. It can provide collision-free inspection of the feature. Three broad approaches to solve the feature accessibility problem were found in the literature:

- The first is a relative approach, which considers a fixed orientation of the work piece and fixtures. It selects a probe orientation by some strategy, and determines whether it is feasible.
- The second approach is to determine all feasible probe orientations, as a subset of all available probe orientations for a given CMM probe, and perform an optimization analysis to select the best orientation.

- The third approach, absolute approach, in which case the orientation is determined after the sequencing of measurement points and the path planning of the probe independently from the part orientation and its environment. The absolute approach is justified by the fact that measurement points accessible with the same probe orientation may generally be grouped in the same set-up.

Limaiem and ElMaraghy (1997b) developed a general method for features accessibility analysis based on the intersection of concentric spherical shells centered at the measurement point. This method is particularly interesting if the characteristics of the probe are not known in advance. In addition, using this method makes it very easy to extract the discrete accessibility domain as a subset of the continuous domain.

Chiang and Chen (1999) proposed a mathematical modeling approach for resolving the accessibility of only through slots. They used the accessibility of the two side surfaces of the slot to obtain probe orientation by using the real geometrical and dimensional relationship of the probe and the slot.

Vafaeseefat and ElMaraghy (2000) presented a methodology to automatically define the accessibility domain of measurement points and tolerance information from a CAD model and then grouped them into a set of clusters using a heuristic algorithm by classifying points based on the maximum intersection between their accessibility domain, into a set of clusters. This methodology could be applied with complex parts since it is computationally efficient and not limited to a particular solid model or surface representations particularly if obstacles such as fixtures and clamps, as well as the probe geometry are taken into consideration.

2.3.3 Collision-free Path Generation

The automation of a collision free path planning using CMM was of interest to many researchers. The path was usually generated, then checked for collision by simulation. If collision is detected, the user modifies the path interactively or a set of heuristic rules is used to move the probe away from the detected interference region.

Yau and Menq (1991 and 1995) presented a hierarchical procedure to detect collisions. The initial path is first determined, and then each individual path segment is checked for collision by the calculation of intersection between the moving CMM and the part. If interference is detected, the trajectory is modified according to some heuristic rules before going to the second level. They simulated the inspection path in a CAD environment before it is carried out by the real CMM. Three CMM components are considered for collisions: Probe tip, probe stylus and CMM column. The probe tip was modeled for simplification as a point instead of a sphere. The probe stylus was modeled as a line instead of a cylinder. The CMM column was modeled by a tube with square cross section the dimension of which is the column diameter.

Lu *et al.* (1995, 1999) developed an integer linear programming model of the distance moved by the probe of the CMM and used Genetic Algorithm (GA) to find the most efficient path to reduce this distance. The collision-free path planning included the large number of testing points and dummy points with no repeated routs. They used an Artificial Neural Network technique to carry out the inspection path management for multi-component inspection. A multiple layer neural network model was developed for the pattern recognition of inspection paths.

Lin and Chow (2001) used the dynamic programming method for planning the measurement sequence for various geometric features of parts consisting of several basic feature elements. They divided the path planning into global and local path planning.

2.4 Sampling

The location and number of data points affect the time of measurement and accuracy of the result. When scanning more points, the fitted surfaces accuracy is better while the time for inspection is longer. To reach a certain measuring accuracy, the relation between the tolerance, geometry features and the number of measuring points at the same time should be taken into consideration. This process is called sampling and it can be divided to two stages:

- The first is to select the number of points.
- The second stage is to choose where those points could be located for better representation of the measured surface.

Ainsworth *et al.* (2000) presented a module for discrete point sampling of sculptured surfaces using touch-trigger probes. The methodology implemented uses the CAD model of the part at each step, with NURBS being the principal modeling entity. Several sampling criteria were proposed. The measurement points were located along an isoperimetric surface curve. The sampling process applies a recursive subdivision algorithm such as chord length, minimum sample density, and surface parameterization.

ElKott *et al.* (2002) developed an algorithm to select an effective sampling plan for the tactile CMM inspection planning of free-form surfaces. The sampling methods presented utilize a NURBS representation of the free-form surface. The developed algorithm falls in two categories: surface feature-based sampling and optimal sampling. The surface feature-based sampling algorithm utilizes user-defined criteria and applies them to locate sample point on the NURBS surface. NURBS surface parameters, such as the surface curvature change, and patch sizes, were used to guide the sampling process. Optimization of the inspection sampling is done using Genetic Algorithm (GA).

Hwang *et al.* (2002) developed a knowledge-based inspection planning system using a hybrid Neuro-Fuzzy method with weight parameters optimized using GAs. The knowledge-based system integrates part geometry information from the 3D CAD file, tolerance information stored in the database, heuristic knowledge of experienced inspection planners and the user input. They determine the number and positions of measuring points. Initially, the Fuzzy rules are prepared by the hybrid Neuro-Fuzzy network where historic inspection planning data such as size of the measurement surface, the degree of tolerance and the number of measurement points that have been utilized for the previous inspection processes, are used to set the Fuzzy variables and Fuzzy membership functions. Each Fuzzy rule has weighting value from 0 to 1. The weighting values of Fuzzy rules are optimized by a GA to find the best values for the constants.

Badar *et al.* (2003) based the sampling optimization technique on the type of tolerance used. For straightness, region-elimination search is used. For flatness two pattern search methods were employed and compared: Tabu search and hybrid search (combination of coordinate search and Hooke-Jeeves search).

2.5 Tolerance Verification Techniques

The inspection process is driven by the tolerance requirements specified by the designer. Tolerance verification is to evaluate the deviation of the measured part from the CAD model. Functional requirements or assembly conditions on a manufactured part are normally translated into geometric constraints to which the part must conform. These constraints are expressed in terms of the American Society of Mechanical Engineers' standard (ASME Y14.5M-1994), Geometric Dimensioning and Tolerancing GD&T. Similar international standards are provided by the International Organization for Standardization's ISO 1101-1983. In these standards, allowable variation of individual and related features is based on the "envelope principle"; that is, the entire surface of the part feature of interest must lie within two envelopes of ideal shape. This is also known as "Taylor's principle". According to the definitions in the ASME Y14.5 standard, the datum feature is determined by the envelope principle, i.e., the tolerance zone of the datum feature is a minimum. Tolerances can be classified to three main types: 1) Coordinate or size tolerance, 2) Form tolerance and 3) Geometric tolerance. Table 2.1 shows the different types of tolerances and their inspection features.

"Hard gauges" are the traditional tools for inspecting geometric features. The envelope principle used in tolerance specifications evolved from the gauging technology. For instance, a Go gauge provides an envelope for checking the maximum material condition, whereas a No-Go gauge provides an envelope for checking the minimum material condition. The process of collecting and interpreting CMM data to inspect geometric features is sometimes called "Soft Gauging". Neither the ASME standards nor the ISO standards specify a method for establishing the (minimum zone), and several different algorithms have been developed for this purpose for various typical features.

Table (2.1) Geometric and Dimensional Tolerances

Types of Tolerances		Point	Line	Surface		
Coordinate [ANSI] (Size Tolerance)	Distance	Point	line	planar		
	Diameter		circle		revolution	
	Radius		circle		revolution	
Form Tolerance	Straightness		line	planar	revolution	
	Flatness			planar		
	Circularity		circle			
	Cylindricity				revolution	
	Profile		curve			free-form
Geometric [ANSI] (Position Tolerance) (Orientation, Location, Run-Out)	O	Angularity		line	planar	
	O	Parallelism		line	planar	
	O	Perpendicularity		line	planar	
	L	Coaxiality	Point	axis		
	L	Symmetry		axis	median plan	
	L	Position	Point	line-axis	planar	
	R	Circular		axis		revolution
R	Total			planar	revolution	

Although many algorithms for the evaluation of tolerances exist, the Least-Squares Method is commonly employed for data fitting in CMM due to its simplicity. The objective of this method is to minimize the sum of squares of deviation of measurement points from nominal features. However, formulation with the Least-Squares Method is inaccurate for tolerance evaluation purpose. The resulting tolerance zone is not in conformance to the standard ASME Y14.5. Therefore, it results in the acceptance of out of tolerance parts and the rejection of parts that are within tolerance specifications.

Ge *et al.* (1992) developed a knowledge-based inspection planner with 5 modules for supporting CIDI (Computer Integrated Dimensional Inspection) and integrated with CATIA: 1) Inspection specification (*GD&T*) module, 2) Automatic Inspection Planning module, 3) CMM verification module, 4) CMM execution module, and 5) Comparative analysis module. In the CMM verification module they used the best fit nonlinear least square method to apply the tolerance verification.

ElMaraghy *et al.* (1990) formulated a minimum zone evaluation model and developed algorithms to evaluate and analyze cylinders inspection data from a CMM to verify tolerance requirements such as size, roundness, runout, cylindricity and straightness of a longitudinal surface element. The optimization process locates the center point or reference axis of the deviation zones. They used unconstrained nonlinear optimization objective function and the Hooke-Jeeve direct search method to adjust the position and orientation of the center of a circle or axis of a cylinder in order to achieve the minimum deviation zone.

Ikonomov *et al.* (1995 and 1997) introduced the virtual measuring gauge as a computerized replacement of a real gauge. They proposed a virtual gauge algorithm to evaluate the geometric relationship between feature and datum features. Geometrical constraints applied to the virtual gauge represent the implicit relationships between features and datum. They modified the Small Displacement Screw method, proposed by Bourdet (1988), with constraint in order to calculate the constraint substitute element for geometrical tolerances verification. The substitute element is calculated by minimizing the distance from the measured data set to the geometrical element after fitting.

Kim and Chang (1996) developed a prototype for the measurement planning system under consideration of geometric tolerances and statistical aspects. They developed 3 modules for the off-line measurement and inspection system; (1) data input module, (2) the measurement-planning module and (3) the statistical analysis module. The scope of the geometric tolerances was limited to position tolerance.

Dowling *et al.* (1997) presented some statistical issues related to tolerances and geometric features inspection using CMM. A variety of techniques have been developed which improve upon the Least-Squares Method, many of which provide the minimum tolerance zone result. However, these methods are mathematically complex and often computationally slow for cases where a large number of data points are to be evaluated.

Carr and Ferreira (1995a and 1995b) formulate the minimum zone problem as a non-linear optimization problem, which is subsequently solved using a sequence of linear programs, which converge to the non-linear optimal solution. They addressed only form tolerance; Straightness and Flatness (1995a), Cylindricity and Straightness of the median line (1995b).

Gou *et al.* (1998) developed a symmetric minimum zone algorithm to unify the formulation and evaluation of datum establishment and orientation tolerances through a geometric theory using orientation constraints. They formulated the problem as a constrained minimization problem. The non-differentiable minimization problem was converted into a differentiable minimization problem with an extended configuration space. This algorithm is simple and computes solutions, which are accurate and consistent with the ASME Y14.5 standard. Then, Gou *et al.* (1999) extended their work from just orientation tolerance to include form and profile tolerances.

Malyscheff *et al.* (2002) modified the support vector machine-learning algorithm, used in either classification or regression problems, in order to identify the minimum enclosing zone for straightness and flatness tolerances. A gradient ascent method is proposed identifying sequentially the solution to the non-convex optimization problem. They compared their results with those obtained by Carr and Ferreira (1995a)

Prakasvudhisarn *et al.* (2003) modified the support vector machine-learning algorithm to a support vector regression algorithm for fitting data to find the minimum zone straightness and flatness tolerances. They solved the resulting non-convex optimization problem sequentially using a gradient ascent. The support vector regression theoretically requires quite a computational time and memory, particularly when the size of data set is large (i.e. range data).

2.6 Registrations and Localization

Localization refers to the determination of positions and orientations of the Design Coordinate System (DCS) of a part with respect to the Measurement Coordinate System

(MCS). However, it is also referred to as registration of design surface with measurement surfaces in some literatures. Localization can be regarded as a two-step process: find the point–point corresponding relationship between measurement and design surfaces; and, solve the rigid body 3D coordinate transformation between these two surfaces to align them into a common coordinate system. The description of the 3D shapes or surfaces for localization is a very basic task. All subsequent operations are based on it. For most of the existing approaches, localization is an iterative process and the calculation of the distance between the digitized surface and the CAD model surface is required at each iteration. This operation which is essential and critical in the localization process is the main time consuming part of the localization process and there are two approaches to implement it:

Point-to-point distance calculation between closest corresponding points from the two surfaces.

Point-to-plane distance calculation method is faster but the problem of searching the plane or closest point from design model is more difficult with surface expressed in parametric form.

In previous decades, localization was achieved by presenting the part at a desired position and orientation, using special tools, fixtures or other part presentation/orientation devices totally dedicated for specific products. This kind of process is usually costly, and time and effort are required to design and manufacture new fixtures. In recent practice, localization has been carried out by mathematically aligning the DCS to the measuring coordinate system by using some initially measured data. This process allowed the use of low precision but general-purpose fixtures in flexible and small batch manufacturing. It has been formulated as the minimization of the sum of the squared distances between the measurement points and the design model with respect to the transformation parameters.

Traditionally, datum is measured to establish a reference frame for the part. This is known as **3–2–1 approach** as shown in Figure 2.2.

- Three (3) points are measured from the first datum to establish a plane.

- Two (2) points are measured from the second datum to establish a second plane perpendicular to the first.
- Finally, one (1) point is measured from the last datum perpendicular to the first two.

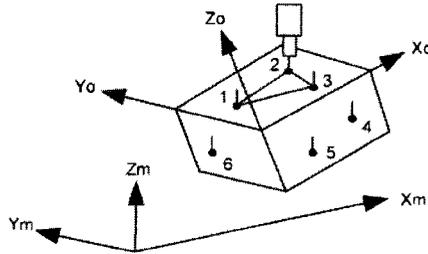


Figure (2.2) 3-2-1 Approach to locate DCS (Menq *et al.*, 1992)

The drawbacks of this approach are:

- The parts are required to have plane surfaces.
- The result is very sensitive to manufacturing errors on the datum and to errors in measurement.
- The solution depends on the selection of points (position and number).

Many authors developed more robust localization approaches to improve this approach for higher accuracy, efficiency and robustness. For example, Bispo and Fisher (1994) investigated localization or the matching of acquired free-form surface image data with the design model. The matching was based solely on the 3D points with an estimation of the pose alignment. ElMaraghy and Rolls (2001) considered the registration to obtain a complete set of measured points for a particular object. They investigated major uncertainty factors that were deemed responsible for discrepancy in registration results. The uncertainty of the sensor was found to be a function of the position that should be measured; hence obtaining a reduced uncertainty could be a goal that should be optimized by the best inspection plan. Fan and Tsai (2001) called the registration process, which was carried out based on human-computer interactions as initial localization between different patches. Then the detailed localization was solved based on the minimization of the objective function, which was the sum of the squared distance between the two surface patches to be studied.

Many approaches for localization are introduced in this chapter. The first approach (*Point approach*) uses points directly for deciding the correspondence by searching the closest point such as Iterative Closest Point ICP or Modified ICP. The second approach (*Feature approach*) differs from the above in selecting surface primitives or features and uses these items to create the corresponding relationship. This approach is based on tree search method or the constrained tree search method.

2.6.1 Point Approach Localization

The Iterative Closest Point ICP algorithm proposed by Besl and McKay (1992) is the main technique for point approach localization. It requires only a procedure to find the closest point on a geometric entity to a given point. It always converges monotonically to the nearest local minimum of a mean-square distance metric. The rate of convergence is rapid during the first few iterations. The algorithm can briefly be sketched as follows. Assuming a reasonable good initial registration given an adequate set of initial rotations and translations with a certain level of "shape complexity", the relative orientation between two datasets is iteratively refined by pairing a number of points on one surface with the closest points on the other surface. Hence, one can globally minimize the sum of squared distances between the point sets over all six degrees of freedom by testing each initial registration.

Delingette *et al.* (1997) applied the localization to the mesh representation based on an ICP approach. The correspondence and registration between the reconstructed surface mesh and the design mesh were done iteratively. The closest points were selected as the corresponding points and the best transformation was estimated based on several different distance criteria such as the median distance between the vertex of the design model and its closest point on the reconstructed model, maximum signed distance, and maximum and median distance at the edges and the corner vertices, for evaluating the shape similarity until a displacement threshold was reached. A set of parameters, relative to the processes from the 3D model digitization to reconstruction of a 3D mesh, were evaluated for their impact on reconstruction accuracy. A modification is done to take into

account the outliers; an algorithm was implemented to remove the vertices that were located too far away. They concluded that this methodology was well suited for the inspection of smoothly curved mechanical parts.

Ainsworth *et al.* (2000) discussed free-form surface inspection using Iterative Closest Point (ICP) for determining transformation. The probe was moved manually to find six (6) or more corresponding points to provide a good estimation of transformation for cases where the design model and measured part were initially grossly misaligned to generate a rough alignment. ICP was then applied to the subsequent registration.

Pottmann *et al.* (2001) also used a modified ICP method to make the surface inspections and comparisons by localizing 3D point clouds from laser scanning to its CAD model. The localization was an iterative process, which was very similar to the ICP process. Instead of moving a point from the measured 3D point cloud toward the possible corresponding point on the design model, they moved this point toward the tangent plane of the design surface at the corresponding point. They claimed that the modified approach converged much faster than the standard ICP approach. As indicated by the author, for low curvature surface regions, this difference on convergence was more obvious. In Pottmann *et al.* (2004), they proposed a different approach than the ICP. This approach relies on local quadratic approximation to the squared distance of the surface to which the point cloud should be registered. The authors also claimed that it leads to faster convergence than ICP.

Guehring (2001) treated two processes of registration and localization. Registration of multiple views of measurements was for surface reconstruction. Localization was to align the reconstructed surface to the design model, for comparison between those two surfaces. All these registrations were based on a modified ICP algorithm. The corresponding points were defined as the point pair that was close in both distance and normal directions based on distance and cosine angular thresholds. For solving the transformation, the rotation matrix was expressed in unit quaternion, and the

transformation was estimated based on the minimization of the covariance-weighted sum of least square of the differences between corresponding points.

Prieto *et al.* (2002) established the corresponding relationship using a modified ICP process, which select the corresponding point based on the evaluation of distance between potential corresponding points and the surface curvature values (Gaussian curvature and mean curvature) at these points. The transformation matrix was found by using quaternion representation. The distance of a point to a NURBS surface was computed as finding a point on the parametric surface such that the distance between the 3D measurement point and the point on the design surface was minimal in the perpendicular direction to the tangent plane at the point from the design surface.

The motion in the ICP is such that the points move in a least squares sense as close as possible. It works with local quadratic approximation to the squared distance, which is very good for points far away from the surface but not the best for points close to the surface.

2.6.2 Feature Approach Localization

The first step to apply localization based on feature is to recognize the features in the part and produce a list of corresponding primitives from measurement range image and design model. This is implemented using two methods; the tree search or the constrained tree search methods. The localization task is then implemented for applying the correspondence between the design model and the digitized model. This is to calculate the transformation parameters that align the two models together using least-square minimization or quaternion.

Faugeras and Hebert (1986) represented the surface in primitives and carried out localization by using the *tree search method*. The transformation was decided by using quaternion. In selecting the primitives for localization, the authors recommended that line primitives should not be parallel and planes should be independent. The localization mainly depended on the existence of planar regions in the object being matched.

Based on the method introduced by Faugeras and Hebert (1986), improvements were made in this approach by Marshal *et al.* (1991). The correspondence was determined by matching the segmented primitives from the design model and measured image data. In this research, the objects contained **planar**, **cylindrical** or **spherical** faces, but only planar faces were used for localization. The estimation of transformation was treated as a least-square minimization problem. If the estimation of the rotational elements of the transformation matrix was not a straightforward minimization, quaternion might be used. However, it has shortcoming such that only planar primitives were used for the localization. As a result, this requires a number of planar primitives on the object studied for the localization.

Brenner *et al.* (1998) used the *constrained tree search approach* to establish the correspondence between the design model and measurement data. The process started with one matching pair between the design model and the measurement image. Once each possible pair was identified, the search went to the next level. The search was a recursive process. In order to control the search time, constraints were used to bind the branching in the tree. Therefore, for each measurement feature, only a subset of design model features was selected as possible matches. During the search, the skipping of features was allowed; if no correspondence was found for a certain measurement feature, this feature was removed from current matching path and the search continued. Once the correspondence was established, the rigid body transformation between the measured data and the design model was estimated.

Unsalan and Ercil (1999) assumed that the alignment between the measurement object and the design model was done beforehand. The inspections in both 2D and 3D situations were studied. 3D data were represented by implicit polynomial surfaces. The inspection activity was then to model the template of the design model by an implicit polynomial. Edges of the image of the measurement object were extracted. Each edge point was tested if it was inside tolerance values.

2.6.3 Comparison between Measured Geometry and CAD model

After aligning the MCS with the DCS, the next step is to determine whether the part coordinates are within tolerance. The deviation, which is the distance between a point from the digitized data and the corresponding point on the design model, is then compared to the specified tolerance.

Patrikalakis and Bardis (1991) firstly selected the maximum among all minimum distances from the localized measurement surface points to the design model. Then, this maximum distance was used to verify if the measurement surface was within the pre-defined bounding surfaces of the tolerance region. If the measurement surface equation was known, the verification was reduced to the interference detection between the localized measurement surface and the bounding surfaces of the tolerance region.

Sahoo and Menq (1991) discussed two methods for distance calculation based on the complexity of the part and the type of surface representation. The first method was *Orthogonal Euclidian Distance*, which was suitable for surfaces represented in either parametric or implicit form. The second method was *Algebraic Distance*, which was suitable for surfaces represented in implicit form. According to the authors, this method worked well for surfaces of planar, quadric and lower order parametric polynomials. For higher order surfaces, this method became computationally expensive. Therefore, the Orthogonal Euclidian Distance method was recommended for higher order surfaces.

Pahk and Ahn (1996) evaluated the difference between the measurement points and the design model in such a way that the correspondence was decided by the closest point concept at first. Then, for every measurement point, the corresponding point on the design surface was calculated based on an iterative subdivision algorithm. Finally, the deviations were obtained.

Kase *et al.* (1999) divided the calculation of the difference between the measurement data and the design model into two categories: local evaluation and global evaluation. The local evaluation was the comparison between points based on the value of their

Extended Gaussian Curvature. A matching rate function was designed to evaluate the local errors. The global evaluation was to extract and evaluate the surface features such as a bend or twist. The relationship between the aggregate normal vectors of the surface features was decomposed into a bent angle and a twisted angle. The differences of bent and twisted angles constituted the global evaluation results.

Fan and Tsai (2001) studied the nearest distance between two sets of point clouds represented in B-spline surfaces. The nearest distance was calculated as the distance between the intersection of the normal to the surface from one of the two patches to another. A so-called direct method, which was a distance minimization process based on the Newton-Raphson method was used.

2.7 Summary and Conclusions

Tactile and laser technology have gained tremendous popularity in manufacturing systems, and their use for inspection is expected to grow more in the future. Manufacturers have expended much effort to produce hardware and software that obtain high-precision measurements. However, the quality of inspection decisions depends just as crucially on the efficiency of data acquisitions as well as the correctness and appropriateness of the data analysis, interpretation and subsequent decisions. Nevertheless, accuracy of inspection results can be affected by many factors (digitization tool, number of digitized points and the fitting technique used). On the other hand, digitization speed, accessibility and sampling are challenges that face the inspector.

A problem with the range sensors is apparent when we want to measure an inner surface, on which it is much more difficult to have a large number of points. That is due to occlusion problems or due to the high incidence angle between the beam and the surface or even due to the limited range of vision of the scanner. One way to overcome such challenge is to use a touch probe. Most internal occluded features can be scanned with special touch probe tips.

The accuracy of a touch probe is much higher than the accuracy of the laser scanner while the inaccuracy resulting from fitting surfaces to small number of sampled points gives advantage for laser sensors. Moreover, large number of points, which cause computation burden is usually reduced by filtering the point cloud. This causes another source of inaccuracy due to the dislocation of the original point in the point cloud to an averaged point in the resulted filtered point cloud. This trade off between the accuracy of touch probe with limited number of points and the inaccuracy of laser scanners with huge number of points urge inspection systems to include both measurement techniques.

From a tolerance perspective, the conformance to tolerance specification is still an unresolved issue in inspection using new tools. There is still a geometrical tolerance verification problem for the current new technology to conform to the tolerance requirements ASME Y14.5M-1994 (American Society of Mechanical Engineers), or ISO standards (ISO 1101-1983 Technical Drawing-Geometric Tolerancing). Nevertheless, definitions of form errors such as (straightness, flatness, circularity, and cylindricity) in these current standards assume perfect (continuous) measurements, not discrete measurements. The evaluation of form errors using CMM relies on discrete measurements. As a result it is not possible to assign statistical confidence level to the estimated form errors or to suggest the stochastically reliable minimum sample size (or the number of measuring points). Current procedures for inspecting geometric forms are not well developed, and there is much room for extension and improvement.

Most CMM verification algorithms are based on the least squares solution, which minimizes the sum of the squared errors, resulting in a possible overestimation of the form and geometrical tolerance. Therefore, although CMM algorithms successfully reject bad parts, they may also reject some good parts. The minimum zone envelope principle overcomes such problem. Many researchers developed and modified techniques for minimum zone tolerance verification.

In localization research, all the techniques in the literature were trying to align the digitized part coordinate system to the design coordinate system through iterative

processes and optimization techniques. All the approaches addressed in the literature were based on *point-to-point* or *point-to-plane matching methods*. Some of the researches were extended to implement the localization based on features to find initial correspondence; however the optimization criteria was based on least square minimization problem and the tolerance verification is performed in a later step. There is a need to include the tolerance verification in the localization process. To be able to complete such tolerance-based localization process, there is a need to develop a segmentation algorithm that keeps the original measured data in a feature belonging format.

Based on the reviewed literature it is clear that optimal inspection planning is still an open ended research area due to the fact that inspection tools are in a continuous advancement on both the technological front as well as the software front. Having said that, a new inspection planning method remains a research challenge and requirement as long as new technologies are introduced and new approaches are being developed to manipulate the data.

3. HYBRID INSPECTION PLANNING

As discussed in chapter one, the developed inspection system is composed of three stages. The first stage, which is the hybrid inspection planning and digitization, is presented in this chapter. This stage includes two sub-modules: knowledge-based sensor selection and inspection tasks sequencer modules. The knowledge rules formulation, the hybrid sequence modeling and optimization are detailed in this chapter. A case study of a water pump is presented to illustrate the developed model. This work is based on Mohib *et al.* (2008).

3.1 Inspection Planning – An overview

The effective planning and execution of the inspection process helps achieve both the time and quality objectives in the CAD-to-Part release processes. Planning has two distinguished levels: Macro- and Micro-level planning (ElMaraghy, 1993 and 2007). At the Macro-level, planning is concerned with identifying the main tasks and their best sequence. Micro-level planning details process parameters, required tools and setups, process time and resources. Macro-level planning is difficult because of its dependence on declarative process knowledge including part geometry, inspection tools, fixtures and technological requirements and also its implied time-dependency represented by the order in which the given features should be inspected. In order to increase the inspection efficiency and effectiveness, a feature-based planning system that utilizes the latest technology makes it possible to plan the combined use of laser scanning and tactile sensing for the geometric and dimensional inspection of complex mechanical work-pieces based on the CAD model and specifications. Laser scanners are usually used with free-form surfaces or large parts when large number of points is to be inspected. Contact sensors are mostly used with regular prismatic shapes due to the accuracy of the acquired data and to the fitting simplicity. Many *complex mechanical parts* include both types of features such as pumps, dies and engine blocks.

In inspection, a Complex Mechanical Part (CMP) can be described as a manufactured part, which includes functional prismatic shapes and free form shapes that

need to be inspected. Figure 3.1 shows a water pump housing that includes prismatic features such as cylindrical holes and cones and some free-form features such as the fluid guide surface, whose geometry has an important function in “Guiding and ensuring a non-turbulent fluid flow”. It is important to verify its shape, which requires the acquisition of an adequate number of points, and would be very time consuming if scanned using traditional touch probing on a sampling basis. First, an examination of this part shows the potential to inspect it using a laser scanner. However, for those features that are occluded for being out of the laser scanner field of view, a tactile probe is needed to acquire the missing points.

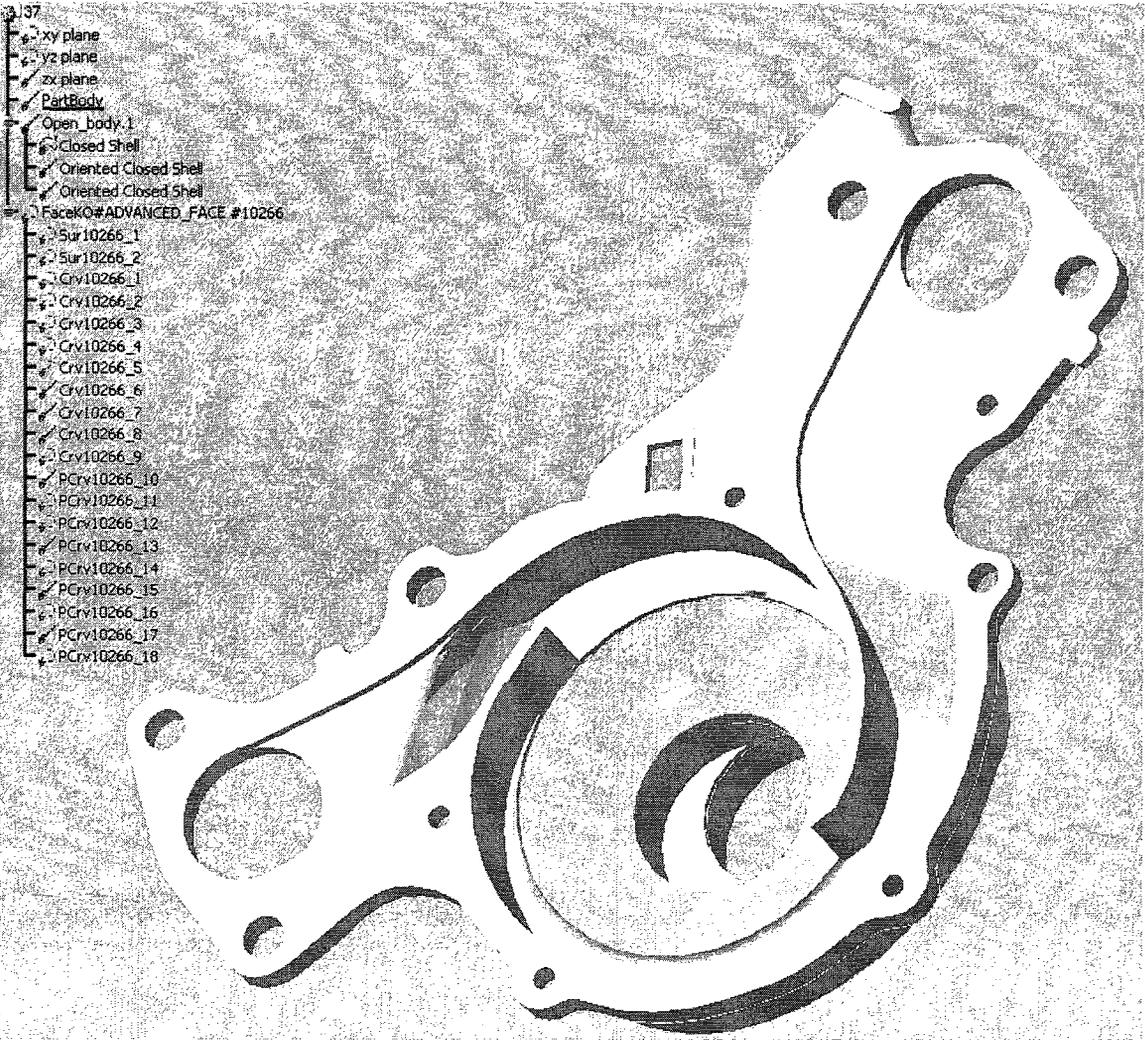


Figure (3.1) Water pump housing containing both prismatic and free form features.

3.2 Related work

Computer-aided inspection planning research for both contact and non-contact sensors has seen significant progress during the last few decades. In this section, different inspection planning techniques are reviewed as well as several attempts to combine different types of sensors.

3.2.1 Inspection planning techniques

Artificial intelligence and knowledge-based techniques such as Expert Systems, Neural Network and Fuzzy Rules were used to plan the inspection process. ElMaraghy and Gu (1987) developed the first expert system for CMM inspection planning. The expert rules were used for clustering and prioritizing features to be inspected. Moroni *et al.* (1998) developed an expert system to generate touch probe configurations and to select the most suitable probe by minimizing the changes of probe and part orientations. Chan and Gu (1993) developed an object-oriented knowledge-based inspection planner, however, the plan optimality was not considered. Lu *et al.* (1999) used an artificial Neural Network technique for multi-component inspection path management where genetic algorithms were applied to reduce the distance moved by the probe to obtain a collision-free path. Hwang *et al.* (2004) proposed a CMM inspection planner to arrange the inspection feature measurement sequence by minimizing the number of part setups and probe orientations using a greedy heuristic and continuous Hopfield Neural Network. Hwang *et al.* (2002) developed a knowledge-based inspection planning system using a hybrid Neuro-Fuzzy method with weight parameters optimized using Genetic Algorithms. Beg and Shunmugam (2002 and 2003) developed an object-oriented planner using Fuzzy Logic to select and sequence part and probe orientations for the inspection of prismatic parts. Ketan *et al.* (2002) developed a feature-based geometric reasoning approach for planning the inspection of prismatic parts. Cho *et al.* (2005) developed a series of heuristic rules by analyzing the features information such as the nested relation and the possible probe approach directions to inspect work pieces having many primitive features.

The use of knowledge-based systems has been successful in analyzing and prioritizing features inspection and other decisions such as generating probe configurations; however, they were combined in many cases with optimization methods, such as mathematical programming, non-traditional optimization, heuristics, etc. to optimize the generated plan.

3.2.2 Multi-sensor inspection

Many attempts have been made to integrate different types of sensors to increase the measurement accuracy. Huang and Zheng, (1996) integrated the computer vision method (photometric stereo approach) with laser displacement sensors to improve the efficiency and precision of the surface digitizing process. The proposed approach utilizes the high speed of the photometric stereo approach and the precision of the laser displacement sensors. Mital *et al.* (1998) proved using statistical analysis that hybrid inspection leads to superior inspection performance and shortens the time taken to reach an accept/reject decision. However, in this case the hybrid inspection system meant using CMM and manual inspection. In reverse engineering, Fang *et al.* (1998) tried to improve the accuracy of the built CAD model by integrating a stereovision with CMM. The results of the stereovision system were used to plan the use of CMM sampled points for better productivity and efficiency. Also, Shen *et al.* (2000) introduced an integrated multiple sensor system, where the developed 3D stationary active vision system was used to guide and control the touch probe for rapid coordinate data acquisition. The integration of a laser scanner with the proposed multiple sensor CMM was suggested. The objective of the multi-sensor integration method, proposed by Li and Liu (2003) was to determine and guide the touch probing points (sampling) from the B-Spline model assumed to be fitted to the rough data of a surface scanned using a vision system. Luo *et al.* (2004) combined the laser interferometer machine vision system with CMM by substituting the tactile probe by a vision camera, where the displacement of the Charge Coupled Device (CCD) camera was measured using the laser interferometer.

Bradley and Chan (2001) proposed a complementary dual sensor approach for reverse engineering applications, where the surface patches are to be scanned using a

laser sensor while the boundaries are digitized using touch probe. However, the edge boundary points are difficult to scan using a touch probe due to the errors caused by the direction of radius compensation.

Bichmann *et al.* (2004) and Haibin *et al.* (2006) integrated the touch probe with a conoscopic sensor. Systems based on conoscopic holography have better accuracy and cost compared with the triangulation technology. However, the use of triangulation technology is widely spread in industrial applications.

Table (3.1) Comparison of hybrid inspection approaches

References	Integrated Sensors	Objective	Approach
Huang and Zheng 1996	Vision + Laser Displacement	Digitization speed	Photometric stereo approach
Mital <i>et al.</i> 1998	Manual + CMM (TouchProbe)	Performance	CMM Programming
Fang <i>et al.</i> 1998	Stereo Vision+ CMM (TouchProbe)	Productivity and efficiency	Stereo Matching
Shen and Menq 2000	3D active Vision + CMM (TouchProbe)	Digitization speed	Feature Recognition
Bradley and Chan 2001	Laser + CMM (TouchProbe)	Boundary definition	Surface fitting
Li and Liu 2003	3D active Vision + CMM (TouchProbe)	Sampling	Simulated annealing
Luo <i>et al.</i> 2004	Laser interferometer + CMM (VisionCamera)	Calibration	Line fitting
Bichmann <i>et al.</i> 2004	CMM (Conoscopic holography + TouchProbe)	Planning	CAD-based (STEP-QDAS)
Haibin <i>et al.</i> 2006	Conoscopic holography + TouchProbe	Planning	CAD-based (STEP-QDAS)

Most of the attempts to integrate different types of sensors aimed to improve the accuracy of the measured point(s), not the overall accuracy of the inspection process and completeness of data (Table 3.1). Effective integration of the different tasks involved in CMM and laser scanner as an example of hybrid (Contact/Non-contact) inspection planning is a key issue in the development of a robust inspection planner. This problem has not been well addressed due to the lack of a formalized and integrated approach for CAD model analysis, resource allocation and measurement operations sequencing. This work presents a feature-based hybrid inspection planner that is capable of automatically determining the best method of measuring given features or parts of features, using the

most suitable type of sensor and ordering the hybrid inspection tasks to inspect complex mechanical parts, which include prismatic and Free-form features.

3.3 Proposed Hybrid Inspection Planning Model

The developed hybrid inspection planning system plans the digitization process of complex mechanical parts using both contact and non-contact sensors in a complementary manner to achieve complete and more accurate inspection results. The developed system consists of two modules: 1) a knowledge-based sensor selection and 2) an inspection task sequencer optimizer. As illustrated in Figure 3.2, inspection knowledge and algorithms, based on the analysis of the human conducted inspection process, are first applied where a new inspection-specific features taxonomy is built to guide the selection of the most suitable inspection method for each feature. Next, a clustering and sequencing module is developed to produce the inspection process plan, where a novel mathematical model based on the popular Traveling Salesperson Problem (TSP) is formulated. In effect, an enabling technology to realize hybrid inspection planning using both knowledge-based systems and optimization methods has been introduced. This enabler equips the planner on the shop floor with appropriate tools to make sound inspection decisions.

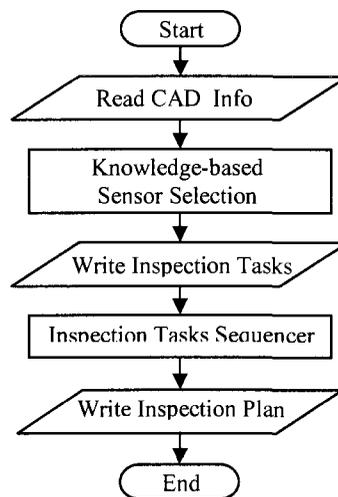


Figure (3.2) Proposed Hybrid Inspection Planning approach flowchart.

The first step of inspection planning is the interpretation of the blue prints/CAD models to gather the relevant design information and the Geometric Dimensioning and Tolerancing (GD&T) specifications. The CAD model may be presented in STEP (STandard for the Exchange of Product data) file format. The STEP file is an international standard for product information representation and exchange, which is used to construct a consistent, integrated information model of the product. However, only geometrical features are represented by the STEP at present where the tolerance information is not captured. No STEP based data format exists which allows conversion-free data exchange between CAD models and the available inspection software systems. To overcome such a problem, some authors have used QDAS (Qualitative Data Analysis Software), an industry standard for data sharing and consistency checking, that can be used to relate inspection data to geometrical features (Bichmann *et al.*, 2004 and Haibin *et al.*, 2006). In this work, the relevant geometric data are extracted from the STEP file and combined with the specified tolerance information, which expresses the designer's intent, into the inspection-specific features taxonomy described earlier. Automation of this input data gathering, interpretation and preparation is not the focus of the current work.

The inspection process planning tasks, using CMM equipped with contact/non-contact sensors, can be summarized in the following three steps: i) sensor selection; ii) sampling and ii) collision-free path generation. These tasks were considered in the literature for single sensor type inspection. We address, here, the same planning tasks for hybrid inspection that utilizes both contact probing and laser scanning. In the next sections, the model parameters, constraints and limitations will be described.

3.4 Knowledge-based Sensor Selection Modeling

The sensor selection module is a knowledge-based system where knowledge is captured in the form of a list of rules, which depends on the available type of sensor and the type of features. Three types of parameters are considered to model the inspection tasks: 1) parameters related to the sensor, 2) parameters related to the part features and 3)

variables related to both of them. The formulation of the knowledge-based sensor selection module is detailed below.

3.4.1 Model Parameters and Assumptions

3.4.1.1 Sensor related parameters - Physical description

The first parameter type, which depends on the physical description of the system, is related to the types of sensors available in the workshop. In the proposed system, a touch probe and a laser strip type sensor are used. The Coordinate Measuring Machine (CMM) used in this work is composed of the body structure, which can be regarded as a Cartesian robot with 3 degrees of freedom (x, y, z) whose end-effector is the probe tip as shown in Figure 3.3. A good analysis of the physical system helps to model the inspection system parameters; the physical description of the system is detailed below.

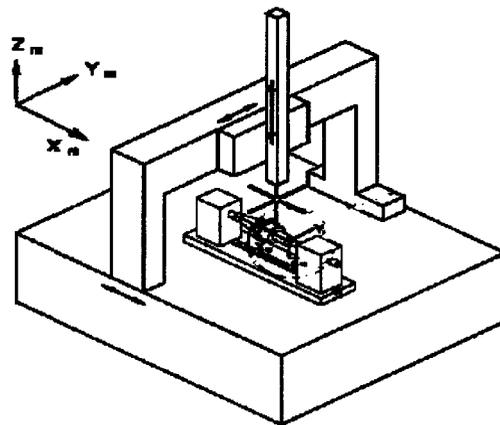


Figure (3.3) CMM Structure.

3.4.1.1.1 Probe Head PH10

The *probe head* that allows a particular spatial orientation of the probe axis is mounted on the end-effector of the CMM. If the probe head is motorized, it is possible to change the orientation during inspection. The used probe holder is the Probe Head Renishaw PH10, as shown in Figure 3.4, that can incline in an angle from vertically down to horizontal 105 degrees around the perpendicular axis (angle A) and rotate 360 degrees around its axis (angle B) designed to position probe tips in horizontal and vertical angular positions to have the most orientations required to measure the part. Both angles

directions have a limitation in steps of 7.5 degrees for a total of 720 discrete positions (15 x 48). These values are used to define the domain for selection of sensor orientation.

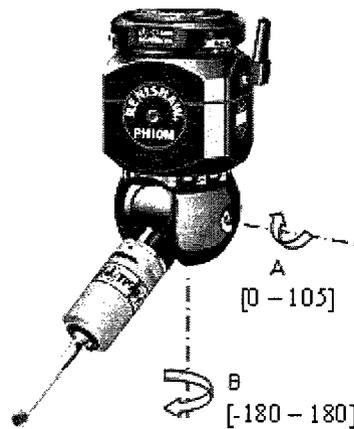


Figure (3.4) Probe Head Angles.

3.4.1.1.2 Touch probe sensor

Figures 3.4 also shows the *electronic touch triggering probes* the most common tools used with the CMM. These are triggering devices consisting of a probe and one or more styli with sensitive elements for indicating the location of the stylus within a chosen coordinate system. When the probe contacts a point on the part, with a very small amount of over travel, a signal is sent to the system. The *touch probe* is the fundamental part of the system. It is the device which signals to the CMM that a contact has been made between the stylus and the work-piece surface by means of the opening of an electric circuit or the deflection of a piezo-electrical material. The touch probe is connected to the probe head through a TP2 Stylus. The *Ruby* is the end tip of the touch probe. It is an extremely hard material, so wear of stylus balls is minimized. It is also a low-density material and its use reduces tip mass and false triggers due to machine motion or vibration.

A *probe extension* can be used to connect the probe head to the probe, expanding the depth of probing, particularly inside large and complicated work-pieces. It is shaped as a multi-diameter cylinder and it is made of steel or aluminum. Also, *Stylus extension bars* provide added probing penetration in deep bores, but they can reduce accuracy due to the loss of rigidity.

3.4.1.1.3 Laser scanner sensor

An alternative sensor to be considered is the strip type laser scanner METRIS LC50, which can be mounted on the Renishaw probe head PH10 as shown in Figure 3.5. The laser scanner consists of a laser beam projector (Emitter) and a CCD camera (Receptor) that detects the reflected laser beam. The laser scanner measures a part by one laser stripe at a time. The laser probe performs the scan by projecting laser stripes along the path, as shown in Figure 3.12, each one of which consists of hundreds of points. The laser beam projector use a moving pinpoint of light using an electronically controlled mirror to illuminate the part. Since a laser beam diverges with distance, scanners generally have a working range from roughly 2.5 cm. to 30 cm. The field of view of the used laser scanner LC50 is a 50x50 mm.² and the Stand off distance is 70 mm. as shown by the specifications in Figure 3.6.

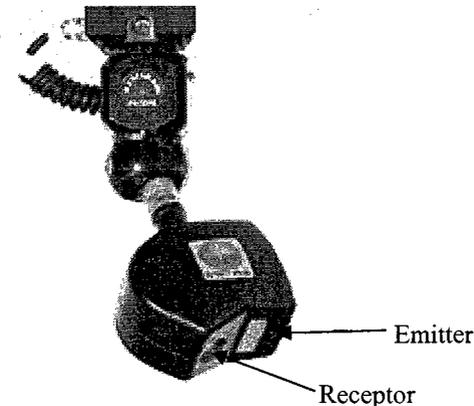


Figure (3.5) METRIS Laser Scanner LC50.

Specifications:	
Weight	290 g.
Dimensions	110x160x68 mm.
Scan Speed	19,200 pts/s
Width of view	50 mm.
Depth of view	50 mm.
Accuracy	15 μ m. (1 σ sphere fit)
Stand off distance	70 mm.
Laser	Class 2

Figure (3.6) METRIS Laser Scanner LC50 Specifications.

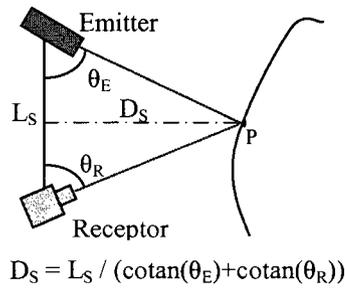


Figure (3.7) Laser Scanner Triangulation Method.

Figure 3.7 explains the theory of optical triangulation for the laser sensor. The detected laser beam is stored as intensity information for each pixel and by going through image processing and triangulation of this information, a coordinate value is assigned for each measured point in a three-dimensional space. Figure 3.8 details how the laser reflection of the laser beam and the part are interpreted by the CCD camera in the receptor.

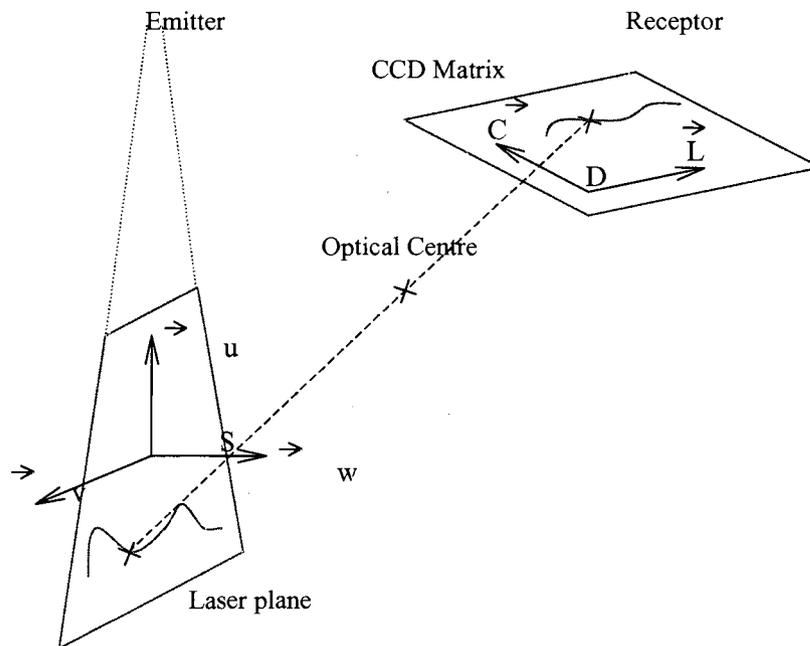


Figure (3.8) Laser Strip Digitization Technology.

3.4.1.1.4 Pros and Cons of Touch / Laser sensors

The two types of sensors described above are considered to be selected to perform inspection tasks. Each type has its pros and cons. The advantages of touch probes are that they are accurate and are also efficient for checking dimensions and tolerances of a well-defined part. Nevertheless, touch probes can reach deep and hidden surfaces. However, the touch probes have some shortcomings:

- They are inherently slow in acquiring point data since they need to make physical contact with a part surface for every point that is sampled.
- They can also deform a part surface if the part is made of soft material. For different work-piece materials with different hardness, the probe triggering force must be adjusted for improved performance. Generally, lower forces are used for soft materials and higher forces for hard materials.
- The bending forces applied to the stylus due to the over travel, which causes lobbing effects.
- Difficult in measuring parts with freeform surfaces.
- Require lengthy planning.

Although the touch probe is more accurate in terms of point accuracy, the laser scanner is preferable to use in some cases due to its ability to collect huge amount of data in much less time. However, the laser scanners have some shortcomings:

- The angle between the incident laser beam and the surface normal at a point being measured should be less than limit angle.
- The measured point should be located within the length of a laser stripe.
- The measured point should be within a specified range of distance from the laser source.
- The incident beam as well as the reflected beam should not interfere with the part itself.
- The laser probe has to be collision-free with the part.
- Surface roughness and reflectance and the ambient illumination influence the accuracy of scanning results.

- The surface measured should be painted with MAGNAFLUX if it is not white colored. Hence the accuracy of measurement is reduced.

A comparison between the advantages (capabilities) and disadvantages (limitations) of tactile sensors and laser scanners is shown in Table 3.2. The improvement and the qualification of the measurement device are not considered here; we mainly address a hybrid system to overcome the shortcomings in both tactile and non-contact measurements. The efficient and accurate planning and mixing of the enormous amount of points captured by the laser scanner with the less dense points digitized with a touch probe is the focus of this research.

Table (3.2) Advantages and Disadvantages of Laser sensors and touch probes

	Touch Probes	Laser Sensors
Advantages	High point accuracy Large range of measurement Can reach most deep and invisible surfaces	Fast Can scan many points in one path Ideal for soft materials and sheet metals
Disadvantages	Slow (time consuming) Inadequate amount of data Soft materials Complex surfaces Requires lengthy planning Requires fixtures	Visibility and accessibility Cannot reach deep and hidden surfaces Results affected by part material (Reflection) Less accurate (camera pixels and resolution)

3.4.1.1.5 Sensor related parameters

Now that the physical characteristics of the system are described, we can define the parameters related to the sensors. First, j is a number that represents the sensor type; for the system in hand a tactile and a strip-type laser sensors are used. Therefore, there are two possible values for j :

$$j = \begin{cases} 1 & \text{Laser} \\ 2 & \text{Touch} \end{cases} \quad (3.1)$$

The probe head orientation “Angles A and B” is considered as a sensor related parameter. It is used to define the angle between the surface orientation and the sensor.

For the system in hand “PH10M”, the orientation is defined by two angles A_j and B_j , which are ranged as follows:

$$A_j \in [0^\circ - 105^\circ] \quad (3.2)$$

$$B_j \in [-180^\circ - 180^\circ] \quad (3.3)$$

The angles A_j and B_j depend on the probe head type, while the values of other parameters such as resolution (Res_j), accuracy (Acc_j) and repeatability (Rep_j) are considered as predefined parameters that belong to the type of sensor selected.

3.4.1.2 Part related parameters - Features Taxonomy

The second type of variables is related to the features of the inspected part. Features are items of interest for an application; they can be numerical such as dimensions or structural such as strings and graphs. The term ‘feature’ in inspection can be defined as the individual measurable properties of the feature being examined. Choosing and discriminating an independent feature is the key to the success of any classification algorithm. Each feature in an inspected part is characterized and described as follows: 1) geometric features, 2) manufacturing features, and 3) inspection features.

3.4.1.2.1 Geometric Features

A geometric feature can be defined by the smallest recognizable canonical or primitive shape, which cannot be further decomposed otherwise it will reduce to meaningless geometric entities such as lines, points and surfaces. Figure 3.9 shows the most common geometric features that are found in typical mechanical parts. Some manufacturing features, such as holes and bosses, can take a cylindrical, rectangular or even spherical shape.

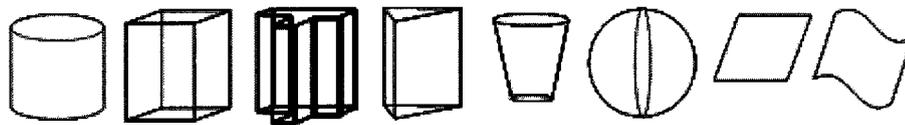


Figure (3.9) Typical Geometric Features.

3.4.1.2.2 Manufacturing Features

Manufacturing features are recognizable shapes where the association between geometry and function can be identified (ElMaraghy H. and ElMaraghy, W., 1994). According to the definition of the STEP AP 224, which is a neutral standard format to allow computer systems to exchange information with each other without using proprietary translation filters (Newman *et al.*, 2007), manufacturing features consists of 16 different categories listed in Figure 3.10. Regular shaped features (also called prismatic features), such as a cylinder in the form of a boss or a hole, a pocket or a slot are introduced as the key elements for associating specific functional meaning to groups of geometric elements (faces, edges and vertices), thus offering the advantage of treating sets of elements as unique entities. Figure 3.11 illustrates the difference in shape for similar types of manufacturing features.

-Boss	-Protrusion	-Planar face	-General outside profile
-Pocket	-Rounded end	-Revolved feature	-Marking
-Hole	-Outer round	-Spherical cap	-Knurl
-Slot	-Step	-Thread	-General volume remove

Figure (3.10) Sixteen types of STEP AP224 defined Manufacturing Features.

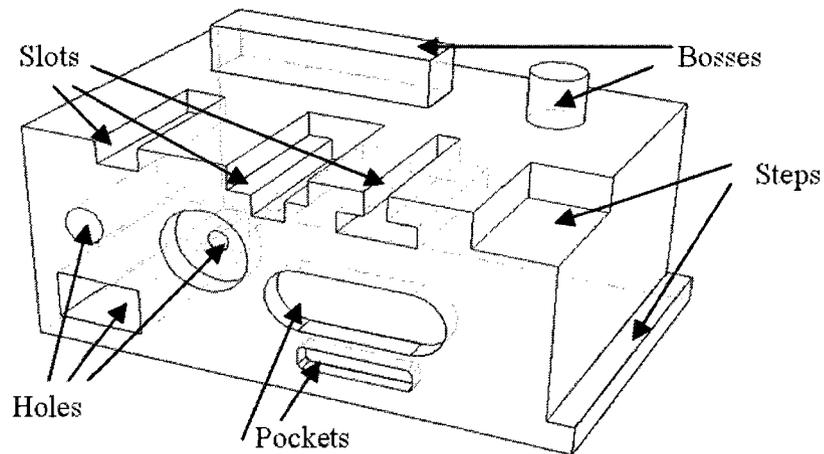


Figure (3.11) Examples of Manufacturing Features.

In addition to geometric and manufacturing features information, the inspection information associated with a feature must be considered. The inspection procedure associated with two similar manufacturing features, but with different geometries, can be

totally different and hence, inspection features have to be defined. For example, a boss with a cylindrical cross section could be inspected for roundness, cylindricity, coaxiality, run-out etc. while a boss with a rectangular cross section would be inspected for flatness, straightness, parallelism, etc. The different inspection features that could be used with the stated geometric and manufacturing features will be introduced in the next section.

3.4.1.2.3 Inspection Features

The third type of feature description in a manufactured part is called the inspection features, which are expressed in terms of the tolerances associated with the manufacturing or geometric features based on the functional requirements as specified by the designer. These tolerance specifications are expressed in the form of a value associated with a symbol on the mechanical drawings. Based on the ISO and ASME Y14.5 standards, the different types of tolerances are classified and illustrated in Figure 3.12. The inspection decision is based on this classification of inspection features.

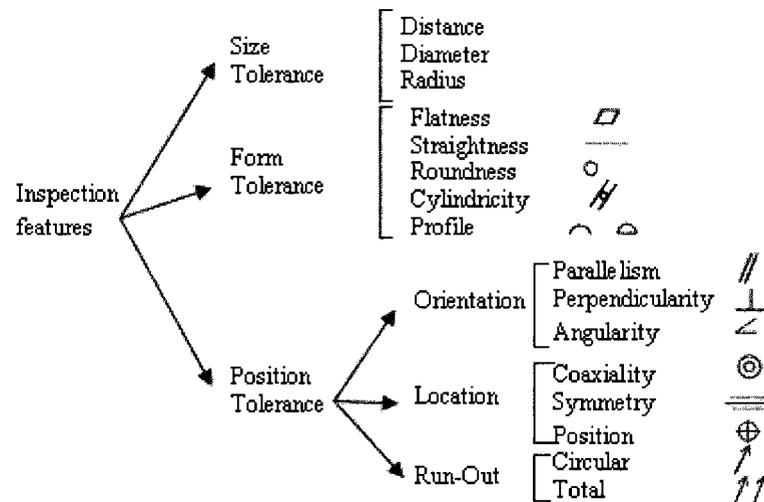


Figure (3.12) Geometric and Dimensional Tolerances Classification.

The geometric, manufacturing and inspection features are related to each other from design throughout manufacturing and inspection processes and cannot be separated. Several attempts to build a feature classification that defines the relationships between the different types of features are found in Beg and Shunmugam (2002) and Yoon *et al.* (2004). However, those classifications established the relationship between different types of features without defining how to use these relationships.

In this research, an inspection-specific features taxonomy is proposed, which embodies knowledge rules and sufficient data for utilizing these relationships and guiding the sensor selection step of the inspection planning process.

3.4.1.2.4 Inspection-specific Features Taxonomy

Usually, a complete CAD model includes all the above-mentioned feature types and related information. None of them alone is sufficient to produce an inspection plan. The proposed inspection-specific features taxonomy is built based on the practical relationships between those types of features in the form of a three-dimensional matrix, which happens to be sparse as shown in Figure 3.13. Each cell, if it exists, includes a set of inspection rules to determine the best sensor to be used to measure the considered feature. The sensor selection depends on three elements: 1) the manufactured feature to be inspected, 2) its shape or its geometric feature and 3) the inspection feature or the specified tolerance to be verified. These elements are grouped in one cell (Cell(inspection, manufacturing, geometric))

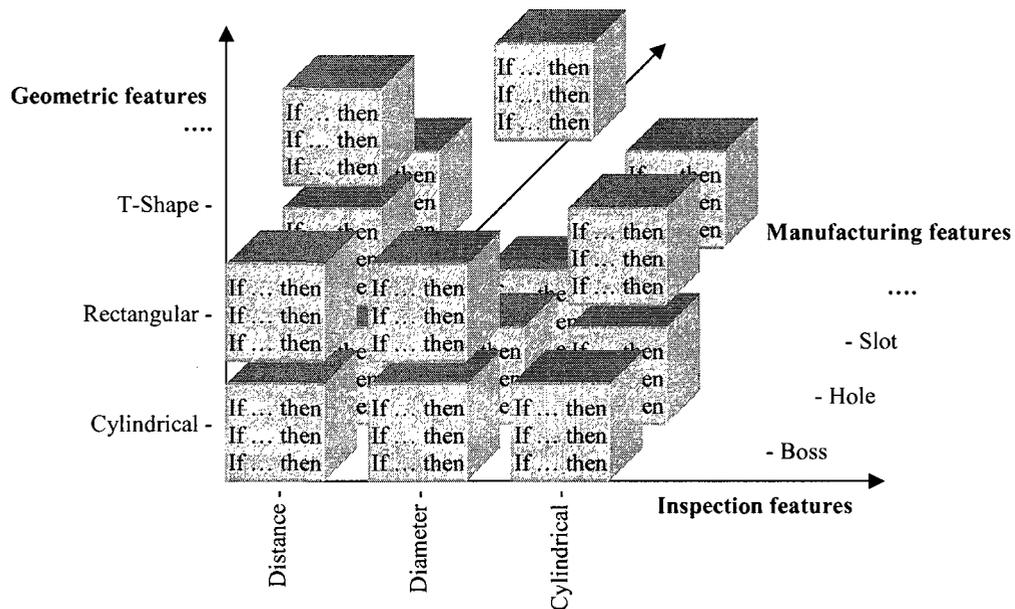


Figure (3.13) Knowledge rules 3D decision Matrix.

An analysis of the inspection process conducted by human operators has been carried out and some inspection procedures have been identified to populate each cell with a list of knowledge rules and adjustable parameters that enable the inspection of the specified feature with available inspection sensors. Section 3.4.2 and 3.4.3 further detail how the proposed taxonomy is used to perform sensor selection where laser and tactile sensing are considered.

3.4.1.2.5 Part-Related Parameters

The inspected part includes a number of features (NF) to be inspected. Each feature is described by the main variables and their associated parameters, such as the feature ID i , the number of repetition of this feature (n_i), Inspection Feature (IF_i) and datum feature ID (DF_i) if it exists, the type of Manufacturing Feature (MF_i), the Geometric Feature (GF_i) and the corresponding geometry parameters such as the Length/Width ratio for a plan, the Diameter/Depth ratio for a cylinder, or the Larger/Smaller-Diameter/Depth ratio in a cone. Other description variables are the Feature Orientation (FO_i), Occlusion (O_i) and the reason behind occlusion. Those are the input feature parameters related to the inspected part as exemplified in Table 3.5.

3.4.2 Sensor Selection Decision Variables

The decision variables of the sensor selection sub-module are the ones relating the inspected feature i with the sensor j to define an inspection operation. They can be represented by a list of inspection operations $\{Op_k\}$ where each operation is defined by a vector $[j_i, A_{ij}, B_{ij}, \alpha_{ij}]$ that includes: j_i , the sensor j to be used to inspect feature i ; A_{ij} and B_{ij} , which are the probe head angles used to inspect feature i using sensor j ; and for a certain part orientation, α_{ij} is the average angle between the probe head orientation and the normal direction to the surface inspected from feature i using sensor j (Figure 3.14). Since the best orientation to digitize a surface is its normal direction, the smaller α_{ij} is, the better is its orientation.

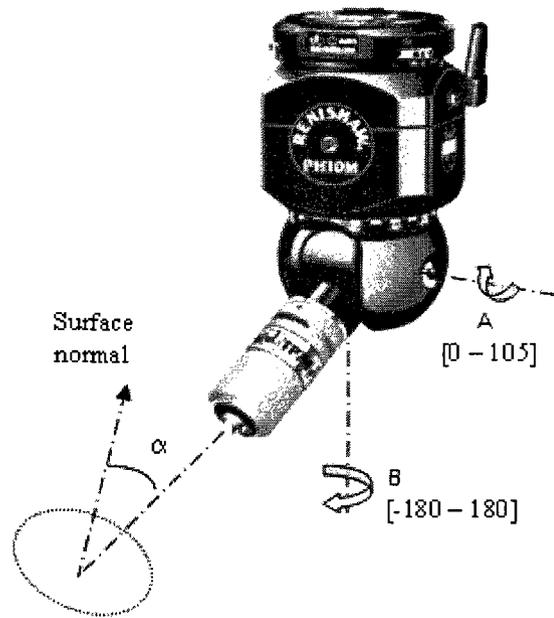


Figure (3.14) Angle α between the probe head orientation and surface normal direction.

The angle α_{ij} depends on the angles A_{ij} and B_{ij} as well as on the part orientation P_{ij} , where P_{ij} is a number that represents the part orientation to inspect feature i with sensor j . This number depends on the part shape and its stability on the table/fixture. All possible orientations are stored in a list from which the sensor selection module selects an orientation and calculates α_{ij} . Since angles A_{ij} and B_{ij} already exist in the decision variable vector, the part orientation P_{ij} is considered as the fourth decision variable to define the inspection operation Op_k . It should not be noted that the part orientation can be common for several features. A key point $(x,y,z)_{ij}$ is also needed to be known to start the inspection operation. This point can be extracted from a sampling procedure as detailed in section 3.4.4.

3.4.3 Knowledge Rules Formulation (Constraints and Limitations)

Traditional inspection methods such as *Jackscrew*, *wobble-plate*, *fixed-plane* or *precision spindle* methods and their setups are chosen based on the part's physical shape (i.e. cylindrical or not) and its size. Moreover, for cylindrical parts, the inspection tool is different for internal (holes) and external (shafts) cylinders (Griffith, 2002). In a complex part with many intricate and interacting features, it is not easy to isolate and inspect each feature according to the traditional methods. Higher technology tools such as CMM and

laser scanner are, hence, used to measure the complex parts to inspect their features. There is no one method that guides the selection and usage of such tools. The knowledge-based sensor selection module starts by analyzing and allocating the part features in the knowledge rules 3D decision matrix (Figure 3.13) to apply the corresponding set of rules. Currently, some endeavors are taking place to robustly and fully automate the process of feature recognition and extraction from different STEP file formats generated from different commercial CAD/CAM systems. Meanwhile, user interaction is still needed to achieve this step.

```
Rule: Cell_Allocation  
IF Feature(i) | GF(i) = cylinder  
           and MF(i) = hole  
           and IF(i) = cylindricity  
THEN  
Feature(i) ∈ Cell(4,2,1)  
END
```

Figure (3.15) Knowledge rules – cell allocation.

Once the feature parameters are identified, the inspected feature is matched with the corresponding description, and positioned in the developed taxonomy as shown in Figure 3.15. Once the feature is located in its appropriate cell, a list of rules is generated. Each cell contains a set of knowledge rules to determine the most suitable probe. These rules are listed based on three main factors (categories):

1. Tolerance specifications
2. Occlusion and accessibility
3. Feature dimensions

3.4.3.1 Tolerance Specifications

An important general rule in sensor selection is that “*It is required to ensure that the measuring instrument’s discrimination is no more than 10% of the total tolerance of the dimension being measured*”. For example, a feature that has a tolerance of 0.01mm. (or ± 0.005) should be measured with an instrument that discriminates to at most 0.001 mm

(Griffith, 2002). Hence, the first constraint to be satisfied when selecting the sensor is the resolution of the sensor. The resolution of the touch probe and the laser sensor depends also on the controller and the software. Both laser and tactile sensors can discriminate to the micro level. An example of tolerance specifications rules can be expressed as shown in Figure 3.16.

```
Rule: Tolerance_Value_Verification  
IF Feature(i) ∈ Cell(12,2,1)  
           and Tol_Val(Feature(i)) < 0.099 mm  
THEN  
           Digitizing(Feature(i)) = Touch  
END
```

Figure (3.16) Knowledge rules – Tolerance Value Verification.

The decision to select a particular sensor type does not only depend on the number of points required to model the feature geometry but also on how many points are needed to verify that the required feature is within its specified tolerance. The primary datum (for a surface) is the plane that passes through the three (or more) highest points on that surface. Using traditional techniques, this can be achieved by applying a planar surface to the datum. Hence, for inspection features that needs a datum, the more points the better the inspection results. Another cardinal rule in geometric tolerances inspection using traditional methods is the fact that “*datums must be fully contacted while measuring the feature and be able to reproduce the specific tolerance zone of the part*”. Hence, one of the limiting rules is “*if a feature is selected to be measured using a sensor, the datum should be measured with the same sensor and it is preferable to be in the same setup to reduce the source of errors and inaccuracy*”. An example of datum verification rules can be expressed as shown in Figure 3.17.

Since the feature characteristics are cell dependent, the rules sets are different for various cells. For example, these rules vary with tolerance types; they depend on the type of tolerance specification associated with the manufactured feature and its value. Consider flatness versus cylindricity; a larger number of acquired points improve the

flatness estimation of the plane surface to be inspected, however, for example to calculate the cylindricity of an engine block cylinder with 9 cm diameter and 11 cm. length, 16 points would be sufficient, which can be acquired using a touch probe. In some cells, such as flatness related features, the default sensor is the laser scanner unless some conditions are satisfied. In other cells such as cylindricity, the touch probe is preferable even though the tolerance requirements may not be tight. Such a rule can be stated as: *A touch probe with a single orientation would reduce the time of changing the laser scanner orientation.* In general, the type of tolerance specification affects the selection of inspection tool; in the next sub-sections, we detail some rules formulated for size and form tolerance types.

```

Rule: Datum_Verification
IF DF(Feature(i)) ≠ 0
    and Digitizing(DF(Feature(i))) = Laser
THEN
    Digitizing(Feature(i)) = Laser
END
IF DF(Feature(i)) ≠ 0
    and Digitizing(DF(Feature(i))) = Touch
THEN
    Digitizing(Feature(i)) = Touch
END

```

Figure (3.17) Knowledge rules – Datum Verification.

3.4.3.1.1 Inspecting Size Tolerance

For non-cylindrical features, size dimensions are generated by two surfaces. Therefore, the MMC boundary of the perfect form is defined by two imaginary planes a MMC distance apart. These two imaginary planes could be simulated by two precise boundaries a MMC distance apart. The traditional techniques to inspect such feature is to use two precision parallels for outside thickness measurements or an adjustable parallel for inside measurements such as a slot with rectangular shape or even T-shape. In the developed approach, the outside thickness can be easily measured using a laser sensor

that can digitize the maximum number of points on each surface to be able to simulate the positioning of the gages on these surfaces, but for the inside measurements the occlusion should be determined first before selecting the sensor. *If the feature is defined as occluded then the tactile probe is selected for both measured and reference features.*

For cylindrical features, the MMC boundary of perfect form is an imaginary cylinder at MMC size. Two types of Manufacturing Features with cylindrical shape are considered; shafts (or external) and holes (or internal). Ring gages and plug gages are used to verify the MMC boundary of perfect form for cylindrical size tolerances since these gages are true cylinders within gage tolerances. The ring/plug gage should encompass the entire shaft at one time or the MMC boundary will not be verified. The LMC is checked locally in various places using a micrometer for the shaft and various hand-held gages for holes such as dial bore gages, telescoping gages and small hole gages. When using a laser sensor or even a contact probe, the problem becomes the fitting of the smallest and the largest cylinder that capture the point cloud in between. If the cylinder is external (shaft), the laser sensor can obtain more points to better fit a cylinder, hence the laser sensor is chosen. In case of a cylindrical hole the diameter to depth ratio should satisfy the non-occlusion condition as described earlier to be able to scan the inside wall of the feature. Otherwise, the tactile probe is selected.

The two shapes of tolerance zone for size tolerance depend on the measured feature. The size tolerance can be applied to all shapes of Geometric Features GF and all Manufacturing Features MF. Hence, *all the cells in the decision matrix that corresponds to Inspection Feature IF = "Size Tolerance" are enabled.*

3.4.3.1.2 Inspecting Flatness

Flatness is a form control that applies to a single continuous surface. Although flatness is never specified to a datum feature, it has an intrinsic datum called an optimum plane. If the feature is required to be inspected for flatness, the DF_i value is equal zero. Flatness cannot be gauged or evaluated in a go/no-go manner; it must be measured. It is understood that when a feature control frame indicates a flatness requirement on a surface

of the part, the measurement must be made with respect to the full indicator movement (FIM) found on an indicator with respect to the optimum plane. The traditional technique to inspect such feature is to use a surface plate, dial indicator with proper discrimination, a set of three jackscrews (or leveling screws, or Indian pins) and a mount for the indicator (height or surface gage). The more measured points, the more accurate is the inspection results. This process is very lengthy and time consuming using a touch trigger probe. *Hence inspecting the flatness of a surface using laser scanner is preferable than using a touch probe or the traditional methods.*

There are three conditions that should be satisfied to be able to use the laser sensor; 1- First, we should be able to place the surface in the field of view of the scanner, 2- The scanner head can be oriented perpendicularly to the surface to measure all the points of the surface or at least within 60° from the surface normal; and 3- The surface should not be shiny and this can be overcome by spraying MAGNAFLUX. *If any of these conditions cannot be satisfied the touch probe is selected and the contact probe conditions are checked.*

The tolerance zone for flatness requirement is defined as two imaginary parallel planes that are the tolerance value apart. Flatness is applied neither to cylindrical or spherical features nor free-form features. *Hence, the corresponding Geometric Features to flatness are rectangular, T-shape, triangular or simply a plane (i.e. z = 2, 3, 4 or 7) and the corresponding manufacturing features are the corresponding values to (y = 1, 2, 3, 4, 5, 8 or 9).*

3.4.3.1.3 Inspecting Straightness

Straightness is also a form tolerance that is not related to a datum. Three possible forms of straightness tolerances: straightness of surface elements, straightness of an axis, and straightness of a center plane. Figure 3.18 shows the different callout for different forms and Figure 3.19 shows the differences in functionality. Straightness of surface elements automatically controls the straightness of an axis. Straightness of surface elements can be applied to cylindrical or non-cylindrical features. The tolerance zone for

straightness of surface elements of cylindrical or non-cylindrical features is the zone between two imaginary and perfectly parallel lines that are the tolerance value apart. Jackscrew and Precision Straightedge Methods are used to inspect both cylindrical and non-cylindrical part while Optical Comparator and Two-equal-block methods are used to inspect only cylindrical part.

The straightness of an axis applies only to cylindrical features while straightness of center plane applies to non-cylindrical features. Those are the only form tolerances that can be gauged when applied at MMC. Differential Measurement and Precision Spindle methods are traditionally used when RFS modifier exist on the callout or understood based on the third tolerance rule which states that if the MMC modifier is not stated in the feature control frame, the RFS is directly understood. There are not many applications for axial straightness control because the tolerance applies only to the axis itself and it creates a virtual condition beyond size limits. Straightness of an axis is applied where the size of a feature is independent from the straightness of that feature such as mating features where there are plenty of clearance and long-size features with limited interface.

The tolerance zone for straightness of an axis of cylindrical features is an imaginary cylinder of tolerance value diameter within which the axis of the controlled feature must lie while for straightness of a center plane of a non-cylindrical part is two parallel planes the tolerance value apart in which the center plane of the part must lie. *To collect data for surface elements type of tolerance, it is necessary to use a touch probe or a point laser sensor* since it is hard to extract a straight line measurement from the point cloud obtained using a strip type laser sensor such as Metris LC50. However, *for straightness of an axis or a center plane, unless occlusion exists, it is better to use laser sensor* to obtain more points to generate the imaginary center line. Occlusion is checked in the next rule category. *The corresponding Geometric Features to straightness are all shapes except spherical and free-form features (i.e. $z = 1, 2, 3, 4, 5$ or 7) and the corresponding manufacturing features are the values corresponding to $y = 1$ to 10 .*

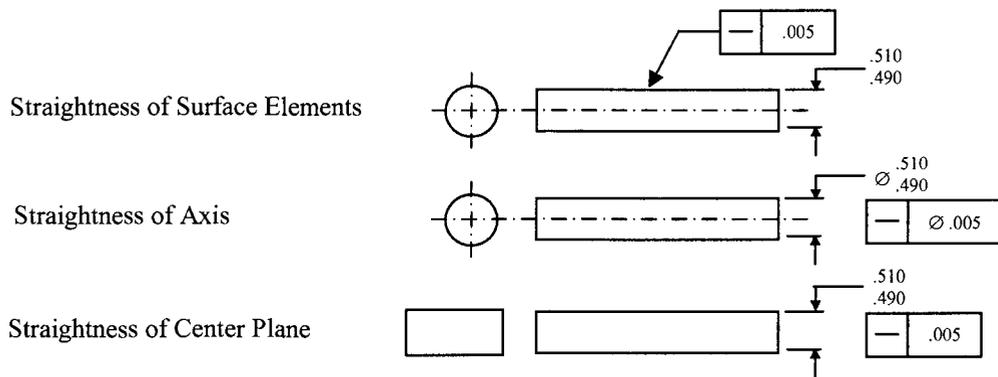


Figure (3.18) Possible Form of Straightness.

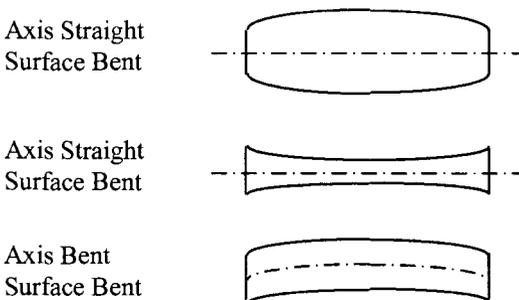


Figure (3.19) Examples of difference between axis and surface element straightness.

3.4.3.1.4 Inspecting Circularity (Roundness)

Circularity is also a form tolerance that is not specified to a datum. It applies only to cylindrical or spherical features and often features that have short axes (such as gaskets, washers and short sleeves) to control the circularity at each circular element and to control the effective feature size. It should also be noted that the tolerance zone is radial not diametric. Standard two point measuring instruments, such as micrometers, calipers, indicating snap gages and other similar instruments falls short to measure the circularity due to the problem of lobes (such as two lobes as oval shape or tri lobe ...). The best traditional instruments to measure circularity are precision spindles (which are very similar to touch probe on a CMM). Since two-point measuring instruments are not recommended due to lobe effect, open setup techniques are used to measure circularity. The outside circularity is traditionally measured by the V-Block method or using a V-Anvil Micrometer to ensure three point contact, while the inside circularity is measured by the Bore or Pneumatic Gage method. Circularity tolerances can be also verified using runout controls with respect to bench centers. A rotary table (rotab) can also be used to

inspect circularity of inside or outside diameters since three points of contact can be established by the rotab jaws.

The tolerance zone for circularity is two concentric circles the difference between their radii is the stated tolerance. It can be concluded that not many points are needed to measure circularity; however they need to be at certain position and at the same height which is hard to guarantee using a strip type laser sensor. *Hence, a few point digitized using a touch probe are better to measure circularity. The corresponding Geometric Features to straightness are all circular shapes such as cylinders, cones and spheres (i.e. $z = 1, 5$ or 6) and the corresponding manufacturing features are the corresponding values to ($y = 1, 3, 5, 6, 7, 10$ or 11).*

3.4.3.1.5 Inspecting Cylindricity

Cylindricity applies only to cylindrical features not associated with a datum reference. Since it includes circularity of all circular elements and straightness of all surface elements, combined into one control, it is considered the most complex form tolerance of all. It is also the most difficult to inspect and most time consuming with traditional sophisticated equipment. The previously described open setups for measuring surface element straightness and circularity at several circular elements and taper per side are usually used to estimate cylindricity. Moreover, total runout with respect to bench centers method can be used to verify if cylindricity is within specifications but only the acceptance decision can be made using this method. The best traditional method for measuring cylindricity is a precision spindle that is also equipped with a precision vertical slide. The tolerance zone for cylindricity requirement is the volume between two concentric cylinders that are apart by the amount of tolerance specified on their radius. *The more the number of measured points the better the cylindricity is verified.* This type of tolerance can be measured using both laser and tactile sensors. However, the use of laser sensors is not preferred with cylindrical features since many changes in sensor orientations are needed. Many researchers addressed the minimum number of measured points using CMM to verify cylindricity. However, in this research, for maximum number of digitized point, *laser scanners are recommended to measure cylindricity*

unless occlusion exists. Section 3.4.3.3 details how occlusion calculation is performed for a cylindrical hole.

3.4.3.1.6 Inspecting Profile Tolerance

There are two types of profile tolerances: Line Profile and Surface Profile. Profile tolerances are often used to control irregular shapes. If datums are not specified for a profile tolerance, the tolerance provides control over shape only (i.e. form). When datums are used, the profile tolerance collectively controls the shape of the feature and the size and/or location of the feature in one tolerance zone. Profile tolerance zone takes the shape of the basic profile (defined by basic dimensions) and are bilateral around that profile unless otherwise specified. Since all profile measurements must be made with the probe (or indicator) at 90° from the tangent line of the surface, a limited number of methods and equipment can be used to measure profile tolerances such as optical comparators (with appropriate overlays), CMM, limit gages, and profile gage designs (or hard tooling). Hence, *the best tool to inspect the line profile tolerance is the touch probe, while the surface profile tolerance is better to be inspected using laser scanner unless the feature or part of the feature is occluded. In such case, the need for hybrid (Laser and tactile sensors) is obvious.*

3.4.3.2 Accessibility / Shadows and Occlusions

Two general constraints that belong to both types of sensors are the accessibility for touch probes and the visibility for laser sensors. Once the touch probe is selected, the accessibility analysis can be performed (Limaiem and ElMaraghy, 1999). The laser scanner visibility of the feature depends on two types of problems: shadows and occlusions. The second set of rules identifies if the feature is external or internal and checks for occlusion problems. Fast occlusion detection was addressed by Qian and Harding (2003) by partitioning positional and normal space.

In this work, the type and shape of the whole feature is considered and the occlusion calculation mainly depends on the geometric feature and its dimensions. In order to have a successful scan, the incident beam from the emitter has to reach the surface; and the

reflected beam by the surface has to be detected by the receptor. The laser scanner can detect a point within the limit of the view angle. The view angle limit is defined as the maximum angle between the axis of the probe and the surface normal of a point. The laser scanner also has a fixed standoff distance and depth of view. The standoff distance is the desired distance that needs to be maintained between the probe and the part surface during scanning, and the depth of view represents the range of allowable deviation from the standoff distance in order for a part surface to be scanned. The laser stripe that touches the part surface also has a predetermined length. The probe travels along the scan path by generating laser strips, each one of which consists of hundreds of points.

The shadow problem means that the laser beam cannot reach the area that the CCD camera can see while the occlusion problem means that the laser beam reaches the surface but the CCD camera cannot see it. This is not because it is out of the field of view of the camera, which is a square 50x50 mm.², but due to the existence of an obstacle between the receptor and the intersection of the laser beam with the part. The shadows and occlusions problems are illustrated in Figure 3.20. Occlusion can be determined directly by the shape type; consider the example of the two T-Shaped slots as shown in Figure 3.11; the rules can be expressed as in Figure 3.21; or it can be calculated by knowing some shape parameters such as feature dimensions.

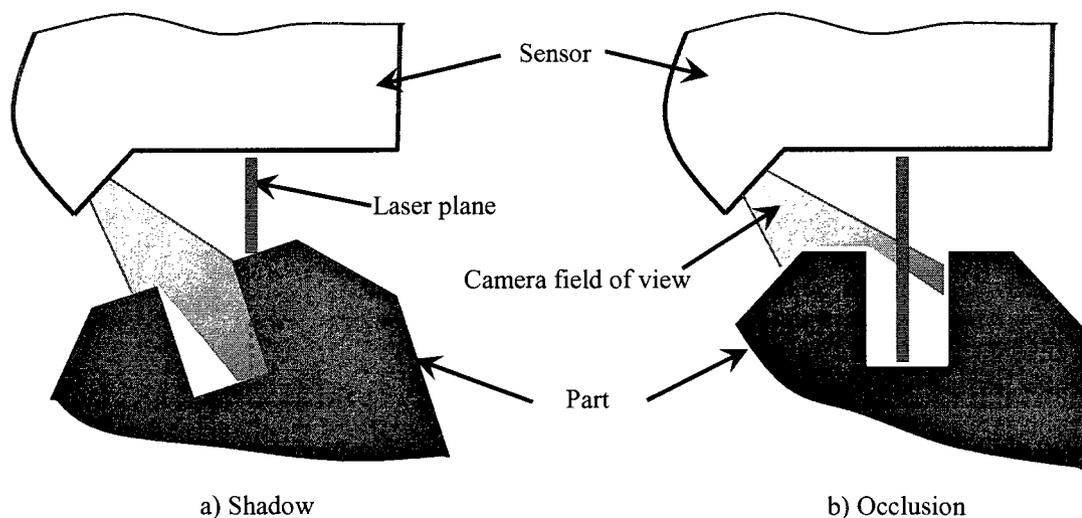


Figure (3.20) Laser Scanner – Visibility.

```

Rule: Occlusion_Test
IF Feature(i) ∈ Cell(3,3,3)
    and External/Open type
THEN
    Digitizing(Feature(i)) = Laser
END
IF Feature(i) ∈ Cell(3,3,3)
    and Internal/Closed type
THEN
    Digitizing(Feature(i)) = Touch
END

```

Figure (3.21) Knowledge rules – Occlusion.

3.4.3.3 Feature Dimensions

The third rule set category addresses the feature dimensions criterion and its parameters, such as shape, size and the dimensions ratios. Consider the example of the cylindrical hole shown in Figure 3.22(a), the intersection line between the laser plane and the part is composed of three segments; two segments on the top surface and the third segment is a curve on the surface of the cylinder. Consider a point on the curve segment inside the hole, the angle between the incident line from the emitter to this point and a line parallel to the axis of the cylinder at this point is called β_E , and the angle between the incident line from the receptor to this point and a line parallel to the axis of the cylinder at this point is called β_R . It is required to calculate the maximum depth in the cylindrical hole that the receptor can scan (i.e. D_h/d_h ratio) where D_h is the hole diameter and d_h is the depth of the curve from the top surface, as shown in Figure 3.22(c).

The minimum Diameter/depth ratio of a cylindrical hole that the laser scanner can digitize is calculated using the following formula:

$$D_h/d_h = \tan(\beta_R), \quad (3.4)$$

where

$$\beta_R = (\beta_E + (90 - \theta_E) + (90 - \theta_R)). \quad (3.5)$$

The angles (θ_E, θ_R) are the incident angles from the emitter and the receptor respectively. The Standoff Distance (D_S) is calculated using the equation

$$D_S = L_S / (\cotan(\theta_E) + \cotan(\theta_R)), \quad (3.6)$$

where L_S is the distance between the receptor and the emitter.

Consider the distance D_S to be the standoff distance prescribed by the laser scanner producer, in this case $D_S = 70$ mm. and $L_S = 50$ mm. Hence, the ratio can be easily calculated. If the measured feature is small, several laser scanner orientations are needed; in this case a touch probe would be a better choice. An example of knowledge rules for a narrow and deep hole can be as shown in Figure 3.23.

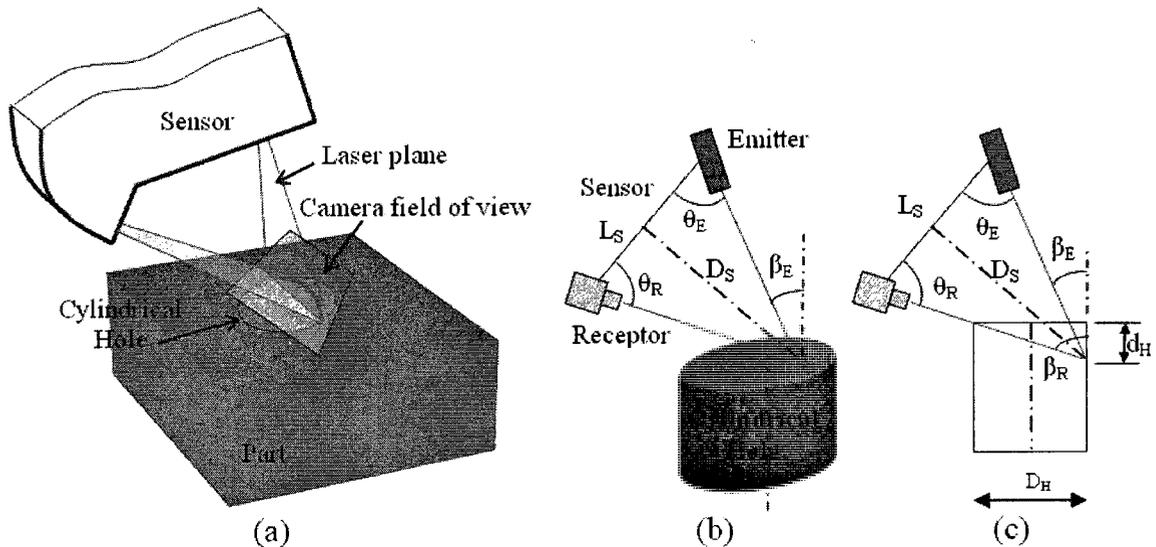


Figure (3.22) Cylindrical hole occlusion calculation.

```

Rule: Dimensions_Test
IF Feature(i) ∈ Cell(3,2,1)
    and Diameter(i) < A
    and Length/Diameter ration > B
THEN
    Digitizing(Feature(i)) = Touch
END

```

Figure (3.23) Knowledge rules – Feature Dimensions.

3.4.4 Sampling

At this point, the type of sensor and its recommended probe head orientation to inspect feature 'i' have been selected for a given part orientation. It is now required to order these inspection tasks. This research is concerned with macro-level planning and sequencing, however some micro-level planning details such as selecting a key point for each feature is required. A key point $(x,y,z)_{ij}$ of a feature i is the Cartesian coordinate of a point, where laser scanner or tactile sensor j starts the digitization process for this feature. This point can be obtained from a sampling process or it can be generated randomly. Sampling is normally used to determine the representative point set to be measured for each feature. The required number and distribution of measurement points depend on the type of sensor used, the size and type of the geometric and manufacturing features as well as the specified geometric tolerance. If the laser sensor is selected to implement the inspection of a feature, in order to simplify the surface information, the target surface is sampled to define the necessary information (ElMaraghy and Yang, 2003). The laser scanning parameters are then determined for each feature. The laser probe performs the scanning by projecting laser stripes along the path. The scanner parameters settings consist of three types of distances to be adjusted: point interval (d_1), strip length (d_2), and overlap distance (d_3). Figure 3.24 illustrates the parameters needed to adjust the sample size when planning a scan with a strip type laser sensor. Changing a scanning direction in this case requires reorientation of the part by adjusting setup fixtures, such as a rotary table or by using a PH10 indexing probe. The density and distribution of the sampled

points must provide good representation of the surfaces to be inspected. If the required digitization tool is selected as a touch probe, then the required number and the distribution of points depends on the type of geometrical and manufacturing feature, as well as the type of geometrical tolerance specified. Several strategies were developed to determine the best sample size and distribution of measured points for given surfaces and geometries (ElKott *et al.*, 2002). For example, to check the straightness of a planar face, a 3x2 set of inspection points are required, if the face is cylindrical then a 6x2 set of points are required (Table 3.3) (Beg and Shunmugam, 2002).

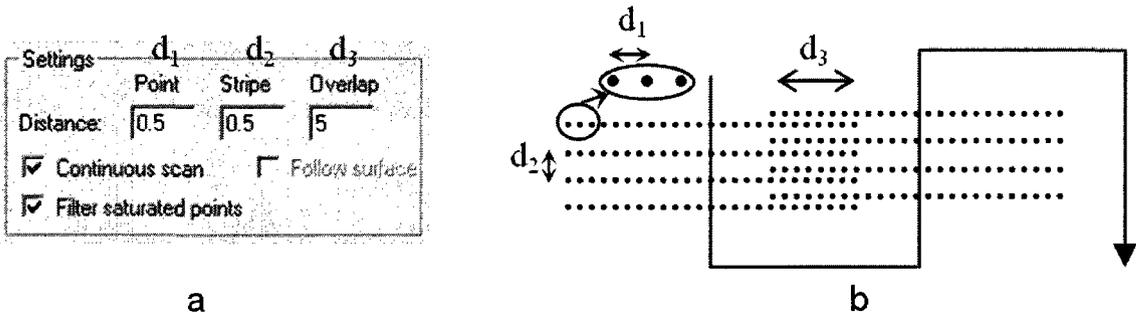


Figure (3.24) (a) Laser Scanner Parameters; (b) Scan Path showing distance parameters.

Table (3.3) Number of points considered for tactile probe (Beg and Shunmugam, 2002)

Inspection Feature	—	▭	○	∕	∩	⊙	<	//	⊥
Planar faces	3x2	3x2			3x2		3x2	4x2	4x2
Cylindrical faces	6x2		6x2	3x5		6x2		6x2	

3.4.5 Knowledge-based Sensor Selection Output

The output from the knowledge-based sensor selector module is a list of digitization operations $\{Op_k\}$. Each operation Op_k is dedicated to digitize feature i for all $i = 1, \dots, NF$ (Table 3.4). j is the sensor used in this operation where j_j equals 1 if the laser scanner is used and equals 2 if the touch probe is used. P_{ij} is a number that represents the part orientation to digitize the feature i using sensor j . A_{ij} , B_{ij} are the probe head angles to digitize feature i using sensor j . $(x,y,z)_{ij}$ is the key point to start the inspection operation. After the type of digitization sensor and its orientation are determined for each feature for a part orientation, the features sharing the common orientations have to be clustered and sequenced (ordered) to minimize changes.

Table (3.4) Knowledge-based Sensor Selection Output

Operation #	Feature	Sensor	Part Orientation	Prob Orientation	Key Point
Op_k	i	J_i	P_{ij}	(A_{ij}, B_{ij})	$(x,y,z)_{ij}$

3.5 Ordering of Inspection Tasks - Modeling and Optimization

The second part of the planning module addresses the *orientation clustering* and *sequencing* of the output set of digitization operations $\{Op_k\}$ from the sensor selector module that needs to be ordered.

3.5.1 Problem Formulation

The objective now is to order a global set of inspection tasks for a given part and sensors in order to minimize the non-digitization effort spent between consecutive operations. This effort can be measured by the time taken to perform these non-digitization tasks composed mainly of: 1) sensor changes, 2) work piece re-positioning, 3) probe orientation changes and finally 4) the total probe rapid traverse time. These four components are modeled as described next.

3.5.1.1 Part Orientation Changes

The first criterion is the minimization of the part orientation changes between consecutive operations:

$$\min C_1 = \Delta(P_{ij})_{mn} \quad (3.7)$$

where $\Delta(P_{ij})_{mn}$ is the effort associated with changing part orientation in order to switch between operations m and n . Part orientation and fixturing changes are usually accompanied by a registration process, where it is needed to measure common points for both orientations. The effort required for part orientation and fixturing changes is expressed in terms of the time taken to perform these tasks. An average of 150 seconds is what it approximately takes to perform this process.

3.5.1.2 Sensor Changes

The second criterion is the minimization of the sensor changes between two consecutive operations:

$$\min C_2 = \Delta(j_i)_{mn} \quad (3.8)$$

where $\Delta(j_i)_{mn}$ is the effort associated with changing sensors in order to switch between operations m and n . Calibration, homing the machine, and registration are necessary tasks required in order to perform sensor changes. Moreover, it is needed to measure common points using both sensors. The effort needed to perform the tasks associated with sensor changes is expressed in this work in terms of the time taken to perform them. The time required to perform sensor changes is 300 seconds on average.

3.5.1.3 Probe Head Orientation Changes

The third criterion is the minimization of the changes of the probe head orientation between two consecutive operations:

$$\min C_3 = \Delta(A_{ij}, B_{ij})_{mn} \quad (3.9)$$

where $\Delta(A_{ij}, B_{ij})_{mn}$ is the effort associated with rotating the probe head, (i.e. to change probe head angles A_{ij}, B_{ij}), in order to switch between operations m and n . The effort exerted to change the probe head orientation during inspection is not as high as the previous two objective function criteria since the calibration needed for each orientation is performed when the sensor is mounted onto the probe head. In other words, the calibration effort needed is already included in the sensor changes objective function component. However, to change orientation, the probe head needs to be moved to a dummy point away from the inspected part and then rotated. A good estimate of the average time taken to perform this operation is 10 seconds.

3.5.1.4 Time taken to travel between successive operations

The fourth criterion is the minimization of the time taken by the probe head to travel between two successive operations:

$$\min C_4 = d((x,y,z)_m, (x,y,z)_n) / V \quad (3.10)$$

where $d((x,y,z)_m, (x,y,z)_n)$ is the distance traveled between features' key points to switch between operations m and n . V is the rapid traverse speed of the CMM head

provided by the CMM vendor. It is used to compute the time taken by the probe to rapidly traverse between successive operations. For the DEA MISTRAL CMM used in this work, the translational rapid traverse speed is 41 m/min.

The four criteria of the objective function are not contradictory and can all be expressed in units of time. Hence, the non-digitization effort to switch between operations m and n can be calculated by incorporating the four objective function components into one overall time objective function:

$$C_{mn} = \sum_{l=1}^4 C_l \quad (3.11)$$

The operations Op_k can be modeled, as shown in Figure 3.25, as nodes $\{m, n, l, \dots, q\}$ and the links between these nodes are C_{mn} , where operation n follows operation m in the sequence of digitization tasks. Since the effort to switch from operation m to operation n is equal to that of switching from n to m , the cost matrix $C = [C_{mn}]$ is symmetric. The process of generating the cost matrix has been automated using a MATLAB script.

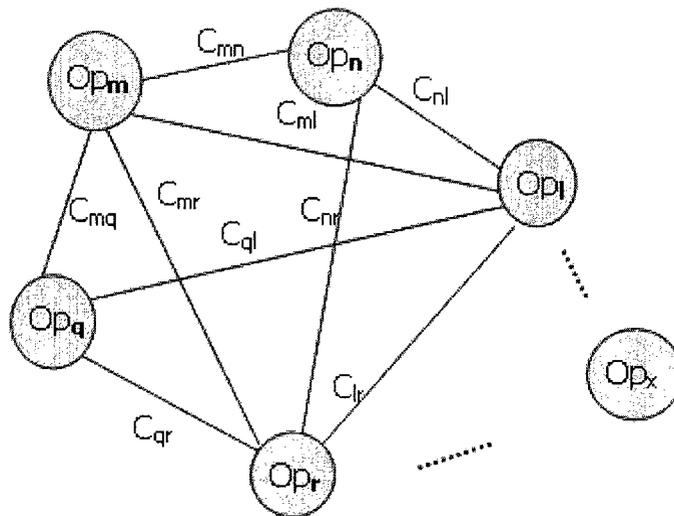


Figure (3.25) Inspection Operations Modeled as Nodes in a Traveling Salesperson Problem (TSP) Model.

3.5.2 Mathematical Model

The problem of ordering K inspection tasks (digitization operations) to verify that the part at hand meets the specified design requirements, is formulated in this work as a Traveling Salesperson Problem (TSP), where each inspection task is modeled as a city that has to be visited once and only once by a salesperson. The objective is to find the optimal tour that would minimize the total distance traveled by the salesperson such that the optimal solution obtained contains no sub-tours. In this case, the total travel to be minimized is that of the inspection tool such that all the inspection tasks would be performed with a minimum total transient time between each two consecutive tasks. The time objective function, as mentioned earlier, is composed mainly of four different components: part orientation changes, sensor changes, probe orientation changes and rapid tool traverse by the inspection tool. Two feasibility constraints have been formulated to ensure that only one route is going into a city (inspection operation Op_k) and only one going out of it. As well as a sub-tour elimination constraint.

Let X_{mn} be 0-1 integer variable, where both indices m and n runs from 1 to K . The value of the decision variable is 1 if the route between digitization operation nodes n and m is taken in the obtained solution tour; otherwise it is zero. The objective is to minimize the time between successive operations, where no digitization task takes place. The objective function is mathematically expressed as follows:

$$\min C = \sum_{m=1}^K \sum_{n=1}^K C_{mn} X_{mn} \quad (3.12)$$

S.T.

$$\sum_{m=1}^K X_{mn} = 1 \quad \forall n \in \{1, 2, \dots, K\} \quad (3.13)$$

$$\sum_{n=1}^K X_{mn} = 1 \quad \forall m \in \{1, 2, \dots, K\} \quad (3.14)$$

$$\sum_{m,n \in S} X_{mn} \leq |\{S\}| - 1 \quad (3.15)$$

$$\forall \text{proper inspection operations sets } \{S\}: 2 \leq |\{S\}| \leq K - 1$$

Equations 3.13 and 3.14 are constraints to ensure the feasibility of the solution tour obtained. Constraint 1 ensures that only one route is going into each node of the obtained solution tour. Constraint 2 ensures that only one route is going out of each node. Equation 3.15 is a constraint to eliminate sub-tour solutions. A sub-tour is a tour that would include only a subset of the nodes to be visited. If sub-tour prevention is not incorporated in a TSP model, sub-tour solutions would very easily result by simply returning a discontinuous solution composed of a number of sub-tours. Each sub-tour represents a separate clustered island of nodes (inspection tasks) with a relatively very low non-digitization time between its nodes. Hence, the optimal solution obtained in the absence of the sub-tour elimination constraint would be an infeasible solution composed of a collection of a number of these separate islands, where the expensive routes between the different islands, due mainly to part orientation and sensor changes cost components, are not included. This is obviously un-desirable as it does not constitute a successive sequence of contiguous tasks.

The sub-tour elimination constraint expressed by Equation 3.15 guarantees that for every subset S of nodes $\{m, n, l, \dots, q\}$ of length $|\{m, n, l, \dots, q\}|$, such that this length is less than or equal $K-1$ and greater than or equal 2, the number of routes connecting these nodes is less than or equal to $|\{m, n, l, \dots, q\}| - 1$. Therefore, no sub-tours could ever be constructed unless it is a complete tour of length n . Finally, it is important to note that the obtained optimal solution is a tour that has no start and no end. Accordingly, the corresponding optimal inspection sequence is obtained by subtracting the route with the highest cost off the tour as shown in Figure 3.26 and hence, establish a start and end of the plan and obtain the inspection path (i.e. sequence of nodes/ inspection operations), to be followed.

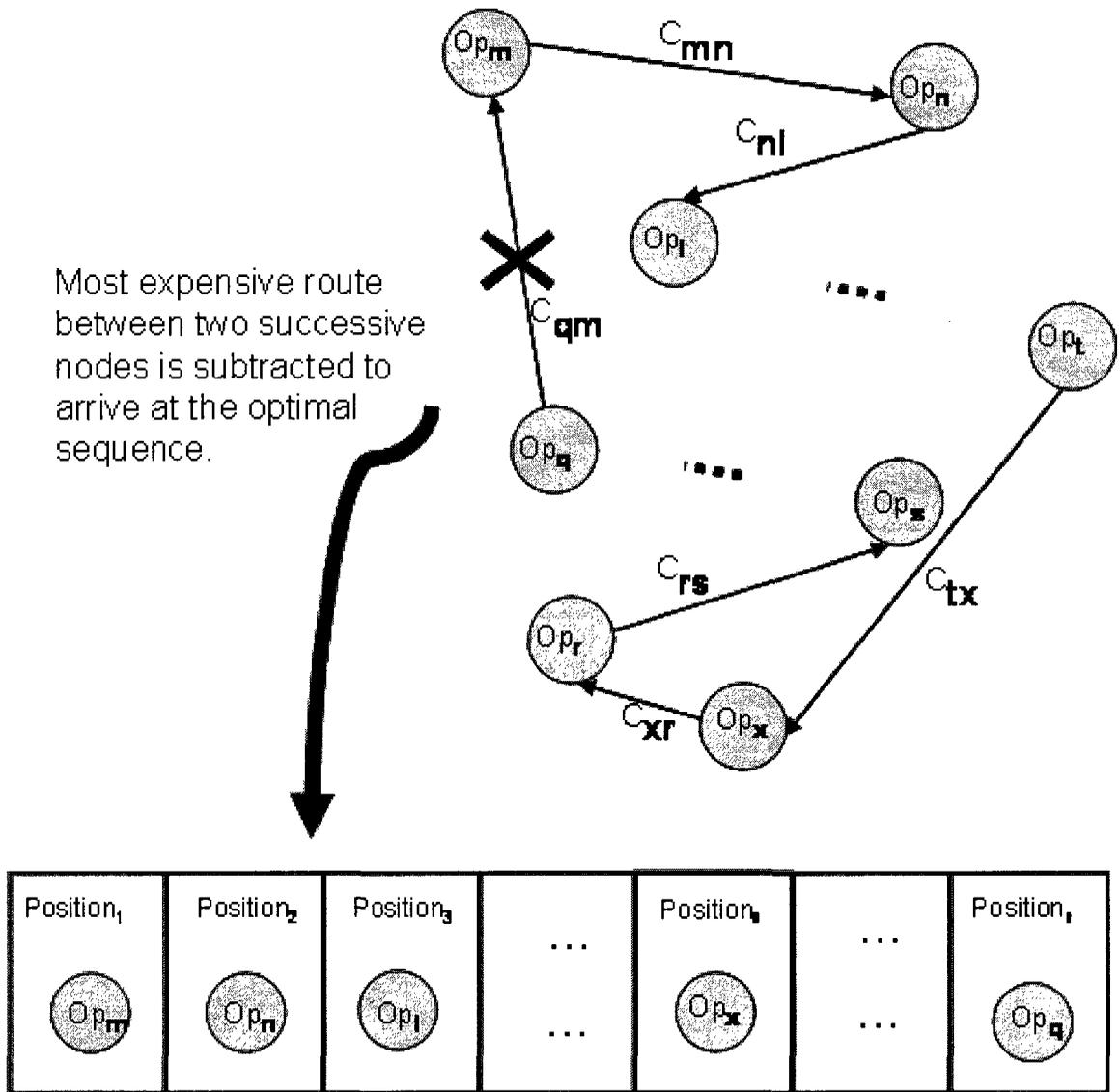


Figure (3.26) Obtaining the required sequence from the solution TSP tour.

3.6 Application – A Water Pump Housing Inspection Case Study

Consider a product to be manufactured using a die as in sheet metal forming or die casting applications. The die is designed based on the part specifications, and then it is produced and inspected. This is an important step in the manufacturing process and is resource intensive. The first lot produced using the die is fully inspected to initially qualify both the product and the manufacturing process. The size of the inspected lot depends on the number of features in the part. If the inspection results are acceptable, the factory proceeds to produce the part. Otherwise, the design of the die is modified. This process is very expensive and time consuming and in some industrial applications, it could take up to a whole year or more. Furthermore, the job may still be lost if the accuracy requirements cannot be met.

An example of such products is the water pump housing shown in Figure 3.1. This product is a complex mechanical part, which includes functional prismatic shapes and free form shapes that need to be inspected (Figure 3.27). The pump housing includes three datums: A- Gasket surface that is assigned a flatness tolerance, B- the cylinder at the bottom and C- the reversed conic hole. It also includes two free form features assigned a profile tolerance with respect to datum A, B and C, five tapped through holes assigned a position tolerance as well as the cone and a stepped hole at the inside diameter of the cone, while the rest of the holes are un-machined clearance holes with loose diameter dimensions. By analyzing the CAD model, the input parameters to the knowledge-based sensor selection module are extracted and presented in Table 3.5. Occlusion was found in three main areas: a) the two free form slides, b) the adjacent walls due to an insufficient incident angle of the laser beam and c) the stepped hole because it is out of the field of view of the CCD camera. These surfaces cannot be fully scanned with the laser scanner.

Table (3.5) Part features information

Feature Name	ID	Shape (GF)	Type (MF)	Tolerance (IF)	Datum	Number
Datum A "Gasket Surface"	01	Plane	Planar face	Flatness		1
Datum B	02	Cylindrical	Hole			1
Datum C	03	Cone	Hole			1
Taped hole	04	Threaded	Hole	Position	ABC	5
Clearance hole	05	Cylindrical	Hole	Size		5
Rotor Cone	06	Cone	Hole	Position	ABC	1
Cone Step	07	Cylindrical	Hole	Size		1
Inlet/Outlet	08	Free Form	Hole	Profile	ABC	2
Slide	09	Free Form	General Volume Remove	Profile	ABC	2
Slide walls	10	Free Form	General Volume Remove	Profile	ABC	4
Prismatic hole	11	Free Form	General Volume Remove	Profile	ABC	1

Feature Name	size	Ratio Type	Ratio Value	Orientation	Occlusion	Reasons
Datum A "Gasket Surface"	23 mm	Length x Width	23x15	Z	No	
Datum B	40 mm	Diameter : depth	10:3	-z	No	
Datum C	13 mm	Big : Small Diameter : depth	13:7:4	-z	No	
Taped hole	5 mm	Diameter : depth	5:6	Z		
Clearance hole	9 mm	Diameter : depth	9:6	Z		
Rotor Cone	80 mm	Big : Small Diameter : depth	8:4:1	Z		
Cone Step	40 mm	Diameter : depth	8:3	Z	Yes	out(FOV)
Inlet/Outlet				Free	No	
Slide				Free	Yes	Hidden
Slide walls				Free	Yes	Hidden
Prismatic hole				Free	No	

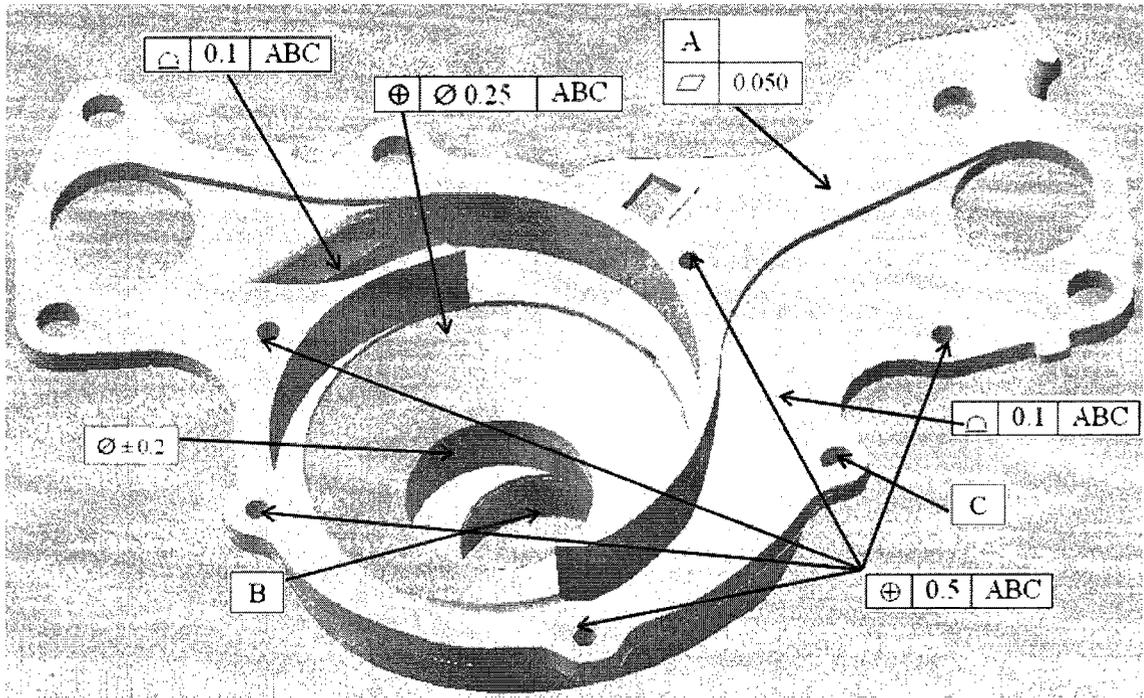


Figure (3.27) Inspection Features in Water Pump Housing.

The output of the knowledge-based sensor selection module applied to the indicated features is a list of 16 inspection operations, as shown in Table 3.6, which represents the recommended inspection tool and their orientations, as well as the inspected part orientation. Two part orientations are considered based on the manufacturing process of the part: Orientation 1, where the flat gasket surface contacts the CMM table, and orientation 2, which is the opposite orientation. Key points corresponding to every inspection feature are arbitrarily chosen.

The order of the inspection tasks is determined by formulating a TSP mathematical model as explained earlier in section 3.5. The effort to switch between operations is represented by the cost matrix shown in Table 3.7. General Algebraic Modeling System (GAMS) language and CPLEX optimization solver were used to model and solve the 0-1 integer TSP formulation.

Table (3.6) Probe and probe orientations selected to inspect the water pump housing

Op	Feature Name i	Recommended Probe		Part Orientation P _{ij}	Probe Orientation		Feature Key Point		
		Sensor	j _i		A _{ij}	B _{ij}	x	y	z
1	Datum A: Gasket Surface	Laser	1	2	0	0	-50	100	0
2	Datum A: Gasket Surface	Touch	2	2	0	0	-50	100	0
3	Datum B: Bottom Cylinder	Laser	1	1	45	45s	50	-25	60
4	Datum B: Bottom Cylinder	Touch	2	1	0	0	50	-25	60
5	Datum C: Reversed conic hole	Laser	1	1	0	0	40	-75	5
6	Datum C: Reversed conic hole	Touch	2	1	0	0	40	-75	5
7	Tapped holes	Touch	2	2	0	0	7	25	0
8	Clearance holes	Laser	1	2	0	0	-10	110	0
9	Rotor cone	Laser	1	2	0	90s	75	20	20
10	Cylindrical step	Touch	2	2	0	0	50	-20	30
11	Innlet/Outlet	Laser	1	2	45	90s	-40	100	0
12	Slide	Laser	1	2	45	45s	-20	70	0
13	Slide	Touch	2	2	0	0	-20	70	0
14	Slide wall	Laser	1	2	45	45s	70	60	0
15	Slide wall	Touch	2	2	0	0	70	60	0
16	Prismatic hole	Laser	1	2	0	0	-10	10	0

Table (3.7) Cost Matrix

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	M	300	160	450	150	450	300.14	0.06	10.22	300.23	10.01	10.06	300.06	10.19	300.19	0.14
2	300	M	460	150	450	150	0.14	300.06	310.22	0.23	310.01	310.06	0.06	310.19	0.19	300.14
3	160	460	M	310	10.11	310.11	460	160	160	460	160	150	460	150	460	160
4	450	150	310	M	300.11	0.11	150	450	460	150	460	460	150	460	150	450
5	150	450	10.11	300.11	M	300	450	150	160	450	160	160	450	160	450	150
6	450	150	310.11	0.11	300	M	150	450	460	150	460	460	150	460	150	450
7	300.1	0.14	460	150	450	150	M	300.13	310.1	0.1	310.13	310.08	0.08	310.11	0.11	300.03
8	0.06	300.06	160	450	150	450	300.13	M	10.18	300.21	10.05	10.06	300.06	10.14	300.14	0.15
9	10.22	310.22	160	460	160	460	310.1	10.18	M	310.07	10.21	10.16	310.16	10.07	310.07	10.13
10	300.2	0.23	460	150	450	150	0.1	300.21	310.07	M	310.22	310.17	0.17	310.13	0.13	300.11
11	10.01	310.01	160	460	160	460	310.13	10.05	10.21	310.22	M	10.05	310.05	10.17	310.17	10.14
12	10.06	310.06	150	460	160	460	310.08	10.06	10.16	310.17	10.05	M	310	0.13	310.13	10.09
13	300.0	0.06	460	150	450	150	0.08	300.06	310.16	0.17	310.05	310	M	310.13	0.13	300.09
14	10.19	310.19	150	460	160	460	310.11	10.14	10.07	310.13	10.17	0.13	310.13	M	310	10.14
15	300.2	0.19	460	150	450	150	0.11	300.14	310.07	0.13	310.17	310.13	0.13	310	M	300.14
16	0.14	300.14	160	450	150	450	300.03	0.15	10.13	300.11	10.14	10.09	300.09	10.14	300.14	M

The optimal order of the inspection operations is {5, 3, 14, 12, 11, 9, 16, 8, 1, 2, 13, 7, 10, 15, 4, 6}. This solution is detailed in Table 3.8, where the different inspection operations sharing the same sensor, part orientation, or probe head orientation are clustered. The inspection operations sequence is obtained by subtracting the most expensive route off the obtained solution tour as explained in section 3.5.2. The objective function value associated with the solution obtained is 641.32 seconds.

Table (3.8) Optimal order of the inspection operations showing identified clusters

Op	Feature Name I	Recommended Probe		Part Orientation P _{ij}	Probe Orientation A _{ij} B _{ij}		Starting Point x y z		
		Sensor	j _i		A _{ij}	B _{ij}	x	y	z
5	Datum C: Reversed conic hole	Laser	1	1	0	0	40	-75	5
3	Datum B: Bottom Cylinder	Laser	1	1	45	45s	50	-25	60
14	Slide wall	Laser	1	2	45	45s	70	60	0
12	Slide	Laser	1	2	45	45s	-20	70	0
11	Innlet/Outlet	Laser	1	2	45	90s	-40	100	0
9	Rotor cone	Laser	1	2	0	90s	75	20	20
16	Prismatic hole	Laser	1	2	0	0	-10	10	0
8	Clearance holes	Laser	1	2	0	0	-10	110	0
1	Datum A: Gasket Surface	Laser	1	2	0	0	-50	100	0
2	Datum A: Gasket Surface	Touch	2	2	0	0	-50	100	0
13	Slide	Touch	2	2	0	0	-20	70	0
7	Tapped holes	Touch	2	2	0	0	7	25	0
10	Cylindrical step	Touch	2	2	0	0	50	-20	30
15	Slide wall	Touch	2	2	0	0	70	60	0
4	Datum B: Bottom Cylinder	Touch	2	1	0	0	50	-25	60
6	Datum C: Reversed conic hole	Touch	2	1	0	0	40	-75	5

In conclusion, the water pump case study has illustrated the need for a hybrid inspection planning system to inspect such intricate shapes; and the results obtained demonstrated the applicability of the overall proposed planning methodology. The occluded surfaces such as the slide and the slide wall are to be inspected using both sensors for completeness while the cylindrical step hole is digitized using only tactile sensor. The bottom cylinder is digitized using only the tactile sensor but it is required to be scanned also using the laser sensor because it is a datum.

The results also show that the sensor changes contributed the most to the value of the formulated time objective function while the effect of the rapid tool traverse time contribution was the most negligible. As far as the sensor selection is concerned, two operation clusters are formed: {5, 3, 14, 12, 11, 9, 16, 8, 1} and {2, 13, 7, 10, 15, 4, 6}. Regarding the part orientation criterion, although only two part orientations were assigned, an extra part orientation change back from orientation 2 to orientation 1 has taken place; this is due mainly to the relatively higher weight of the sensor selection criterion. It is also shown that for laser sensor, similar probe orientations are grouped together successively. Finally, as for the computational complexity, the TSP is well known that it is NP-complete problem; hence, heuristics and non-traditional optimization techniques would be used to solve large problems.

3.7 Summary and Conclusions

In this chapter, the developed hybrid inspection planning model was described. A new integrated feature-based inspection planning system for hybrid contact/non-contact inspection has been developed. The proposed approach overcomes the shortcomings of both techniques; namely touch probe and strip type laser scanner. It consists of two modules: inspection method selection and inspection operations sequence optimizer/planner. Its application and use in industry can lead to reducing product development cost and increasing quality of manufactured goods. It has the advantage of improving adaptability to changing products due to its automated planning characteristics and ease of use and implementation.

The factors that affect the sensors selection were analyzed and included in an inspection-specific features' taxonomy organized in the form of a 3-D matrix. Each matrix cell contains the planning knowledge rules for the corresponding class of features considered. The rules are developed based on features related parameters, such as occlusion, dimensions and tolerances specifications. The proposed approach enables the inspection planner to match features in the CAD model with cells in the developed features taxonomy and apply the corresponding sensor selection rules. The most suitable

sensor for each feature is then selected while minimizing the time and effort in terms of changing sensors, its orientation and the corresponding part orientation.

A Traveling Salesperson Problem TSP is then applied for the first time to the macro-level inspection planning problem to optimally order the inspection tasks. Each inspection operation is conceptually modeled as a node in an undirected graph-based network of nodes and edges, where the optimum inspection plan is represented by a complete tour of the visited nodes less the one most expensive route. The process of generating the cost matrix between pairs of different consecutive operations has been automated. The developed 0-1 integer mathematical model is solved using the CPLEX solver and the GAMS algebraic modeling language.

The applicability of the developed overall hybrid inspection planning methodology to complex mechanical parts is demonstrated using an industrial case study of a water pump housing. Laser inspection integrated with tactile sensing is proven successful in digitizing parts, which are difficult to be completely digitized using a single type of sensor. The results demonstrated the capacity of the developed feature-based inspection planner for hybrid sensing systems to: 1) plan the inspection of prismatic, free form and complex mechanical parts such as water pumps, dies and moulds using combined tactile and non-contact sensors, 2) complete the acquisition of required data for parts with accessibility and occlusion problems for one or both sensors, and 3) improve the accept/reject decisions accuracy of the inspection process. It also illustrated the effectiveness and benefits of using the proposed TSP sequence planning method to optimize the inspection process and to minimize non-digitization effort and time.

The proposed sensor selection knowledge-based system is limited to a single tolerance specification per feature. However, the consideration of more than one tolerance control per inspected feature is worthy of investigation in the future. The developed inspection-specific features taxonomy captures the knowledge of the human planner. It can be extended by adding new rules and features applied in different industries. The developed system can be integrated with CAD models to automate the

process of input data collection and interpretation, as well as with downstream applications for quality related performance analysis and decision making.

4. CAD-BASED SEGMENTATION OF UN-ORGANIZED POINT CLOUD FOR INSPECTION

Once we obtain a registered point cloud from both types of sensors the next step is to divide it into sub-point clouds, where each sub-point cloud represents the measured points for a certain meaningful and functional feature. This feature is to be later verified for tolerance requirement satisfaction. Segmentation is the process of partitioning such 3-D range data into non-intersecting homogeneous subsets. In this chapter, the process of segmenting a 3D *un-organized* point cloud to prepare the measured points for localization and tolerance verification is presented.

4.1 Managing Un-Organized Point Clouds

In many industries, the problem of segmentation of an un-organized point cloud is still an intriguing problem with several different applications for model reconstruction such as reverse engineering, computer visualization and animation. Non-contact digitization technologies such as laser scanning are progressing quickly. These advances include improvements in accuracy, in speed and in the amount of generated points. This, in turn, encouraged the inspection systems, which used to use only touch probes mounted on a Coordinate Measuring Machine (CMM), to incorporate and benefit from these technologies.

In inspection, although the CAD model of the inspected part exists, features in the measured point cloud must be detected in order to determine their characteristics and compare them with the CAD model to verify that the detected features meet their functional and tolerance requirements. When using contact probes on a CMM, the relationship between the measured points and the inspected feature is known. When using laser scanner or any non-contact sensor, the sample points such as range data, which belong to different features, are obtained collectively from one scan. The scan lines and the scan path usually cover multiple inspected features.

For complex parts with intricate complex topology and/or complex geometry, which are difficult to inspect and verify using traditional inspection methods such as Jackscrew, wobble-plate, fixed-plane or precision spindle (section 3.4.3), sample points from one view are not enough to characterize the object shape. Multiple views with high density and high accuracy need to be taken with multiple scanning processes to cover all the features of interest. Hence, obtaining an organized point cloud that captures many details is difficult and sometimes impossible, particularly when the part contains deep holes and special features in different orientations. The scanned coordinate points resulting from different scan must then be merged and registered into a single point cloud that would most likely be *un-organized*.

Based on the input point clouds, a triangular mesh structure is used to interpolate the sample points and linearly approximate the object shape. The developed mesh, therefore, substitutes the measured points with triangles. The segmentation process would then be applied to divide the mesh and identify features. Mesh segmentation is used in several applications such as collision detection, computer visualization and animation; however, this mesh representation is not adequate for inspection purposes because smooth surfaces are represented in an inaccurate way by many planar triangles with discontinuous normal direction, which contributes to loss of inspection accuracy and leads to errors in inspection decisions. An increasing demand for effective and direct manipulation of such huge un-structured (*un-organized*) point clouds is necessary in order not to lose the accuracy of the measured points for inspection purposes.

When inspecting a mechanical part, the measured point cloud is compared to the CAD model. However, some features, not the whole part, are required to be verified for different types of tolerances. The features in the point cloud must then be detected in order to determine their characteristics and compare them with the CAD model. Hence, there is a need to segment (divide) the 3D un-organized point cloud into meaningful components, which represent the inspected features. Most of the current segmentation techniques start from an image, an organized point cloud, or a mesh to generate and reconstruct a model for the digitized part.

The quality of the segmentation process affects the quality of the inspection decisions. A novel segmentation algorithm is proposed based on the feature geometric parameters from the CAD information. It analyzes un-organized point clouds containing multiple features, identifies the individual features and produces a set of sub-point clouds accordingly for inspection purposes. The contribution of this research lies in its ability to work directly with an un-organized point primitives used to represent the measured part without explicit construction of a mesh. This approach leads to efficient usage of storage and computing resources to segment a point set. Moreover, the output of the developed algorithm is the actual measured points; not a representative triangle or substitute points. This, in turn, improves the accuracy of the inspection results. Another advantage of the proposed algorithm is its simplicity and ease of implementation.

4.2 Related Work

Automatic segmentation of a 3-D point cloud is a complex iterative process where the original point set is logically divided into meaningful subsets, one for each surface, such that each subset contains just those points sampled from a particular surface of interest. Widely diverse methods for segmentation differ according to the quality of measurement, quantity of points, geometric characteristics of the part and amount of human interaction required. Attene *et al* (2006) recently conducted a comparative study of only five segmentation techniques. The comparison included the extraction of correct segments, the boundaries between segments, the type of segmentation (Hierarchical / multi-scale), the sensitivity to pose, noise and tessellation, and the asymptotic complexity and control parameters. The study showed that there is no perfect segmentation algorithm; each algorithm has benefits and drawbacks.

Most of the existing segmentation techniques are classified as either edge-based, region-based or hybrid. Region-based methods usually partition the image into surface regions (Besl and Jain 1988, Sapidis and Besl 1995, Yamazaki *et al.* 2006), while the edge detection techniques are intended to isolate discontinuities in both depth and surface orientation (Huang and Menq 2001, Woo *et al.* 2002, Alberts 2004, Meyer and Marin

2004, Lavoue *et al.* 2005). A hybrid edge-based and region-based segmentation approach for range image was introduced by Yokoya and Levine (1989). A modification of the hybrid segmentation procedure was proposed by AlRashdan *et al.* (2000) where the region-based and edge-based approaches were integrated using neural networks. Either edge-based or region-based segmentation schemes can be further classified according to either: 1) the format of the input data, 2) the propagation technique within the measured points or 3) the separation method to detect the feature boundaries.

4.2.1 Input Data Format

There are two types of input data format for the segmentation process, which can be classified as Range Image (RI) or Range Data (RD). Range Images are 2-D images (pixels) where the image is viewed as a piecewise smooth surface and the z coordinate corresponding to a pixel x and y is considered a third characteristic. In a range image, this value represents the distance to a physical surface from a reference surface, while in intensity images, it represents the number of visible photons incident at this point in the focal plane of the camera (Besl and Jain 1988, Yokoya and Levine 1989, Sapidis and Besl 1995).

Range Data (RD) segmentation is based on the segmentation of 3-D digitized data i.e. a point cloud captured by a range finder such as laser scanners or a Coordinate Measurement Machine (CMM), where the data format is a list of the 3-D coordinates of the measured points. The structure of the RD heavily depends on the digitization process and can be described by two structures. The first structure is an organized point cloud, an image style, which describes almost parallel digitization profiles over a 2½D object (AlRashdan *et al.* 2000) or a CT-Scan (Delingette *et al.* 1997), or a scan stripes (Woo *et al.* 2002, Patane and Spagnuolo 2002). An organized point cloud could also be obtained using a tactile scanner (Alberts 2004). When the input data from scanning process is ordered, the segmentation process follows the scanning lines. AlRashdan *et al.* (2000) split the 3D point cloud in halves to obtain a 2½D object and grouped the data in a grid. Alberts (2004) considered data type delivered by tactile sensors in a scan path such as the Cyclone. Woo *et al.* (2002) and Patane and Spagnuolo (2002) considered each scan path

that consists of a series of line segment described by a set of points. The registration of multiple scans is performed after the estimation of normal directions and storing the point cloud into a point data structure which includes the (x,y,z) coordinates and the (x,y,z) of the normal components (Woo *et al.* 2002).

The second structure is an *un-organized* (Un-Structured) point cloud obtained from several scans and merged into a single point cloud (Katz and Tal 2003, Meyer and Marin 2004, Benko and Varady 2004, Lavoue *et al.* 2005). This data format provides precise and dense data, which is good, but it includes redundant information and is considered one of the bottlenecks in data processing. The pros and cons of point cloud merging and combining single-view models were mentioned in (Benko *et al.* 2001). The range data in this structure is usually tessellated to form a triangular mesh before segmentation. Katz and Tal (2003) proposed a hierarchical mesh decomposition that proceeds from coarse to fine triangles. Each node in the hierarchy tree is associated with a mesh of a particular patch and the root is associated with the whole input object. The input data format in Meyer and Marin (2004) is a polygonal representation of the surface that facilitates the selection of the start point and its vicinity. The surface segmentation is then performed on this mesh. Benko and Varady (2004) represented a point region by a connected set of triangles. A point neighborhood with the minimum number of points is determined by the adjacent rings of triangles around the selected point. Lavoue *et al.* (2005) addressed the 3D triangle meshes in general and particularly they focused on optimized triangulated CAD meshes.

It can be seen that the *un-organized* structure, where a triangulated mesh is usually constructed, is the most common and the most challenging structure to deal with for segmentation purposes. In inspection of complex mechanical parts, it is needed to compare the measured point of some features with the CAD model. Hence, it is required to keep the actual measured points in the segmentation output and to deal directly with point clouds. The results of the digitization process are usually un-structured range data format. The objective is to have the output as point cloud not in the form of a mesh or a re-constructed surface. Yamazaki *et al.* (2006) introduced a segmentation approach that

deals directly with unorganized point sets. Their algorithm is suitable to segment natural object for reverse engineering application where features are defined using topological approach. However, this process lacks the knowledge of the inspected features in the CAD model and their geometrical and dimensional characteristics. Orazi and Tani (2007) presented a post alignment segmentation process by projecting scanned points on CAD surfaces where local differences between scanned and nominal surfaces are evaluated by comparing curvatures of corresponding points.

4.2.2 Propagation Techniques

The segmentation process usually starts with a seed feature and then propagates through the point cloud by using region growing techniques such as neighborhood search (connectivity) and differential geometry (homogeneity). Besl and Jain (1988) developed an iterative region growing technique using variable-order surface fitting that uses spatial coherence of the data to organize pixels into meaningful groups for subsequent visual processes. A modified version of the region growing technique for image segmentation, used in Besl and Jain (1988), was later presented by Sapidis and Besl (1995). The input points are given in the form of “grid” dense range image, constructed using a matrix for the heights of surface points above a plane. The algorithm is characterized by a two-stage region growing strategy and a set of simple rules where the points lie on a regular grid to eliminate a large number of topological and geometric operations included in the initial version of the region growing technique from Besl and Jain. AlRashdan *et al.* (2000) developed a back propagation network based on the sign of the Gaussian and mean curvature, as in Besl and Jain (1988) and Yokoya and Levine (1989), defined features by eight different primitives. Simulated range data was used to train the network to select a threshold value.

In the segmentation technique developed by Huang and Menq (2001), the combinatorial manifold mesh grows from its boundary edges by sequentially choosing the best point for each boundary edge and by updating the manifold structure with an appropriate topological operation such as vertex joining, ear attaching or bridge linking. The best point is selected if its projection onto the tangent plane falls inside a fan-shaped

region with two bounding angles specified by two incident boundary edges. A common neighbor criterion was proposed to select the best candidate point to be connected to a boundary edge, which is neighboring both end points of this edge.

Lavoue *et al.* (2005) classified vertices according to the values of their principal curvatures; then a triangle growing and labeling operation is performed. When a seed triangle is encountered, a new region is created, containing this triangle, associated with a new label and a curvature value. Then a recursive process extends this region. This growing algorithm is repeated for every other triangle marked as seed and still unlabeled.

Meyer and Marin (2004) constructed a step-by-step edge called “Absoid fitting” that starts from a list of start points, then the angle between two adjacent facets is calculated so that the two nodes of this edge are put in the start point list if it is greater than a given threshold. A point cloud representative of its vicinity is then selected to fit a paraboloid. The extent of the vicinity is then adapted. The absoid fitting gives a point that is located on the edge, as well as the edge direction and curvatures.

For ordered-structure point cloud, which takes into account the scan path, the propagation process usually follows the sequence of the scan lines. A path adaptive triangulation algorithm was proposed by Alberts (2004). Patane and Spagnolo (2002) introduced a local displacement algorithm that follows the scan lines based on a sequence of local updates, where, at each iteration, the data set is slightly modified. In the Octree method developed by Woo *et al.* (2002), the propagation starts by designating one cell in the leaf nodes as a seed cell. This seed cell grows up by merging adjacent cells by checking the homogeneity conditions (average normal values). If two cells have a common parent and one of the vertices in the cell is shared with other cells a connectivity test is performed to search the adjacent cells. In order to merge the cells, both the connectivity and homogeneity conditions should be satisfied.

It can be seen that the propagation direction in the segmentation process highly depends on the neighborhood search (connectivity) where common neighbors define the

region growing direction. Depending on a threshold value, angles between the normal at each point and the one at its neighbor define their belonging to the same feature.

4.2.3 Separation Techniques

Separation techniques search for borders and discontinuities in the point cloud. In region-based techniques, where a sequence of variable polynomial order surfaces are fitted on measured points, the region growing process is terminated when the region growing has converged or when all the polynomials have failed to fit the seed region (Besl and Jain 1988). AlRashdan *et al.* (2000) used Laplacian filter, based on the second derivative of the range data, at each point, to detect step edges, which corresponds to surface discontinuities while surface normal values with the calculated eight neighboring pixels were used to find roof edge points.

Benko *et al.* (2001) used a non-iterative “Direct Segmentation” approach based on the fact that it is possible to compute local characteristic quantities (e.g. normal direction) within the interior face. This characterizes the planarity of the point neighborhood. An extension to this work is presented in Benko and Varady (2004) where subdivision is performed using a sequence of different types of statistical tests (filters) and indicators.

In the edge-based techniques proposed by Huang and Menq (2001), the directional curvature across each mesh edge is estimated and compared with the directional curvature of adjacent vertices in the same direction. The mesh edges located on or near the border curves are then identified as border edges based on their curvature characteristics. Lavoue *et al.* (2005) detected sharp edges and vertices by analyzing the curvature tensor for each vertex and extracting the principle curvature and directions values. Then, they subdivide the sharp triangles by adding a new vertex at the center. Woo *et al.* (2002) introduced a different edge-based segmentation approach that uses an Octree-based 3-D grid splitting process where separation is done by elimination of cells.

Meyer and Marin (2004) adapted Huang and Menq’s (2001) approach in their Absoid fitting technique where the growing process is stopped by the calculated edges.

They defined a 1st order discontinuity for surface with edges. The step-by-step constructed edge process stops when: 1) the edge loops back, 2) the curvature radiuses of the edge become smaller than a given threshold (usually near a corner) or the angle between two flanks becomes greater than a given threshold (widening-out of the edge). Katz and Tal (2003) considered fuzzy boundaries where no sharp edges can be found in the measure part. First they compute a fuzzy decomposition using an iterative clustering scheme and then the exact boundaries are constructed between the components using minimum cut algorithm.

Alberts (2004) considered the information about scan paths to allow reconstructing creases and ridges more reliably. The information from the scan paths is used to define the boundaries of surfaces to create a triangular mesh. By assuming that the sampling points in the scan paths are linearly arranged in the input data set according to their occurrence on the scan paths, the different scan lines are studied to detect some features. Only sharp bends, as shape features of the scan paths, were considered. They are detected by studying the z profile of a scan path while the behavior of a path with respect to the x and y coordinates is ignored.

It can be seen that most of the separation methods or edge detection, in which object discontinuities are detected, depend on calculating the normal vector at each vertex then calculating the angle between adjacent points. Borders are detected when the value of this angle exceeds the threshold value.

Most articles found in the literature attempted to develop segmentation methods by exactly fitting surfaces to precisely find edge points or curves, which are time consuming and difficult. Nevertheless, the segmentation is usually based on organized point cloud or meshed surfaces. Table 4.1 summarizes the different techniques for segmentation. In the general case, segmentation cannot be fully automated and iterative procedures and/or user interactions are necessary, especially in the presence of free form surfaces with regular shape features found in complex mechanical parts. At present, the segmentation of digitized data is performed interactively, where the operator defines the approximate

locations of part edges and surface boundaries in the digitized data. Attene *et al* (2006) in their comparative study suggested that since segmentation algorithms can neither be formalized nor measured mathematically an empirical and semantic basis for research should be provided to recognize specific parts and their role. When segmentation is used for inspection, a surface is fitted and the value of the fitting parameters is compared to the design value. However, in inspection, it is needed to calculate the deviation of the points from the corresponding surface and compare this deviation to the value of the specified geometric and dimensional tolerance i.e. inside the tolerance zone. Hence, it is necessary to preserve the measured points in their simple and explicit format.

Table (4.1) Comparison of different segmentation techniques

References:	Approach	Input data	Tools	Propagation
Besl and Jain (1988)	Region-based	Range Image	Sign Gaussian & Mean curvature	Region Growing
Yokoya and Levin (1989)	Hybrid	Range Image	Sign Gaussian & Mean curvature Jump & Roof	Region Growing
Sapidis and Besl (1995)	Region-based	Range Image	Threshold Selection	Region Growing
AlRashdan <i>et al.</i> (2000)	Hybrid	2½-D Organized Point Cloud	Gaussian & Mean curvature Laplacian filter Neural Network	Ordered
Patane and Spagnuolo (2002)	Edge-based	2½-D Organized Point Cloud	Slice oriented curvature threshold	Hierarchical approach
Woo <i>et al.</i> (2002)	Edge-based	3-D grid	Octree grid	Ordered
Benko <i>et al.</i> (2001)	Edge-based	Meshed Surface	Direct Segmentation Removing Triangles	Ordered
Benko and Varady (2004)	Region-based	Meshed Surface	Direct Segmentation Test Hierarchy	Ordered
Alberts (2004)	Edge-based	Touch / Point Laser	Z Profile	Ordered

4.3 Proposed Segmentation Algorithm

As discussed earlier, most of the existing 3-D point cloud segmentation techniques use either region-based, edge-based or hybrid approaches. Since edge-based techniques lack information about surfaces, the region-based segmentation approach is employed in this work to extract measured meaningful features. Furthermore, it is difficult to extract the exact edge points because the scanned data are made up of discrete points and edge points are not always or explicitly included in the scanned data. The problem is, then, how to identify and exclude border points from adjacent features. The proposed algorithm presents a decomposition algorithm of a 3-D point cloud into near constant curvature features. We address particularly complex mechanical parts with CAD models; natural objects are not considered in this work. The developed region-based segmentation algorithm relies on the theory of differential geometry where the discontinuities and curvature changes in the range data are first detected. Then, particular geometrical tests are performed to generate the segmented features using different types of indicators such as regional indicator, geometric indicator and propagation indicator. Figure 4.1 shows the flowchart of the developed segmentation algorithm.

The inputs to this algorithm are: 1) a 3D un-organized point cloud, and 2) CAD Info such as the number of features NF , a seed point SP_{SF} for each feature, a continuity measure K_{SF} for each surface feature (Threshold value). It should be noted that a feature can have two threshold values in two different directions (K_{SFu} , K_{SFv}). Consider the example part in figure 4.2 (a), which includes planar, cylindrical and freeform surfaces. Figure 4.2 (b) illustrates the generated sub-point clouds after segmentation, where each feature is represented by a color. The given point cloud should be first stored and ordered in a Point Data Structure (PDS), where points with potential maximum number of neighbor points are then selected based on CAD info as seed points as well as a threshold value for each feature. The iterative region-based segmentation algorithm is then applied to divide the point cloud. Finally, a feature rectification subroutine is applied to adjust the obtained sub-point sets into meaningful features. These steps are detailed below.

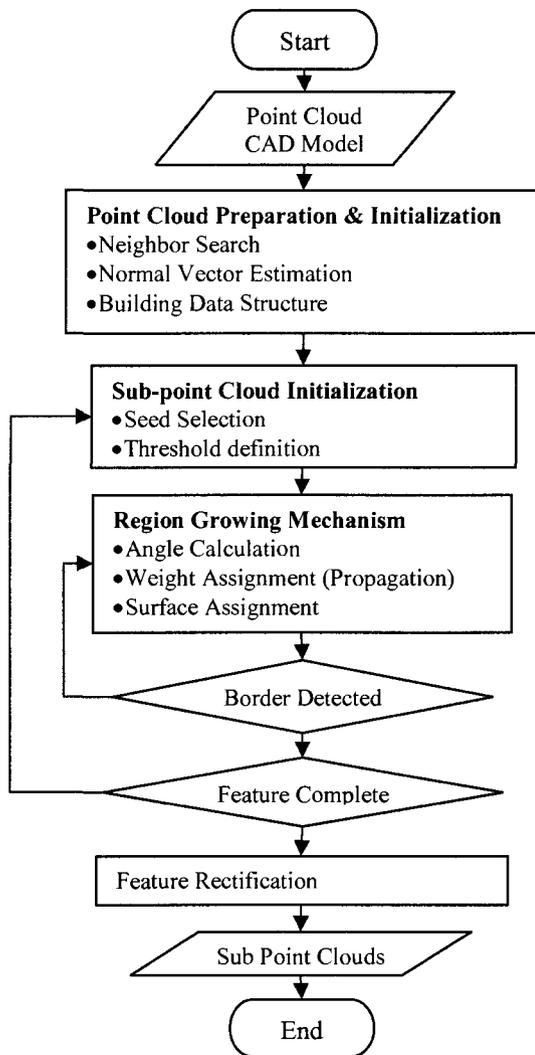
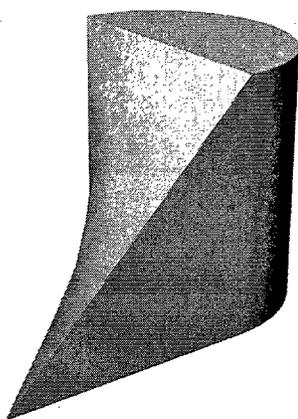
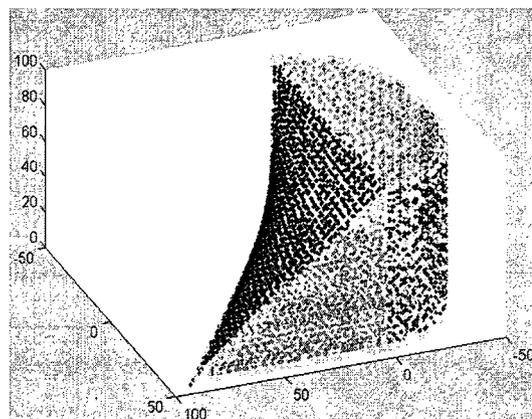


Figure (4.1) Proposed Inspection Specific Segmentation flowchart.



(a) CAD model



(b) Segmented point cloud

Figure (4.2) Example Part.

4.3.1 Point Cloud Preparation and Initialization

The scanned data obtained from multiple views obtained by multiple scanning process using non-contact scanners consist of a number of points that only include an unorganized three-dimensional coordinates (x, y, z) points on the surface of an object. It is difficult to obtain any geometric information of a part directly from such raw data. A point by itself cannot identify a feature. A surface feature can be determined by a point and its neighbors. To get the geometric information of a point and its neighbors, it is required to link P_i to its neighbor points. Given a point set $\{P\}$, the first phase of the segmentation algorithm includes the storage and organization of the measured points, searching neighbor points, estimating curvature characteristics, and preparing the point cloud for segmentation without constructing a triangular mesh (Figure 4.3).

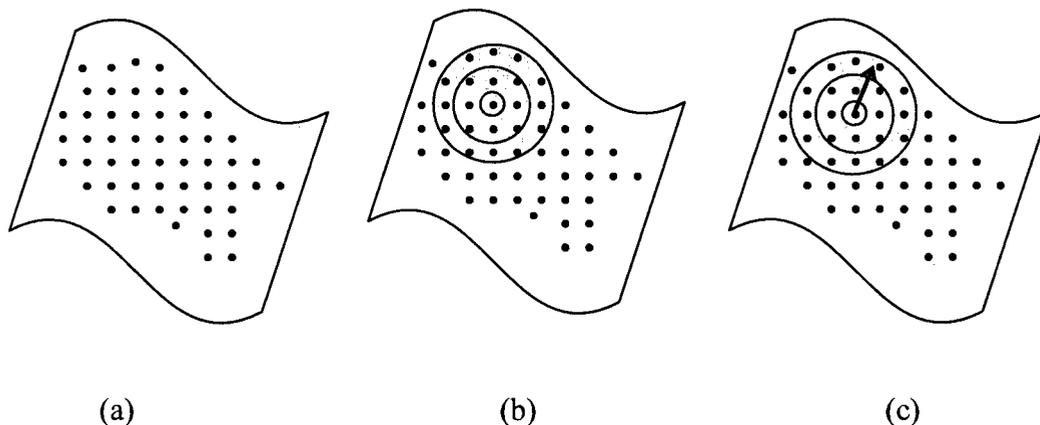


Figure (4.3) Point cloud preparation a) point cloud b) two spheres and c) normal vector estimation.

4.3.1.1 Neighborhood Search

Neighborhood search can be based on a constant number of surrounding points or based on a constant distance between points. Yamazaki *et al* (2006) determined the neighbor points in a 3D point cloud by a constant number of points and defined seven neighbor points as the best number in a three dimensions space. In this research, neighbor points are defined in terms of the closest Euclidean distance. The best neighborhood function is considered as a sphere where the distance between the point P_i and its' far neighbor in all directions is constant. To overcome the problem of over-segmentation and under-segmentation, two levels of neighborhood function are chosen. Two spheres (S_1 ,

S_2) are assigned to the selected point P_i as shown in Figure 4.5. The first sphere S_1 with radius R_1 is used to determine the neighboring points close to point P_i and is used to determine the normal direction at point P_i . R_1 is strictly determined by the density of the point cloud and can be equal to twice the distance between two consecutive points.

$$R_1 = 2 d \tag{4.1}$$

$$\text{where } d = \max(d_1, d_2) \tag{4.2}$$

d is the distance between two consecutive points in the point cloud. d_1 and d_2 are extracted from the scanner setup as illustrated in Figure 4.4. In such case, under-segmentation due to smoothly connected features can be avoided. The second sphere S_2 with radius R_2 encloses a larger neighborhood points for point P_i to widen the search space, which improves the selection of the propagation direction using weights and avoids the problem of under-segmentation. To find appropriate general threshold values for an object with variable point density and noise distribution is a difficult task and several iterations may be needed until the proper values are found. Several trials to avoid the problem of over-segmentation due to noise were executed by selecting a value for R_2 in the interval $[1.5R_1, 3R_1]$, i.e. $1.5 R_1 \leq R_2 \leq 3 R_1$. Besides accumulated experiment, a good sensitivity analysis with statistical indicators can better tune these parameters. Based on numerical experimentation results, the best value for R_2 is found to be:

$$R_2 = 2 R_1 \tag{4.3}$$

By comparing the distance D_{ij} between the point P_i and its neighbor point P_j with R_1 and R_2 , a list of indices of the neighborhood points is assigned to point P_i . A point is considered an outlier if it has no neighbors within S_1 .

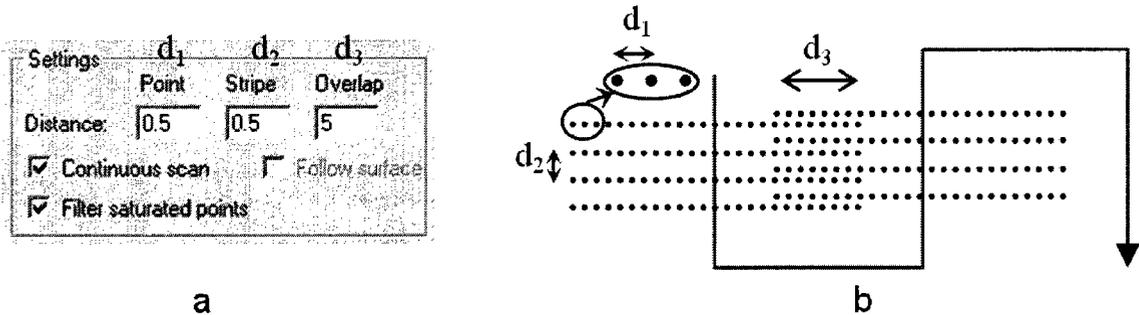


Figure (4.4) d_1 and d_2 from laser scanner parameters.

4.3.1.2 Normal Vector Estimation

In estimating the normal vector, triangulation methods are usually used regardless of the point data type. In such case the normal vector is defined by the normal direction of the triangle that composes the triangular mesh at that point. In the developed algorithm, the point normal is not limited to three points and is defined differently. It is the normal direction of the plane surface that can be fitted to the point P_i and its neighboring points in the sphere S_1 . A normal vector is then computed and assigned to point P_i in the data structure. A graphical illustration of point P_i , its neighbors, normal vector and the two spheres (S_1 , S_2) is presented in Figure 4.5.

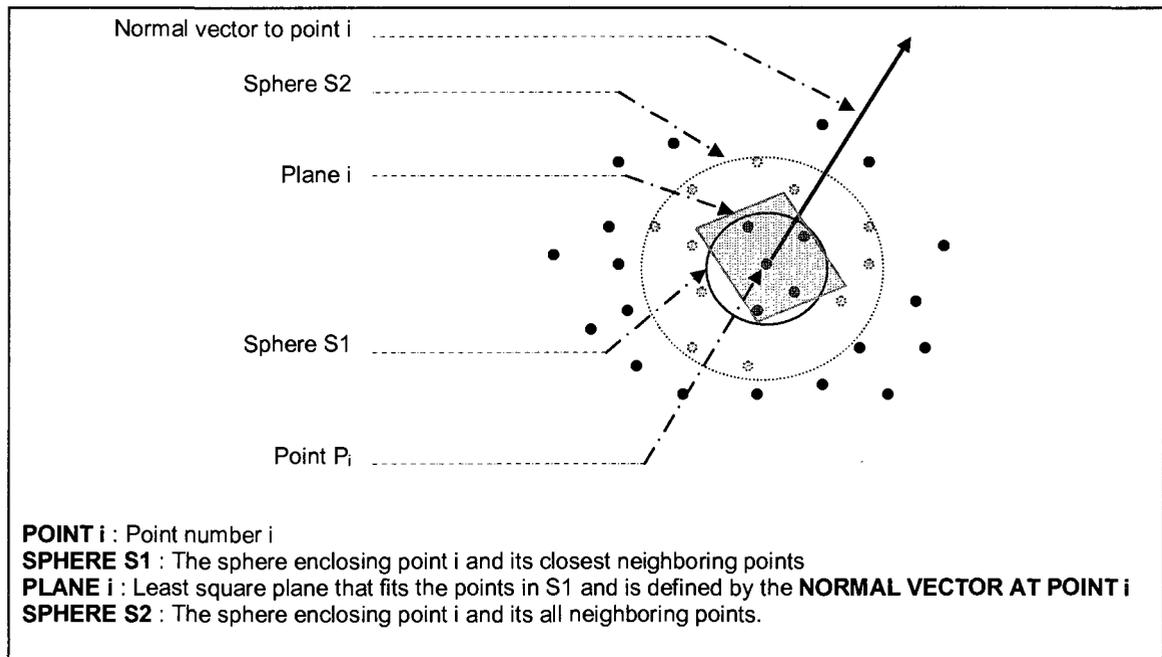


Figure (4.5) Normal vector estimation at point P_i .

4.3.1.3 Point Data Structure (PDS) Initialization

A Point Data Structure (PDS), which includes information about each point in the point cloud, is generated. This data structure is updated along with the segmentation process (Figure 4.6). The algorithm starts with the initialization of two parameters; weight and surface feature parameters for all points. The *weight parameter* directs the segmentation propagation process from point P_i to point P_j and is called *propagation indicator*. The value assigned to this parameter is a unit value (1), which will be modified

from one iteration to the next. The *surface feature parameter* is a parameter to identify, for each point in the point cloud, to which surface feature the point P_i belongs. This value starts with zero (0) since no feature is identified at the beginning of the segmentation process.

Point Data Structure for Point P_i:	
Point Index: i	
Point coordinates: x,y,z	Normal vector: n_i
Weight of the point: W_i	Feature that point i belongs to: SF_i
List of points indices in the sphere: $S1$	List of points indices in the sphere: $S2$
Neighboring Points' indices: NP_i	

Figure (4.6) Point Data Structure (PDS).

4.3.2 Sub-point Cloud Initiation (New Feature)

Based on the information from the CAD file, characteristics of the inspected features such as the number of features NF , the surface feature number SF , the seed points SP_{SF} , and surface feature continuity K_{SF} are obtained. Orazi and Tani (2007) suggested a segmentation process based on the projection on the CAD model where a pre-alignment of the point cloud with the CAD model is performed a priori. In the proposed algorithm, the segmentation process starts by assigning the point P_i to the first surface feature SF . The segmentation then proceeds by comparing the normal vector of a point to the normal vectors of its neighbors, which becomes the current point for subsequent iteration.

In region-based segmentation, a *seed point* SP_{SF} is selected for each surface feature SF to start the propagation. Seed selection is a great challenge in the segmentation algorithms since all consequent steps depend on the selected seed. In mesh segmentation, different characteristics were proposed to select the seed triangles (Lavoue *et al.* 2005). There is no automated procedure to select seed points from the point cloud since there is no other information that belongs to the points in the point cloud except the (x, y, z) values. For a direct segmentation of a point cloud, Yamazaki *et al.* (2006) proposed a method for seed selection based on the geodesic distance computation to determine a sink, which they called super-node. They declared that this process is the most time and memory consuming process within the segmentation process. However, this method is useful for segmenting natural shape objects. For the inspection of complex mechanical

parts that include prismatic and free-form features, with known CAD model, a predetermined seed point selection is more efficient. However, this process is human (user) dependant and is performed manually. This process is performed interactively following the guidelines illustrated in Figure 4.7. Based on Morse theory, the centrality of a point within a surface is a key to define a critical point (Yamazaki *et al.*, 2006). Hence, a seed point SP_{SF} must be selected in the middle of the surface, not on the edge, and with lots of neighboring points. It should be noted that a point with expected large number of neighbors might be located on, or near, the edge of the surface. This point should be avoided in the selection process. Figure 4.8 shows a good and a bad selection of a seed point.

Characteristics of the seed point selection:

- Middle of the surface
- Not on/near the edge
- Visible point - With neighbor points
- Not noisy point

Figure (4.7) Seed Point Selection Characteristics.

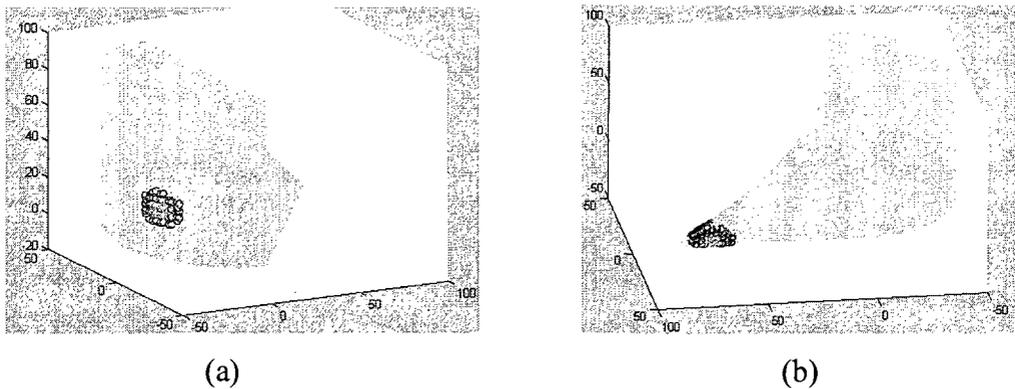


Figure (4.8) (a) good seed point and (b) bad seed point.

A **threshold value** K_{SF} is also needed to define the boundary of the segmented features. If the value of the angle between the normal directions at two neighbored points is higher than the threshold value, then the two points belong to different surfaces. A fixed metric threshold is not desirable for automatic boundary detection because sampling density varies with different object and different measuring methods (Huang

and Menq 2002). AlRashdan *et al.* (2000) automated threshold selection by Neural Network where simulated range data were used to train the network. The value of the threshold for surface SF which defines the limit of the surface can be determined by feature characteristics and is represented here as the continuity (derivative) value of the surface. This value is extracted from the CAD info based on the type of geometric feature.

It should also be noted that features with multiple faces such as rectangular or T-shape features would require multiple sub-seed points for each face which will be merged together at the end of the segmentation process to define the sub-point cloud of the inspected feature.

4.3.3 Region Growing Mechanism

Usually, in region-based segmentation techniques, the process starts with a seed from a labeled range image (Besl and Jain 1988, AlRashdan *et al.* 2000, etc.). The image is divided into eight different primitive surfaces that depend on the sign of the Gaussian and the mean curvature (i.e. peak, pit, ridge, saddle ridge, valley, saddle valley, flat, and minimal). The region growing technique is then used iteratively on the labeled images starting with a seed region approximated by a bi-variant polynomial. First the seed region is approximated by a first order polynomial (planar surface). If the planar fit is acceptable according to a statistical test, the seed region grows on the planar surface fit. Otherwise, a second order polynomial is tried. If it is accepted, the region will grow in the same fashion, but with a higher order polynomial. Those iterations stop when the region growing has converged or when all the polynomials have failed to fit the seed region. This process is efficient for partitioning one free form surface into several smaller surfaces. The purpose of the developed segmentation algorithm is to allocate each point in the point cloud to its corresponding feature with known polynomial order.

In a similar approach, the developed algorithm starts from the first point (seed), chosen as described earlier to be the current point P_i , and then proceed in an iterative region growing approach. The propagation direction depends on a weight parameter

calculated from the angle difference between normals. First, the angle between the normal vector direction n_j for all non-segmented neighbor points P_j and the normal direction n_i at the current point P_i is calculated as described in section 4.3.3.1. Hence, a weight is assigned to P_j and adjusted in the PDS. The best neighbor point is then selected to become the current point based on the highest weight value. A common neighbor for the current point and a previous segmented point with less deviation of normal would be selected as the best neighbor. In case two or more non-segmented neighbor points have the same weight, the non-segmented point with less Euclidean distance to the current point is selected as the best neighbor. The feature SF is then assigned to the neighbor point (i.e. the point become segmented).

It should be noted that, at each iteration, the current point P_i belongs to the current Surface Feature SF and the iterative region growing mechanism search for the best neighbor to be assigned the same SF. The surface features SF are prioritized (ordered) as follows: Regular shape surface (with less polynomial order) are considered first. To reduce the computation time and complexity, features with bigger areas are selected first so that a bigger number of points are selected and not to be used several times.

4.3.3.1 Angles Calculations

In the extracted neighbor structure of each point, the orientation angles of its neighbors in the local tangent plane are calculated and the neighbors are sorted based on their orientations differences. The angle α_{ij} , in equation 4.4, is the angle between the two normal vectors for two neighbor points i and j (Figure 4.9).

$$\alpha_{ij} = \cos^{-1} \left(\frac{n_i \bullet n_j}{|n_i| |n_j|} \right) \quad (4.4)$$

The two points belong to the same surface feature if the angle α_{ij} is smaller than the predefined threshold value K_{SF} . The point P_j is a border point or belongs to the adjacent surface feature when the angle α_{ij} is greater than K_{SF} . It should be noted that α_{ij} can be also bidirectional along with the threshold K_{SF} .

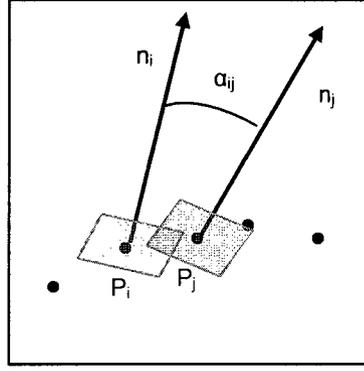


Figure (4.9) angle α_{ij} estimation.

4.3.3.2 Assigning Weights (Propagation Direction)

A common neighbor criterion was introduced by Huang and Menq (2002) to propagate within the point cloud. The common neighbor criterion states that the best candidate point to be connected to a boundary edge in their combinatorial mesh reconstruction method is a neighbor of both end points of this edge. Similarly, a weight-based method is developed in this work, to propagate within the point cloud through a *propagation indicator*, where the propagation direction depends on a weight factor calculated from the angle α_{ij} so that the point with normal direction closer in orientation to the current point is the best to select for the next iteration (Figure 4.10). The weight of each neighboring point, that doesn't belong to a surface, in the sphere S_1 is calculated based on the angle between the normal vector n_j at the non-segmented neighbor point P_j and the normal vector n_i at point P_i . The value of the weight W_j for the non-segmented neighbor point P_j is increased by the value of W_{ij} where

$$W_{ij} = \begin{cases} 1 - \sin \alpha_{ij} & \text{if } \alpha_{ij} \leq K_{SF} \\ 0 & \text{Otherwise} \end{cases} \quad (4.5)$$

The value of the weight W_j increases at each iteration. The propagation through the point cloud is then directed toward points that have the least difference between their normal (i.e. highest weight value). This, in turn, directs the propagation toward *common neighbor*. This is to maintain consistency in the same surface and to avoid the generation of extra non-meaningful sub-point clouds (over-segmentation). In case the propagation

process reaches a border point, the process restarts from the closest non-assigned surface point, with highest weight, to the seed point SP_{SF} of surface feature SF.

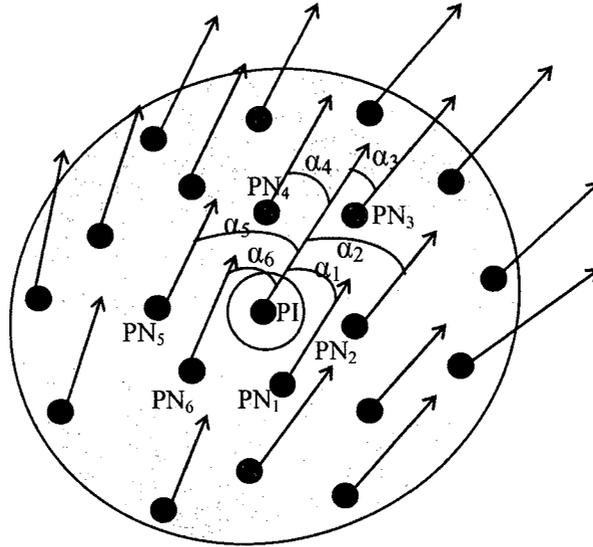


Figure (4.10) angle α_{ij} in all directions.

4.3.3.3 Surface Assignment

The number of features NF in the CAD model is defined in the second input data: “CAD Info”, as discussed earlier. Each feature is represented by a feature ID, a seed point SP_{SF} and a threshold value K_{SF} . Surface assignment starts with assigning the value of the surface feature SF to the seed point SP_{SF} . A point P_j is added to the Surface Feature SF (i.e. the same surface value is assigned to the point P_j) if the angle α_{ij} between its normal vector value n_j and the normal vector value n_i of the current point P_i that belongs to the surface feature SF is less than the threshold value K_{SF} . When the propagation process reaches the border of the surface, a clear change in the value of α_{ij} is noticed. Hence, the weight value of the current point is switched back to the initial value: one (1), so that the point is considered when assigning surface to the adjacent feature. The next iterated point is selected based on its closeness to the seed point of the current surface.

4.3.4 Loop Conditions

The segmentation loop ends with two consecutive tests which are performed to iterate the surface assignment process to the point cloud: first, a *Border Detection test*,

still in the same surface feature SF, is performed as a *geometric indicator* to the limit of the surface. The closest weighted point to the seed point is then selected to restart the iteration. If two or more weighted points have the same distance from the seed, the one with the highest weight is selected as described earlier.

Second, a *Complete Feature test* is performed as a *regional indicator* that all the points, which belong to the feature SF have been analyzed. This test checks if all the weights of non-segmented points are equal to the initial value: one (1). Then it proceeds to the next surface feature (SF + 1). The algorithm stops when all surface features have been addressed and assigned points.

4.3.5 Feature Rectification

The last step in the segmentation algorithm is the process of merging sub-point clouds to form meaningful features for inspection. The segmentation output is refined based on information from the CAD model as well as tolerance specifications. The developed segmentation method extracts near constant curvature simple surfaces from the 3D point cloud, which results in several sub-point clouds. However, the significance of the corresponding features is not considered when applying the surface assignment process. Our purpose is to obtain clean sub-point clouds where each sub-point cloud represents a meaningful feature with tolerance requirements to be verified. As discussed earlier, a meaningful feature, such as a T shape slot, may be composed of several surfaces that cannot be considered in one surface assignment process. Hence, a further step is needed to merge predefined sub-point cloud together. Consider for example that a parallelism tolerance specification is specified between two surfaces, the two point clouds representing the two surfaces are hence linked together by this parallelism constraint. After rectification of linked features, the output of the segmentation process is a set of NF sub-point clouds.

The rest of non-segmented points in the point cloud are considered as a border region. A border region can be extracted using filters, contour tracking algorithm, or Absoid fitting as described in related work (Meyer and Marin, 2004). However, this

process is not considered here since our objective is not to reconstruct features; but to identify points to be inspected to check if they are within tolerance for certain surfaces. Moreover, this will increase the computational time and complexity of the algorithm.

4.4 Feature recognition in a point cloud

In inspection planning applications, the features in the CAD model are known. However, in other applications of model reconstruction such as reverse engineering and computer visualization and animation, there is a need to recognize features in the point cloud. A feature recognition algorithm based on Medial Axis Transform (MAT) has been developed to recognize a feature from its skeleton to achieve this goal. The geometric features are recognized by comparing their computed Medial Axis (Amenta *et al.* 2001) with a set of previously identified feature signatures. This section is based on the work of Mohib *et al.* (2006). The different geometric features that exist on each obtained non-planar point cloud are detected by studying each skeleton characteristic using a skeleton creation tool (Amenta *et al.* 2001). The proposed methodology for automatic recognition of geometric primitives (e.g. cylinder, sphere, cone and round slot) in a 3-D point cloud consists of the following steps:

- Planarity test is done to exclude planar surfaces since planes have no skeleton,
- The skeleton for each non-planar point cloud is analyzed to determine whether the surface represents a geometric primitive or a free form.

4.4.1 Planarity Test and Skeleton Computation for Non-Plane Surfaces

A planarity test is applied to each sub-point cloud to determine if it represents a plane and exclude all planar surfaces. The Planarity test is done by comparing the normal vectors of all points in the sub-point cloud. If the orientation of normal vectors at all points within the point cloud varies within a certain threshold, taking into account the noise level, then this point cloud forms a plane. If the sub-point cloud is not a plane, an algorithm is applied to find its skeleton using a tool called Powercrust (Amenta *et al.* 2001) developed at the University of Texas to compute the Medial Axis Transform (MAT) of a 3-D point cloud.

Figure 4.11 shows an example output from the Powercrust software for two connected cylindrical shapes. The input point cloud is presented in Figure 4.11 (a). Figure 4.11 (b) shows the Medial Axis obtained when applying the Powercrust tool to the point cloud. Figure 4.11 (c) presents how the Medial Axis can characterize the shape of the point cloud and how it can be affected by the existence of several features (the bend at the end of the two straight lines). This shows the importance of segmentation to decouple the effect of features on each other.

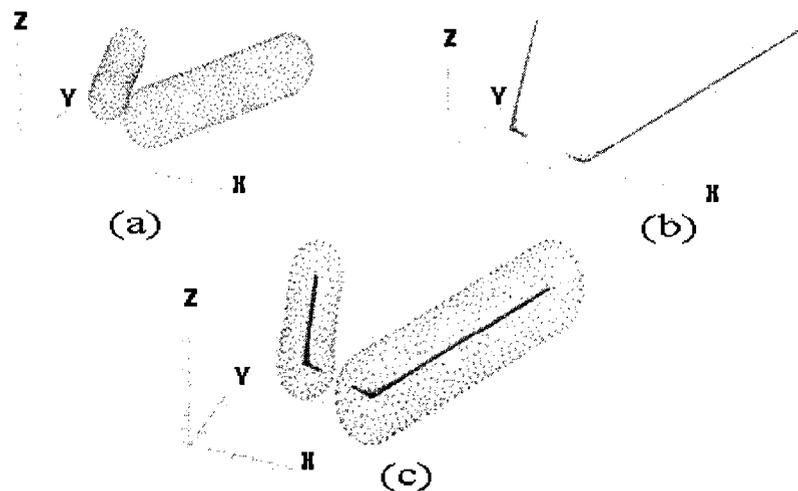


Figure (4.11) Medial Axis Transform Example using Powercrust.

One of the outputs of this program is an ASCII file in column format that represents (x,y,z) coordinates of the points of the MAT (which will be called the skeleton shape) and the 4th column represents the distance between a given point of the skeleton and the nearest point of the point cloud (which will be called the Distance parameter).

4.4.2 Geometric Primitive Recognition (GPR)

An engineered geometric primitive indicates a unique shape characteristic, which the desired part should possess, realized as a consequence of applying some manufacturing processes. Each geometric primitive has a signature, which can be represented by a shape skeleton computed using Powercrust. Figure 4.12 illustrates different examples of geometric primitive entities' signatures / Powercrust representation.

Point Cloud	PowerCrust		Entity
	Shape	Distance	
		Constant	Cylinder
		Increase along the shape	Cone
		Constant	Sphere
		Constant	Round Slot

Figure (4.12) Signatures of some geometric primitives.

A primitive such as a cylinder, cone, sphere or round slot can be described by two parameters: a) skeleton shape parameter, and b) Distance parameter. For example, the skeleton shape for a cylinder can be a fitted straight line where the Distance parameter all over the MAT is constant (in this case it is equal to the radius of the cylinder). A cone can be represented by its axis as the shape parameter where the Distance parameter increases (or decreases) linearly along this axis (a uniformly varying radius). For a sphere, the skeleton shape will be many points with almost the same coordinate (x,y,z) that can fit in a sphere with a threshold radius and the Distance parameter will be constant for all the points. A round slot entity can be defined by a fitted plane and a constant Distance parameter.

The Geometric Primitive Recognition (GPR) algorithm identifies the non-planar point cloud by a sequential comparison to the signatures listed before. Several fitting tests are made to determine the nature of the skeleton where the input is the points of the skeleton shape obtained by Powercrust. The flowchart in Figure 4 illustrates these steps.

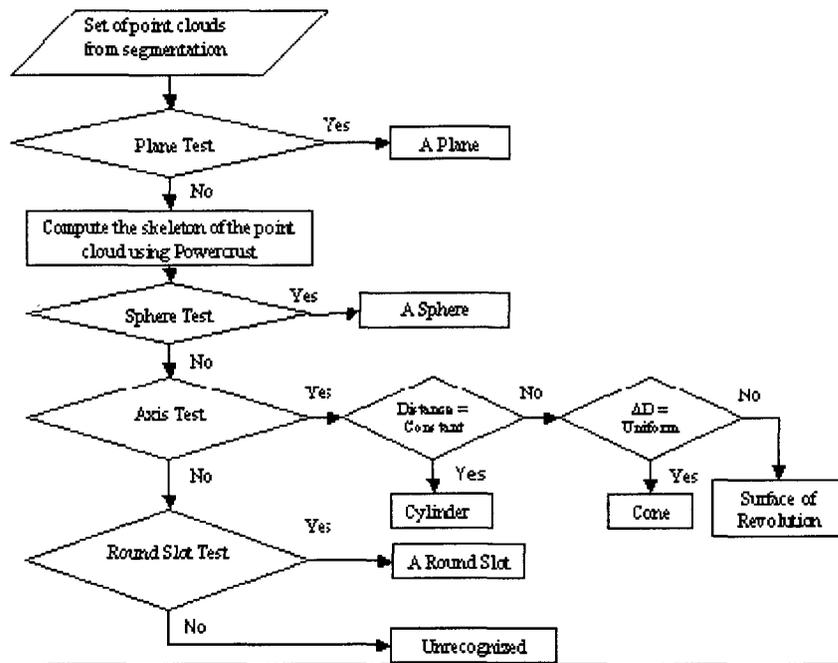


Figure (4.13) Geometric Primitives Recognition flowchart.

The first test is called the ‘Sphere Test’. If the points of the skeleton fit in a small sphere with a threshold radius and the 4th column is almost constant (within 5% error), the skeleton represents a sphere. Otherwise, an axis is fitted to these points and the standard deviation between these points and the fitted axis is computed, ‘Axis Test’.

If this standard deviation is larger than 5% of the average of the 4th column, then the skeleton is not a straight line and the next test ‘Round Slot Test’ is performed. If this skeleton represents a straight line the following test ‘Distance Variation Test’ is applied to the 4th column, which represents the Distance parameter.

There are three possibilities: a) If the Distance parameter is constant the skeleton represents a cylinder, b) If it varies uniformly the skeleton is a cone, else, c) the skeleton represents a surface of revolution. If the skeleton is not a straight line, a plane is fitted to the points and the standard deviation between the points and the plane is computed. If this value is smaller than 5% of the average of the fourth column, then the skeleton represents a round slot entity. Otherwise, none of the form features in the data-base is found by GPR

algorithm. The surface is then labeled unrecognized and it can be a free form surface or an undefined geometric feature not yet included in the database.

Now that all the primitives of the point cloud are classified, the next step consists of defining the parameters of the recognized primitives. For example, the cylinder is described by its axis as the least square line fitted to the points of the skeleton and its radius which is equal to the average of the Distance parameter. For a cone, its axis is the least square line fitted to the points of the skeleton and the slope is equal to the average of the following ratio:

$$(D_i - D_0) / L_i \quad (4.6)$$

Where D_i is the Distance parameter of the point P_i of the skeleton, D_0 is the Distance parameter of the point P_0 , which is the point at the beginning of the skeleton and L_i is the distance between P_0 and P_i as shown in Figure 4.14.

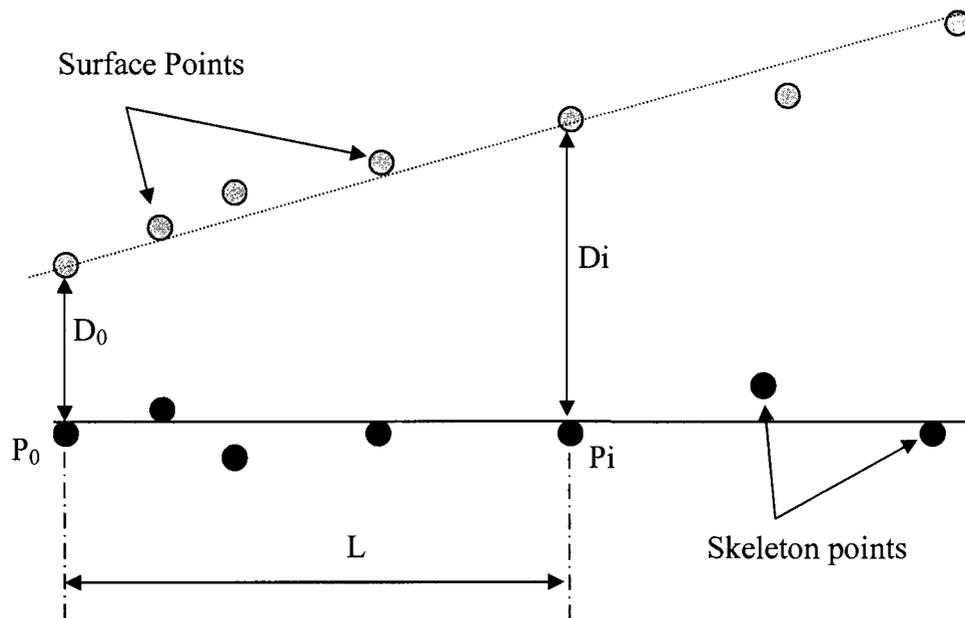


Figure (4.14) Cone slope calculation.

A sphere is defined by its center, which is equal to the center of mass of the skeleton and its radius is calculated as the average of the Distance parameter. A round slot is formed by two parallel planes and two half cylinders. The two side planes are parallel to the least square plane fitted to the skeleton and the distance between them and the

skeleton is equal to the average of the Distance parameter. The two half cylinders are fitted similar to a full cylinder as described above. Their axes are the least square lines fitted to the border of the skeleton as shown in Figure 4.15.

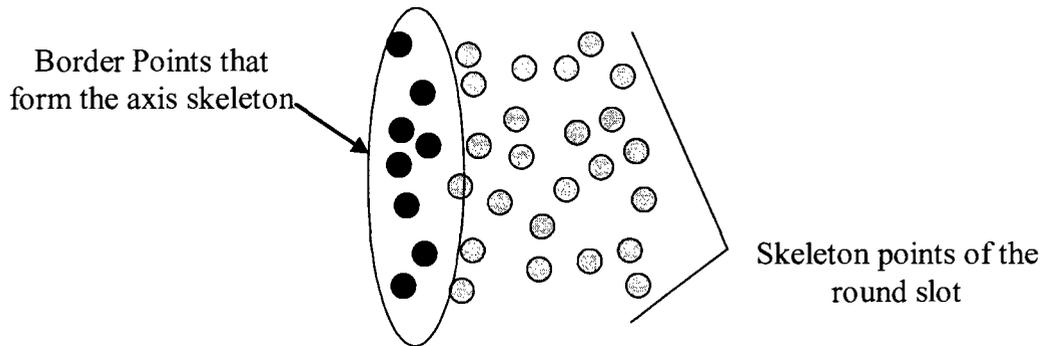


Figure (4.15) Axis of the round slot half-cylinder.

4.5 Summary and Conclusions

A novel 3D un-organized point cloud segmentation algorithm for inspection of complex mechanical parts has been developed and presented in this chapter. Segmentation is usually used as a pre-processing operation to prepare the point cloud for inspection. The results of the currently used segmentation algorithms are fitted surfaces, not the actual measured 3D coordinate points, which in turn reduces the accuracy of the inspection decisions. This chapter proposes an algorithm to segment a 3D un-structured point cloud into 3D sub-point clouds and assign each sub-point cloud to its corresponding feature on the CAD model. The developed region-based segmentation of the point cloud is based on curvature variation and results in a set of sub-point clouds for segmented features. The problem of over and under segmentation is avoided by implementing a two spheres neighborhood functions and by adding limitation to the propagation technique. A feature refinement module has been developed to merge sub-point clouds and produce meaningful components.

The main advantage of the proposed algorithm is that it deals directly with un-organized point data sets resulting from multiple scans without the need for meshing or preprocessing. Moreover, the output is the original measured data allocated to the

corresponding inspection feature. The proposed segmentation algorithm is developed for inspection purposes. However, it can be used in other applications such as reverse engineering, computer graphics and animations. For industrial parts having quadric surfaces, such as planes, cylinders and cones, this method can be applied efficiently regardless of the type of point. In this case, more attention should be given to borders (fuzzy – fillet (not sharp edge)) as well as variable densities and missing data.

It should also be noted that neighborhood calculation is the most time consuming step in the segmentation process and model reconstruction techniques. Efficient algorithm such as Grid-Octree, proposed by Woo *et al.* (2002), would be of great help to speed up the segmentation process as well as further research to tune up the selection and limits of parameters such as R_1 and R_2 .

The capability of the proposed segmentation algorithm, integrated with the latest in scanning devices and technologies, is illustrated using a point cloud having distinct characteristics. This new point set segmentation approach has potential applications in a whole spectrum of engineering problems with a major impact on inspection, reverse engineering and rapid product development.

5. TOLERANCE-BASED LOCALIZATION

Now that each point in the point cloud and the feature in the CAD model to which it belongs has been identified, it is required to check if the sub-point cloud, which includes all the measured point for this feature, satisfies the tolerance requirements. As explained in chapter one, the point cloud should be first aligned to the CAD model in order to calculate the deviation of the points from the CAD model; then the tolerance is verified. This chapter proposes a one step localization, to replace the traditional two-steps process, where the alignment is performed based on satisfying the geometric and dimensional tolerance requirements.

5.1 Localization and Tolerance Verification – An overview

Traditional inspection methods such as *Jackscrew*, *wobble-plate*, *fixed-plane* or *precision spindle* methods and their setups are chosen based on the part's physical shape (i.e. cylindrical or non-cylindrical) and its size and they are dedicated for simple parts with one or two features. In a complex part with many intricate and interacting features, it is not easy to isolate and inspect each feature according to the traditional methods. Higher technology tools such as CMM and laser scanner are, hence, used to measure the complex parts to inspect their features. Hence, a localization process is needed before verifying the tolerance. The localization process in inspection is the process of rotating and translating the Measurement Coordinate System (MCS) of the measured points to match the coordinate system of the design model (Design Coordinate System (DCS)) in order to obtain the best alignment. It is also referred to, in some literature, as the registration of design surfaces with measurement surfaces; however, this is not technically accurate since registration is the process of aligning two MCS of two paired point clouds to obtain a complete point cloud. The results of the localization process enable the inspector to compare and calculate the deviation between the measured and the CAD models and thus check if the measured surface is within tolerance (Li and Gu, 2004). The part is then accepted or rejected based on the tolerance verification results. Existing localization techniques have produced good results in inspecting sculptured

surfaces (i.e. airfoil (Pahk and Ahn, 1996), Masks (Besl and McKay, 1992, Pottman *et al.*, 2004), statues (Chen and Medioni, 1992, Masuda and Yokoya, 1995)) when a large amount of points are measured. These algorithms, when applied to mechanical parts with regular and free-form features associated with tolerances, need to be modified to account for tolerances at earlier stages. This, in turn, will speed up the inspection process while maintaining good and valid inspection decisions.

The Geometric and Dimensioning Tolerance (GD&T) verification process is used in industries to examine the conformity of the manufactured part specifications defined at the design stage. GD&T standards such as ASME Y14.5 have been used in practice for many years. In these standards, allowable variation of individual and related features is based on the “envelope principle”; that is, the entire surface of the part feature of interest must lie within two envelopes of the ideal shape and the amount of tolerance apart. This is also known as “Taylor’s principle”. These standards are easily applied using hard gage technology while there is little guidance for how tolerances should be verified using flexible and programmable technologies such as Coordinate Measuring Machines (CMM) and laser scanners. Ikonomov *et al.* (1995) introduced the virtual measuring gauge as a computerized replacement of a real hard gauge. Different approaches presented in the literature such as least mean squares and minimum zones were developed to verify the tolerances on manufactured parts using CMM data.

In this chapter, a new Iterative Minimum Zone (IMZ) localization algorithm has been developed to account for form tolerance in each iteration. Two main challenges are usually faced in the localization process: First, to find the correspondence relationship between the measured points and the design features; and then, to solve the 3D rigid body coordinates transformation of the MCS to align the two surfaces into a common coordinate system. Usually, features that need to be verified for tolerances can be expressed mathematically in terms of the DCS based on the information from the Computer Aided Design (CAD) model. In the developed iterative process, the correspondence between the measurement data and CAD model is estimated in each iteration by projecting points on the CAD model. The minimum zone is achieved by

rotating and translating the MCS of the measured point set to be placed within the mathematically defined tolerance zone volume. The objective is to find the rotation angles and the translation values of the measured points while finding a minimum zone value that satisfies the required tolerance value specified by the designer. Different types of form tolerance such as straightness of a median line, straightness of a surface line, flatness and cylindricity have been considered.

5.2 Related work

In previous decades, localization was achieved by presenting the part at a desired position and orientation, using special tools, fixtures or other part presentation/orientation devices totally dedicated to specific products. This process is usually costly, and time and effort are required to design and manufacture new fixtures. This type of localization is implemented prior to digitization (Pahk and Ahn, 1996, Wang and Lin, 1997). In recent practice, localization has been carried out by mathematically aligning the Measuring Coordinate System (MCS) to the Design Coordinate System (DCS) using features correspondence.

5.2.1 Correspondence Search

The correspondence search is a key issue in finding the best transformation. It can be established by selecting points or calculating distinct features of one object and locating the same ones in the other object (CAD model in case of localization). The search for correspondence can be characterized by a neighborhood search algorithm where the correspondence is determined by the closest Euclidean distance (Sahoo and Menq 1991, Besl and McKay 1992). Chen and Medioni (1992) formed correspondence by projecting selected (control) points onto the model in the direction of their normal vectors rather than selecting the closest points. These methods necessitate that the CAD model and the point cloud are in a good initial relative position and orientation to each other.

The change of geometric curvature and approximate normal vector of the surface formed by a point and its neighborhood can be also used to determine the possible correspondence (Bae and Lichti 2008). Sharp *et al.* (2002) proposed a method based on

Euclidean invariant features: curvature, second order moment and spherical harmonics. Patrikalakis and Bardis (1991) used pairs of the Gaussian and mean curvature at three different non-collinear locations to define correspondence. Li and Gu (2005) used a so called feature attributes (internal and external attributes) to establish correspondence such as area, Gaussian, mean and principle curvatures. Curvatures are computed from the first and second derivatives, and thus they are easily affected by the noise contained in the range image.

Since correspondence search algorithms generally have a large potential space to search through, several approaches to reduce the computational complexity have been used. The basic principle of the tree search algorithms, such as (k-dimensional binary search tree) KD-Tree (Zhang 1994) and (Balanced Box Decomposition) BBD-Tree (Arya *et al.* 1998), is that a node is taken from a data structure; then, its successors are examined and added to the data structure. The tree is then explored by manipulating the data structure in different orders for instance level by level.

5.2.2 Rigid Body Transformation

The other challenge in the localization process is the optimization of the 3D rigid body transformation variables. It has been formulated as the minimization of the sum of the squared distances between the measured points and the design model with respect to the transformation parameters. Traditional optimization techniques that use derivatives such as Newton (Chen and Medioni 1992, Jinkerson *et al* 1993), Newton Raphsone, (Sahoo and Menq 1991, Tucker and Kurfess 2003 a, b, Fan and Tsai 2001), Quasi-Newton (Patrikalakis and Bardis 1991) and Gauss-Newton (Tucker and Kurfess 2003 a and b) have been used to determine such transformation. Tucker and Kurfess (2006) proved that the Gauss-Newton or Gauss-Secant methods are better choices strictly for speed considerations. However, these techniques are usually computationally intensive. When a large number of measured points are involved, as in laser scanning, these algorithms are not desirable (Huang and Gu 1998).

An approach based on instantaneous kinematics and local quadratic approximants to the squared distance function has been proposed by Pottman *et al.* (2004). Evolutionary random based (Global/near optimal optimization) techniques such as genetic algorithms have been used to address the registration problem (Silva *et al.* 2007). They developed surface interpenetration measures to find correspondence between multiple views of range data. This simultaneous or global registration method is better than sequential pairwise registration of multi-view range data. These techniques can reach the optimal value but with extensive computations.

Arun *et al.* (1987) developed a closed form solution algorithm to find the least-squares solution based on the Single Value Decomposition (SVD) of a 3x3 matrix. The explicit solution obtained by SVD was employed to calculate the desired 3D rigid body transformation (Huang and Menq 2002). This method was shown to be powerful and computationally quick especially for problems higher than three dimensions. A quaternion-based algorithm has been derived for closed form solution of the least square matching problem (Horn 1987). Besl and McKay (1992) proposed the Iterative Closest Point (ICP) algorithm using Horn's algorithm. Each iteration of the algorithm contains the establishment of point correspondence and the rigid motion estimation using a unit quaternion. Unit quaternion and dual quaternion (Section 5.4) have been widely used for 3D rigid body transformation due to its fast convergence (Zhang 1994, Masuda and Yokoya 1995, Guehring 2001, Langis *et al.* 2001 and Shi *et al.* 2006). Once the correspondence between points is known, the transformation is done using the maximum eigenvalue of a 4x4 cross covariant matrix.

Rusinkiewicz and Levoy (2001) classified the variants of ICP algorithms and evaluated their effect on the speed with which the correct alignment was reached. They compared the convergence characteristics of several ICP variants based on the selection of points, matching points, weighting of pairs, rejecting pairs, error metric and minimization. Eggert *et al.* (1997) compared the accuracy, robustness, stability and efficiency of four major algorithms for 3D rigid body transformation. The SVD and the quaternion-based closed form solution algorithms achieved better results for large

number of points. (Eggert *et al.* 1997). Neither quaternion-based nor SVD methods have been, to the authors' knowledge, applied in tolerance verification algorithms. For 2D and 3D localization problems, the quaternion-based algorithms are preferred over SVD method since reflections are not desired (Besl and McKay 1992).

5.2.3 CMM-based Tolerance Verification techniques

Although many algorithms for the evaluation of tolerances exist, the least-squares method is commonly employed for data fitting in inspection using CMM due to its simplicity. The objective of this method is to minimize the sum of squares of deviation of measurement points from nominal features. However, the formulation used with the least-squares method is inaccurate for tolerance evaluation purposes. The resulting tolerance zone is not in conformance to the standard ASME Y14.5. Therefore, it results in the acceptance of out of tolerance parts and the rejection of parts that are within tolerance specifications. A variety of techniques have been developed which improve upon the least-squares method, many of which provide the minimum tolerance zone result (Dowling *et al.* 1997). However, these methods are mathematically complex and often computationally slow particularly for cases where a large number of data points are to be evaluated.

Form tolerances such as straightness, flatness and cylindricity for parts measured with a CMM can be verified using minimum zone approaches (Carr and Ferreira 1995 a, b, Gou *et al.* 1998, 1999, etc.), support vector regression (Malyscheff *et al.* 2002, Prakasvudhisarn *et al.* 2003), or approximations (Weber *et al.* 2002). All these algorithms are characterized by their approach to best fit the toleranced shape to the measured points.

Carr and Ferreira (1995a and 1995b) formulated the minimum zone problem as a non-linear optimization problem, which is subsequently solved using a sequence of linear programs and converges to the non-linear optimal solution. They addressed form tolerances such as straightness of a surface line and flatness (1995a), cylindricity and straightness of the median line (1995b).

A recent review of available inspection methods of free-form surfaces with and without datums may be found in Li and Gu (2004). Patrikalikas and Bardis (1991) extended the concept of the ball-offset regions to model position tolerance of curved surfaces for CAD/CAM information exchanges. The tolerance is verified after localization by intersection detection with the approximated surfaces. Barcenas and Griffin (2001) presented a statistically-based technique using jackknifing for geometric tolerance verification. Localization parameters were considered to ameliorate the fixturing errors between the CMM reference frame and the machine tool to optimize a generalized representation for super-quadrics. Prieto *et al.* (2002) checked the dimensional, angular, and geometric tolerances by aligning the point cloud to the CAD model of the part and then segmenting the point cloud into different surface patches by using the CAD model. Finally, the specified tolerances for free-form surfaces were verified by calculating the perpendicular distance between each 3D point and the NURBS surfaces. As a result, the tolerance verification for a surface is affected by the localization results of other surfaces.

In modern manufactured parts, different types of tolerances are usually required to be verified on one part. The produced part is digitized using contact or non-contact techniques; then the localization process aligns the point cloud to the CAD model, hence, these tolerance requirements are verified. Based on the previous review, tolerance specifications were not considered in current localization techniques as an optimality criterion. Failure to do so can lead to errors in the inspection results. This chapter presents an approach to satisfy this need by introducing an integrated approach that combines both localization processes with tolerance verification techniques.

5.3 Mathematical Formulation for Correspondence

In Computer Aided Design (CAD) modeling, the data of prismatic shapes are usually represented in the form of vertex/nodes and links. In this section, methods for computing the closest point on selected prismatic features such as lines, planes and cylinders to a given 3D point are described. Sample representations, in the generated STEP file format

using CATIA, of the simple prismatic part shown in Figure 5.1, are listed in Figure 5.2. These features are represented as point coordinates and vector directions. In this work, three prismatic features are considered; a line, a plane and a cylinder. The extracted data from the STEP (STandard for the Exchange of Product data) file will be used to estimate the closest point on the selected features to a given point.

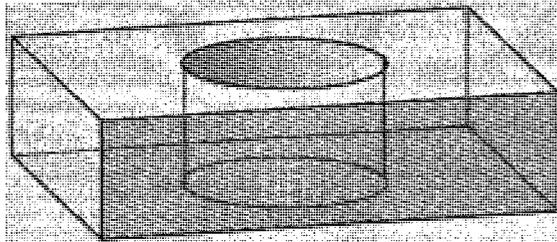


Figure (5.1) A simple prismatic shape.

```

Line:
#64=EDGE_CURVE("#61,#63,#59,.T.);
#59=LINE('Line',#56,#58);
#56=CARTESIAN_POINT('Line Origine',(-40.,30.,12.5));
#58=VECTOR('Line Direction',#57,1.);
#57=DIRECTION('Vector Direction',(0.,0.,1.));
#61=VERTEX_POINT('Vertex Point',#60);
#60=CARTESIAN_POINT('Cartesian Point',(-40.,30.,0.));
#63=VERTEX_POINT('Vertex Point',#62);
#62=CARTESIAN_POINT('Cartesian Point',(-40.,30.,25.));

Plane:
#55=PLANE('Plane',#54);
#54=AXIS2_PLACEMENT_3D('Plane Axis2P3D',#51,#52,#53);
#51=CARTESIAN_POINT('Axis2P3D Location',(-40.,30.,0.));
#52=DIRECTION('Axis2P3D Direction',(-1.,0.,0.));
#53=DIRECTION('Axis2P3D XDirection',(0.,-1.,0.));

Cylinder:
#178=CYLINDRICAL_SURFACE('generated cylinder',#177,17.5);
#177=AXIS2_PLACEMENT_3D('Cylinder Axis2P3D',#175,#176);
#175=CARTESIAN_POINT('Axis2P3D Location',(0.,0.,12.5));
#176=DIRECTION('Axis2P3D Direction',(0.,0.,1.));

```

Figure (5.2) Sample information in STEP file.

5.3.1 Closest point on a line to a point in space

A line is an ideal zero-width, infinitely long, perfectly straight curve containing an infinite number of points. This line can belong to a plane or it can be the resulting intersection of two planes of an inspected part. It can also be the center line of a cylinder or it can belong to the surface of the cylinder and parallel to the center line. In CAD models a 3D line is usually defined by two points (vertices); the starting point and the end point, such as the Cartesian points numbers 60 and 62 in Figure 5.2. In Euclidean

geometry, exactly one line can be found that passes through any two points. Consider two points: \vec{A} and \vec{B} , the line L that passes through them can be described by the point \vec{A} and a direction $(\vec{B} - \vec{A})$. Now, consider a point \vec{P}_i near the line L as shown in Figure 5.3. The closest point \vec{P}'_i on the line L to the point \vec{P}_i can be calculated as follows:

$$\vec{P}'_i = l' \frac{\vec{B} - \vec{A}}{|\vec{B} - \vec{A}|} + \vec{A} \quad (5.1)$$

$$\text{where } l' = (\vec{P}_i - \vec{A}) \cdot \frac{(\vec{B} - \vec{A})}{|\vec{B} - \vec{A}|}$$

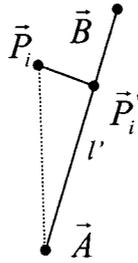


Figure (5.3) Closest point projected on a line to a point in space.

5.3.2 Closest point on a plane to a point in space

A plane can be described as a theoretical surface, which has infinite width and length, zero thickness, and zero curvature. A plane can be defined by either three points, a point and a direction or a direction and the distance from the origin. There is no unique way to find the closest point on a plane to a point outside the plane. This can be done by intersection of a line and a plane, using vectors or by minimization and partial derivatives. In this work, we use the vector approach to account for the position of the point with respect to the normal direction of the plane. To find the closest point \vec{P}'_i on the plane P to the point \vec{P}_i knowing a point \vec{A} and the normal direction \hat{T} to this plane where \hat{T} is a unit vector, it is required to find the distance between this point and the plane. This can be done by substituting the point in the equation of the plane. The question now is on which side of the plane the point lies (Figure 5.4). There are two cases: (a) the point is on the same side as the normal direction and (b) the point is on the

opposite side. By using vectorial equations this problem can be overcome by calculating the distance l' between the point \vec{P}_i and the plane \mathbf{P} as $l' = (\vec{P}_i - \vec{A}) \cdot \hat{T}$. The point is on the same side of the normal direction if the value of l' is positive and on the opposite side if the value is negative. The direction of the vector between the point \vec{P}_i and the closest point on the plane \vec{P}_i' is determined based on the sign of l' and can be calculated by $\vec{r}_{PP'} = l' \hat{T}$. If the value of l' is positive the vector $\vec{r}_{PP'}$ will be pointing out of the plane. If the value of l' is negative the vector $\vec{r}_{PP'}$ will be pointing to the plane. In both cases, the coordinates of point \vec{P}_i' can be easily obtained by subtracting the two vectors:

$$\vec{P}_i' = \vec{P}_i - \vec{r}_{PP'} \quad (5.2)$$

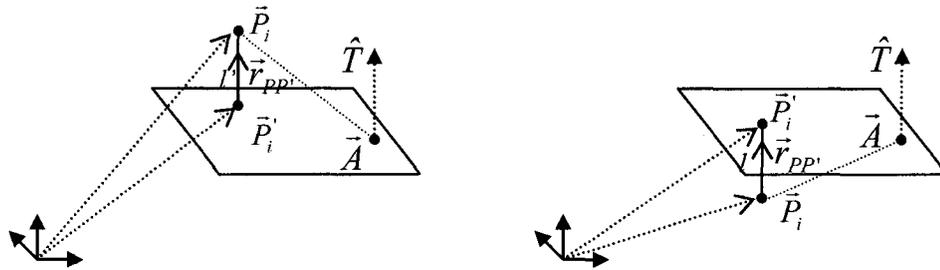


Figure (5.4) Closest point projected on a plane to a point in space.

5.3.3 Closest point on a cylinder to a point in space

A cylinder is one of the most basic curvilinear geometric shapes. It can be defined as the surface formed by the points at a fixed distance (radius) from a given straight line (the axis of the cylinder). It can be defined by the axis direction, a point on the axis, a radius and a length.

Consider a cylinder defined by a point \vec{A} , radius r and a direction \hat{T} where \hat{T} is a unit vector (Figure 5.5). To find the closest point \vec{P}_i' on the surface of the cylinder to the point \vec{P}_i , we need to find the point \vec{P}_{iL}' on the center line that is the closest to the point \vec{P}_i first. Hence, the closest point \vec{P}_i' on the surface of the cylinder to the point \vec{P}_i can be calculated by:

$$\vec{P}'_i = \vec{P}'_{iL} + r \frac{\vec{P}_i - \vec{P}'_{iL}}{|\vec{P}_i - \vec{P}'_{iL}|} \quad (5.3)$$

where $\vec{P}'_{iL} = \vec{A} + l' \hat{T}$ and $l' = (\vec{P}_i - \vec{A}) \cdot \hat{T}$

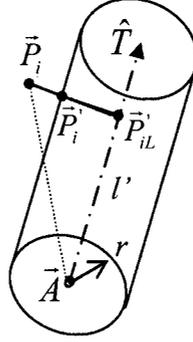


Figure (5.5) Closest point projected on a cylinder to a point in space.

5.4 Quaternion

Quaternion, in mathematics, is a non-commutative extension of complex numbers in the form of $\vec{q} = [w + xi + yj + zk]$ which is often considered as a scalar value w and a 3D vector $(x, y, z)^T$. A unit quaternion \vec{q} , used to perform a rigid body rotation, can be represented by a 4-D vector $\vec{q} = (q_1, q_2, q_3, q_4)^T$ where q_1 presents the scalar value w and $q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1$. The relationship between quaternion and the axis-and-angle representation can be defined to specify a rotation of angle θ around a unit vector \hat{u} (Wheeler and Ikeuchi, 1995). The unit quaternion \vec{q} can be easily calculated using the relations:

$$q_1 = \cos \frac{\theta}{2} \quad (5.4)$$

$$(q_2, q_3, q_4) = \left(\sin \frac{\theta}{2} \right) (u_x, u_y, u_z) \quad (5.5)$$

The formula for the corresponding orthogonal (Euclidean) 3x3 rotation matrix R generated by a unit quaternion \vec{q} can be defined by

$$R = \begin{bmatrix} 1 - 2(q_3^2 + q_4^2) & 2(q_2q_3 - q_1q_4) & 2(q_2q_4 - q_1q_3) \\ 2(q_2q_3 - q_1q_4) & 1 - 2(q_2^2 + q_4^2) & 2(q_3q_4 - q_1q_2) \\ 2(q_2q_4 - q_1q_3) & 2(q_3q_4 - q_1q_2) & 1 - 2(q_2^2 + q_3^2) \end{bmatrix} \quad (5.6)$$

The rotation matrix R is used to apply the 3D rigid body transformation in the developed localization algorithm.

5.5 Minimum Tolerance-Zone Models Formulations Applied to Form Tolerance

Form tolerances such as straightness, flatness, circularity (roundness) and cylindricity are applicable to individual features or elements of single features (ASME Y14.5 1994). A domain of possible tolerance zones can be described for form tolerance specifications addressed in this dissertation as well as other toleranced features. The mathematical formulae prescribed by the ASME Y14.5.1 (ASME Y14.5.1 1994) for some form tolerance zones are presented in Appendix A. These zones are shaped, located and oriented in space by the value of the tolerance size t , and two representative vectors; position vector \vec{A} (3D point) and a zone direction vector \hat{T} (unit vector), which are different for various form tolerances. In this section, these tolerance zones MZ_t , defined by the set of points $\{P\}$, which need to be minimized, are derived based on their shape, size, location and orientation.

Unlike traditional techniques, which were used to fit the surface over the inspected points, the developed algorithm rotates the measured data to ensure minimum zone fit of the inspected points. In this section, based on the ASME standards, a formula is derived to define the zone function MZ_t to be minimized. We focus here on four types of form tolerance: straightness of a median line, straightness of surface line, flatness and cylindricity.

5.5.1 Straightness of a median line

For a median line, the straightness tolerance zone to be minimized is a cylindrical volume zone. This zone should include all the derived measured points \vec{P}_i . To minimize this zone, it is required to rotate and translate these points and then to calculate the maximum radial distance r_{max} between all the points \vec{P}_i and the median line. The median line is defined from the CAD model by a point \vec{A} and a direction \hat{T} (unit vector). In

Figure 5.6 to 5.9, open circles denote measurement points while filled circles indicate CAD point. The dashed lines represent the cylindrical enclosing minimum zone where r_{max} is the radius. For a given orientation of the measured points, the radial distance r_i can be calculated as the magnitude of the cross product of \hat{T} and the relative vector $(\vec{P}_i - \vec{A})$.

$$r_i = \left| \hat{T} \times (\vec{P}_i - \vec{A}) \right| \quad \forall \vec{P}_i \in \{P\} \quad (5.7)$$

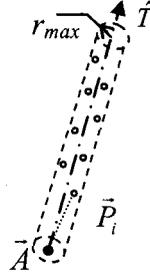


Figure (5.6) Minimum Zone for Straightness of Median Line.

Hence, the value of the maximum radius r_{max} of the minimum enclosing cylindrical zone can be obtained by

$$MZ_{St.} = r_{max} = \max(r_i) \quad \forall r_i \in \{R\} \quad (5.8)$$

where R is the set of all the values of r_i .

5.5.2 Straightness of a surface line

For surface line elements, the tolerance zone to be minimized is the area formed between two parallel lines in the cutting plane defining the line element. All measured points should lie between these two lines (Figure 5.7). The distance d_i is the distance between any measured point \vec{P}_i and a nominal line that divides this area in half. The nominal line is represented by the point \vec{A} and the direction unit vector \hat{T} . The distance d_i can be calculated by:

$$d_i = \left| \hat{T} \times (\vec{P}_i - \vec{A}) \right| \quad \forall \vec{P}_i \in \{P\} \quad (5.9)$$

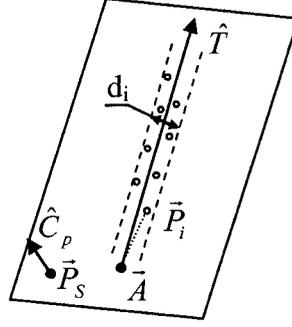


Figure (5.7) Minimum Zone for Straightness of Surface Line.

For straightness of a surface line, there are three constraints to make sure that the measured points lie on the cutting plane (Appendix A). Consider \hat{C}_p the unit normal vector to the cutting plane defined as being parallel to the cross product of the cutting vector and the mating surface normal at \vec{P}_s . In the proposed approach, the cutting plane is known from the CAD model; hence, constraints A.4 and A.5 are satisfied by default. Only the first constraint is applicable because we only rotate the measured points. The point \vec{P}_s is not needed and the point \vec{A} can be used instead as shown in equation 5.10. However, this constraint is hard to satisfy due to the inaccuracy of measured points. Hence, the error generated due to the un-satisfied constraint has to be minimized, or the points that do not satisfy this condition are considered outliers and rejected. This error can be calculated by:

$$e = \hat{C}_p \bullet (\vec{P}_i - \vec{A}) \quad \forall \vec{P}_i \in \{P\} \quad (5.10)$$

To minimize the surface line straightness tolerance zone area, it is required to minimize the maximum value of the distance function d_i on both sides. Since the nominal line lies in the middle, then twice the value of $(\max(d_i))$ should be minimized by definition. Hence the minimum zone can be calculated by

$$MZ_{st} = d = 2(\max(d_i)) \quad \forall d_i \in \{D\} \quad (5.11)$$

where $\{D\}$ is the set of all the values of d_i .

5.5.3 Flatness

The flatness tolerance zone to be minimized is the volume formed between two parallel planes separated by the tolerance size such as the two dashed-line planes in Figure 5.8. All measured points should lie between these two Planes. The nominal plane obtained from the CAD model is the plane parallel to both planes and divides the distance between them in half such as the solid-line plane (Figure 5.8). This nominal plane is defined by a point \vec{A} and the normal vector \hat{T} (unit vector) perpendicular to the plane. The distance d_i is the distance between any measured point \vec{P}_i and a nominal plane and can be calculated by the absolute value of the dot product of \hat{T} and the vector $(\vec{P}_i - \vec{A})$.

$$d_i = \left| \hat{T} \bullet (\vec{P}_i - \vec{A}) \right| \quad \forall \vec{P}_i \in \{P\} \quad (5.12)$$

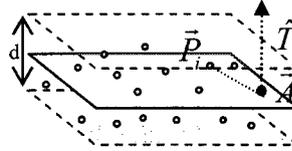


Figure (5.8) Minimum Zone for Flatness Tolerance.

To minimize the flatness tolerance zone volume between the two planes, it is required to minimize the maximum value of the distance function d_i on both sides of the plane. Since the nominal plane should lie in the middle of the two parallel planes, it is needed to minimize twice the maximum value of the distance function d_i . (i.e. to minimize the value of $\max(d_i)$);

$$MZ_{fl} = d = 2(\max(d_i)) \quad \forall d_i \in \{D\} \quad (5.13)$$

where D is the set of all the values of d_i .

5.5.4 Cylindricity

The cylindricity tolerance zone to be minimized is the volume formed between two coaxial cylinders separated by the tolerance size. All measured points \vec{P}_i should lie between these two cylinders, which satisfies the condition A.7. Consider the nominal cylinder from the CAD model defined by a point \vec{A} and the normal vector \hat{T} (unit vector) axis of the cylinder (Figure 5.9). r_i is the value of the distance between the point \vec{P}_i and

the axis \hat{T} (unit vector). Then r_i can be calculated as the absolute value of the cross product of the unit vector of the axis \hat{T} and the relative vector $(\vec{P}_i - \vec{A})$.

$$r_i = \left| (\hat{T} \times (\vec{P}_i - \vec{A})) \right| \quad \forall \vec{P}_i \in \{P\} \quad (5.14)$$

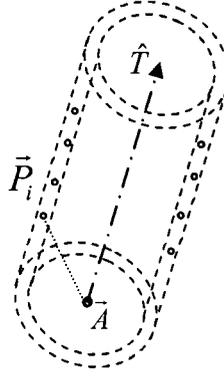


Figure (5.9) Minimum Zone for Cylindricity Tolerance.

To satisfy the tolerance requirements, this value should be between the minimum and the maximum value of the tolerance specified by the designer. The minimum cylindrical zone with axis \hat{T} that includes all points \vec{P}_i can be stated as

$$MZ_{Cyl} = \delta r = \max(r_i) - \min(r_i) \quad (5.15)$$

5.6 Proposed Iterative Minimum Zone Localization Algorithm

Now that the methods for computing the closest point on cited geometric shapes to a given point from the point cloud and the methods for computing the minimum zone have been derived, the Iterative Minimum Zone (IMZ) localization algorithm will be described. It is clear that computing the formulated zones MZ_t in equations (5.8, 5.11, 5.13 and 5.15) to be minimized is inherently a nonlinear optimization problem. In earlier works regarding form tolerance verification techniques using the minimum zone approach, the idea was to rotate the zone direction vector and change the zone locating position vector to minimize the tolerance zone, while keeping the measured points fixed. The derived model was, then, linearized and simplified to be solved (Carr and Ferreira 1995 a and b).

From a localization perspective, the CAD coordinate system is fixed while the measured coordinate system is rotated in six degrees of freedom to minimize the mean-square distance metric between matching points from the measurement and CAD model. All localization techniques are based on finding correspondence between points and minimizing the difference between them. These works stem from the registration techniques that merge different point clouds into one. However, in inspection, the objective is different. The inspector is requested to assess the conformance to the design requirement(s) by verifying the specified tolerances. Hence, in the developed localization process, the objective is to minimize the zone within which all measured points should lie, formulated as a constraint satisfaction problem. The general localization steps can be simply described as

- Start with a good initial alignment
- Find correspondence between points
- Apply rotation and translation
- Iteratively refine by pairing a number of points on one surface with the closest points on the other surface
- Globally minimize the sum of squared distances between point sets over all six degrees of freedom

The decision variable, in localization, is usually a vector of the six degrees of freedom variables $[\theta_x, \theta_y, \theta_z, \delta_x, \delta_y, \delta_z]$ that represents the 3D rigid body transformation of the measured points to be aligned with the CAD model, where $(\theta_x, \theta_y, \theta_z)$ are the rotation angles around axis (x, y, z) respectively to form the rotation matrix R and $(\delta_x, \delta_y, \delta_z)$ are the translation values along these axes to form the translation vector L . The function to be minimized is the sum of the squared error between the transformed measured points and the corresponding CAD model point set as shown in the function expressed in equation 5.16:

$$f(R, L) = \frac{1}{N_p} \sum_{i=1}^{N_p} \|\bar{x}_i - R \times \bar{p}_i - L\| \quad (5.16)$$

Where x_i are the CAD model points and \vec{P}_i are the measured points. N_p is the number of corresponding points from both sets.

In this work, the unit rotation quaternion $\vec{q}_R = (q_1, q_2, q_3, q_4)^T$, described in section 5.4, is chosen to form the rotation matrix R as in Besl and McKay (1992). This allows obtaining a closed-form solution to verify the form tolerance. The quaternion representation makes the minimization of the sum of the squared error equation equivalent to the maximization of a quadratic form of a unit quaternion. Let the complete localization vector \vec{q} be the decision variables to be optimized and denoted by $\vec{q} = \begin{bmatrix} \vec{q}_R \\ \vec{q}_t \end{bmatrix}$ where $\vec{q}_t = (q_5, q_6, q_7)^T$ is the translation vector. The flowchart in Figure 5.10 details the steps of the developed algorithm.

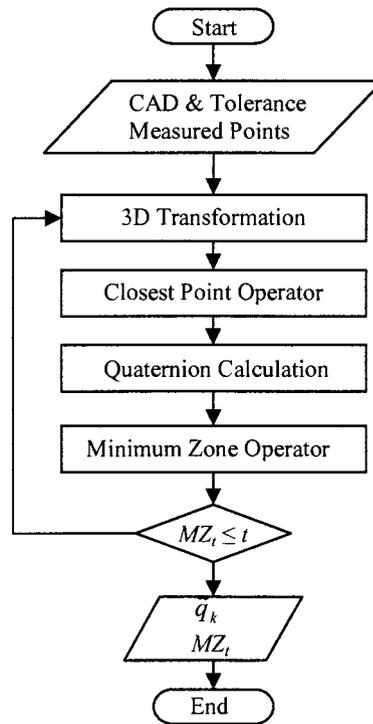


Figure (5.10) Iterative Minimum Zone Localization flowchart.

Four sub-algorithms were developed: 1) 3D rigid body transformation, 2) closest point, 3) quaternion calculation and 4) minimum zone. The inputs to the developed

algorithm are 1) the measured point cloud $\{P_0\}$, 2) the prismatic feature information extracted from the CAD model $\{X\}$ such as coordinate point \bar{A} , directional vectors (\hat{T}, \hat{C}_p) and scalar values (r, l) that represent the median shape of the form tolerance, 3) the type of the form tolerance F to be verified and the tolerance value t . The number of points in the point cloud will be denoted by N_P which is the same as the number of the corresponding points.

5.6.1 Algorithms Formulation

For a given value of the quaternion \bar{q}_k , the rotation matrix R can be calculated using equation (5.4) and applied to the original set of points $\{P_0\}$ to obtain the rotated and translated point set $\{P_k\}$ as follows

$$\bar{p}_k = R \times \bar{p}_i + q(5:7) \quad \forall \bar{P}_i \in \{P_0\} \quad (5.17)$$

Let P' denote the resulting set of the closest points on the CAD model to the set of the transformed set points $\{P_k\}$ and let C be the closest point module. The values of P' depend on the form tolerance type F and can be computed as described in section 3.1. Hence the set of the corresponding points P' from the CAD model can be calculated by:

$$P' = C(\{P_k\}, \{X\}, F). \quad (5.18)$$

The optimal rotation matrix is given by the unit quaternion that is the unit eigenvector corresponding to the maximum eigenvalue of the 4x4 matrix $Q(\Sigma_{pp'})$ whose components are generated from the cross covariance matrix between the given pairs of point sets as shown below.

$$Q(\Sigma_{pp'}) = \begin{bmatrix} tr(\Sigma_{pp'}) & \Delta^T \\ \Delta & \Sigma_{pp'} + \Sigma_{pp'}^T - tr(\Sigma_{pp'})I_3 \end{bmatrix} \xrightarrow{\text{eigenvector}(\max(\text{eigenvalue}))} \bar{q}_R = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \quad (5.19)$$

where $\Sigma_{pp'} = \frac{1}{n} \sum_{i=1}^n (\bar{p}_i - \bar{\mu}_p)(\bar{x}_i - \bar{\mu}_{p'})^T$ is the cross covariant matrix between the measured point set and the corresponding projected points on the CAD model and $\bar{\mu}_p = \frac{1}{n} \sum_{i=1}^n \bar{p}_i$ and $\bar{\mu}_x = \frac{1}{n} \sum_{i=1}^n \bar{p}'_i$ are the corresponding means of the point sets. Δ is a column vector composed of the cyclic components of the matrix resulting from subtracting the transposed cross covariant matrix from itself, and denoted by $\Delta = [A_{23} \ A_{31} \ A_{12}]$ where $A_{ij} = (\Sigma_{pp'} - \Sigma_{pp'}^T)_{ij}$. $tr(\Sigma_{pp'})$ is the trace of the matrix $\Sigma_{pp'}$ (i.e. the sum of the elements on the main diagonal) and I_3 is the 3x3 identity matrix.

The optimal translation vector is obtained by the difference between the transformed means of the measured point set and the CAD one.

$$\bar{q}_T = \bar{\mu}_{p'} - R(\bar{q}_R)\bar{\mu}_p \longrightarrow \bar{q}_T = \begin{bmatrix} q_5 \\ q_6 \\ q_7 \end{bmatrix} \quad (5.20)$$

The value of the objective function is the minimum zone value MZ_t calculated for a given position and orientation of the transformed point set $\{P_k\}$. The computation of the minimum zone differs with different form tolerances as described earlier and can be computed as presented in section 4. For generality, the objective function module will be denoted by MZ where

$$MZ_t = MZ(\{P_k\}, \{X\}, F, t) \quad (5.21)$$

5.6.2 Iteration Loops

Let the iteration number be denoted by k and set $k = 0$. The iteration is initialized by setting $\{P_0\} = \{P\}$ and $\bar{q} = (1,0,0,0,0,0)^T$. The localization vectors are defined relative to the initial set P_0 so that the final localization step represents the complete transformation. Hence, the resultant is the 3D rigid body transformation between the original point cloud and the final iteration. The steps accomplished in each iteration can be described as follows:

1. Compute the closest point set P' using the closest point operator C.
2. Apply the 3D rigid body transformation to $\{P\}$ to obtain $\{P_k\}$.

3. Compute the minimum zone MZ_t using the minimum zone operator MZ .
4. Compute the quaternion vector \bar{q}_k as described above.
5. There are two conditions to terminate the iterations:
 - a. when the value of MZ_t is less than the value of t which indicates that the part is accepted (constraint satisfaction problem).
 - b. when the value of MZ_t doesn't change for five successive iterations (optimization problem).

Steps 1 to 5 are repeated until convergence within the tolerance zone t is satisfied, otherwise the measured points are out of the tolerance limit and the part is rejected; i.e. no orientation or position for the measured points exist where they can be all included in the specified tolerance zone.

5.7 Experimental Numerical Verification

The developed localization algorithm was tested and validated using benchmark data sets with known value of deviation that were published in previous papers (Carr and Ferreira 1995 a and b). The data sets are listed in Appendix B. The CAD information needed by the proposed Iterative Minimum Zone IMZ algorithm was extracted and presented in Tables 5.1-5.4.

Table (5.1) CAD data for straightness of median line

Set	Tx	Ty	Tz	Ax	Ay	Az
1	1	0	0	0	0	0
2	1	0	0	0	0	0

Table (5.2) CAD data for straightness of surface line

Set	Tx	Ty	Tz	Ax	Ay	Az	Cpx	Cpy	Cpz
1	1	0	0	-50	0.003	0	0	0	1
2	-0.9748	-0.2229	0	4	2.931	0	0	0	1
3	-1	0	0	2.2261	-0.0025	0	0	0	1
4	-0.9459	0.2782	0.1669	0.4896	0.4519	0.426775	0	0.5145	0.8575

Table (5.3) CAD data for flatness of a plane

Set	Tx	Ty	Tz	Ax	Ay	Az
1	0	0	1	0	0	0.005
2	0	0	1	0	0	0.002
3	0	0	1	1.6726	1.2968	0.0014
4	0	0	1	0	600	0.00438

Table (5.4) CAD data for cylindricity

Set	Tx	Ty	Tz	Ax	Ay	Az	Radius	Length
1	0	0	1	0	0	0	30.0005	60
2	0	0	1	0	0	0	60	30
3	1/√3	1/√3	1/√3	1	2	-3	50	100

In the developed IMZ algorithm the CAD model data set is fixed and the point cloud is rotated and translated. The resulting quaternion vector \vec{q} that represents the 3D rigid body transformation and the minimum zone MZ_t value are presented in Tables 5.5-5.8. Also, Tables 5.5-5.8 also present the minimum zone results obtained by Carr and Ferreira as well as nominal values for certain sets.

Table (5.5) IMZ algorithm and benchmark results for straightness of median line

set	\vec{q}	New MZ_t	Published Results	Differences %
1	[-0.7706, 0.6374, 0.0001, -0.0, -0.0, -0.0026, 0.0005]	0.0084	0.0151364 0.01569	-0.44505 -0.46463
2	[0.0918, -0.9958, 0, 0.0002, -0.0, -0.1387, 0.4776]	0.7239	0.9001114 0.9839	-0.195766213 -0.264254497

Table (5.6) IMZ algorithm and benchmark results for straightness of surface line

Set	\bar{q}	New MZ_t	Published Results	Differences %
1	[1, 0, 0, 0.0, -0.0, -0.0026, 0]	0.0036	0.002666	0.350338
2	[-1, 0, 0, 0.0066, 0.0916, -0.4005, 0]	1.1035	0.8578577	0.286343877
			0.8968	0.230486173
			0.88	0.253977273
3	[-1, 0, 0, 0.0001, -0.0, 0.0011, 0]	0.0017	0.0013110	0.296720061
			0.0013110	0.372997712
4	[0.4853, -0.7924, -0.2063, -0.3067, -0.0074, 0.2980, 0.5012]	0.0000012854	0.0000011349	0.1335097

Table (5.7) IMZ algorithm and benchmark results for flatness of a plane

Set	\bar{q}	New MZ_t	Published Results	Differences %
1	[1, 0, 0, 0, 0, 0, 0.0009]	0.0028	0.0025	0.12
2	[1, 0, 0, 0, 0, 0, -0.0032]	0.0059	0.0048636	0.213093182
			0.004864	0.212993421
3	[-1, -0.00002, 0.00009, 0, 0, 0, 0.00045]	0.00271	0.0026273	0.031477182
			0.002817	-0.037983671
4	[1, 0, 0, 0, 0, 0, 0.0047]	0.00923	0.00876	0.053652968

Table (5.8) IMZ algorithm and benchmark results for cylindricity

Set	\bar{q}	New MZ_t	Published Results	Differences %
1	[-1, 0, 0, 0, 0, 0, 0]	0.001	0.001	0
2	[-1, 0.0000076, -0.00002, 0, 0.001116, -0.00374, 0]	0.1966	0.19667	-0.000355926
			0.21197	-0.072510261
			0.18396	0.068710589
3	[-1, -0.000002, 0.000003, 0, -0.000101, -0.000062, 0.000163]	0.00983	0.00983	0
			0.01037	0.00054
			0.00941	0.00042

5.8 Discussions

The developed Iterative Minimum Zone (IMZ) localization algorithm was applied and compared to sample data from the literature and the results prove its validity to verify form tolerances such as straightness, flatness and cylindricity. Both; the new IMZ localization algorithm and the published solutions satisfy the ASME mathematical definitions of the tolerance zones presented in Appendix A. The developed algorithm applies the minimum zone concept for form verification with less computation. By comparing the obtained minimum zone values from the IMZ localization algorithm to previous results for these data sets, it shows that the results from the above formulation are within 0.5 % differences from previously published results. The reason for these differences is due to the different approaches used to solve the problem; in this work, the problem is solved as a constraint satisfaction problem not as an optimization problem, which reduces computation time and produces “within tolerance” results (i.e. accept/reject decision) that are acceptable from inspection perspective. The minimum tolerance zone used to be estimated by any optimization method that uses explicit vector gradients, which would require at least seven operators’ evaluations for each numerical gradient evaluation. Any optimization method that doesn’t use explicit vector gradient estimates requires literally hundreds to tens of thousands of Closest Point C and Minimum Zone MZ_t sub-algorithms calculations. This extra effort to reach the optimum value is not needed to reach the decision of accepting or rejecting a part. In this algorithm, the approximate minimum zone is attained by applying a constraint satisfaction problem modeled as a closed form solution problem which reduces the computation effort. The IMZ localization algorithm allows us to move from a given starting point to local minima relatively quickly in comparison with other possible alternatives. Each iteration requires only one evaluation of the Closest Point C and Minimum zone MZ_t operators: the most expensive computation.

Moreover, from localization perspective, the inspection decision is reached in fewer steps. Inspectors used to align the measured points with the CAD model in one step and verify tolerance in the next step. This algorithm verifies tolerance and applies localization in one step.

The developed algorithm needs neither derivatives nor preprocessing of 3D data except for the removal of outliers. It also doesn't need to select or sample points from the CAD model for correspondence. However, the 3D point cloud should be first filtered and segmented so that each sub-point cloud represents a feature.

The developed algorithm was applied to four types of form tolerances for algorithm verification purpose. However it can be applied to other types of tolerance as long as they can be expressed in a mathematical form. It can be also applied to more complex part with different type of features such as dies and pumps.

One of the disadvantages of the proposed method is that it uses mathematical representation of the CAD model without taking into consideration the boundary of the surface. This leads to insufficient translation when applied to single type of tolerances. Consider cylindricity for example, the length of the cylinder extracted from the CAD model is not used in calculation. The cylinder considered has infinite length, which is the case for tolerance verification standards. Hence, the excess adjustment in translation value \bar{q}_r to reach a complete matching of the whole measured points with the CAD model is not necessary for tolerance verifications.

It should also be noted that the starting position for rotation and translation of the measured point cloud is a key issue to be able to reach a valid answer since projection is used to determine the correspondence between the measured points and the CAD model. However, the developed algorithm is capable of computing the minimum zone value as long as the point set is close to the CAD model but with more iterations. The closed form solution drive the localization vector \bar{q} , in each iteration, to be defined relative to the initial set $\{P_0\}$ so that the final localization operation represents the complete closed form transformation.

5.9 Summary

This chapter presented a novel localization technique that accounts for tolerances limitations specified by the designer. The developed algorithm incorporates the minimum tolerance zone estimate into the traditional localization technique as an optimality criterion. It was formulated as a closed form constraint satisfaction problem using unit quaternion. Acceptable results are obtained in shorter time than traditional tolerance verification techniques that seek the optimal value of the minimum zone, which is not needed for inspection decisions. Although, localization techniques differs from tolerance verification techniques, the proposed algorithm was implemented to different types of form tolerances and verified through benchmark examples with known tolerance values from the literature. The results show that the developed IMZ localization algorithm has the potential to be applicable to other types of geometric tolerances.

6. CONCLUSIONS

6.1 Conclusions

Globalization, increased products customization and the quest for competitive advantages are but a few of the many challenges manufacturing enterprises are increasingly facing now and in the future. The presented work addresses the challenges arising due to the increased complexity of parts, their frequent variation due to customization and introduction of new technologies as well as customers' insistence on having high quality products. This environment created a need for efficient dimensional and geometric verification methods that can be easily and quickly adapted to the changing features of parts and products. Such methods and tools would be very useful at the early product development and certification stage as well as quality assurance programs for manufactured products. The complexity of some of the newly developed products featuring intricate sculptured geometries, high accuracy and tight tolerances call for using advanced measurement technologies and require the use of more than one measuring method to help make the correct accept/reject decisions. The frequent changes in products require and justify the automation of several aspects of this quality verification function.

Quality assurance depends on a sound and coherent inspection system in terms of planning, execution, manipulation and analysis of the results which should be based on the designer requirements and specifications. In this thesis, a hybrid inspection system has been proposed and developed to improve the completeness of the acquired data, keep its accuracy during its manipulation and improve the accuracy of the inspection decisions taken. Under this proposed hybrid inspection methodology, the following conclusions could be made for the developed planning, segmentation and localization models and methods.

Developed Planning Method

1. Tactile sensors with their low digitization speed are not best suited for the current manufacturing environment with its more and more complex part

designs. Non-contact sensors on the other hand are not as accurate as contact scanners and fall short to reach occluded or shadowed areas in the measured part. The developed hybrid (contact/non-contact) inspection planning model overcomes the shortcomings of both techniques; namely touch probe and strip type laser scanner.

2. Several attempts were found in the literature to build a feature classification that defines the relationships between the different types of features. One of the main contributions in this planning method is the development of a detailed and rich feature taxonomy in the form of a 3D decision matrix that facilitate the sensor selection process based on formulated knowledge rules listed in each cell of the decision matrix. These rules differ from a cell to the other.
3. The knowledge-based system was not just formulated based on the physical description of the inspection system, but also on the part attributes. The three axis of the feature taxonomy matrix, which determines the rules set, are the inspection features, geometric features and the manufacturing features of the inspected part. The formulated knowledge rule system is function of three independent factors: the type of tolerance, the feature dimensions and the occlusion and accessibility calculations. All the information that can affect the sensor selection is included in the developed features taxonomy to better assist in determining the most appropriate sensor for the inspected feature.
4. Although the hybrid (contact/non-contact) inspection concept was proposed in the literature, a detailed planning procedure to combine two different sensors was missing. A new mathematical model for planning the integration of different types of sensors which includes the effort to switch between inspection operations has been developed.
5. Formulating the changes in sensors, sensors' orientations, and part's orientations with the Traveling Salesperson Problem (TSP) offers localized optimal plans and hence minimizes the effort to switch between the inspection tasks.

6. The applicability of the developed overall hybrid inspection planning methodology to complex mechanical parts is demonstrated using an industrial case study of a water pump housing. Laser inspection integrated with tactile sensing is proven successful in digitizing parts, which are difficult to be completely digitized using a single type of sensor.
7. Results demonstrated the capacity of the developed feature-based inspection planner for hybrid sensing systems to: a) plan the inspection of prismatic, free form and complex mechanical parts such as water pumps, dies and moulds using combined tactile and non-contact sensors , b) complete the acquisition of required data for parts with accessibility and occlusion problems for one or both sensors, and c) improve the overall inspection process by arriving at accurate inspection decisions of accepting/rejecting parts. It also illustrated the effectiveness and benefits of using the proposed TSP sequence planning method to optimize the inspection process and to minimize non-digitization effort and time.
8. Results also show that the sensor changes contributed the most to the value of the formulated time objective function while the effect of the rapid tool traverse time contribution was the most negligible.

Developed Segmentation Method

9. A segmentation process that divides the measured point cloud into meaningful segments (sub-point clouds) corresponding to the features to be inspected, is needed to perform a tolerance-based localization process. Current segmentation algorithms deal with mesh representations and the associated loss of accuracy compared to the one of the original measured points. A segmentation algorithm that deals directly with the measured point cloud and yet produces the same measured points, but divided based on the inspected features information from the CAD model, was developed to accomplish the previously mentioned goal.
10. One of the main contributions in this algorithm is its ability to manipulate *un-organized* point cloud data obtained from different types of sensors with

different orientation without the need for a tessellated mesh or any other preprocessing operations such as filtering and removing of redundant points and, at the same time, keeps the original measured points all the way through the segmentation process.

11. The main advantage of the developed algorithm is that the output data allocated to the corresponding inspection feature keep the accuracy of the measured points, and hence, lead to more accurate inspection decisions.
12. The threshold value that limits the segmented surfaces, under- and over-segmentation are different challenges that face current segmentation algorithms due to variation in sampling density with different objects and different measuring methods. The developed segmentation algorithm presents a novel approach to overcome the under- and over-segmentation problems by implementing a two sphere neighborhood functions and adding limitation to the propagation technique. The threshold value is determined based on feature characteristics (continuity) from the CAD model.
13. A weight-based approach was developed in the propagation process that adopts a common neighbor criterion, which in turn maintains consistency in surface segmentation and avoids the generation of extra non-meaningful sub-point clouds (over-segmentation).
14. Although, features are known in the CAD model for inspection application, a feature recognition algorithm was developed to identify features from their skeleton, based on medial axis transform, for reverse engineering and computer vision applications.

Developed Localization Method

15. Current localization algorithms are applied to the point cloud as a whole, while tolerances are being specified for some features. The deviation from the CAD model is then calculated and compared to tolerance values. The deviation values of toleranced features are affected by the value of the localization results of non-toleranced features. Moreover, some tolerances associated with some features such as cylindricity, straightness or flatness are

not related to other features in the part. Current localization techniques are not best suited for the current manufacturing environment with its complex parts. A novel localization algorithm that accounts for tolerance specifications specified by the designer was developed. The developed algorithm integrates the tolerance verification as an optimality criterion in the localization process.

16. From a tolerance perspective, the minimum zone criterion has demonstrated and edge over the least mean square evaluation methods, which are used in current localization techniques, to comply with the international standards such as ASME Y14.5. The tolerance-based localization algorithm is developed based on the minimum zone criterion.
17. The minimum zone tolerance verification algorithm is formulated as a constraint satisfaction problem which reduces computation time and effort and produces “within tolerance” results (i.e. accept/reject decision) that are acceptable from inspection perspective. The minimum tolerance zone estimated by any explicit vector gradients optimization method would require at least seven operators’ evaluations for each numerical gradient evaluation. Any optimization method that doesn’t use explicit vector gradient estimates would require literally hundreds to tens of thousands of Closest Point C and Minimum Zone MZ_i sub-algorithms calculations. This extra effort to reach the optimum value is not needed to reach the decision of accepting or rejecting a part. In the developed algorithm, the approximate minimum zone is attained by applying a constraint satisfaction problem modeled as a closed form solution problem, which reduces the computation effort and the solution is obtained in fewer steps and shorter time. The IMZ localization algorithm allows us to move from a given starting point to local minima relatively quickly in comparison with other possible alternatives. Each iteration requires only one evaluation of the Closest Point C and Minimum zone MZ_i operators: the most expensive computations.
18. One of the main contributions of this tolerance-based localization algorithm is that the inspection decision is reached in fewer steps and the effects

(consequences) of non-toleranced features on the localization results are avoided. Inspectors used to align the measured points with the CAD model in one step and verify tolerance in the next step. This algorithm verifies tolerance and applies localization in one step, which in turn, speed up and improves the inspection decisions.

19. The advantage of the developed tolerance-based localization algorithm is that it requires neither derivatives nor preprocessing of 3D data except for the removal of outliers.
20. The developed Iterative Minimum Zone (IMZ) localization algorithm was applied and compared to benchmark data from the literature and the results were analyzed and compared to the published results and prove its validity to verify form tolerances such as straightness, flatness and cylindricity. Both; the new IMZ localization algorithm and the published solutions satisfy the ASME mathematical definitions of the tolerance zones. By comparing the obtained minimum zone values from the IMZ localization algorithm to previous results for these data sets, it shows that the results from the developed algorithm are within 0.5 % of previously published results. The reason for these differences is due to the different approaches used to solve the problem.

6.2 Significance

This dissertation addresses a problem that arises due to the increased aesthetic and functional complexity in products as well as the advances in inspection systems technology and the need to manage these technologies cost effectively and with the least disruption of the production activities and their associated high cost. A novel hybrid inspection methodology was developed to achieve these goals. The merit of the developed methodology can be summarized in the following points.

1. The developed planning model generates an optimal plan that combines both tactile and non-contact measurement tools to maximize the benefit and utilization of new technologies and overcome their shortcomings while

satisfying the different tolerance requirements specified by the designer. Its application and use in industry can lead to reducing product development cost and increasing quality of manufactured goods. It has the advantage of improving adaptability to changing products due to its automated planning characteristics and ease of use and implementation. The reduction of the human intervention will in turn reduce the associated errors of the inspection operator.

2. One of the main benefits of the developed methods is its ability to capture the knowledge of the inspection planner and enable the selection of the most appropriate tool for measurement and reduce unnecessary effort during the digitization process. A unified inspection-specific features taxonomy that captures the knowledge of the human planner was developed to help automate both inspection tools selection and planning the inspection tasks. A detailed plan with sequenced inspection tasks is generated to facilitate and speed up the process of producing complete and accurate measured points.
3. A 0-1 integer mathematical programming model has been formulated and applied to solve the hybrid inspection planning problem. The model is a formulation of the classical TSP, where the process plan is modeled as a tour less the most expensive leg. This is the first time this exact-optimal model has been applied to the hybrid inspection planning problem.
4. Although hardware exists for conducting hybrid measurements, mainly touch probes and laser scanners, the data manipulation and interpretation of these hybrid measurements is not well developed. The methods developed throughout the thesis have helped better define more rigorous means for inspection and verification using hybrid measurement.
5. An accurate automated treatment of inspection data from hybrid measurement system leads to better inspection results. The benefit of the developed segmentation algorithm is that it deals directly with *un-organized* point cloud data digitized using non-contact sensors combined with the data received from the contact probe and eliminates the associated errors. Moreover, the output of the segmentation algorithm which is the input to the

localization algorithm is the actual measured points of inspected features not a substitute representation. This reduces the amount of error in data exchange and produces better and accurate results.

6. The new developed tolerance-based localization algorithm, which integrates the tolerance verification as an optimality criterion in the localization process, overcomes the shortcomings of current localization algorithms, which lead to the rejection of good parts. The iterative minimum zone localization algorithm benefits from the powerful techniques of localization while being able to conform to tolerance standards such as ASME and ISO standards.
7. The main benefits of the developed segmentation and tolerance-based localization algorithms are the improvement of inspection decisions to not to reject good parts that has been rejected due to misleading localization results. The better and more accurate achieved inspection decisions will lead to less scrap and waste in the manufacturing system which in turn will reduce the product price and improve the company potential in the market.

6.3 Future Work

The results of this thesis provide a useful basis for further research in the area of hybrid inspection planning, range data manipulation, and tolerance verification techniques. The following issues are suggested for further research and investigation as extensions of the developed research:

- a. The sensor selection knowledge-based system is limited to a single tolerance specification per feature. Some features may require more than one tolerance control callout. The consideration of more than one tolerance control per inspected feature is worthy of investigation in the future.
- b. The developed inspection-specific features taxonomy captures. The knowledge of the human planner. It can be extended by adding new rules and features applied in different industries.

- c. The developed system can be integrated with CAD models to automate the process of input data collection and interpretation, as well as with downstream applications for quality related performance analysis and decision making.
- d. Neighborhood calculation in the developed segmentation algorithm is the most time consuming step. More efficient algorithms such as Grid-Octree would be of great help to speed up the segmentation process as well as further research to tune up the selection of parameters and their limits.
- e. Numerical experimentations have been conducted to select the most appropriate values for the radii of the two-sphere neighborhood function. A good sensitivity analysis with statistical indicators can better tune these parameters.
- f. The tolerance-based localization algorithm was developed for verification of different types of form tolerances. A similar approach may be used to develop algorithms for other types of geometric tolerances.

REFERENCES

- Ainsworth, I., Ristic, M. and Brujic, D., 2000, "CAD-based measurement path planning for free-form shapes using contact probes", *International Journal of Advanced Manufacturing Technology*, vol. 16, n1, pp. 23-31.
- Alberts, C. P., 2004, "Surface reconstruction from scan paths", *Future Generation Computer Systems*, vol. 20, pp. 1285-98.
- AlRashdan, A., Motavalli, S. and Fallahi, B., 2000, "Automatic segmentation of digitized data for reverse engineering applications", *IIE transactions*, vol. 32, pp. 59-69.
- Attene, M., Katz S., Mortara, M., Patane, G., Spagnuolo, M. And Tal, A., 2006, "Mesh segmentation – a comparative study", *Proceedings of the IEEE International Conference on Shape Modeling and Applications (SMI'06)*, 14-16 June 2006, pp. 7-18.
- Amenta, N., Choi, S. and Kolluri, R., 2001, "The power crust", *Proceedings of Solid Modeling Sixth ACM Symposium on Solid Modeling and Applications*; Ann Arbor, MI, pp. 249-60.
- Arun K. S., Huang, T. S. and Blostein, S. D., 1987, "Least-Squares Fitting of Two 3-D Point Sets", *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI-9)*, Sept. 1987, vol. 9, n5, pp. 698-700.
- Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R. and Wu, A. Y., 1998, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions", *Journal of the ACM*, vol. 45, n6, pp. 891-923.
- ASME Y14.5M-1994: Dimensioning and Tolerancing, American Society of Mechanical Engineers (ASME), New York, 1994, ISBN: 0791822230
- ASME Y14.5.1M-1994: Mathematical Definition of Dimensioning and Tolerancing Principles, American Society of Mechanical Engineers (ASME), New York, 1994, ISBN: 0791822524
- Badar, M., Raman, S., Pulat, P., 2003, "Intelligent search-based selection of sample points for straightness and flatness estimation", *Journal of Manufacturing Science and Engineering*, vol. 125, pp.263-70.
- Bae, K. H. and Lichti, D. D., 2008, "A method for automated registration of unorganized point clouds", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 63, pp. 36-54.
- Barcenas, C. C. and Griffin, P. M., 2001, "Geometric tolerance verification using superquadrics", *IIE Transaction*, vol. 33, pp. 1109-20.

- Beg, J. and Shunmugam, M. S., 2002, "An object oriented planner for inspection of prismatic parts", *International Journal of Advanced Manufacturing Technology*, vol. 19, n12, pp. 905-16.
- Beg, J. and Shunmugam, M. S., 2003, "Application of Fuzzy Logic in the selection of part orientation and probe orientation sequencing for prismatic parts", *International Journal of Production Research*, vol. 41, n12, pp. 2799-815.
- Benko, P., Martin, R. and Varady, T., 2001, "Algorithms for reverse engineering boundary representation models", *Computer Aided Design*, vol. 33, pp.839-51.
- Benko, P. and Varady, T., 2004, "Segmentation methods for smooth point regions of conventional engineering objects", *Computer Aided Design*, vol. 36 n5, pp.11-23.
- Besl, P. and Jain, R. 1988, "Segmentation through variable order surface fitting", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 10, n2, pp. 167-91.
- Besl, P. and McKay, N., 1992, "A method for registration of 3-D shapes", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 14, n2, pp. 239 - 56.
- Bichmann, S., Glaser, U. and Pfeifer, T., 2004, "User-friendly optical metrology in production engineering", *Proceedings of SPIE*, vol. 5457, pp. 432-40.
- Blais, F., 2004, "Review of 20 years of range sensor development", *Journal of Electronic Imaging*, vol. 13, n1, pp. 231-43.
- Bourdet, P., 1988, "A study of optimal criteria identification based on the small displacement screw model", *Annals of CIRP*, vol. 37, pp. 503-6.
- Bradley, C. and Chan, V., 2001, "A complementary sensor approach to reverse engineering", *Transaction of the ASME*, vol. 123, pp. 74-81.
- Brenner C, Bohm J, and Guhring J., 1998, "An experimental measurement system for industrial inspection of 3D parts", *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 3521, pp. 237-47.
- Bispo E., Fisher R., 1994, "Free-form surface matching for surface inspection", *IMA Conf Mathematics Surfaces*; pp. 119–36.
- Carr, K. and Ferreira, P., 1995 a "Verification of form tolerances Part I: Basic issues, flatness, and straightness", *Precision Engineering*, vol. 17, pp. 131-43.
- Carr, K. and Ferreira, P., 1995 b "Verification of form tolerances Part II: Cylindricity and straightness of a median line", *Precision Engineering*, vol. 17, pp. 144-56.

- Chan, K. and Gu, P., 1993, "An Object-oriented knowledge-based inspection planner", IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, vol. 2, pp. 646-649.
- Chen, Y. and Medioni, G., 1991, "Object modeling by registration of multiple range images", IEEE International Conference on Robotics and Automation, vol. 3, pp. 2724-9.
- Chen, Y., Wang, Y. and Yang, Z., 2004, "Towards a haptic virtual coordinate measuring machine", International Journal of Machine Tools and Manufacture, vol. 44, n10, pp. 1009-17.
- Chiang, Y., Chen, F., 1999, "CMM Probing Accessibility in a Single Slot", International Journal of Advanced Manufacturing Technology, vol. 15, pp. 261-7.
- Cho, M. W., Lee, H. and G.S., Y., 2005, "A Feature-Based Inspection Planning System for Coordinate Measuring Machines", International Journal of Advanced Manufacturing Technology, vol. 26, n9-10, pp. 1078-87.
- Clark, J., 2000, "Implementing non-contact digitization techniques within the mechanical design process", Sensor Review, vol. 20, n3, pp. 195-201.
- Delingette, H., Seguin, O., Perrocheau, R. And Menegazzi, P., 1997, "Accuracy evaluation of 3D reconstruction from CT-scan images for inspection of industrial parts", Proceedings of the International Conference on Quality Control by Artificial Vision (QCAV'97), Le Creusot, France, May 1997, pp. 80-85.
- Dowling, M., Griffin, P., Tsui, K. and Zhou, C., 1997, "Statistical issues in geometric feature inspection using coordinate measuring machines", Technometrics, vol.39, n1, pp. 3-16
- Eggert, D. W., Lorusso, A., Fisher, R. B., 1997, "Estimating 3-D rigid body transformations: a comparison of four major algorithms", Machine Vision and Applications, vol. 9, pp. 272-90.
- ElKott, D. F., ElMaraghy H.A., ElMaraghy, W.H., 2002, "Automatic sampling for CMM inspection planning of free form surfaces", International Journal of Production Research, vol. 40 n11, pp. 2653-76.
- ElMaraghy, H. A. and Gu, P. H., 1987, "Expert system for inspection planning", CIRP Annals, vol. 36 pp. 85-9.
- ElMaraghy, H. A. 1993, "Evolution and future perspectives of CAPP", CIRP Annals vol. 42, pp. 739-51.

ElMaraghy, H. A. and ElMaraghy, W. H., 1994, "Computer-aided inspection planning (CAIP)", in *Advances in Feature Based Manufacturing*, eds. J.J. Shah, M. Mantyla, D. Nau, Elsevier Publishers, Amsterdam, ISBN No.: 0 444 81600 3, Chapter 16, pp. 363-96.

ElMaraghy, H. A. and Yang, X., 2003, "Computer aided planning of laser scanning of complex geometries", *CIRP Annals*, vol. 52, n1, pp. 411-4.

ElMaraghy, H. A. 2007, "Reconfigurable process plans for responsive manufacturing systems", *Digital enterprise technology: Perspectives and future challenges*, Editors: Cunha, PF and Maropoulos, PG, Springer Science, ISBN: 978-0-387-49863-8: 35-44.

ElMaraghy, W., ElMaraghy, H., and Wu, Z., 1990, "Determination of actual geometric deviations using coordinate measuring machine data", *Manufacturing Review*, vol. 3, n1, pp. 23-9.

ElMaraghy, W. and Rolls, C., 2001, "Design by quality product digitization", *CIRP Annals*, vol. 50, n1, pp. 93-96.

Fan, K. and Tsai, T., 2001, "Optimal shape error analysis of the matching image for a free-form surface", *Robotics and Computer Integrated Manufacturing*, vol. 17, pp. 215-22.

Fang, Y., Chen. K. N. and Lin, Z. H., 1998, "Stereo vision and CMM integrated intelligent inspection system in reverse engineering", *SPIE Conference on Machine Vision Systems for Inspection and Metrology VII*, pp. 115-22.

Faugeras O., Herbert M., 1986, "The representation recognition and localization of 3-D objects", *International Journal of Robotic Research*, vol. 5, n3, pp. 27-52.

Ge, Q., Chen, B., Smith, P., Menq, C., 1992, "Tolerance specification and comparative analysis for computer-integrated dimensional inspection", *International Journal of Production Research*, vol. 30, n9, pp.2173-97.

Guehring, J. 2001, "Reliable 3D surface acquisition, registration and validation using statistical error models", *Third International Conference 3-D Digital Imaging and Modeling (3DIM'01)*, Quebec City, Canada, May 28–June 1. pp. 224-31.

Gou, J., Chu, Y., Wu, H. and Li, Z., 1998 "Geometric formulation of orientation tolerances", *Proceedings of the IEEE International Conference on Robotics & Automation*, vol. 3, pp. 2728-33.

Gou, J., Chu, Y., Li, Z., 1999, "A geometric theory of form, profile, and orientation tolerances," *Precision Engineering*, vol. 23, pp. 79-93.

Griffith, G. K., 2002, "Geometric Dimensioning and Tolerancing: Applications and Inspection," Prentice Hall.

- Haibin, Z., Junying, W., Boxiong, W., Jianmei, W. and Huacheng, C., 2006, "A PLM-based automated inspection planning system for coordinate measuring machine", Proceedings of SPIE, vol. 6358, pp. 1-6.
- Horn, B., 1987, "Closed-form solution of absolute orientation using unit quaternions", of the Optical Society of America A (Optics and Image Science), vol. 4, n4, pp. 629-42.
- Huang, J. and Menq, C. H., 2001, "Automatic data segmentation for geometric feature extraction from unorganized 3-D coordinate points", IEEE Transactions on Robotics and Automation, vol. 17, n3, pp. 268-79.
- Huang, J. and Menq, C. H., 2002a, "Combinatorial manifold mesh reconstruction and optimization from unorganized points with arbitrary topology", Computer Aided Design, vol. 34, n2, pp. 149-65.
- Huang, J. and Menq, C. H., 2002b, "Automatic CAD model reconstruction from multiple point clouds for reverse engineering", Journal of Computing and Information Science in Engineering, vol. 2, pp. 160-70.
- Huang, X. and Gu, P., 1998, "CAD-model based inspection of sculptured surfaces with datums", International Journal Production Research, vol. 36, n5, pp. 1351-67.
- Huang, C. C. and Zheng Y. F., 1996, "Integration of vision and laser displacement sensor for efficient and precise digitizing", IEEE Transaction on Industrial Electronics, vol. 43 n3 pp. 364-71.
- Hwang, C. Y., Tsai, C. Y. and Chang, C. A., 2004, "Efficient inspection planning for coordinate measuring machines", International Journal of Advanced Manufacturing Technology, vol. 23, n9-10 pp. 732-42.
- Hwang, I., Lee, H. and Ha, S., 2002, "Hybrid Neuro-Fuzzy approach to the generation of measuring points for knowledge-based inspection planning", International Journal of Production Research, vol. 40, n11 pp. 2507-20.
- Ikonomov, P., Okamoto, H., Tanaka, F. and Kishinami, T., 1995, "Inspection method for geometrical tolerance using virtual gauges", IEEE International Conference on Robotics and Automation, 21-27 May, vol. 1, pp. 550-5.
- Ikonomov, P., Tanaka, F. and Kishinami, T., 1997, "A model of the constraint substitute element for geometrical tolerance verification", International Journal of Machine Tools Manufacturing, vol. 37, n10, pp. 1465-73.
- Jinkerson, R. A., Abrams, S. L., Bardis, L., Chryssostomidis, C., Clement, A., Patrikalakis, N. M. and Wolter, F. E., 1993, "Inspection and Feature Extraction of Marine Propellers", Journal of Ship Production, vol. 9, pp. 88 -106.

- Kase, K., Makinouchi, A., Nakagawa, T., Suzuki, H., and Kimura, F., 1999, "Shape error evaluation method of free-form surfaces", *Computer Aided Design*, vol. 31, n8, pp. 495-505.
- Katz, S. and Tal, A., 2003, "Hierarchical mesh decomposition using Fuzzy clustering and cuts", *ACM Transactions on Graphics*, vol. 22, n 3, pp. 954-61.
- Ketan, H. S., Al-Bassam, M. A., Adel, M. B. and Rawabdeh, I., 2002, "Modelling integration of CAD and inspection planning of simple element features", *Integrated Manufacturing Systems*, vol. 13, n7 pp. 498-511.
- Kim, S. and Chang, S., 1996, "Development of the off-line measurement planning system for inspection automation", *Computers and Industrial Engineering*, vol. 30, n3, pp. 531-42.
- Langis, C., Greenspan, M. and Godin, G., 2001, "The parallel iterative closest point algorithm", *Proceedings of the Third International Conference on 3D Digital Imaging and Modeling (3DIM)*, Quebec City, Quebec, Canada, May 28 – June 1, 2001.
- Lavoue, G., Dupont, F., and Baskurt, A., 2005, "A new CAD mesh segmentation method, based on curvature tensor analysis", *Computer Aided Design*, vol. 37, pp. 975-87.
- Li, Y and Gu, P., 2004, "Free-form surface inspection techniques state of the art review", *Computer Aided Design*, vol. 36, pp. 1395-417.
- Li, Y and Gu, P., 2005, "Inspection of free-form shaped parts", *Robotics and Computer Integrated Manufacturing*, vol.21, pp. 421-30.
- Li, Y. F. and Liu, Z. G., 2003, "Method for determining the probing points for efficient measurement and reconstruction of freeform surfaces", *Measurement Science and Technology*, vol. 14, pp. 1280-8.
- Limaïem, A. and ElMaraghy, H., 1997a, "Automatic planning for coordinate measuring machines", *Proceedings of the IEEE International Symposium on Assembly and Task Planning*, pp. 243-8.
- Limaïem, A. and ElMaraghy, H., 1997b, "A general method for analyzing the accessibility of features using concentric spherical shells", *International Journal of Advanced Manufacturing Technology*, vol. 13, pp.101-8.
- Limaïem, A. and ElMaraghy, H., 1998, "Automatic path planning for coordinate measuring machines", *Proceedings IEEE International Conference on Robotics and Automation*, vol. 1, pp. 887-92.
- Limaïem, A. and ElMaraghy, H. A., 1999, "CATIP: A computer aided tactile inspection planning system". *International Journal Production Research*, vol. 37, n2, pp. 447-65.

- Limaiem, A. and ElMaraghy, H. A., 2000, "Integrated accessibility analysis and measurement operations sequencing for CMMs", *Journal of Manufacturing Systems*, vol. 19, n2, pp. 83-93.
- Lin, Z., and Chow j., 2001, "Near optimal measuring sequence planning and collision-free path planning with a dynamic programming method", *International Journal of Advanced Manufacturing Technology*, vol. 13, pp. 29-43.
- Lin, Z. C., Lin, W. S., 2001, "The application of Grey theory to the prediction of measurement points for circularity geometric tolerance", *International Journal of Advanced Manufacturing Technology*, vol. 17, n5, pp. 348-60.
- Liu, J., Ng, H., Yamazaki, K. and Nakanishi, K., 2003, "Autonomously generative CMM part programming for in-process inspection", *Journal of Manufacturing Science and Engineering, Transactions of the ASME*, vol. 125, n1, pp. 105-12.
- Lu, C., Morton, D., Myler, P. and Wu, M., 1995, "Artificial intelligent (AI) inspection path management for multiple tasks measurement on coordinate measuring machine (CMM): an application of Neural Network technology", *IEEE International Engineering Management Conference*, pp. 353-7.
- Lu, C. G., Morton, D., Wu, M. H. and Myler, P., 1999, "Genetic algorithm modeling and solution of inspection path planning on a coordinate measuring machine (CMM)", *International Journal of Advanced Manufacturing Technology*, vol. 15 n6 pp.409-16.
- Luo, P. F., Pan, S.P. and Chu, T. C., 2004, "Application of computer vision and laser interferometer to the inspection of line scale", *Optics and Lasers in Engineering*, vol. 42, pp. 563-84.
- Malamas, E., Petrakis, E., Zervakis, M., Petit, L., Legat, J., 2003, "A survey on industrial vision systems, applications and tools", *Image and vision computing*, vol. 21, pp. 171-88.
- Malyscheff, A. M., Trafalis, T. B., Raman, S., 2002, "From support vector machine learning to the determination of the minimum enclosing zone", *Computers and Industrial Engineering*, vol. 42, pp.59-74.
- Marshall, A, Martin, R. and Hutber, D., 1991, "Automatic inspection of mechanical parts using geometric models and laser range finder data", *Image and Vision Computing*, vol. 9, n6, pp. 385-405.
- Masuda, T. and Yokoya, N., 1995, "A robust method for registration and segmentation of multiple range images", vol. 61, n3, pp. 295-307.
- Menq, C. H., Yau, H. T. and Lai, G. Y., 1992, "Automated precision measurement of surface profile in CAD-directed inspection", *IEEE Transactions on Robotics and Automation*, vol. 8, n2, pp. 268-278.

Merat, F., Radack, G., Roumina, K. and Ruegsegger, S., 1991, "Automated inspection planning within the rapid design system", IEEE International Conference on Systems Engineering, 1-3 Aug. pp.42 – 48.

Meyer, A. and Marin, P., 2004, "Segmentation of 3D triangulated data points using edges constructed with a C1 discontinuous surface fitting", Computer Aided Design, vol. 36, pp. 1327-36.

Mital, A., Govindaraju, M. and Subramani, B., 1998, "A comparison between manual and hybrid methods in parts inspection," Integrated Manufacturing Systems, vol. 9, n6, pp. 344-9

Mohib, A., Azab, A., ElMaraghy, H., 2008, "Feature-based Hybrid Inspection Planning: A Mathematical Programming Approach", International Journal of Computer Integrated Manufacturing (IJCIM) Special Issue on: Integrated Products / Processes and Systems in Changeable Manufacturing, vol. 22, n1, In Press.

Mohib, A., Remy, S., ElMaraghy, H., 2006, "Recognition of Geometric Primitives Using Medial Axis Transform," 16th CIRP International Design Seminar, Kananaskis, Alberta, July 16-19, 2006.

Moroni, G., Polini, W. and Semeraro, Q., 1998, "Knowledge based method for touch probe configuration in an automated inspection system", Warsaw, Poland, Elsevier.

Newman, T., Jain A., 1995, "A survey of automated visual inspection", Computer Vision and Image Understanding, vol. 61, n2, pp. 231-62.

Newman, T.S., Ali, L, Brail, A, Brecker, C, Klemn, P, Liu, R, Nassehi, A, Nguyen, VK, Proctor, F, Rosso, RSU, Stroud, JI, Suh, SH, Vittr, M, Wang, L, Xu, XW, 2007, "The evolution of CNC technology from automated manufacture to global interoperable manufacturing", CIRP 2nd International Conference on Changeable, Agile, Reconfigurable and Virtual Production, 22-24 July 2007, Toronto, Canada, pp. 75-101.

Orazi, L. and Tani G., 2007, "Geometrical inspection of designed and acquired surfaces", International Journal of Advanced Manufacturing Technology, vol. 34, pp. 149-55.

Pahk, H. J. and Ahn, W. J., 1996, "Precision alignment technique for parts having thin features using measurement feedback iterative method in CAD/CAI environment", International Journal of Machine Tools Manufacturing, vol. 36, n2, pp.217-27.

Patané, G. and Spagnuolo, M., 2002, "Multi-resolution and slice-oriented feature extraction and segmentation of digitized data", ACM Symposium on Solid and Physical Modeling, Proceedings of the seventh ACM symposium on Solid modeling and applications (SM'02), June 2002, Saarbrucken, Germany, pp. 305-12.

- Patrikalakis, N. M. and Bardis, L., 1991, "Localization of rational B-spline surfaces", *Engineering with Computers*, vol. 7, pp. 237-52.
- Pottmann, H, Wallner J, Leopoldseder S., 2001, "Kinematical methods for the classification, reconstruction and inspection of surfaces", *Proceedings of the 1er Congrès National de Mathématiques Appliquées et Industrielles (SMAI 2001)*, 28 May – 1 June 2001, Pompadour, France.
- Pottmann, H., Leopoldseder, S. and Hofer, M., 2004, "Registration without ICP", *Computer Vision and Image Understanding*, vol. 95, pp. 54-71.
- Prakasvudhisarn, C., Trafalis, T. B., Raman, S., 2003, "Support vector regression for determination of minimum zone", *Transactions of the ASME*, vol. 125, pp. 736-9.
- Prieto F, Redarce T, Lepage R, Boulanger P., 2002, "An automated inspection system", *International Journal of Advanced Manufacturing Technology*, vol. 19, pp. 917–25.
- Qian, X. and Harding, K.G., 2003 "Partitioning positional and normal space for fast occlusion detection", *Proceedings of the ASME Design Engineering Technical Conference*, v 2 B, *Proceedings of the 2003 ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. vol. 2: 29th Design Automation Conference, 2003, pp. 737-44.
- Rusinkiewicz, S and Levoy, M., 2001, "Efficient variants of the ICP algorithm", *IEEE Third International Conference on 3D Digital Imaging and Modeling (3DIM 2001)*, pp. 145-52.
- Sahoo, K. C. and Menq, C. H., 1991, "Localization of 3-D objects having complex sculptured surfaces using tactile sensing and surface description", *Journal of Engineering for Industry*, vol. 113, pp. 85-92.
- Sapidis, N. and Besl, P., 1995, "Direct construction of polynomial surfaces from dense range images through region growing", *ACM transactions on Graphics*, vol. 14, n2, pp. 171-200.
- Sharp, G. C., Lee, S. W. and Wehe, D. K., 2002, "ICP registration using invariant features", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, n1, pp. 90-102.
- Shen, T. S., Huang, J. and Menq, C. H., 2000, "Multiple-sensor integration for rapid and high-precision coordinate metrology", *IEEE/ASME transaction on Mechatronics*, vol. 5 n2 pp. 110-21.
- Shi, Q., Xi, N., Chen, Y. and Sheng, W., 2006, "Registration of point clouds for 3D shape inspection", *Proceedings of the International Conference on Intelligent Robots and Systems*, October 9-15, 2006, Beijing, China. pp. 235-40.

- Silva, L., Bellon, O. R. P. and Boyer, K. L., 2007, "Multiview range image registration using the surface interpenetration measure", *Image and Vision Computing*, vol. 25, pp. 114-25.
- Tucker, T. M. and Kurfess, T. R., 2003a, "Newton methods for parametric surface registration. Part I. Theory", *Computer Aided Design*, vol. 35, pp. 107-14.
- Tucker, T. M. and Kurfess, T. R., 2003b, "Newton methods for parametric surface registration. Part II. Experimental validation", *Computer Aided Design*, vol. 35, pp. 115-120.
- Tucker, T. M. and Kurfess, T. R., 2006, "Point cloud to CAD model registration methods in manufacturing inspection", *Journal of Computing and Information Science in Engineering*, vol. 6, pp. 418-21.
- Unsalan C. and Ercil, A., 1999, "Automated tolerance inspection by implicit polynomials", *Proceedings of the IEEE International Conference on Image Analysis and Processing (ICIAP '99)*, Venice, Italy, September; 1999. pp. 1218–25.
- Vafaeseefat, A. and Elmaraghy, H., 2000, "Automated accessibility analysis and measurement clustering for CMMs", *International Journal of Production Research*, vol. 38, n10, pp. 2215-31.
- Wang, L. and Lin, G., 1997 "A vision-aided alignment datum system for coordinated measuring machines", *Measurement Science Technology*, vol. 8, pp. 707-14.
- Weber, T., Motavalli, S., Fallahi, B. and Cheraghi, S. H., 2002, "A unified approach to form error evaluation", *Journal of the International Societies for Precision Engineering and Nanotechnology*, vol. 26, pp. 269-78.
- Wheeler M. D. and Ikeuchi, K., 1995, "Iterative estimation of rotation and translation using the quaternion", *Technical Report CMU-CS-95-215*, Carnegie Mellon University.
- Woo, H., Kang, E., Wang, S. and Lee, K., 2002, "A new segmentation method for point cloud data", *International Journal of machine tools and manufacture*, vol. 42, pp. 167-78.
- Yamazaki, I., Natarajan, V., Bai, Z. and Hamann, B., 2006, "Segmenting point sets", *Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006 (SMI'06)*, 14-16 June 2006, pp. 6-15.
- Yang, Z. and Chen, Y., 2005, "Inspection path generation in haptic virtual CMM", *Computer Aided Design and Applications*, vol. 2, n1-4, pp. 273-82.
- Yau, H. and Menq, C., 1991, "Path planning for automated dimensional inspection using coordinate measuring machines", *Proceedings IEEE International Conference on Robotics and Automation*, vol. 3, pp. 1934-9.

- Yau, H. and Menq, C., 1995, "Automated CMM path planning for dimensional inspection of dies and molds having complex surfaces", *International Journal of Machine Tools & Manufacture*, vol. 35, n6, pp. 861-76.
- Yokoya, N. and Levine, M., 1989, "Range image segmentation based on differential geometry: A hybrid approach", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol.11, n6, pp. 643-9.
- Yoon, G.S., Kim, G.H., Cho, M.W., Seo, T.I., 2004, "A study of on-machine measurement for PC-NC system", *International Journal of Precision Engineering and Manufacturing*, vol. 5, n1, pp. 60-68.
- Zhang, Z., 1994, "Iterative point matching for registration of free-form curves and surfaces", *International Journal of Computer Vision*, vol. 13, n2, pp. 119-52.
- Ziemian, C., Medeiros, D., 1997, "Automated feature accessibility algorithm for inspection on a coordinate measuring machine", *International Journal of Production Research*, vol. 35, n10, pp. 2839-56.

APPENDIX A

TOLERANCE ZONE DEFINITIONS

This appendix is provided to give detailed information about the mathematical definitions of form tolerances (ASME Y14.5.1M, 1994).

A.1 Straightness of a derived median line: Cylindrical volume

Definition: Let \hat{T} be the zone direction vector of the straightness axis, \vec{A} be the zone locating position vector which is a point on the straightness axis and t is value of the tolerance size (diameter of the cylinder), then the zone is defined by the set of points {P} where

$$|\hat{T} \times (\vec{P} - \vec{A})| \leq \frac{t}{2} \quad (\text{A.1})$$

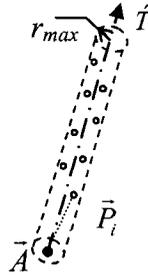


Figure (A.1) Straightness Tolerance Zone of a Median Line.

A.2 Straightness of a surface line: Area between two parallel lines

Definition: Let \hat{T} be the zone direction vector of the straightness axis, \vec{A} be the zone locating position vector which is a point on the straightness axis, t is value of the tolerance size (perpendicular distance between the two lines), \hat{C}_p is the direction vector of the cutting plane and \vec{P}_s is an arbitrary point on the cutting plane, then the zone is defined by the set of points {P} where

$$|\hat{T} \times (\vec{P} - \vec{A})| \leq \frac{t}{2} \quad (\text{A.2})$$

and

$$\hat{C}_p \bullet (\vec{P} - \vec{P}_s) = 0 \quad (\text{A.3})$$

$$\hat{C}_p \bullet (\vec{A} - \vec{P}_s) = 0 \quad (\text{A.4})$$

$$\hat{C}_p \bullet \hat{T} = 0 \quad (\text{A.5})$$

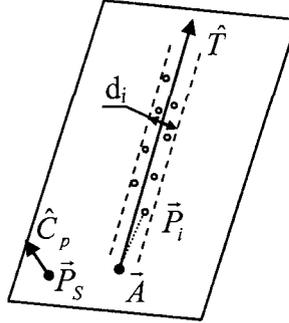


Figure (A.2) Straightness Tolerance Zone of a Surface Line.

A.3 Flatness: Volume between two parallel planes

Definition: Let \hat{T} be the zone direction vector which is normal to the planes, \vec{A} be the zone locating position vector which is a point on mid-plane and t is value of the tolerance size (perpendicular distance between the two planes), then the zone is defined by the set of points $\{P\}$ where

$$|\hat{T} \bullet (\vec{P} - \vec{A})| \leq \frac{t}{2} \quad (\text{A.6})$$

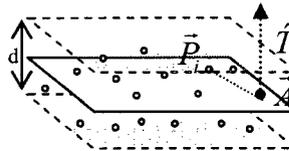


Figure (A.3) Flatness Tolerance Zone.

A.4 Cylindricity: Volume between two coaxial cylinders

Definition: Let \hat{T} be the zone direction vector which is of the two cylinders' axis, \vec{A} be the zone locating position vector which is a point on the axis and t is value of the

tolerance size (radial distance between the two cylinders), then the zone is defined by the set of points $\{P\}$ where

$$\left| \hat{T} \times (\bar{P} - \bar{A}) \right| \leq \frac{t}{2} \quad (\text{A.7})$$

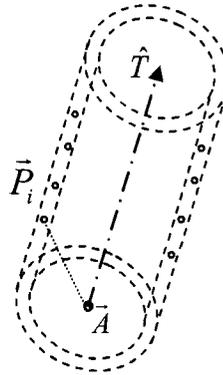


Figure (A.4) Cylindricity Tolerance Zone.

APPENDIX B

DATA SETS FOR LOCALIZATION ALGORITHM

This appendix is provided to give the 3D coordinate data for form tolerance evaluation in the localization algorithm (Carr and Ferreira, 1995 a, b).

Straightness of median line dataset: 1

X	Y	Z
000	-0.003	0.005
100	0.009	-0.012
200	0.015	-0.025
300	0.025	-0.038
400	0.029	-0.040
500	0.036	-0.048
600	0.042	-0.059
700	0.051	-0.062
800	0.059	-0.074
900	0.072	-0.088

Straightness of median line dataset: 2

X	Y	Z
000	0.410	0.000
254	0.000	0.124
508	-0.108	0.205
762	-0.170	0.306
1016	-0.112	0.352
1270	-0.068	0.387
1524	-0.050	0.326
1778	-0.150	0.248
2032	-0.302	0.256
2286	-0.286	0.1
2540	-0.220	-0.068
2794	-0.180	-0.262
3048	-0.148	-0.558
3302	-0.078	-0.740
3556	-0.178	-0.899
3810	-0.220	-0.956
4064	-0.272	-0.942
4318	-0.334	-0.928
4572	-0.266	-0.880
4826	-0.126	-0.980

5080	0	-1.040
5334	0.150	-0.830

Straightness of surface line dataset: 1

X	Y	Z
-50	0.003	0
-25	0.005	0
0	0.002	0
25	0.001	0
50	0.002	0

Straightness of surface line dataset: 2

X	Y	Z
1	2.428	0
2	2.891	0
3	3.445	0
4	2.931	0
5	3.895	0
6	4.196	0
7	4.497	0
8	4.662	0
9	4.545	0
10	4.303	0

Straightness of surface line dataset: 3

X	Y	Z
0.2845	-0.0034	0
0.6600	-0.0032	0
1.2041	-0.0030	0
1.4994	-0.0035	0
1.8494	-0.0036	0
2.2261	-0.0025	0
2.5724	-0.0028	0
2.9076	-0.0026	0
3.2548	-0.0031	0
3.4142	-0.0031	0
3.6307	-0.0029	0
3.9237	-0.0029	0
4.2647	-0.0028	0
4.5122	-0.0028	0

4.8150	-0.0027	0
5.1334	-0.0027	0
5.3603	-0.0030	0
5.6534	-0.0032	0
5.9058	-0.0020	0
6.0774	-0.0019	0
6.2962	-0.0019	0
6.5240	-0.0019	0
6.7114	-0.0017	0
6.9996	-0.0019	0
7.2076	-0.0017	0

Straightness of surface line dataset: 4 (Cutting plane normal = [0 0.5145 0.8575])

X	Y	Z
0.027333	-0.02638	-0.06218
4.556534	4.615587	4.654955
0.489564	0.45189	0.426775
9.348999	9.34492	9.3422
8.26204	8.355867	8.418418
5.244509	5.347951	5.416913
3.825358	3.842696	3.854255
6.913624	6.82318	6.762885
8.418161	8.503897	8.561055
6.560642	6.518368	6.490185
9.077879	9.127793	9.161068
0.45633	0.492422	0.516482
6.378744	6.275567	6.206783
3.70484	3.603447	3.535851
7.24276	7.210923	7.189698
0.767879	0.687045	0.633155
2.755516	2.69952	2.662189
4.753316	4.800613	4.832144
3.591212	3.594045	3.595934
9.929707	9.022041	9.083597
9.048268	9.044845	9.042563
3.189687	3.190953	3.191797
2.709189	2.615105	2.552382
0.725137	0.749481	0.76571
2.77399	2.767739	2.763572
4.596485	4.691021	4.754045
7.650097	7.581216	7.535295
1.273748	1.23415	1.207751
8.707659	8.658024	8.624934
7.295553	7.213906	7.159475

Flatness dataset: 1

X	Y	Z
0	0	0.005
100	0	0.004
200	0	0.001
300	0	0.002
400	0	0.002
0	100	0.004
100	100	0.003
200	100	0.003
300	100	0.002
400	100	0.002
0	200	0.003
100	200	0.004
200	200	0.002
300	200	0.001
400	200	0.002

Flatness dataset: 2

X	Y	Z
0	0	0.002
0	25	0.005
0	50	0.006
0	75	0.008
0	100	0.009
25	0	0.005
25	25	0.007
25	50	0.008
25	75	0.009
25	100	0.012
50	0	0.006
50	25	0.007
50	50	0.008
50	75	0.009
50	100	0.011
75	0	0.007
75	25	0.007
75	50	0.006
75	75	0.007
75	100	0.009
100	0	0.007
100	25	0.006
100	50	0.006

100	75	0.006
100	100	0.008

Flatness dataset: 3

X	Y	Z
0.2556	0.2994	0.0005
1.4992	0.3371	0.0013
2.6656	0.3726	0
3.5978	0.4009	0.0005
4.6241	0.4321	-0.0007
4.5989	1.2640	0.0001
3.4451	1.2289	0.0008
2.7096	1.2066	0.0004
1.6726	1.2968	0.0014
0.5273	1.2620	0.0009
0.1683	2.1413	-0.0002
0.9906	2.1663	0.0010
2.5485	2.1801	0.0008
3.4605	2.1369	0.0011
4.8632	2.1795	-0.0017
4.8401	2.9417	-0.0014
3.6557	2.9058	0.0012
2.4224	2.8683	0.0012
1.3839	2.8368	0.0011
0.4966	2.8098	-0.0002
0.4672	3.7751	-0.0008
1.6709	3.8116	0.0010
2.8864	3.8486	0.0006
3.7562	3.8750	0.0008
4.6746	3.9029	-0.0003

Flatness dataset: 4

X	Y	Z
0	0	0.00279
0	100	0.00294
0	200	0.00249
0	300	0.00224
0	400	0.00219
0	500	0.00313
0	600	0.00438
100	0	0.00117
100	100	0.00091

100	200	0.00024
100	300	-0.00006
100	400	-0.00007
100	500	0.00056
100	600	0.00188
200	0	-0.00084
200	100	-0.00111
200	200	-0.00183
200	300	-0.00223
200	400	-0.00224
200	500	-0.00184
200	600	-0.00052
300	0	-0.00195
300	100	-0.00224
300	200	-0.00320
300	300	-0.00378
300	400	-0.00361
300	500	-0.00359
300	600	-0.00207
400	0	-0.00202
400	100	-0.00253
400	200	-0.00339
400	300	-0.00410
400	400	-0.00429
400	500	-0.00418
400	600	-0.00263
500	0	-0.00117
500	100	-0.00217
500	200	-0.00328
500	300	-0.00392
500	400	-0.00422
500	500	-0.00438
500	600	-0.00263
600	0	-0.00067
600	100	-0.00108
600	200	-0.00244
600	300	-0.00313
600	400	-0.00366
600	500	-0.00391
600	600	-0.00221
700	0	0.00082
700	100	0.00018
700	200	-0.00131
700	300	-0.00212
700	400	-0.00260
700	500	-0.00267

700	600	-0.00069
800	0	0.00255
800	100	0.00219
800	200	0.00056
800	300	-0.00031
800	400	-0.00066
800	500	-0.00059
800	600	0.00135
900	0	0.00429
900	100	0.00438
900	200	0.00298
900	300	0.00253
900	400	0.00224
900	500	0.00285
900	600	0.00438

Cylinder dataset: 1 (Radius=30.0005; Length=60)

30.000000	0.001475	7.892267
1.056360	-29.981396	27.519008
-29.366428	-6.132937	13.137551
28.698913	8.739131	40.731883
-12.893256	-27.088078	56.081574
-22.315616	20.050269	31.164982
14.611799	-26.201056	2.074327
28.323328	9.888835	31.782012
-14.262380	-26.392880	0.461891
-22.304724	20.062385	4.010534
-26.057635	14.866056	41.206363
-25.432673	-15.911603	55.826190
17.043966	-24.688119	31.615727
25.129457	16.386287	39.235138
-25.917641	15.108801	42.071436
25.362190	-16.032710	45.731882
-2.344940	29.908214	2.847871
-2.620160	-29.885360	19.694054
-20.170149	-22.206503	45.384629
29.952444	-1.688520	21.920320
30.001000	0.001475	7.892267
1.056359	-29.981395	27.519008
-29.367407	-6.133141	13.137551
28.699869	8.739422	40.731883
-12.893685	-27.088981	56.081574
-22.316359	20.050938	31.164982
14.612286	-26.201929	2.074327

28.324273	9.889165	31.782012
-14.262855	-26.393767	0.461891
-22.305468	20.063053	4.010534
-26.058504	14.866552	41.206363
-25.433521	-15.912134	55.826190
17.044534	-24.688942	31.615727
25.130294	16.386834	39.235138
-25.918505	15.109304	42.071436
25.363036	-16.024245	45.731882
-2.345018	29.909211	2.847871
-2.620248	-29.886356	19.694054
-20.171721	-22.207243	45.384629
29.953442	-1.688577	21.920320

Cylinder dataset: 2 (Radius=60; Length=30)

X	Y	Z
60.051121	0.002953	3.946134
-57.932024	15.399312	15.983017
57.432130	17.488707	20.365942
55.022756	-23.936632	11.505062
29.180100	-52.423113	1.037163
-58.861558	-11.113569	20.134482
-44.597179	40.113733	2.005267
-23.247383	-55.406652	17.669299
34.041568	-49.309081	15.807863
-34.084135	-49.427745	12.479981
50.684216	-32.022045	22.865941
57.318676	17.619539	22.082457
-40.408130	-44.485701	22.692315
-39.838370	44.994386	7.411167
-10.261352	-59.146784	22.600675
53.919844	26.493193	18.949042
-8.540012	59.442972	13.092342
-59.369089	8.361285	7.133233
-38.029817	46.404843	4.995216
47.946099	-35.925380	27.276243

Cylinder dataset: 3 (Radius=50; Length=100)

X	Y	Z
-11.820859	50.421254	-15.817382
42.403448	-6.693162	56.567707
10.366902	80.249947	26.965969

18.527457	61.577469	-13.680418
23.930322	23.878386	-41.820643
66.363729	0.636729	49.246025
-3.608026	-24.493246	39.678687
75.507564	20.208045	6.298139
48.919097	55.614254	-13.266609
65.713317	2.841028	3.498858
46.632786	80.517454	4.866333
13.598993	83.519129	30.375000
84.570573	18.219363	28.224203
2.322453	-10.802862	51.268799
82.820384	38.516367	9.148307
3.553158	75.111087	30.738097
-5.898713	21.390330	60.097056
30.009532	-24.696147	35.870356
-3.793621	-14.263808	46.897322
58.357492	87.161327	11.960644
33.207329	64.844079	-10.665479
34.461290	41.806234	94.623903
26.871029	3.103967	39.482460
-4.153639	67.427229	23.451422
22.371000	47.845956	88.060867
67.398986	16.520701	79.062822
79.257377	49.418921	4.727043
-37.543275	31.718373	8.573268
49.576671	65.965076	-6.501629
96.781947	53.421231	22.908004
-18.623157	23.988046	47.691608
58.416292	-4.557784	48.525368
48.408528	15.833662	81.511728
31.694971	-2.169579	63.538387
-18.366214	2.837799	46.415679
81.087477	11.573666	46.319607
57.311572	-9.096050	38.123767
68.593970	33.580936	-6.118165
89.036231	21.722310	35.086999
3.141412	52.730721	67.919265

VITA AUCTORIS

Ahmed Mohib received his B.Sc. and M.Sc. degrees from Cairo University, Egypt in Mechanical Engineering with an Industrial Engineering focus. He is currently a doctoral candidate at University of Windsor, Canada in the department of Industrial & Manufacturing Systems Engineering. He is also employed by the Intelligent Manufacturing Systems (IMS) Centre as a research engineer. Beside inspection planning, his research interests also span Manufacturing Systems, Maintenance Planning, Nontraditional Optimization, and Control and vibration theory. A list of his up-to-date publications is given below:

Refereed Journal Publications:

- Mohib, A., Azab, A., ElMaraghy, H., 2008, "Feature-based Hybrid Inspection Planning: A Mathematical Programming Approach", International Journal of Computer Integrated Manufacturing (IJCIM) Special Issue on: Integrated Products / Processes and Systems in Changeable Manufacturing, vol. 22, n1, In Press.
- Youssef, A.M.A., Mohib, A. and ElMaraghy, H.A., "Availability Assessment Of Multi-State Manufacturing Systems Using Universal Generating Function", Annals of the CIRP, Vol. 55, n1, 2006, pp. 445-448.
- Shalaby, M.A., Gomaa, A.M., Mohib, A.M., "A genetic algorithm for preventive maintenance scheduling in a multi-unit multi-state system", Journal of Engineering and Applied Science, Vol. 51, n4, 2004, pp. 795-811.

Refereed Conference Publications:

- Mohib, A.; ElMaraghy, H.A., 2009, CAD-based Closed-form Solution Algorithm to Verify Cylindricity, CAT 2009, 11th CIRP International Conference on Computer Aided Tolerancing "Geometric Variations within Product Life-Cycle Management", March, 26th and 27th 2009, Annecy, France.
- Mohib, A.; ElMaraghy, H.A., 2007, Framework for Feature-based Hybrid Inspection Planning for Complex Mechanical Parts, CARV International Conference on Changeable, Agile, Reconfigurable and Virtual Production, Toronto, Ontario, Canada.
- Mohib, A.; Remy, S; ElMaraghy, H.A., 2006, Recognition of geometric primitives using medial axis transform, The 16th CIRP International Design Seminar, Design and Innovation for a sustainable Society, Kananaskis, Alberta, Canada.