Queen Mary, University of London

School of Electronic Engineering & Computer
Science

MODEL BASED 3D VISION SYNTHESIS AND ANALYSIS FOR PRODUCTION AUDIT OF INSTALLATIONS

Thesis submitted to University of London

in partial fulfilment of the

requirements for the degree of

Doctor of Philosophy

Karthikeyan Vaiapury

Supervisor: Prof. Ebroul Izquierdo

London, 2013.

Dedication

I would like to dedicate this thesis to all my gurus.

Acknowledgements

Declaration

I hereby declare that this dissertation is entirely the result of my own work, it arises out of my own research, and I have made full acknowledgements of the work and ideas of any other people who are cited in the thesis, or who have contributed to it.

Abstract

One of the challenging problems in the aerospace industry is to design an automated 3D vision system that can sense the installation components in an assembly environment and check certain safety constraints are duly respected. This thesis describes a concept application to aid a safety engineer to perform an audit of a production aircraft against safety driven installation requirements such as segregation, proximity, orientation and trajectory. The capability is achieved using the following steps. The initial step is to perform image capture of a product and measurement of distance between datum points within the product with/without reference to a planar surface. This provides the safety engineer a means to perform measurements on a set of captured images of the equipment they are interested in. The next step is to reconstruct the digital model of fabricated product by using multiple captured images to reposition parts according to the actual model. Then, the projection onto the 3D digital reconstruction of the safety related installation constraints, respecting the original intent of the constraints that are defined in the digital mock up is done. The differences between the 3D reconstruction of the actual product and the design time digital mockup of the product are identified. Finally, the differences/non conformances that have a relevance to safety driven installation requirements with reference to the original safety requirement intent are identified. The above steps together give the safety engineer the ability to overlay a digital reconstruction that should be as true to the fabricated product as possible so that they can see how the product conforms or doesn't conform to the safety driven installation requirements. The work has produced a concept demonstrator that will be further developed in future work to address accuracy, workflow

and process efficiency. A new depth based segmentation technique GrabcutD which is an improvement to existing Grabcut, a graph cut based segmentation method is proposed. Conventional Grabcut relies only on color information to achieve segmentation. However, in stereo or multiview analysis, there is additional information that could be also used to improve segmentation. Clearly, depth based approaches bear the potential discriminative power of ascertaining whether the object is nearer of farer. We show the usefulness of the approach when stereo information is available and evaluate it using standard datasets against state of the art result.

Contents

4   3D PACT (Production Audit Compare Tool) for discrepancy checking   91

List of Figures

List of Tables

Glossary

3DHT 3D Hough Transform. 15, 70

3DMT 3D Measurement and Testing. 61, 64

ACAT Advanced Constraints Analysis Tool. 15

AIV Acceptable Installation Volume. 2, 12, 20, 28, 40, 41, 121, 123, 125, 127–130

ARP Aerospace Recommended Practices. 5, 7

BP Backprojection. 52

BPE Back Projection Error. 64

CAD Computer Aided Design. 25, 27, 28, 40

CAE Computer Aided Engineering. 28, 40

CAM Computer Aided Manufacturing. 28, 40

CCA Common Cause Analysis. 4, 7

CCD Charge Coupled Device. 23

COTS Commercial Off The Shelf. 19, 43

CS Consensus Set. 76

CSG Constructive Solid Geometry. 47

CV Computer Vision. 8, 15, 97, 100, 111, 112

DLT Direct Linear Transform. 25, 51

DMU Digital Mockup. 1, 2, 4, 7–9, 11, 12, 15, 17, 92, 123, 133

DSD Disparity Distribution. 163

FAL Final Assembly Line. 44

FPSS Focused point based SLR stereo. 11, 43, 68, 114, 118

GMM Gaussian Mixture Model. 162, 165, 166

GUI Graphical User Interface. 92

ICP Iterative Closest Point Algorithm. 28, 40, 98, 100, 112

k-NN k-Nearest Neighbourhood classifier. 109

LIFE Local Invariant Feature Extraction. 37, 43, 48, 57, 89

LS Least Square. 98

MISSA More Integrated Systems Safety Assessment. 7, 17, 19, 44, 49

MRF Markov Random Field. 34

MSR Microsoft Research. 168

MSS Minimal Sample Sets. 75

MVS Multi View Stereo. 93

PACT Production audit compare Tool. 15, 41

PAMT Production audit measurement Tool. 14, 15, 41, 43, 48, 49

PMVS2 Patch based Multiview Stereo. 28, 40, 93

PSA Production Safety Audit. 10

PSS Point based SLR stereo. 69

PSSA Preliminary System Safety Assessment. 4, 7

RANSAC Random Sample Consensus. 70, 76, 77, 79, 81

SAE Society of Automative Engineers SAE International. 4, 5, 7

SIFT Scale Invariant Feature extraction. 19, 37, 57, 60, 71

SLR Single Lens Reflex. 11, 114, 115, 117

SURF Speeded Up Robust Features. 19, 37, 57

SVM Support Vector Machine. 106, 109

ToF Time of Flight. 24–26

XML Extendable Markup Language. 92, 94, 125, 133

Chapter 1

Introduction

In the past few years, there has been a tremendous development and modernization in the aircraft industries. However, in practice there is still a human subject intervention needed to find faults in the aircraft installation engines as a part of the production audit process. This is accomplished during the verification and validation phase in order to support the aircraft level modelling and further analysis. Advanced sensing techniques and image processing tools offer an appealing prospect of accurate 3D and 4D model generation of complex scenes and objects. In fact, manufacturing industries are showing compelling interest towards three dimensional scene synthesis and analysis due to its multifaceted potential applications ranging from real time recognition, verification, vehicle guidance etc. Using images taken from different perspectives or at different times, 3D and 4D representation of the scene can be built. This thesis investigates to develop a complete and novel 3D based solution for supporting installation production safety audit in the industrial environment. The aim is to take accurate 3D measurements of the fabricated sections of an installation from captured images and compare those measurements to the required constraints defined within the specification model of the installation in the DMU (Digital Mockup). The fabricated parts are assembled and machine parts built according to the pre-defined specifications. A sample aircraft and assembly environment is shown in Fig. 1.1. As mentioned by Webel et al. (2007), physical mockups are used to verify feasibility and ease the planning and they are fabricated based on digital data [86]. The physical mod-

els often do not longer match with digital data due to either the modifications made by model maker or fabrication inaccuracy. Furthermore, we address the problem of discrepancy checking by a new paradigm DMU based 3D vision and analysis which is concerned with building accurate 3D models with the semantic metadata of installation engine parts by using the corresponding base CATIA (Computer Aided Three Dimensional Interactive Application)/CAD(Computer Aided Design) installations. A new unified system application for the production audit in an aerospace industry is proposed in this thesis that serves three key tasks.

- Verification of the distant locations between datum points of installation with/without reference to planar surface model.

- Identification of the discrepancies between computed 3D world point cloud model and digital mock-up model of an assembly.

- Verification of the advanced installation constraints such as AIV (Acceptable Installation Volume) contains, intersect, disjoint test and trajectory test.

In spite of the facts that the first and subsequent two tasks are functionally independent, commonly they are related in terms of assisting the production audit task. First process involves the verification of manufactured parts to be within some predefined threshold range using a calibrated stereo camera with human interaction to select the matching disparity points. The distance between datum points and with respect to planar surfaces is used for the audit purpose. We describe the system flow, plus validate the technique via a number of experimental datasets. Second process involves model matching between the computed 3D world model and digital mock-up model. Usually, the computer aided geometry model is built before the actual installation. The discrepancy analysis between objects and geometric model is demonstrated. The third process involves the procedure which checks whether the computed 3D point cloud is within AIV or trajectory. In this thesis, we present a new cost-effective and robust framework for 3D measurement audit, discrepancy checking and installation constraint analysis

that uses input from digital camera and semantic metadata knowledge available from geometry models which can be used for verification tasks.

1.1 Concept Description : Production safety audit

A systems installation is the result of a design activity that consists of positioning the physical system components inside the aircraft. In the early design phase, the systems installation is defined by geometrical 3D models that define approximately the shape, the size and the location of each system. When the design progresses, the 3D models become more accurate, eventually taking the shape of the detailed product 3D drawings that are used to manufacture the individual parts. These latest models look like ideal representations of the system. (they do not take into account flexibility due to gravity, etc.) At each design phase, the installation shall fulfil various geometrical requirements. Some of these requirements are justified by the expected system or aircraft performance. For instance, the total length of cables shall be minimized in order to optimize the aircraft weight. The aim is to annul high danger and minimize the low danger situation of components placed in an installation environment (please refer Chapter. 5 for more details). 3D computer aided design tools like CATIA are currently used to support the definition and verification of the system installation. During design specification, safety analysis leads to the identification of installation requirements that need to be respected. One such example is in the case of performing a particular risk analysis. The analysis identifies conditions where claims of independence made by systems that employ redundant architectures are challenged as a consequence of the application of some identified particular risk model. It becomes necessary to redesign or mitigate e.g. by relocating the relevant equipment, or providing structural protection etc. It is also important that the installation constraints that have been identified, in the end, are respected in order to ensure the claim of independence is ensured. There should be a methodology to confirm that this is the case that need to be achieved by inspection of the production aircraft at different stages of fabrication and assembly. It is necessary for quality/safety

inspections of the complex installation environments to be carried out. With the recent advancements of 3D vision techniques, it is now becoming feasible to use these techniques to support the inspection and measurement of the fabricated products. These techniques can be used to carry out the measurements, to guide an inspector in identifying the exact locations related to safety constraints, and to also highlight where faults or inconsistencies within the installation may exist as a consequence of some constraint or segregation threshold being exceeded thereby supporting safety analysis and offering a considerable reduction in the time to carry out an inspection.

The research is part of MISSA (More Integrated Systems Safety Assessment), a research project joint funded by the European Commission 7th Framework Program ACP7-GA-2008-212088 [11]. This project aim to develop methods and tools to help safety engineers to collect, navigate, and manage information, structure their arguments, express their ideas, and most importantly find solutions to problems in an efficient, auditable and exhaustive way.

The objective of the task is to design and implement a production safety audit capability that:

1. Can identify configuration differences between the DMU and the captured fabricated product

2. Can identify and take an accurate enough relative measure of the position of the various objects present in the two images with respect to some datum and to each other, thereby aiding in the quality and safety inspection

3. Can check whether part of the safety analysis framework that will link the inspection task and audit results to the reasoning behind the safety related installation requirements.

Basically the capability fits in the existing safety process in the following way. Referring to the Fig. 1.2, a system installation is specified by executing various design iterations at around the interaction point between PSSA (Preliminary System Safety Assessment), the development of the systems architecture, CCA (Common Cause Analysis), and the allocation of requirements to hardware. CCA is a part of the aircraft systems safety analysis as defined by SAE (Society of

(a)

(b)

Figure 1.1: (a) Sample aircraft [5] and (b) an assembly environment [6]

Automative Engineers) ARP(Aerospace Recommended Practices) 4754 and SAE
ARP 4761.
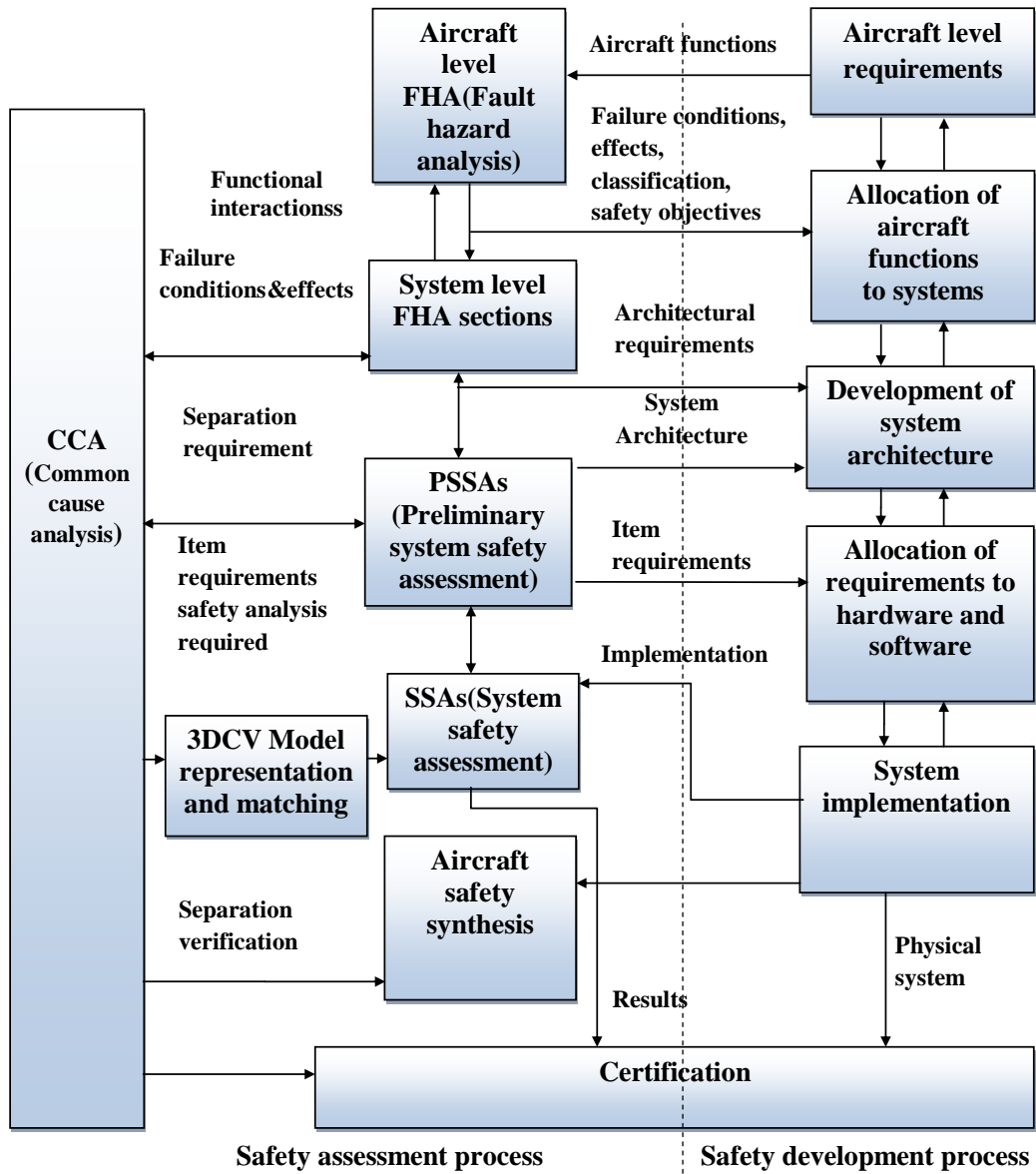
Figure 1.2: SAE (Society of Automative Engineers) ARP (Aerospace Recommended Practices) 4754 [11]

SAE 4754 is the aerospace safety standard issued by Aircraft and system development and safety assessment committee which discusses the certification aspects of highly-integrated or complex systems installed on aircraft, taking into account the overall aircraft operating environment and functions. As defined in [14], the term "highly-integrated" refers to systems that perform or contribute to multiple aircraft-level functions. The term "complex" refers to systems whose safety cannot be shown solely by test and whose logic is difficult to comprehend without the aid of analytical tools. SAE ARP 4761 is the standard which discusses about guidelines and methods for conducting the safety assessment process on civil airborne systems and equipment. CCA generates a number of derived requirements that impose installation constraints that must be respected. During design time, this simply means ensuring that the geometry model has constraints such as segregation, proximity, and orientation defined that represent these installation constraints and that the installation constraints are respected. When the aircraft is fabricated, it is cross checked against the design model for correctness using various measurement techniques. The PSSA and CCA part of the safety process delivers three types of analyses, amongst others that are in that region of the process that influence the installation of systems equipment and that are relevant to the MISSA project (More Integrated Systems Safety Assessment) [11]. The first is the functional safety analysis that identifies amongst other things, the lines of redundancy between system elements that independently provide a specific safety critical function. From this the safety cut-sets are identified. Subsequently, various particular risk analyses are carried out that help identify an installation policy for equipment that need to be kept segregated to protect the claims of independence from being invalidated by particular risk fragments. Likewise, the zonal safety analysis inspects the proposed installation to ensure that other safety rules related to zones and means of installation are respected. This leads to the sets of requirements that are represented within the DMU as segregation, proximity, and orientation constraints:

1. Segregation : distance between two reference points measured in a plane that is projected onto a plane perpendicular to the observation vector from a point of origin

2. Proximity : absolute distance between two measurement reference points

3. Orientation : e.g. above, below, beside.

Fig. 1.2 shows the traditional system safety process with the addition of one block labelled as 3D CV model representation and matching. The explanation above describes how the production safety audit capability fits into the current process.

1.1.1 Process outline and research objective

The overall process outline diagram is shown in Figure. 1.3. Usually before constructing any actual installation, a computer base DMU model of the prototype is created. In the early design phase, the systems installation is defined by geometrical 3D models (DMU).

Check the real world installation against installation constraints segregation, proximity, orientation

| Accurate 3D measurement distance between parts of the installation components with respect to reference surface, angular analysis planes, edges | Discrepancy checking between real world installation and Digital Mockup (DMU) model generated using CATIA Knowledge | Projection of the model, Identify differences, non conformances with reference to original safety requirement intent | Acceptable Installation Volume (AIV), Trajectory testing |

Figure 1.3: Process overview

The DMU approximately defines the shape, the size and the location of each of a systems equipment and routings within the aircraft structure. Through the

evolution of the design process, the 3D models become more accurate, eventually giving the shape of the detailed product. The DMU is used to manufacture the individual parts. These models are ideal representations of the systems installation (It does not show how flexibility due to gravity may affect the installation of wire bundles, etc.) The DMU is used by safety engineer to carry out the particular risk analysis that is mentioned in earlier Sect. 1.1. The result is that installation requirements are encoded within the DMU that need to be respected in order for the fabricated product to conform to the safety requirements. Quality checks are carried out at each stage of fabrication to ensure the fabricated product falls within acceptable tolerances for every measurement and constraint identified by the safety process. Eventually, people involved in testing verify different parts of the actual installation model with that of this reference computerized base model to check for the discrepancy checking. Currently, the safety engineer has to measure the distance between different objects and checks whether it complies the safety requirements. The way industrial technique used is to design drawing doing individual part measurements assemblies using laser measurement techniques. It is able to provide only single point measurement distance unlike in 3D actual representation of the real world.

The above has disadvantages such as that it takes a lot of time to set up, a lot of equipment is needed to collect and process the measurements, the technology is affected by the material that is being measured and by the finish that the material has, e.g.

1. A shiny, reflective surface will reflect light based measurement technique.

2. A black surface will absorb the rays of a light based measurement technique.

To complicate matters, frequently it is necessary to make measurements within areas where the assembly or the DMU is not complete, or where the nature of the material used in the product means that accurate measurements are not possible but rather the part needs to be shown to lie within an acceptable installation envelope, e.g. wire harness attachment points are precisely installed to within a tight tolerance though the actual wire harnesses are installed compared to a much more approximate tolerance not to forget to mention that a wire harness

does not have a regular shape that can have a clear defined datum to work from. Furthermore, manufacturing industries currently rely on 3D scanning techniques.

There is a need to provide a means to facilitate the process of auditing the fabricated product against the DMU features or constraints that have been created to address safety related installation requirements. The new PSA (Product Safety Audit) tool should be robust to all the variables that affect the accuracy of the installation and that facilitates the process of auditing by linking each specific audit point to the original requirement that has raised the constraint that is being checked, both for traceability purposes and also as a means to ensure that the measurement taken makes sense. This same tool can be used to guide an installer to ensure that their work conforms to the design specification, and to motivate them by giving them a clear understanding of the intent behind the requirement that they are trying to conform to. A maintainer or supportability engineer can also benefit from this tool when it is necessary to inspect damaged equipment to ensure that they have identified all aspects of the design that any damage has invalidated.

Using images taken from different perspectives or at different times, 3D and 4D representation of the scene can be built. This in turn can lead to important applications in a wide range of validation tasks. The exemplar or prior knowledge about the structure of industry part component models could be potentially utilized. A goal of this research is to automate the production audit process using computer vision techniques in 3D space.

The research questions formulated from the process outlined in Fig. 1.3 are summarized.

1. How to design a system that performs discrepancy checking between 3D vision model of the actual installation to the original digital mock-up, with a focus on the safety driven installation constraints, such as segregation, proximity and orientation (various means of measurements)

2. How to use the geometry model knowledge for 3D point cloud processing, safety engineer input (query) for production audit ?

3. How to provide various means of measurements : between datum points

10

with reference to planar reference?

4. How to perform projection onto the 3D digital reconstruction of the safety related installation constraints, respecting the original intent of the constraints that are defined in the digital mock-up.

5. How to identify the differences/non conformances that have a relevance to safety driven installation requirements with reference to the original safety requirement intent.

1.2 Contributions and publications

1.2.1 Contributions

The key research idea is the new concept of using semantic DMU knowledge intelligence for discrepancy checking, supporting model based 3D vision analysis for production audit relevance to aircraft SAE safety standard. To be more specific, hypothesis mentioned in Sect. 1.2.1.1 are novel and discussed in the thesis with supporting substantial arguments. An use case scenario is defined which comprises both actual and faulty assembly setup. For example, a component in faulty assembly setup is shifted by 1 cm and the framework need to identify this discrepancy automatically. The semantic information such as component label, size, width, height, color, orientation (rotation, translation) are used for 3D point cloud processing. The concept has been illustrated for solving the discrepancy checking problem in 3D space using DMU (Digital Mockup) knowledge for a sample use case assembly setup (please refer Chapter 4 for more details).

A software package has been developed based on the proposed novel framework that constitute different modules such as calibration, stereo measurement, 3D model computation using domain knowledge, and 3D model fitting.

1.2.1.1 Hypothesis statements

- H1: User input from the safety engineer based on FPSS Focused point SLR (Single Lens Reflex) stereo can be used for accurate 3D measurements

(point, planar, angular) at mm level while ensuring reliability of the measurements using the backprojection criterion (more details refer Chapter 3)

- H2: Use semantic knowledge of the model (intelligence from DMU model) as key information for discrepancy checking (DMU model generation, point cloud segmentation such as color, connected component analysis, pose estimation using datum and object etc.) (more details refer Chapter 4)

- H3: AIV, trajectory, contains, intersect and disjoint test can be performed on the point cloud model to check whether installation satisfies constraints (more details refer Chapter 5)

- H4: Using extra depth information in existing graph cut mechanism, segmentation can be improved.(more details refer Annexure I)

1.2.2 Publications

1.2.2.0.1 Patent Karthikeyan Vaiapury, Anil Aksay, Xinyu Lin, Ebroul Izquierdo, Queen Mary and Westfield University of London; Chris Papadopoulos, Airbus UK Ltd.07379, A Vision Based Audit Method and Tool that Compares a Systems Installation on a Production Aircraft to the Original Digital Mock-Up, with a Focus on Safety Driven Installation Constraints, such as Segregation, Proximity and Orientation Installation Optimization (to be submitted)

1.2.2.0.2 Book Chapter E. Izquierdo and K. Vaiapury, Applications of Video Segmentation, in *Video Segmentation and Its Application*s, K. N. Ngan and H. Li, Eds. New York, pp. 145-157, NY:Springer New York, 2011. (published)

1.2.2.0.3 Journal

1. K.Vaiapury, A.Aksay, X.Lin, E.Izquierdo and C.Papadapoulous, A vision based audit method and tool that compares a systems installation on a production aircraft to the original digital mock-up, *SAE International Journal of Aerospace*, 4(2), pp. 880-892, 2011, doi:10.4271/2011-01-2565.(published)

2. K.Vaiapury, A.Aksay, X.Lin and E.Izquierdo, Model based 3D Vision and analysis for Production Audit purposes, *Infocommunication Journal, Scientific Association for Infocommunications (HTE)*, Vol.3, No.2, pp. 1-8, September 2011, ISSN 2061-2079 (published)

3. K.Vaiapury, A.Aksay, X.Lin, E.Izquierdo and C.Papadapoulous, A new cost effective 3D measurement audit and model comparison system for verification tasks, *Special Issue on 3D Imaging and Video, Multidimensional Systems and Signal Processing, Springer Netherlands*, pp.1-47, issn.0923-6082, DOI: 10.1007/s11045-012-0200-9, 2012 (published)

4. K.Vaiapury, E.Izquierdo and C.Papadapoulous, Linking installation constraints and checking in production audit (to be submitted)

1.2.2.0.4 Conference, Exhibition/Demo, Technical Document

1. K.Vaiapury, A.Aksay, X.Lin, E.Izquierdo and Chris Papadopoulos, A vision based audit method and tool that compares a systems installation on a production aircraft to the original digital mock-up, *SAE AeroTech Congress and Exhibition*, Toulouse, France, October 18-21, 2011 (published)

2. K.Vaiapury, A.Aksay, and E.Izquierdo, GrabcutD: improved grabcut using depth information, *In Proceedings of the 2010 ACM workshop on Surreal media and virtual cloning (SMVC '10)*, ACM, New York, NY, USA, 57-62. (published)

3. K.Vaiapury and E.Izquierdo, A OFDP Framework in Model based Reconstruction, *The 12th International Asia-Pacific Web Conference,(APWEB)*, Busan, Korea, pp.424-429, 6-8 April 2010, doi: 10.1109/APWeb.2010.79 (published)

4. A. Aksay, V. Kitanovski, K. Vaiapury, E. Onasoglou, J. D. Perez-Moneo Agapito, P. Daras, E. Izquierdo, Robust 3D Tracking in Tennis Videos, *Summer School ENGAGE-Immersive and Engaging Interaction with VH on Internet*, Switzerland, September 13-15, 2010

5. K.Vaiapury, A.Aksay, X.Lin and E.Izquierdo, Measurement and Discrepancy Checking in 3D Space, *Digital Shoreditch Festival*, 5-7 May 2011, Queen Mary University of London, London. (http://digitalshoreditch.com/) (presented)

6. Marco Bozzano, Alessandro Ferranti, Karthikeyan Vaiapury, Anil Aksay, Xinyu Lin, Ebroul Izquierdo, Antonella Cavallo, Chris Papadopoulos, Development Description Report for Installation Optimisation Issue B and Development Description Report For Production Audit, *MISSA Project D 3.81 Technical Document*, 2011

1.3 Thesis outline

This thesis is organized into five chapters. In Chapter 1, a brief introduction to the production audit process and existing limitations in an industrial environment is described and discussed. This research work describes the 3D vision assisted production safety audit process that takes measurements of the fabricated sections of an installation from captured images and uses model knowledge to compare the measurements to the required constraints defined within the specification model of the installation in the Digital Mock-Up.

The literature survey made on 3D model based vision techniques related to the production audit is provided in Chapter 2. The survey is presented in two key fields related to this research such as user guided measurement and testing, discrepancy checking of installations. In order to acquaint with the background knowledge, multiview sensing and calibration is discussed.

A proposed framework for 3D PAMT (Production audit measurement tool) that uses input from a digital camera for the verification tasks is discussed in Chapter 3. A 3D based measurement system with capabilities that aims to assist the safety personnel and verify whether the following segregation constraints are duly respected is presented a) distance between datum points of interest, b) distance between points with respect to planar surface. The advantage of optimally using the combination of both point and automatic disparity coupled with planar surface detection is demonstrated.

In Chapter 4, a new cost-effective and robust framework for 3D PACT (Production audit compare tool) that uses input from a digital camera and semantic metadata knowledge available from geometry models which can be used for verification tasks is presented. The discrepancy checking and analysis of CV model with DMU is carried out to conform whether the installation requirements are met. Furthermore, 3D feature extraction and classification has also been studied to complement the verification tasks. Ideally, the framework act as proof of concept for safety analysis and verification and has been tested with a controlled environment dataset for model matching. 3D object structures with respect to other objects position in the scene can be extracted. In future, experiments would be conducted in real industry setup.

The problem of advanced safety installation constraints and its relation to production audit is addressed in Chapter 5. The objective is to link the installation optimisation constraints file to the ACAT (Advanced constraint analysis tool) and to perform an automated check that the installation constraints were respected. The conclusions and the directions of future work are given in Chapter 6. Further, the research contributions that have been discussed in the thesis are briefly summarized.

In Annexure I, the description of user manuals for the proposed production audit software such as 3D PAMT, 3D PACT is provided. Further, an additional software for constraint viewing has been described.

3DHT (3D Hough Transform) for planar surface detection is described in Annexure II. Finally, in Annexure III, a new depth based segmentation technique GrabcutD which is an improvement to existing Grabcut, a graph cut based segmentation method is proposed. The goal is to extract pixel accurate object silhouettes from the multiple views of an object that can then be used to generate 3D convex hulls for the objects. Conventional Grabcut relies only on colour information to achieve segmentation. However, in stereo or multiview analysis, there is an additional information that could be also used to improve segmentation. Clearly, depth based approaches bear the potential discriminative power of ascertaining whether the object is nearer of farer. We show the usefulness of the approach when stereo information is available and evaluate it using the standard datasets against state of the art result.

Chapter 2

Background and existing techniques

The major drawback of the existing production audit process in the aerospace industry is that manual inspection is needed in such complex environments. The aim of the research is to provide an in-depth analysis of the potential of model based computer vision technology to complement the aircraft level modelling and analysis with a focus on system safety assessment. Visual sensing and analysis tools could be potentially deployed for the validation (and potential optimization) of an installation. 3D representations of the objects based on prior known geometry model information can be built in a robust manner for further use in the validation and recognition tasks. In this work, it is aimed to reconstruct the installation parts of a plant assembly using knowledge based CATIA model information and provide the safety engineer, accurate 3D locations of the discrepancies or faults of installation parts thereby promoting the safety analysis and considerable reduction in the time consumption.

During a production audit, these 3D locations can be used to check whether certain installation safety constraints are duly respected. Installation safety constraints are discussed elaborately in Sect. 2.1.2. The target of the work is to create an early working prototype of the concept and to carry out an in-depth analysis of the shortfalls of the prototype, i.e. to discover what is required in order for the prototype to be accepted for intended use. By starting with a prototype

based on the today's existing techniques and adapting them to this concept, the vision of what will be delivered becomes more apparent, as do the limitations of the application that need to be overcome. Once a working prototype is available, it becomes easier to imagine spin-off uses for the application. This task is very well aligned with the overall goal of MISSA [11] since it provides an additional independent (vision-based) method to accelerate the convergence toward an optimal system specification that conforms to the safety constraints. The application can be achieved by building an accurate 3D representation from the multi-view images captured of the actual product and using the addition of semantic metadata originating from the digital mock-up (e.g. description of the primitives that make up the parts within the digital model, such as cylinders, cubes, rectangular boxes, etc.) in order to build the DMU based 3D reconstruction. The 3D reconstruction is then used to replicate the measurement constraints that also exist in DMU in order to test if the measurement constraints still conform to the requirements that they represent. Though the main target application is safety analysis during aircraft manufacturing, the potential outcomes can be used in a wide-range of applications during and after the aircraft building. Example: during an in-service aircraft maintenance inspection of the damaged structure. The tools can be developed that can sense the installation environment in 3D space and able to perform actions such as:

1. Stitching together the multiple scans to form a composite scan of a region of interest.

2. Identification of the basic geometric shapes from the scan data such as: points, planes

3. Various filtering tools to clean the data set of noisy data.

4. Various means to measure the distances or angles between identified objects, e.g. angle between two planes, angle between edges.

There are various ways to automate some of the above functionality.

The objective of this chapter is to understand a) the industrial scenario where the proposed solution primarily needs to be deployed, requirements and limita-

tions, b) literature review in 3D measurement, 3D discrepancy checking, installation constraint checking and analysis with the perspective of production audit process and c) basics of 3D reconstruction. The remainder of the chapter is organized as follows. In Sect. 3.2, we describe the application concept development process. The various requirements and challenges are discussed. In Sect. 2.1, the survey of image processing techniques with respect to industrial requirement is provided. Specifically, the literature review of 3D measurement and discrepancy checking are discussed in Sect. 2.1.3 and Sect. 2.1.4. The literature survey for installation constraints checking is provided in Sect. 2.1.5. A brief introduction about model based 3D vision methods are described in Sect. 2.2. 3D reconstruction methods are discussed in Sect. 2.3. The summary of this Chapter is provided in Sect. 2.5.
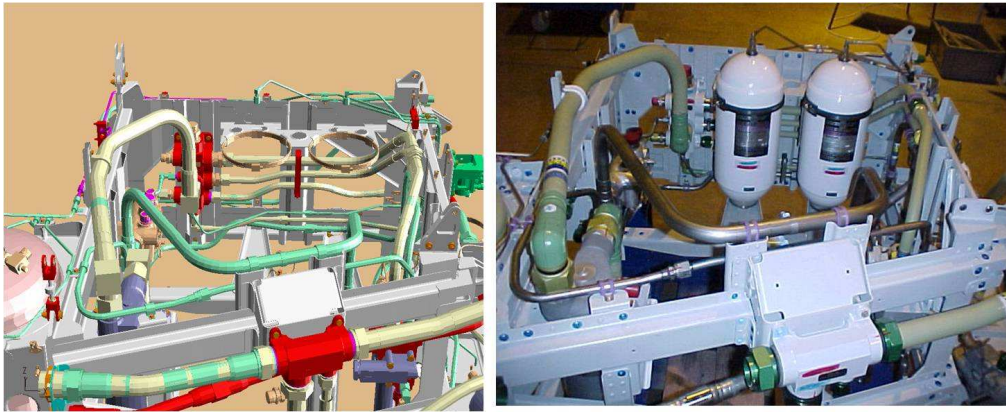
Figure 2.1: Screenshot of a sample model installation (left) and related picture of the actual installation (right)

2.1 Survey of image processing techniques with respect to industrial requirement

A significant initial step in the algorithm development process was to understand which image-processing techniques could be applied and which techniques needed further development before they could be used for the purpose intended. A se-

ries of mock-ups of a measurement scenario were captured and used for each of the stages of image processing needed to perform the types of measurements and comparisons that were intended as a result of the developed application. Various stereo correspondence (disparity) estimation methods with specific attention to extraction of features that are invariant to rotation, scaling, and translation irrespective of the view change was studied. For example, SIFT by Lowe 2004 [52], SURF by Baye 2008 [18] are a few such techniques. A review of available techniques and COTS tools was performed to see if there already existed some of the capabilities that were needed. A number of tools e.g. from Geomagic [8], Faro [7], Kubit [9], were found that went a significant way towards achieving the objectives but none of the tools achieved the complete capability that could verify safety related installation constraints as outlined in Section. 1.1.1. These companies were contacted to see if there was anything on the horizon that might deliver what was needed. The companies indicated that some of the requirements were in the long-term development plans for their products, i.e. after MISSA was finished and some expressed an interest to keep contact in the event that there is an opportunity to collaborate in future activities. As the COTS tools were not open source, it was decided to focus on state of the art techniques from the research community.

2.1.1 MATHSAT3D and basic definitions

MathSAT(3D [21] has essentially been a system that takes in input a set of geometrical shapes S and a set of constraints C and resolves the installation optimization problems over S and C. Each shape described in S can be of one of the following three kinds:

1. Volume: This represents a containment volume V, that is possible volumes where the components can be installed. For each containment volume an initial positioning has always to be specified and it cannot be changed during the solution calculus.

2. Trajectory: This represents a risk trajectory T, that is trajectory from which a possible risk for the installation can be derived (for example, trajectories of fragments derived from an explosion). Also, for each trajectory

requires an initial positioning and it is never modified during the solution calculus.

3. Component: This is an installation component C, i.e. a physical component that has to be installed.

2.1.2 Installation constraints

The constraints lead to the installation optimization which

1. Check the object is installed/setup in an AIV, else risk area

2. Check the trajectory path

3. Check enough distance between the components

It is necessary to ascertain the spatial relationship and that no objects are in the trajectory path (to avoid collision during problems). For each of the constraints, the system needs to be able to check whether it is satisfied. As an illustration
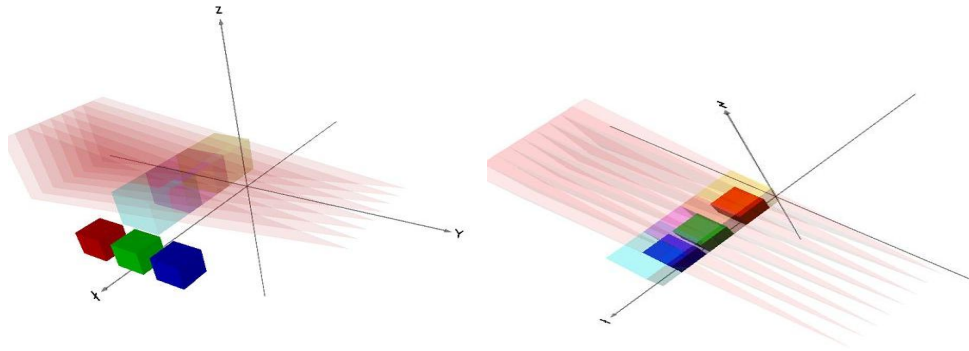
Figure 2.2: Representation of assembly (Mathsat3D)

example shown in Fig. 2.2, the 3 cubes (brightly colored) are the installation components. The relatively bigger size cubes are AIVs. Red color protruding

Tools/features	3D measurement tool palette [3]	Meshlab [10]
F1	Yes	Yes
F2	No	No
F3	No	No
F4	No	No
F5	Yes	No
F6	Yes	No
F7	No	No

Table 2.1: Feature comparison in state of the art measurement tools; F1: distance between positions, F2-planar surface detection, F3-distance between point and planar surface analysis, F4-angular analysis between planes, F5-3D radial dimension, F6-angular analysis between edges, F7-discrepancy checking

shapes are trajectories. Ideally, in an installation environment, every component should satisfy certain safety rules that it is within acceptable installation volume and trajectories.

The survey is presented for the different functionalities such as 3D measurement, 3D discrepancy checking and advanced installation constraint checking that are related in terms of assisting the production audit.

In Table 2.1, a feature comparison study is done for the existing state of the art 3D measurement utilities. MeshLab [10] is an open source and extensible system for processing and editing the 3D models. Adobe Acrobat [3] reads 3D models that readers can move, turn, zoom in on, and examine part by part. It can be noted that none of the tools have all the feature capabilities ($F1 - F7$).

2.1.3 User guided 3D measurement and checking of installations for production audit

The existing methods of measurement systems can be categorized into three broad categories based on scanners, stereo vision, projection using coded structured point light etc (refer Fig. 2.3). Nair has classified 3D vision techniques such as stereo vision, laser triangulation, time of flight and projected light [31]. Nan et al. (2010) [58] have made a study on 3D measurement technology for apparels and accessories. 3D scanning systems are used to capture the point clouds by using a laser-based range finding technique. The point clouds are then used to

analyze the given environment [44].
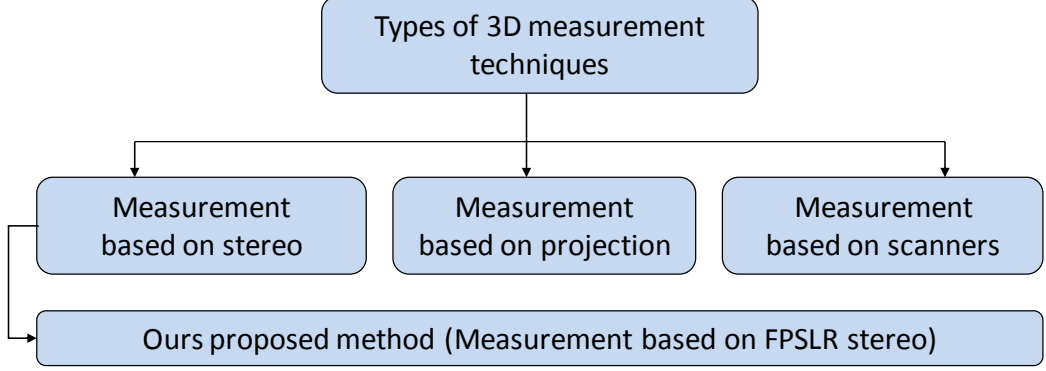
Figure 2.3: Types of 3D measurement techniques

2.1.3.1 Measurement based on stereo

Binocular stereo vision uses two cameras separated by a short distance. For a simple stereo vision system, depth is given by

$$Z = f\frac{b}{d} \tag{2.1}$$

where f is the focal length, b is the distance between two cameras(baseline) and d is the disparity between corresponding points. In order to suit the real world 3D applications, the system need to consider distortion, aberration which requires calibration either using checker board pattern or grid of dots [31]. More details regarding stereo triangulation is provided in Sect. 3.4.1.2 of Chapter 3. Nagatomo et al. (2009) [57] have proposed calibration error-tolerant 3D measurement method using stereo vision which exploits the scaling relationship of triangles. The methodology is used for the systems where the robotic system that vibrates or moves. The relationship between defined depth (H) and measured depth $(H^{'})$ is defined as

$$H^{'} = 0.005H^2 + H \tag{2.2}$$

As mentioned by the authors, system suffer the problems such as a) lens aberration b) close range measurement when the distance between camera and object is small. In general, the images need to have sufficient detail, and the objects sufficient texture or non-uniformity so that features can be identified and differentiated [31]. A way of addressing this problem is illuminating the scene with structured lighting or profilometry system [31].

2.1.3.2 Measurement based on projection (3D Profilometry using phase shifting and multifrequency heterodyne principle)

3D Profilometry system is composed of projector, CCD camera and target objects. 3D measurement can be achieved using viewpoint coded structured light generated by some means of projection to reconstruct the 3D shape of an object (Shi et al. 2009) [87]. Phase shifting and multi-frequency heterodyne principles are applied to the images in order to enable measurement. As stated by Kim et al. 2009 [49], the sinusoidal fringe pattern with single frequency is projected by the projector to target objects. Then the image of target objects together with fringe pattern on the surfaces are captured and saved by CCD cameras. Three cameras are used to obtain multi-frequency fringe images. As stated by Nair 2012, sophisticated algorithms are required to extract depth information and quickly make decisions [31]. It is noted that mostly the works mentioned above and industries rely on scanner technology for inspection tasks. Most of the industrial components such as pipes, wires are not well textured which makes feature detection and matching cumbersome for image processing techniques. Likewise dark or highly polished surfaces of the subject matter make laser based ranging techniques very noisy. This motivated us to prefer user guided 3D measurement process which uses the inspection point of interest from the safety engineer itself as key input for further processing with stereo images. The approach that was developed in this research work was based on image processing techniques that are assisted by user intervention to identify common features where they are not immediately apparent due to the lack of texture. Further 3D measurement system should have the capability to provide measurement with a reference model.

2.1.4 Discrepancy checking of installations for production audit

Discrepancy analysis is a process of estimating how much deviation is between the actual 3D installation and corresponding geometry model. As stated by Kahn et al. 2010 , discrepancy checking is used to ensure whether 3D model matches the real geometry [48]. The discrepancy checking can be broadly categorized into three types a) augmented discrepancy checking 2D space , b) 3D image discrepancy checking via analysis by synthesis approach using ToF cameras and c) ours proposed method: discrepancy checking on 3D point cloud using DMU knowledge via synthesis and analysis approach (refer Fig. 2.4). There exist several approaches for 3D model matching, however the work which is close to our research are Georgel et al. (2007) [38], Kahn et al. 2010 [48]. The authors [38] propose an augmented reality solution for discrepancy check for identifying the differences between a planned 3D model and the corresponding built items in the real world. Initially, anchor plates are used as reference information to obtain pose in the coordinate system of the 3D model (refer Sect. 2.1.4.1). Kahn et al. (2010) perform discrepancy checking using the time of flight cameras which is discussed in Sect. 2.1.4.2.
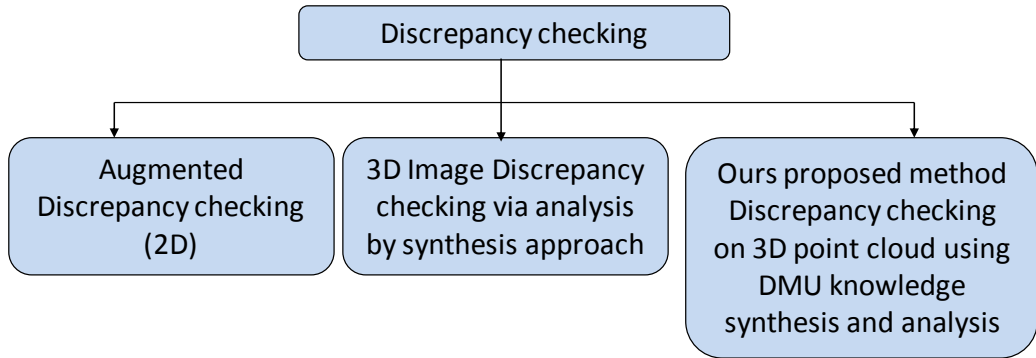
Figure 2.4: Types of discrepancy checking

2.1.4.1 Augmented discrepancy checking (2D space)

Georgel et al. 2007 have performed discrepancy checking using 2D images based on the idea of superimposition of the images from a plant assembly with the CAD model developed during planning phase [38]. The registration process is based on identifying anchor plate and then match with anchor plates. The anchor plate is segmented by the following strategy. Initially, user selects area around anchor plate. The techniques such as a) canny edge detector to detect edges, b) hough transform to reconstruct incomplete borders are used. The homography H is computed using DLT algorithm based on correspondence between anchorplates in image (Ap^{2D}) and 3D (Ap^{3D}). The authors have defined the mean projection error to ensure the quality of homography. Specifically, the following equation need to be minimized.

$$e(H_k) = \frac{1}{4n} \sum_{i=1}^{n} \sum_{j=1}^{4} \left\| Ap_{ij}^{2D} - H_k Ap_{ij}^{3D} \right\|^2 \tag{2.3}$$

where Ap_{ij}^{2D} is the jth corner of ith 2D anchor plate. Ap_{ij}^{3D} is the corresponding 3D point, n is the number of anchor plates. Once pose is obtained, an augmented CAD is created by positioning the image into the 3D view. Upon positioning, a transparency level is used to view the deviation which is estimated using 2D information. Georgel et al. 2007 has used 2D camera images and discrepancy checking is performed in 2D space [38]. However, we use 3D information from multiview images and perform robust matching in 3D space.

2.1.4.2 3D image discrepancy checking via analysis by synthesis (ToF cameras)

Kahn et al. 2010 have performed discrepancy checking via analysis by synthesis approach using the ToF cameras [48]. The method is based on a) synthetic 3D image, b) time of flight camera 3D image and c) difference based on depth value. In order to estimate the pose of ToF cameras, they install an extra camera which is calibrated using Schiller method [48]. They use relative transformation (δR,

δT) between the two cameras.

$$R_{ToF} = R_{Cam2D}\delta R \tag{2.4}$$

$$t_{ToF} = R_{Cam2D}t_{Cam2D} + \delta t \tag{2.5}$$

The depth value d_{cam} of a pixel (p_x, p_y) in 2D image coordinate system is calculated as

$$d_{cam} = \frac{-2z_{far}z_{near}}{z'(z_{far} - z_{near}) - (z_{far} + z_{near})} \tag{2.6}$$

d_{cam} is then transformed to 3D point $p_{CCS}(x, y, z)$ in the camera coordinate system.

$$p_{CCS} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} (p_x - c_x)\frac{1}{f_x}d_{cam} \\ (p_y - c_y)\frac{1}{f_y}d_{cam} \\ d_{cam} \end{pmatrix} \tag{2.7}$$

Euclidean distance is used to calculate the difference between the 3D value of the synthetic 3D image and 3D measurement of the time of flight camera at the same pixel. For visualization of discrepancy, 2D camera image is augmented with semi transparent RGB image. The transparency of each pixel in the difference visualization image is set such that pixels visualizing close distances have a higher transparency than pixels at positions where there is a large discrepancy between the 3D model and the real measurements o is the opacity factor.

$$\alpha_{dist} = \sqrt{(x_r - x_s)^2 + (y_r - y_s)^2 + (z_r - z_s)^2}o \tag{2.8}$$

where (x_r, y_r, z_r) represents the 3D point in synthetic image and (x_s, y_s, z_s) represents the 3D point at corresponding pixel in 3D image by ToF cameras. The colors of discrepancy image is as shown in Table 2.2 and the idea is to set the transparency level of each pixel in difference image such that pixels visualizing close distance exhibit high transparency than pixels at positions where the discrepancy between 3D model and real measurements is larger.

Indeed, there are few commercial applications like Geomagic [8] which at-

3Dvalue	r	g	b	a
$z_r, z_s \neq 0, z_s \geq z_r$	255	0	0	α_{dist}
$z_r, z_s \neq 0, z_s < z_r$	255	255	0	α_{dist}
$z_r = 0$ or $z_s = 0$	0	0	255	α_{noref}

Table 2.2: Colors of semitransparent discrepancy image

tempts for quality inspection using the CAD and scanned point cloud model. The method provides the dimensions and datum information by verifying every feature on the part and comparing it to a 3D CAD model. As mentioned earlier, unlike the existing works, distinct nature of our work is as follows.

1. we use 3D information from the multiview images;

2. we use semantic metadata knowledge available from the geometry models that evolve after installation optimization; and

3. we perform the discrepancy checking and analysis.

The common goal in all works is to identify where each of the individual parts match or doesn't match the geometry model information so called discrepancy identification. This will assist the safety engineer to see and understand where the problems lie and make necessary steps or modifications. Rabbani et al. (2005) has used Hough transform for cylinder detection in 3D point clouds [65]. Attene et al. (2006) has proposed a hierarchical mesh segmentation based on fitting the primitives such as planes, spheres, cylinders etc [16]. Since all installation objects cannot be completely represented by a limited set of geometry primitives, there is a need for a more generic framework which can fit multiple shape and model matching. Further, we have key geometric knowledge information which is already available such as in CAD and CATIA which can be used. This motivated us to design a framework which can a) perform multiple shape matching b) use model knowledge for processing. Ip et al. (2006) has made an exhaustive study on using 3D object classification using different classifiers for discriminating the manufacturing processes [44]. More details regarding classifier usage in discrepancy identification is provided in Chapter 4. With regard to computer aided geometry models, they are usually available in formats such as CAD DXF,

CATIA etc. In this research, we use CATIA which is a format used for conceptualization, design (CAD), manufacturing (CAM), and engineering (CAE). Most of the 3D information used in inspection tasks is obtained either using scanner/ToF. There exists several approaches for the purpose of 3D reconstruction of a given environment (refer Sect. 2.3 for more details). We used PMVS2 (Furukawa and Ponce 2010) [37] for the experiments conducted regarding model matching in this paper. From the reconstructed 3D cloud, individual objects can be segmented based on available semantic metadata knowledge. If the segmented cloud is noisy, further processing is done using component analysis. We also investigate and extract 3D shape discriminative feature information and use in training a classifier which can complement the available geometric knowledge in object identification. Upon the classification of objects, the model alignment can be done using an Iterative Closest Point (ICP) algorithm.

2.1.5 Installation constraint checking and analysis

The problem of conflict or collision between the two equipments and the problem of containment by the correct AIV or the relevant equipment of an idealization are problems that have been solved by FBK [21]. The problem that we address in this research work is.

1. Does the 3D point cloud model of an assembly environment raise a conflict or collision?

2. Is the 3D point cloud model contained by the correct AIV (Acceptable Installation Volume)?

3. Is the 3D point cloud model components are in trajectory path?

Up to our knowledge, there is no work available in the literature so far that checks the 3D point cloud model against the model which evolved after installation optimization process and verify the set of installation constraints such as AIV contains, intersect, disjoint, trajectory are met with a final 3D point cloud assembly. Now, model based 3D vision techniques are introduced in Sect. 2.2.

2.2 Model based machine vision in 3D space

This section discusses model based vision concept available in the literature. Machine vision is a research discipline devoted for machine analysis and recognition started about two decades ago. As stated by Zhao et al 1989, the underlying goals of machine recognition [89] are

1. to see what objects are present in the scene

2. to find precise positions and orientations of the objects

Also, a core general idea of all machine recognition approaches is to hypothesize and verify. Model based recognition is based on the assumption of having a library of models mostly required for training and classification [43]. Often, the representation of model is in the form of geometric properties of objects. Model based recognition is the task of searching consistent matches of the real world model features and geometry model features. Since the study is related to aircraft domain, the related exemplar or prior knowledge that we can better make use of is explored.

Generally, an instance of a generic wide-body aircraft model is assembled out of cylindrical and conical primitives with circular and polygonal cross-section. The central cylinder so called fuselage and two symmetrical cylinders (wings) are usually present in a simplistic airplane model. The target is to build an accurate 3D representation, 3D object structure and recognition using scenes with degrees of complexity similar to sections of an aircraft fuselage. The sample aircraft model with fuselage, wings and rudder component is shown in Fig. 2.5. Further, bright sidewalk region in the upper middle of the image provides strong support for the edges of the aircraft wings [56]. In order to have a complete description of the three-dimensional structure of each model, we should be able to perceive objects that could be constructed out of parts [56]. That is, either one of the cases is true

1. whole object is a transformation (projection) of a prior known model

2. whole object can be broken into constituent parts.
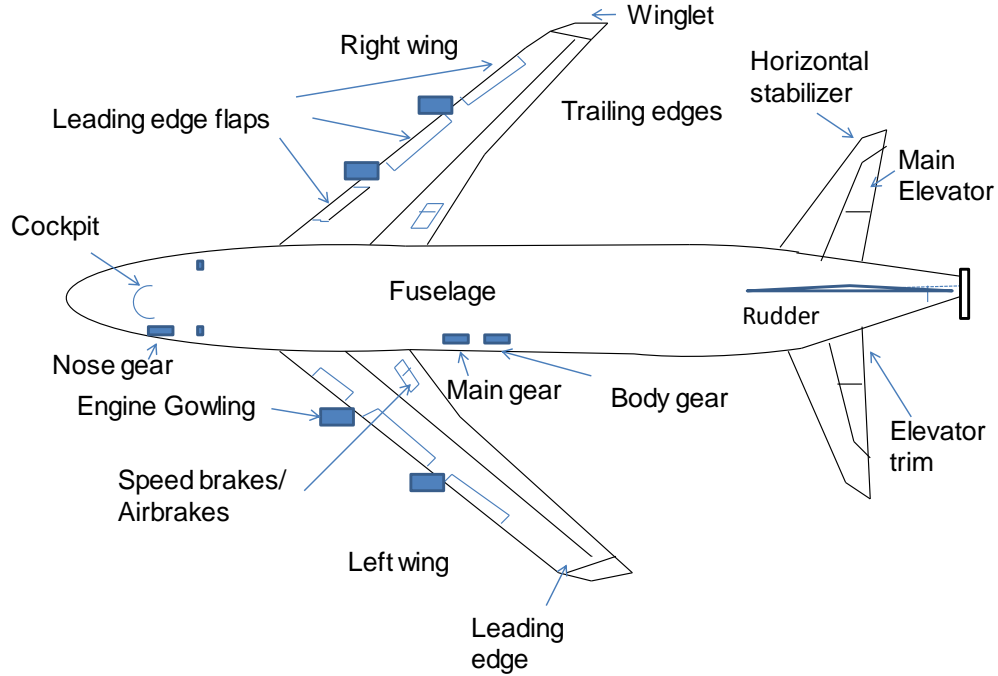
Figure 2.5: Model instance of an aircraft

A sample CATIA model representation of the toyhouse is shown in Fig. 2.6. The main problems in processing this data are segmenting out reliable primitive portions from the data and matching these primitives to those in a stored geometric model of an object [53]. The two methodologies that are available to recognize 3D objects in 2D images are a) shape based (segmentation) and b) texture based (object texture) [39]. Mundy et al. (2006) has done an exhaustive review of key advances in geometric era and enunciated the paradigm shift made from the formal geometry and prior models to statistical learning models based on appearance features [56]. The appearance based method includes calculating edge information, color histogram, texture information etc. Some of the commonly noticed structural shapes include pipes, tubes, cylinders, canal surfaces, and polyhedron etc. The recognition of manufactured parts has been attempted using a planar model [61]. The set of point and curve features are extracted by a bottom up processing methodology. For example, toy house shown in Fig. 2.6 is made up of pyramids and cubes. This can be modelled using an algebraic basis. The combination of geometry and symbolic algebraic constraints can lead
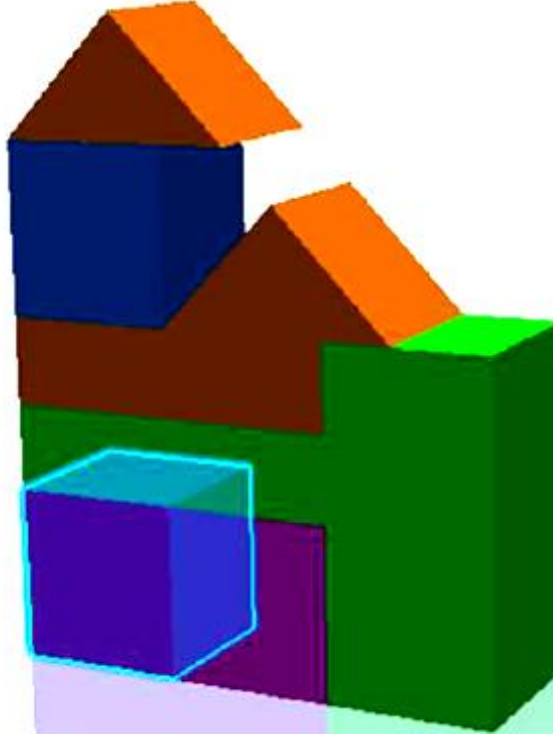
Figure 2.6: CATIA model of a toyhouse

to accurate image interpretations [24]. The target is to make three dimensional interpretations from 2D images (refer Chapter 5 for more details). For example, if the cylinder is represented by a generalized cone, a class of all cylinders of volume 5 (in some units) can be represented by two constraints as shown in the following equation.

$$5 \geq CLR^2\pi \qquad (2.9)$$

Where CL and R are cylinder length and radius respectively. It is also noted that ribbon and GC are also yet another useful representation. The available existing recognition systems include ACRONYM in which a 3D dock model of the submarine is created using generalized cylinder as shown in Fig. 2.7.

A large fraction of the manufactured objects is designed using CAD models and hence described by the primitive geometric elements such as planes, spheres
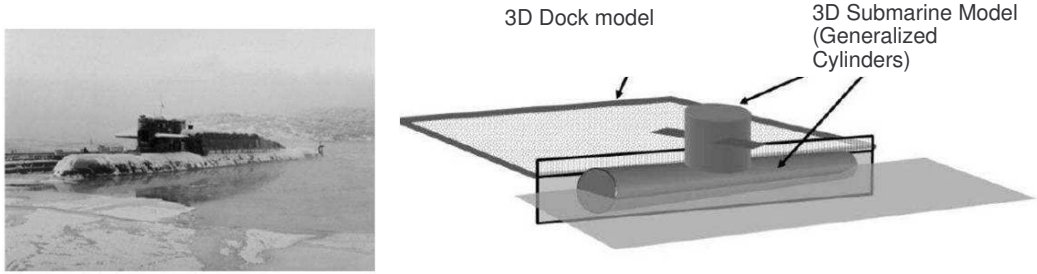
Figure 2.7: Acronym: a submarine and 3D dock model [56]

etc. In fact, more complex shapes could be represented by the geometric descriptions such as triangular mesh or polynomial patches [56]. Model based approaches are also investigated in medical, face recognition domain. Abdelrahim et al 2010 has used shape model priors in medical domain for 3D reconstruction of teeth [15]. Park et al 2007 has used model based face recognition in video based on predefined facial features using active appearance model and face pose estimation [59]. Rabbani et al. (2005) has used hough transform technique for automatic detection of cylinders in point clouds which are obtained from scanners [65]. (refer annexure. II for more details regarding hough transform) Grzegorzek et al. (2010) [39] has used local wavelet features for classification and localization of 3D objects in 2D digital images. The problem addressed is the estimation of object pose in 2D images. It is clearly mentioned in [36] that the recognition performance is not just dependent on the invariant structure but rather better segmentation techniques . A proposed depth based segmentation technique is discussed later in Annexure III. Finding the intensity driven features such as affine patches are vital since invariant regions provide a stable description of the objects and facilitates the reliable segmentation of object parts which would aid the object recognition process. The integration of affine patches with associated geometric constraints lead to impressive performance in complex scenes [56]. Some of the issues that can be considered for better segmentation [36] are:

1. Forming image segments: It is a first step in the recognition process where regions that are of coherent color and texture are partitioned.

2. Fitting lines to the edge points: The machined objects contain lines where the plane faces meet and circles. It is also suggested to find edges in any

image initially and then followed by fitting the lines and circles to them. The estimation of parameters of the lines and correspondence between lines and points need to be established. This method of fitting lines to the edge points is useful for the DMU model generation in 3D space which is discussed in Chapter 4.

3. Fitting fundamental matrix to a set of feature points: This relates to correspondence between a set of feature points under different views.

The whole idea is to effectively use prior or exemplar knowledge for better recognition by removing the outliers. In order to obtain the 3D reconstructed model of the installation environment, we performed a brief literature review in the field of 3D reconstruction of the installations.

2.3   3D reconstruction techniques

The aim of performing this review is to understand the available 3D reconstruction techniques using images so that assembly setup can be reconstructed to perform discrepancy checking. The research goal of 3D reconstruction from multiview images depends on specific domain; however most common requirement always centres around accuracy, admissible time, reliability, clearly defined assumptions, environment conditions and testing with practical real world application. A robust framework for stereo image analysis, 3D modeling and view point synthesis has been provided by Izquierdo et al. (1998, 2000) [46], [47]. 3D structure (eg. structure from motion) can be extracted using disparity results and camera parameters. As stated by Mckinley et al. (2001) [54], there are three approaches to 3D reconstruction using stereo image pairs.

1. The first, and the most familiar, is where both the intrinsic and the extrinsic parameters of the stereo system are known.

2. In the second case, only the intrinsic parameters are available.

3. The third case is the one where both intrinsic and extrinsic parameters are unknown, but a sufficiently large number of 3D object points are known. [27].

The process of recovering 3D shapes from multiple calibrated images can also be called as image based modelling [37]. The commonly used techniques for the 3D reconstruction tasks can be categorized as shape from shading, shape from silhouettes (visual hulls), structure from motion, scanning, kinect based reconstruction, plenoptic capture etc. etc. Shape from shading is the process of recovering the 3D shape of a surface through the analysis of the brightness variation in a single image. Photometric stereo by Vogiatzis et al. 2010 is based on the light source direction and can be used to get high quality reconstructions [85]. The methodology incorporates estimation of light directions coloured light (variant of three source photometric stereo) and intensities. Lambertian surfaces are the surfaces which exhibit diffuse reflectance i.e reflecting light in all directions and its brightness is proportional to the energy of incident light. It can be written as the product of strength of light source E_o, albedo of the surface A and foreshortened area.

$$I_L = R = E_o A cos\theta_i \tag{2.10}$$

where R is the reflectance map which is the cosine of the angle between the unit vector s in the light direction and the normal vector n. θ is the angle between surface normal and source direction. The above equation can be written as

$$I_L = R = E_o A \vec{N}.\vec{S} \tag{2.11}$$

where $\vec{N}$ is the local surface normal and $\vec{S}$ is the light source. This method may be optimal for cases where specific individual objects need to be reconstructed and certain indoor environment such as reconstructing single objects such as toy, statue shapes or even dancer. Seitz et al. (2006) has made a study and classified multi-view stereo reconstruction algorithms under categories such as 3D volume surface extraction, visual hull (space carving), depth maps etc [70]. 3D volume surface extraction is based on extracting a surface from 3D volume which is based on cost function where voxels that cost below a threshold are in the same pass. In order to extract the surface, techniques such as MRF, max flow, multiway graphcut are used.

Visual hull based approaches rely on silhouette and recovered camera informa-

tion. It is also a subset of structure from motion approach. As stated by Forbes et al. 2004, visual hull is usually constructed from the multiple contours and corresponding camera parameters such as camera pose [35]. A sample visual hull of elephant captured using seven silhouettes is shown in Fig. 2.8. The visual hull method includes space carving and level set methods [70]. The space carving method generally starts from initial volume and shrink inward. Visual hull based techniques are based on space carving methods which operates by removing pixels that are not photo consistent. The voxels are deleted based on energy minimization function. Photo consistency measure can be defined in scene space or image
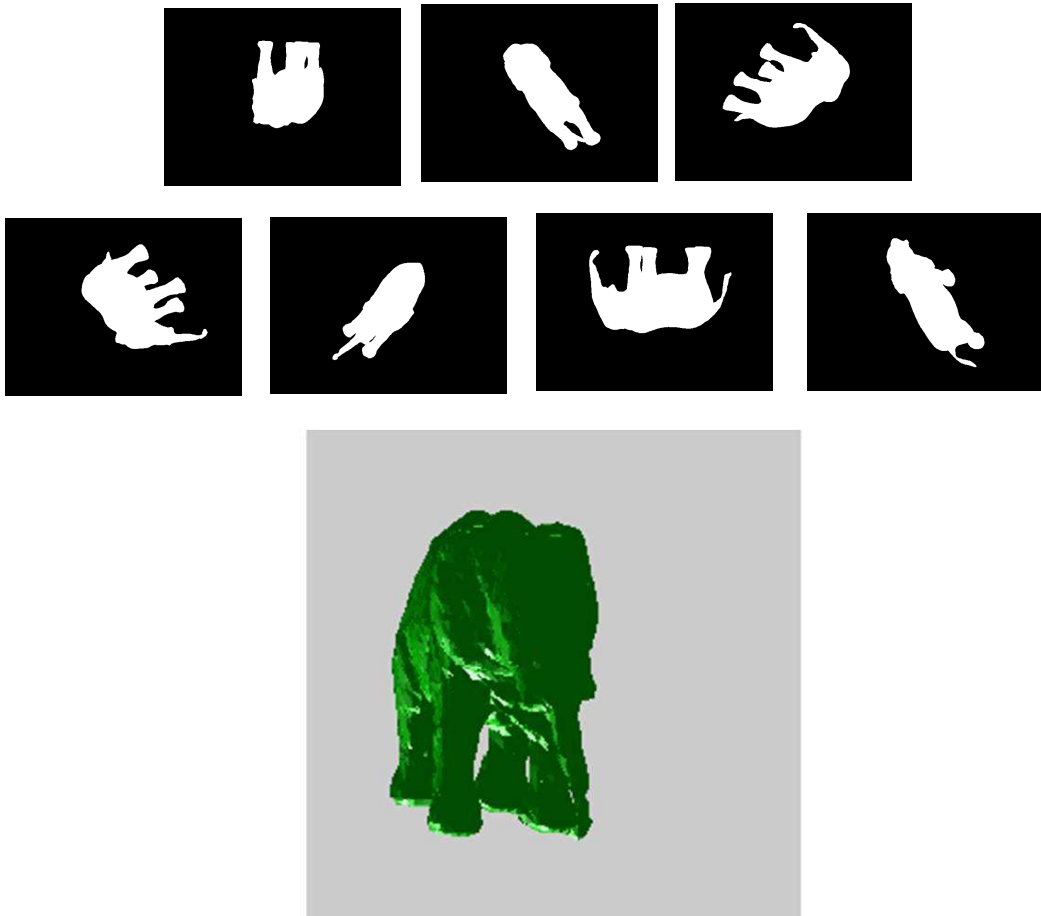
Figure 2.8: Visual hull of elephant [35]

space. Scene space works by taking a point, patch, or volume of geometry, projecting it into the input images, and evaluating the amount of mutual agreement

between those projections. A simple measure of agreement is the variance of the projected pixels in the input images. The other methods compare images using window matching methods such as the sum of the squared differences or normalized correlation. Image space methods use the scene geometry estimate to predict a different view by warping images from one view point. The difference between space carving methods and level set method is that latter can also locally expand if needed to minimize an energy function. However, space carving method starts from a large initial volume and shrink inward. The level set methods minimize a set of partial differential equations based on a volume.

Visual hull techniques are better for reconstruction of the high curvature or thin structures whereas not good for the low surface texture in which surface tends to bulge out [70].

Structure from motion such as PMVS2 by Furukuwa et al. 2010 [37], Insight3d [12] which performs 3D reconstruction from multiview images using key points and auto calibration methods but the outliers need to be removed to make it readily available for a particular application for example surgery, industrial, surveillance etc. The filtering tools using segmentation and model fitting could be used to alleviate the outlier problems. Further instead of using auto calibration methods, photometric calibration can be used which could improve 3D reconstruction quality. (please refer Chapter 4 for more details)

There is also research towards using line or curve based methods than just point based matching at disparity level. Fabbri et al. 2010 used curve based multiview stereo reconstruction based on image curve content to 3D curve sketch [32]. The key difference is unlike usage of traditional interest points, curve information is used to perform reconstruction and bundle adjustment. The explicit curve information can be used where interest points are not present. The line and plane information are also used by Li et al. 2010 [50]. There are many works such as reconstruction from line geometry, curves (instead of points) 3D curve sketches, regularisation of shading, shapes, reconstruction from single image, reconstruction fusing image and range data etc. Creating 3D from 2D line drawings has been accomplished using either depth or plane based approaches whose purpose is to handle more complex objects than state of the art methods. Infact, these techniques have been used in many applications such as cultural heritage for

example, building rome in a day etc.

Plenoptic camera as stated by Shijagurumayum et al. 2010 has an array of microlens and used to record information from all possible view points within lens aperture [71]. The advantage is the estimation of depth information of pixels, no need for geometric, color calibration, frame synchronization etc. The disadvantage includes low resolution due to defocus of microlens image, smaller baseline which leads to the degradation of depth estimation accuracy [60] [71]. The less spatial resolution is due to spatio angular light field sampling [60].

The gap between either high quality object based or structure from motion reconstruction and industrial application level usage is filled by model based knowledge or scene geometry information. In particular, in industries, deviation analysis or collision detection can be performed using stereo reconstruction unlike laser scanners that are still used for testing purpose works such as Chunmei et al. 2009 [29]. As stated by Seitz et al. (2006), all stereo algorithms assume view independent intensities (Lambertian scenes) [70]. While there are features such as color, shape and texture, they have their own limitations. For example, color is not sensitive to the direction and scale changes of an image. Also, texture information is not truly captured when the light or reflection is present whereas shape based features depend on the segmentation. These features are not invariant to rotation, affine distortion, scaling etc.

In this research we use features that are robust over a wide range of affine distortion change in 3D view,noise, illumination change, scaling, rotation etc. The usage of geometric primitive shape knowledge and robust image characteristics (LIFE) are explored. Inspite of the fact that LIFE algorithms such as SIFT (Scale Invariant Feature extraction Technique) [52], SURF (Speeded Up Robust Features) [18] are able to produce only sparse correspondence, they are used since they are robust to geometrical and photometric transformations and invariant towards translation, rotation and scaling. Further SIFT enable high discrimination for finding corresponding points and has been widely used for many vision applications such as non identical duplicate video detection [83]. Morever, the key target is to define a reference planar surface using enough number of distinct key correspondences between stereoviews with a plane fitting algorithm (for more details refer Section. 3.4.1.4).

Bundle adjustment can provide a very accurate sparse estimate [34] of the object structure. 3D information from the multiview images can be obtained using a structure from motion approach such as Bundler by Snavely et al. [73], [74]. There are few works regarding 3D reconstruction from single image (Saxena et al. 2007) [68] and even from single silhouette using a probabilistic generative method. Their strategy is to fuse both the monocular(texture variations, gradients, color, haze) and stereo triangulation cues. et al. 2010) has modeled 3D deformable shape variations and infers 3D shapes from a single silhouette [28]. The main issue is regarding the accuracy of 3D estimation. We need mm level of accuracy in this research (refer Chapter 3 for more details).

2.4 Applications and properties

Though the main target application is safety analysis during aircraft manufacturing, the envisaged study and potential outcomes (3D models with semantic metadata) can be used in a wide-range of applications during and after aircraft building. The following properties are studied in order to build a system prototype and to determine the trade-off between efficiency, generality and accuracy.

1. **Segregation:** Safety constraints need to be respected to avoid effects/impact such as stress and pressure on other machine parts.

2. **Generality:** A system has to be generic and also scope needs to be well defined so that it is suitable for addressing class of real world problems. There needs to be a separation between domain dependent modules and domain independent module of the system.

3. **Representation:** When there are lots of features and constraints, representation has a modular organization that facilitates indexing into model library.

4. **Accessibility:** It is concerned with whether a description can be computed easily given a representation.

5. **Uniqueness:** The class of shapes and uniqueness of the description of each shape is studied and also to find the discriminating power of each shape description.

6. **Sensitivity:** The sensitivity is related to resolution with respect to small variations in shape parameters.

7. **Control:** This refers to use of features and constraints of representation in the core recognition process.

8. **Model validation:** The output from the model is cross checked with reference data from the production environment.

2.5 Conclusions

In this chapter, we briefly described the literature review in key fields related to production audit such as a) 3D measurement, b) 3D discrepancy checking and c) linking installation constraints to production audit. Firstly, it is noted that most of the existing works for production audit and industries rely on scanner technology for inspection tasks. Most of the industrial components such as pipes, wires are not well textured which makes feature detection and matching cumbersome for image processing techniques. Likewise dark or highly polished surfaces of the subject matter make laser based ranging techniques very noisy. This motivated us to prefer user guided 3D measurement process which uses the inspection point of interest from the safety engineer itself as key input for further processing with stereo images. The approach that was developed in this research work was based on image processing techniques that are assisted by user intervention to identify common features where they are not immediately apparent due to the lack of texture. Further 3D measurement system should have the capability to provide measurement with a reference model. Secondly, the common goal in the existing works and commercial applications like Geomagic [8] is to identify where each of the individual parts match or doesn't match the geometry model information so called discrepancy identification. This will assist the safety engineer to see and

understand where the problems lie and make necessary steps or modifications. Unlike the existing works, distinct nature of our work is as follows.

1. we use 3D information from the multiview images;

2. we use semantic metadata knowledge available from the geometry models that evolve after installation optimizationto perform the discrepancy checking and analysis.

In this research, we used CATIA which is a format used for conceptualization, design (CAD), manufacturing (CAM), and engineering (CAE). Most of the 3D information used in inspection tasks is obtained either using scanner/ToF. We used PMVS2 (Furukawa and Ponce 2010) [37] for the experiments conducted regarding model matching in this paper. From the reconstructed 3D cloud, individual objects can be segmented based on available semantic metadata knowledge. If the segmented cloud is noisy, further processing is done using component analysis. We also investigated the use of 3D shape discriminative feature information in training a classifier which can complement the available geometric knowledge in object identification. Upon the classification of objects, the model alignment can be done using an Iterative Closest Point (ICP) algorithm. Finally, the problem of conflict or collision between the two equipments and the problem of containment by the correct AIV or the relevant equipment of an idealization are problems that have been solved by FBK [21]. The problem that we address in this research work is.

1. Does the 3D point cloud model of an assembly environment raise a conflict or collision?

2. Is the 3D point cloud model contained by the correct AIV (Acceptable Installation Volume)?

3. Is the 3D point cloud model components are in trajectory path?

Up to our knowledge, there is no work available in the literature so far that checks the 3D point cloud model against the model which evolved after installation optimization process and verify the set of installation constraints such as

AIV contains, intersect, disjoint, trajectory are met with a final 3D point cloud assembly. In next Chapter 3, we describe the proposed 3D production audit measurement and testing (3DPAMT) approach. A proposed model matching solution (3DPACT) using CATIA knowledge is discussed in Chapter 4.

Chapter 3

3D PAMT (Production Audit Measurement Tool) for industrial verification

3.1 Introduction

The concept of 3D measurement plays key role in the industrial verification tasks and it can be used as post processing step to get the precise measurement once the component having discrepancy is identified. This Chapter describes a concept application to aid a safety engineer to perform an audit of a production aircraft against safety driven installation requirements. The capability is achieved using image capture of a product and measurement of distances between datum points within the product with/without references to a planar surface. The above step gives the safety engineer a means to perform measurements on a set of captured images of the equipment they are interested in. Recently, 3D vision and analysis have attracted manufacturing industries for safety checking and advanced production audit analysis. From industrial safety engineer perspective, the distance between any datum inspection point of interest and/or with reference to planar surface is useful and important to check whether safety constraints are duly respected. In this Chapter, we provide a robust production audit framework

targeting industrial applications using a) user guided input from test engineer (point based disparity), b) optimal combination of focused point and LIFE based disparity, c) outliers removal using back projection and d) distance between 3D measurements with respect to planar surface. The usefulness of the approach is described with substantial results. By assisting safety testing engineer with automated checking and feedback in minimal time, necessary precautionary steps can be taken so that their installation safety and quality constraints are adhered properly. Specifically the hypothesis statement of the Chapter is

H1: User input from the safety engineer based on FPSS Focused point SLR stereo can be used for accurate 3D measurements(point, planar, angular) at mm level while ensuring reliability of the measurements using the backprojection criterion.

The remainder of the Chapter is organized as follows. In Sect. 3.3, we provide the advantages of proposed 3DPAMT. The proposed approach and framework description is provided in Sect. 3.4. In Sect. 3.4.1, we describe 3D distance calculation between points of interest and 3D stereo triangulation in Sect. 3.4.1.2. In Sect. 3.4.2, we discuss distance measurement between 3D points and planar surfaces. The various techniques for planar detection is discussed and results are compared. The angular analysis between planes and edges are discussed in Sect. 3.4.3 and 3.4.4. The summary of the Chapter is provided in Sect. 3.6.

3.2   Application concept development process

A work-study was performed as a part of the MISSA project by partners in [4] [21] to understand the industrial context and needs. Ad hoc tools were prototyped to test the application of various state of the art techniques and to develop some new techniques to try to achieve the objectives of the project. Meanwhile, COTS light and sound based scanning measurement devices and the recommended software tools were trialled on a wooden mock-up of a physical installation and on various types of equipment [4] [21]. A prototype tool-set was produced and tested on various simple installations to prove the workflow and to validate the ideas behind the toolset. Finally, an evaluation to test the accuracy that was achieved was performed and future directions for improvement of the concept were identified

and discussed with potential future partners [4] [21].

3.2.1 Understanding the industrial context for using the device

In order to have a good idea of the requirements for a device that satisfied the measurement objectives and could be used within an industrial context meant that the necessary stakeholders had to be involved.

3.2.1.1 Visiting the final assembly line

As a starting point, a meeting was organized by [4] with FAL (Final Assembly Line) management and with zonal safety engineers. The meeting with the FAL management, amongst other things, involved a thorough walk through a possible inspection area on a live production aircraft to understand the actual environment.

It was understood that the practical aspects of accurate data acquisition were not the only obstacles. Health and safety, privacy, production process and accountability (what happens if during scanning an area, damage is sustained that is not noticed and so is not reported) concerns were seen as potentially more significant obstacles. This led to the identification of a list of industrial requirements related to the environment that the measurements would be performed in, regarding the objects that would be measured, as well as performance targets regarding achieved measurement accuracy based on the accuracy needed at the FAL for things that needed to be checked. It was recognised that the MISSA 3D production audit process would have to be worked into the official production process for it to be used. The issue of accountability and damage reporting would be considered during such a process integration activity and so would be handled outside the scope of MISSA.

Figure 3.1: Assembly wooden mockup [6]

3.2.1.2 Experience in using the current state of the art equipment and software

Companies specializing in performing measurements using various lasers, white light and ultrasonic scanning equipment were invited by [4] to demonstrate their products on a set of test specimen in order to experience the existing state of the art scanning measurement processes first hand and to see the level of measurement accuracy that could be achieved. The process of performing a scan, regardless of the technology, processing the scanning information so that measurements could be performed, and finally carrying out specific measurements was studied carefully. The various parts of aircraft model is shown in Fig. 2.5. Various devices were used for measuring five objects [21].

1. The leading edge of a wooden mock-up of an aircraft wing and the various installed equipment.

2. A black composite panel.

3. A highly polished shiny metallic wing, wind tunnel model.

4. A transparent stereo lithographic wing, wind tunnel model.

5. A mat painted fuselage panel cut out with the machine cut shiny metal edges exposed, with riveted joints and bare metal riveted repair plates.

The accuracy of measurement using the light and sound based measurement techniques were heavily dependent on the user skill during data pre-processing (cleaning) step before performing a measurement. Once the cleaning step was complete, then during the measurement step, it was just as important that identifiable features were present and correctly selected to perform the measurement. Measurements, such as perpendicular offset from a plane depended on the accurate definition of a plane, making sure that the points that were selected to define a plane did not include outlier points that would give a poor definition of a plane. As long as the various stages of pre-processing and measurement were performed well, then it was possible to achieve significantly better measurement accuracy than what was required for the types of measurement needed. As a consequence of using any of the scanning techniques above, once a measurement process is complete, an engineer is left with not only confirmation of the measure they are interested in but also the raw dataset, the processed data, as well as the actual measure that was performed; and so a complete record of the audit are present that can be visited and used at a later stage to repeat the measurements or to perform modified measurements if there is a need. It was concluded that the duration for the devices to perform a scan was acceptable regardless of the accuracy obtained with the device, but duration to study the environment, to set up and calibrate the equipment, to make sure that the measurements of interest could be performed from the scanning results, all activities that would be performed regardless of which technique was used, actually took much longer. The main factors concerning the equipment were related to health and safety, due to

the use of laser technology, the use of flash equipment, since some industrial environments contain flash sensors for fire protection, and to scanning sensor noise due to the nature of the surfaces that were being scanned. E.g. dark, shiny, translucent surfaces did not scan easily or accurately. The idea of accuracy was also affected by the fact that the aircraft is a structure that changes shape every time that, somebody boards to take a measurement, the temperature changes or more equipment is installed. Each time the flexing of an aircraft is different as the loading conditions evolve as the aircraft is assembled. The trialling activity helped qualify the requirements.

3.2.2 Challenges

There are restrictions that need to be considered while performing a measurement and testing process within a complex installation environment. The safety engineers make measurements within areas where access or space is limited, limited exposure of light, no exposed sparks, low power etc. Furthermore, there should not be any permanent markers and no systems that might have an affect on "health and safety" are allowed. People cannot be captured as well. Hence any equipment used for safety measurement should consider the above requirements. As such only photographic techniques using low cost cameras were considered. The amount of effort involved is much less in terms of logistics required to use a camera when compared to the usage of scanner solutions. There may be discrepancies during the production process which leads to faulty installations. As an example, one can visualize that two cylinders like structures present in an actual installation is not present in the corresponding CATIA or geometric model (refer Fig. 2.1). To be specific, the task beforehand is to design and implement a solution that can a) identify defaults (in this case cylinders) thereby aiding the verification and validation process, b) provide accurate position of the 3D object structures. The project is challenging since it is stated in the literature (Rabbani et al. 2004) [64] that only 85% of the objects in industrial installations can be approximated by CSG primitives such as planes, spheres, cones, cylinders. It would be increased to 95% if toroidal surfaces are considered. Now, survey of image processing techniques with respect to industrial requirement is provided in

Sect. 2.1.

3.3 Advantages of proposed 3D PAMT

The unique features of the 3D Measurement tool are the automated calibration utility, the zoom in option (to help selecting points accurately) and optimal combination of point based disparity and Local Invariant Feature Extraction (LIFE) based disparity which complements each other. These two techniques enable the measurement between any combination of two points, as well as the distance between any points and the perpendicular distance to a defined plane. The portability feature of the digital cameras is worthy factor to be considered since the working environment during production audit is too complex where sometimes even difficult to carry scanners due to constrained free space especially lot of cables. In terms of time complexity, for 3D measurement and analysis, we process only minimal information i.e., stereo to achieve high level accuracy. The algorithm run time analysis and discussion is provided in Section. 6.1.1. A brief case study is provided that demonstrates safety driven installation requirements having been achieved not only in design but have also been maintained through production. The safety driven installation requirements are met through this 3D stereo vision based audit process.

3.4 Proposed approach and framework description

The PAMT framework provided in Fig. 3.2 aims to solve two distinct issues: a) distance between any datum points of interest, b) accurate distance measurement between 3D points and planar surface. In the former method, user provides points of interest with which distance constraints need to be checked. The 3D distance between the points is calculated and accuracy is typically achieved in mm level (refer Table. 3.13, Table. 3.6 for more details). The latter requires components such as plane detection, point based disparity and triangulation. This portion requires reasonable 3D cloud using automatic correspondence to facilitate planar

surface detection. In this part, we perform outlier removal based on backprojection which is discussed in detail in Sect. 3.4.1.4. The plane detection techniques are discussed in Sect. 3.4.2. With thus obtained plane and any user point of interest, 3D distance with respect to planar surface is estimated. The flowchart for 3DPAMT is provided in Fig. 3.3 and would be discussed elaborately in succeeding sections. The screenshot of implemented MISSA [11] 3D PAMT: a complete framework for 3D measurement and inspection and user manual is provided in Annexure I.
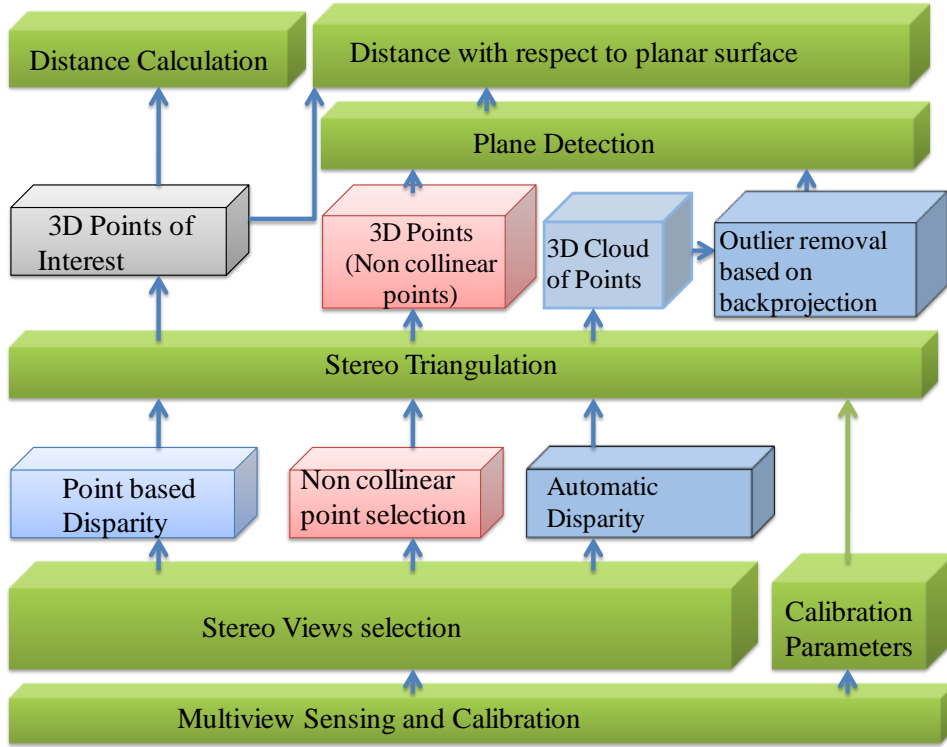
Figure 3.2: MISSA 3D PAMT (Production audit measurement tool)

3.4.1   3D distance calculation between points of interest

3.4.1.1   Calibration

Camera calibration in the context of three-dimensional machine vision is the process of determining the internal camera geometric and optical characteristics
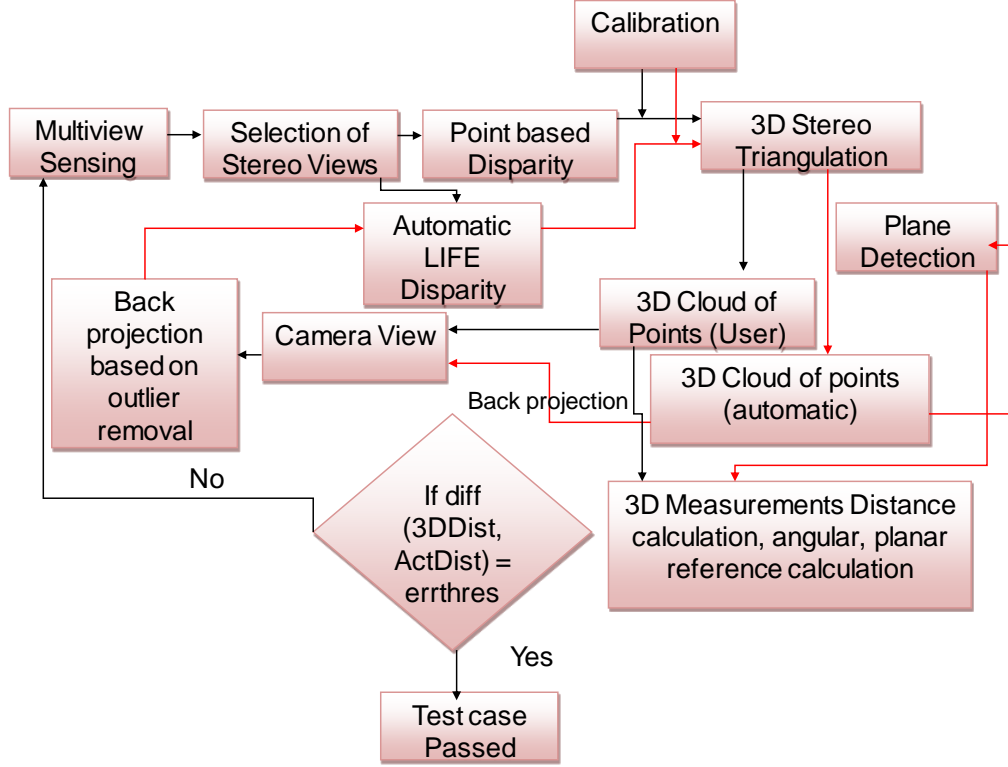
Figure 3.3: Measurement audit component module

(intrinsic parameters) and/or the 3D position and orientation of the camera frame relative to a certain world coordinate system (extrinsic parameters). Calibration is needed for at least two reasons: a) to estimate the epipolar geometry of the system, in order to constrain the matching search along the epipolar lines and b) to reconstruct the 3D structure of objects after correspondence estimation (Izquierdo and Ohm 2000) [47]. Furthermore, it is also stated that although epipolar lines may be estimated for unregistered images using exclusively stereo image information, camera parameters are required in the process of depth reconstruction. Calibration broadly fall under two categories such as as photometric and autocalibration. Photometric calibration uses calibration object such as checkerboard pattern. This can be achieved by using different views of checker board using the same camera set-up. By using structure from variety of angles, the task is to find relative location and orientation of camera for each image and intrinsic

parameters. There are three traditional algorithms for photometric calibration which uses a checkerboard pattern including (Tsai 1987) [79], Heikkila and Silven (1997) [42] and Zhang (2000) [88] calibration method. Auto calibration does not need any calibration object and the camera parameters are estimated from the images. We used Zhang (2000)'s calibration method for estimating extrinsic parameters [88] since the technique requires the planar pattern to be observed at a few (at least two) different orientation and does not need any equipment with two or three orthogonal planes. The method contains the following key steps:

1. The homography is calculated between the image patterns and the model using labeled features using DLT (Direct Linear Transformation) algorithm

2. Estimation of intrinsic and extrinsic parameters

3. Estimation of distortion parameters

4. Refine all parameters using optimization technique

In practice, it is advised to use 7 or more images for better calibration results [22]. There should be at least minimum two orientations (Zhang 2000) [88]. The calibration error is calculated as

$$calib_{err} = \left[\frac{\sum (x_i - x_i^{'})}{n}, \frac{\sum (y_i - y_i^{'})}{n} \right] \tag{3.1}$$

where $x_i^{'}$ and $y_i^{'}$ represent backprojected 2D points and x_i, y_i represent the checkerboard corner points. The number of checker board corners of calibration grid is $n = (width - 1)(height - 1)$, (as only inner corners are used for the calibration process).

3.4.1.2 3D SLR stereo triangulation and backprojection

As stated by (Izquierdo and Ohm 2000) [47], two corresponding points represent the projection onto the image planes of the same object point. 3D position is the intersection of both viewing lines and can be estimated using the coordinates of its projection in both images and the camera parameters. The stereo reconstruction

problem can be formulated in terms of maximum likelihood estimation(inference problem) as follows.

$$\hat{X} = \arg\max \sum_{j=1}^{n_c} \log \Pr(x_j|P, M_j) \tag{3.2}$$

where M_j is the camera projection matrix of the jth camera. $x_{j\,j=1}^{n_c}$ is the projection (z_l, z_r) as shown in Fig. 3.4 in $n_c \geq 2$ calibrated cameras. This can be rewritten as

$$\hat{P} = \arg\max \sum_{j=1}^{n_c} \log \Pr(x_j|P, K_j[R_j|t_j]) \tag{3.3}$$

P need to be manipulated until the predictions $BP(P, M_j)$ agree with the data x_j where BP is the backprojection (refer Sect. 3.4.1.2.1). Triangulation is the key process behind reconstruction which is a process of determining the 3D location of a point by measuring angles to it from known points at either end of a fixed baseline, rather than measuring distances to the point directly. The point can then be fixed as the third point of a triangle with one known side and two known angles. As shown in Fig. 3.4, we retrieve P in space from observed projection $z_l = (u_l, v_l)$ and $z_r = (u_r, v_r)$ onto image planes. u, v represent the coordinate system used in computers or digitized image. It can also be noted that triangulation is not possible if P lies on O_l, O_r or $z_l = z_{el}$ and $z_r = z_{er}$ where z_{el} and z_{er} are epipoles. Given two 3×4 camera projection matrices M_l, M_r and z_l, z_r which represents the corresponding points in stereo images, then mathematically triangulation can be written as a function

$$P = \tau(M_l, M_r, z_l, z_r)_{i=1:n_c}, M_i = K_i[R_i|t_i] \tag{3.4}$$

where i is the index representing the number of cameras n_c. K is the calibration matrix. R_l and R_r represent the rotation matrixes of an object relative to the first camera and to the second camera. The rotation between them R_{lr} can be calculated as

$$R_{lr} = R_r R_l^{-1} \tag{3.5}$$

R_{lr} can be written as

$$R_{lr} = \begin{pmatrix} cos\alpha cos\gamma + sin\alpha sin\beta sin\gamma & cos\beta sin\gamma & -sin\alpha cos\gamma + cos\alpha sin\beta sin\gamma \\ -cos\alpha cos\gamma + sin\alpha sin\beta cos\gamma & cos\beta cos\gamma & sin\alpha sin\gamma + cos\alpha sin\beta cos\gamma \\ sin\alpha cos\beta & -sin\beta & cos\alpha cos\beta \end{pmatrix}$$

(3.6)

where α, β and γ are rotation angles around X, Y and Z axis. Similarly given two translation vectors t_l and t_r, translation between two cameras can be obtained as

$$T_{lr} = t_r - R_{lr}t_l$$

(3.7)

The 3D position of a point P can be reconstructed from the perspective projection of M on the image planes of the cameras, once the relative position and orientation of the two cameras are known. Let $X_l' = (X_l, Y_l, Z_l)$ and $X_r' = (X_r, Y_r, Z_r)$ represent the 3D world coordinate points of point P in left and right camera coordinate systems.

$$U_l = \frac{X_l'}{Z_l} = [u_l \ v_l \ 1]$$

(3.8)

and

$$U_r = \frac{X_r'}{Z_r} = [u_r \ v_r \ 1]$$

(3.9)

are the coordinate vectors of perspective projection of P on the image. X_l and X_r are related by the rigid motion equation as

$$X_l' = R_{lr}X_r' + T_{lr}$$

(3.10)

$$U_l Z_l = R_{lr} Z_r U_r + T_{lr}$$

(3.11)

$$[U_l - R_{lr}U_r]\begin{bmatrix} Z_l \\ Z_r \end{bmatrix} = T_{lr}$$

(3.12)

With each of the two cameras, we get linear equations in unknown coordinates of P, which can be written as $AP = T_{lr}$ where $A = [U_l - R_{lr}U_r]$ is 3×2 matrix involving projection matrix M_l, M_r of the camera [33]. In order to find the best
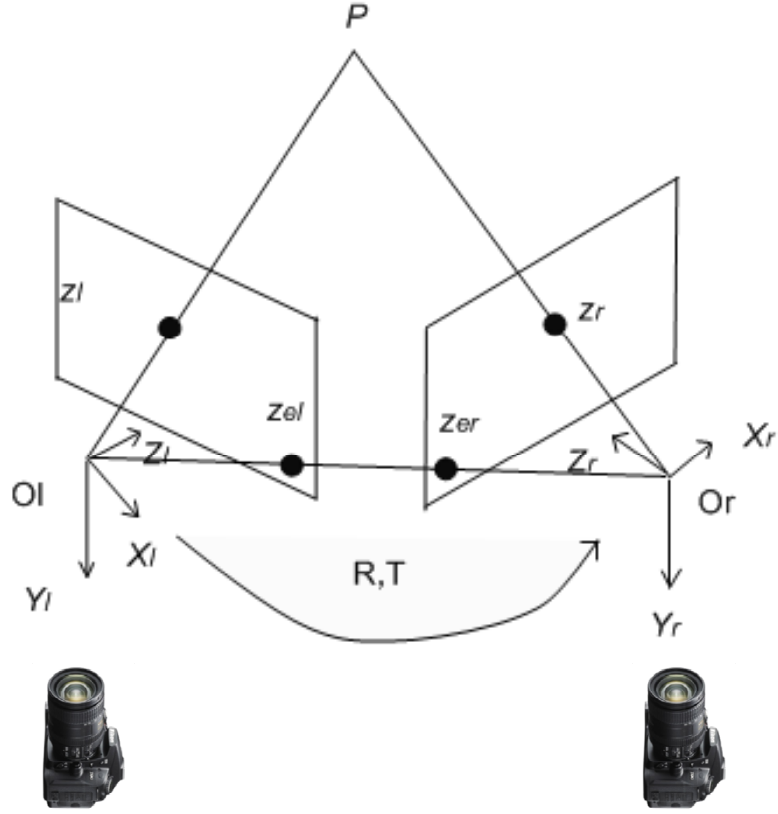
Figure 3.4: Epipolar stereo geometry

reconstructed 3D point, linear method minimizes the criterion $\|AP - b\|^2$ with respect to P. P can be determined as

$$P = (A^T A)^{-1} A^T b \tag{3.13}$$

where $A^T A$ is nonsingular and $(A^T A)^{-1} A^T$ is the pseudo inverse of A.

3.4.1.2.1 Backprojection (BP): 3D points can be projected onto the image plane using perspective transformation which is described as below. World coordinates (X, Y, Z) are transformed to image coordinates using perspective projection. Let $\vec{m} = \begin{bmatrix} u & v & 1 \end{bmatrix}$ be coordinates of projection points in pixels and

$\vec{M} = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}$ be coordinates of 3D world points

$$sm\vec{} = I_m \begin{bmatrix} R^{3 \times 3} & t^{3 \times 1} \\ \vec{0} & 1 \end{bmatrix} \begin{bmatrix} \vec{M} \\ 1 \end{bmatrix} \tag{3.14}$$

$\begin{bmatrix} R & t \end{bmatrix}$ describes camera motion around the object or rigid motion of objects in front of camera. Considering tangential distortion coefficients (p_1, p_2).

$$x_v = x_u + [2p_1 y_u + p_2(r^2 + 2x_u^2)] \tag{3.15}$$

$$y_v = y_u + p_1(r^2 + 2y_u^2) + 2p_2 x_u \tag{3.16}$$

u and v are updated as

$$u = \alpha_x x_v + u_0, v = \alpha_y y_v + v_0 \tag{3.17}$$

x_u and y_u are distorted points; x_v and y_v are corrected points. k_1, k_2 and k_3 are radial distortion coefficients, p_1, p_2 are tangential distortion coefficients, f_x, f_y are focal length in pixel units.

3.4.1.2.2   Correspondence estimation: Disparity is a vector field mapping from one stereo to another (refer Fig. 3.5). Let (x_l, x_r) and (y_l, y_r) be the corresponding points in left and right images respectively. The horizontal disparity is a function of depth $(x_r - x_l)$ and vertical disparity is a function of the camera geometry $(y_r - y_l)$.

3.4.1.3   Point based disparity

Point based disparity is based on an inspection audit by test engineers from the perspective of testing and analysis of any datum points of interest. Initially, the user will select the corresponding points in 2D images. For each selected point, the more focused window of point selected is shown (refer Fig. 3.6). With thus obtained disparity (point of interest from inspection perspective) and calibration parameters, we perform triangulation to get 3D points. Since disparity is obtained
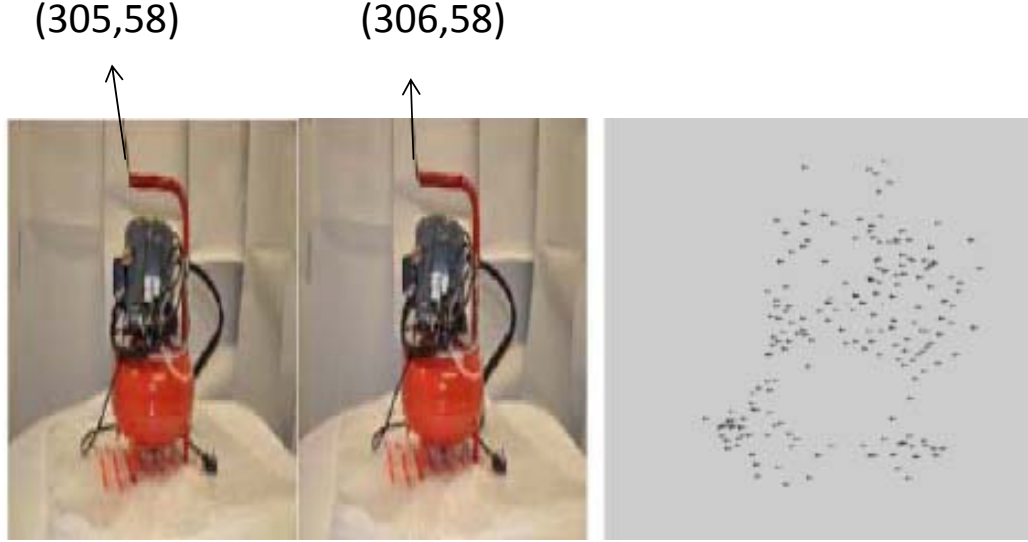
Figure 3.5: Disparity

manually, there is no need for outlier removal and mm level accuracy can be achieved (refer Fig. 3.21 and Table 3.6 for more details). As stated by Takahashi et al 2010, in general the captured stereo image file cannot provide a reasonable measurement result without any additional information since an image file has no information about digital zooming [76]. Images are captured with high resolution digital cameras with size of 2160×3840 pixels. When displaying the image on screen, it is resized to fit on the screen and selecting points with resized images is not accurate enough. In order to solve this problem, we provide a two-step selection process. Let $w = (x1, y1)$ represents the point that user has selected in the first image and corresponding point $x = (x2, y2)$ in second image respectively. In the subsequent window, the user will be shown an image with a portion $(y1 - \delta : y1 + \delta, x1 - \delta : x1 + \delta)$ where $\delta = 150$ in our setup. The points selected now in focused images be $m = (u1, v1)$ and $n = (u2, v2)$ respectively. Then the points x_1' and y_1' for the first image are updated as follows: $x1' = w(1) + (m(1) - \delta)$, $y1' = w(2) + (m(2) - \delta)$ Similarly, the corresponding points in second image are updated as $x2' = x(1) + (n(1) - \delta)$, $y2' = x(2) + (n(2) - \delta)$.
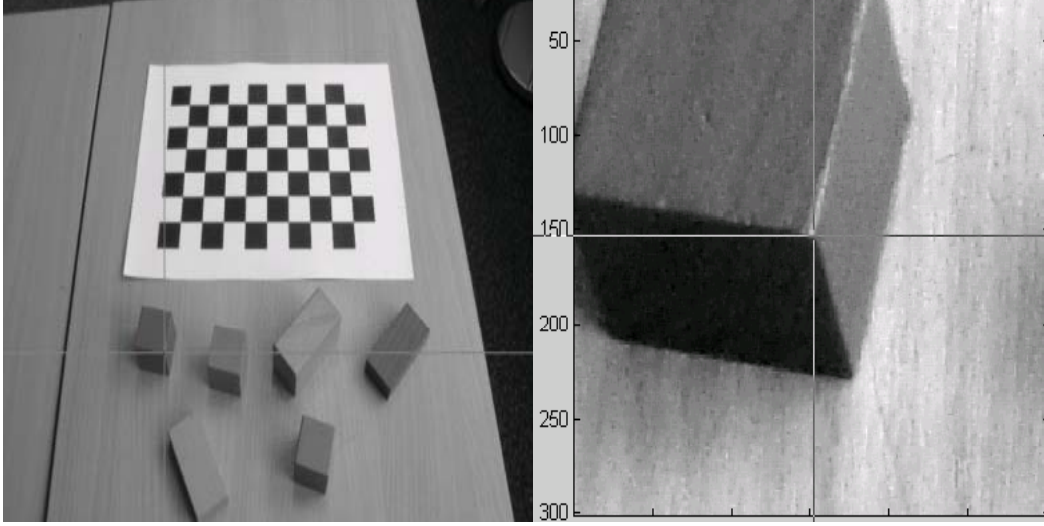
Figure 3.6: Corresponding point selection in stereo images

3.4.1.4 LIFE based disparity and outlier removal based on backprojection

Local Invariant Feature Extraction (LIFE) techniques such as SIFT (Lowe 2004) [52], SURF (Bay et al. 2008) [18] can be used to get auto correspondence. The advantage of this method is its invariant capability towards translation, rotation, scaling etc. (refer Fig. 3.7 to Fig. 3.8 as an example). The aim of getting automatic correspondence is to get enough points to facilitate planar surface detection of the environment which can be used for safety analysis. In this research work, we used SIFT and triangulated with calibration information. In order to ensure all reconstructed points are in line with real world points, we perform outlier removal based on backprojection. The two stereo images and thus obtained 3D cloud is shown in Fig. 3.9. In general, outliers are removed based on fundamental matrix. After backprojection, we calculate the distance between the selected point in the image and the backprojected coordinates. If the distance is above certain threshold, then we reject that point. But, this rejection was added for automatic correspondences. We use Local Invariant Feature Extraction technique such as SIFT (Lowe 2004) [52] for demonstrating the process. Let $D_{c1(2D)} = \{Xc1_i, Yc1_i\}_{i=1}^n$ and $D_{c2(2D)} = \{Xc2_i, Yc2_i\}_{i=1}^n$ represent the arrays of 2D disparity points in two camera (c) views. i represent the index and n is
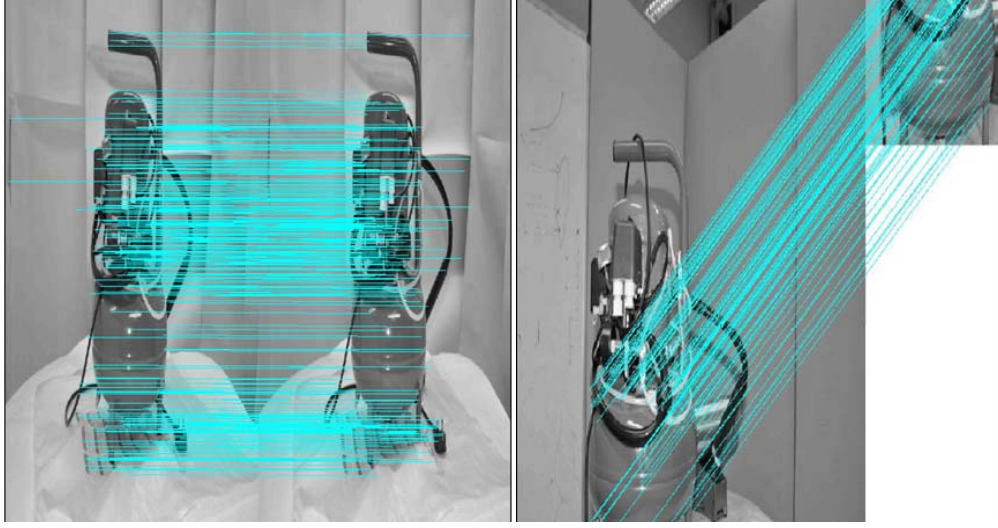
Figure 3.7: Automatic stereo correspondence using SIFT
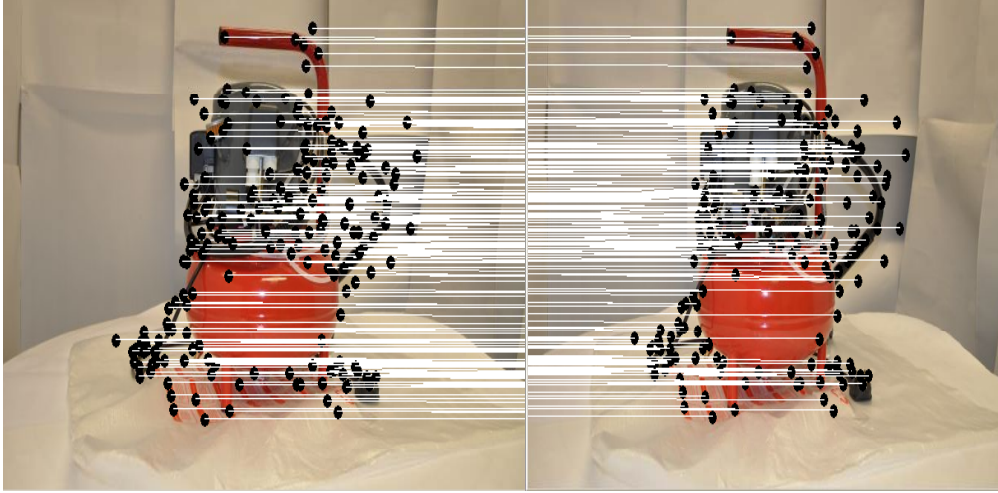
Figure 3.8: Automatic stereo correspondence using SURF

the number of stereo correspondences between images. $c1, c2$ denote the left and right views of stereo images. These points are triangulated to obtain real 3D points $P_{reconst(3D)} = \{X_i, Y_i, Z_i\}_{i=1}^{n}$. After 3D calculation and backprojection, for each of the corresponding camera views, we have backprojected points as D'_{c1} and D'_{c2}. Though we use multiview images (more than 7) in the calibration process as suggested in [22], we only use 2 camera views for correspondence estimation

and measurement tasks. For the stereo camera views, error between actual and re-projected 2D points is calculated as

$$e_{c1} = D'_{c1} - D_{c1}, e_{c2} = D'_{c2} - D_{c2} \tag{3.18}$$

The outlier removed 3D points $P'_{reconst(3D)}$ as

$$P_{reconst(3D)} = P_{reconst}(intersect(k, l)) \tag{3.19}$$

where

$$k = find(e_{c1}(:, 1) < bperr_{thr} \quad and \quad e_{c1}(:, 2) < bperr_{thr}) \tag{3.20}$$

$$l = find(e_{c2}(:, 1) < bperr_{thr} \quad and \quad e_{c2}(:, 2) < bperr_{thr}) \tag{3.21}$$

Now, the outlier removed 2D points can be found using

$$D_{c1}^{or} = D_{c1}(intersect(k, l), :) \tag{3.22}$$

$$D_{c2}^{or} = D_{c2}(intersect(k, l), :) \tag{3.23}$$

Intersect is defined as the set of points in 2D or 3D space that represent common elements in the set. The measurements are not reliable for cases where backprojection error is high. The probability that random data from 3D cloud selected has less backprojection error (bperr) is

$$\Pr(X) = \omega(1 - \phi) + \delta\phi; \phi = \frac{\left| P'_{reconst(3D)} \right|}{\left| P_{reconst(3D)} \right|} \tag{3.24}$$

$||$ represent cardinality of inliers and outliers. ω is the probability of data having less backprojection error given the data is outlier. δ represents the probability of data having less backprojection error given the data is inlier which is 1. Now, the probability of data being outlier given observed (*obs*) data is having less
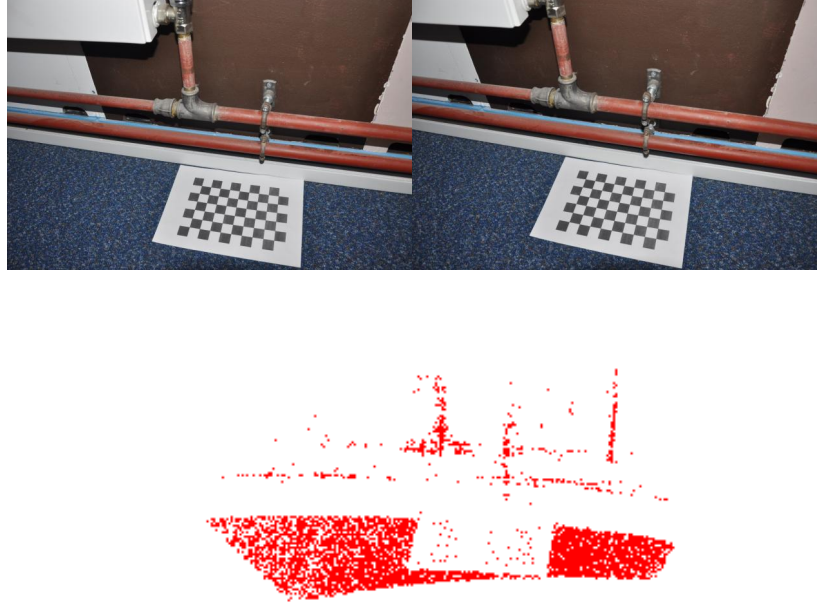
Figure 3.9: Stereo images and point cloud

backprojection error is

$$\Pr(P_{reconst(3D)outlier} \mid obs_{<<bperr}) = \frac{\omega(1-\phi)}{P(X)} \tag{3.25}$$

For example, consider the case of 3D cloud of pipe (Fig. 3.10) with total number of 8105 points, using our model we have inliers of 8091 and 5791 for back−projection error less than 10 and 1 pixels respectively. It can be noted that for later case, inliers 71% and outliers 29%. $P(X)$ as mentioned in equation 21 can be calculated as $(0.5)(0.29) + (1)(0.71)$ which is 0.855. Similarly, the probability of data being outlier can be obtained as $(0.5 \times 0.29)/0.855 = 0.169$.

The sample backprojected result of 3D cloud obtained using SIFT onto 2D image is shown in Fig. 3.11 and Fig. 3.12.

bp = 29, Npts = 8098   bp = 15, Npts = 8093   bp = 10, Npts = 8091

bp = 5, Npts = 8074   bp = 2, Npts = 7182   bp = 1, Npts = 5791
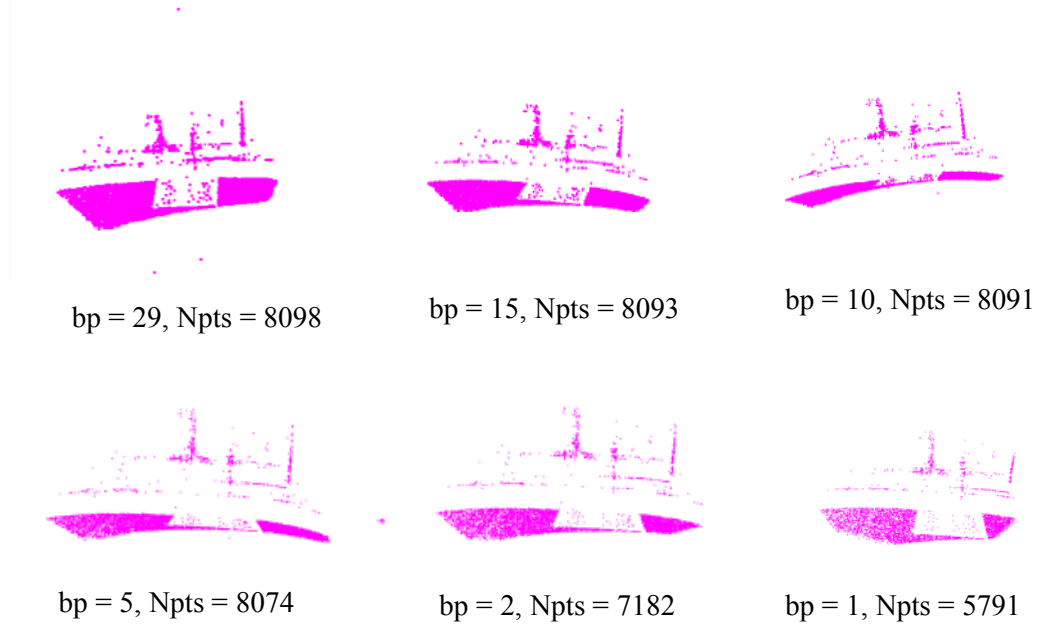
Figure 3.10: 3D cloud result corresponding to different bp errors (bp-back−projection error, Npts-number of 3D points)

3.4.1.5 Experimental results: distance between points of interest

In order to validate the framework, we experimented on different categories of datasets (3DMT) that we created as there is no standard benchmark dataset publicly available that could be used for verification tasks. We have posted our dataset online at [2]. In fact, the MISSA project partner [4] has tested the tool exhaustively on industrial parts such as composite panel, fuselage, metallic wing and have obtained high accuracy. Those results are not included in this thesis due to copyright of the images of industrial parts. In this thesis, we report results with pyramids, basic blocks, pipes, compressor, foyer etc.

3.4.1.5.1 Dataset:Pyramids and basic blocks. The results comprising selected points, 3D cloud of points and backprojected points for pyramid and basic block dataset is shown in Fig. 3.13 and Fig. 3.14. As one can see from the results, the distance between estimated 3D points using a given framework is close to the
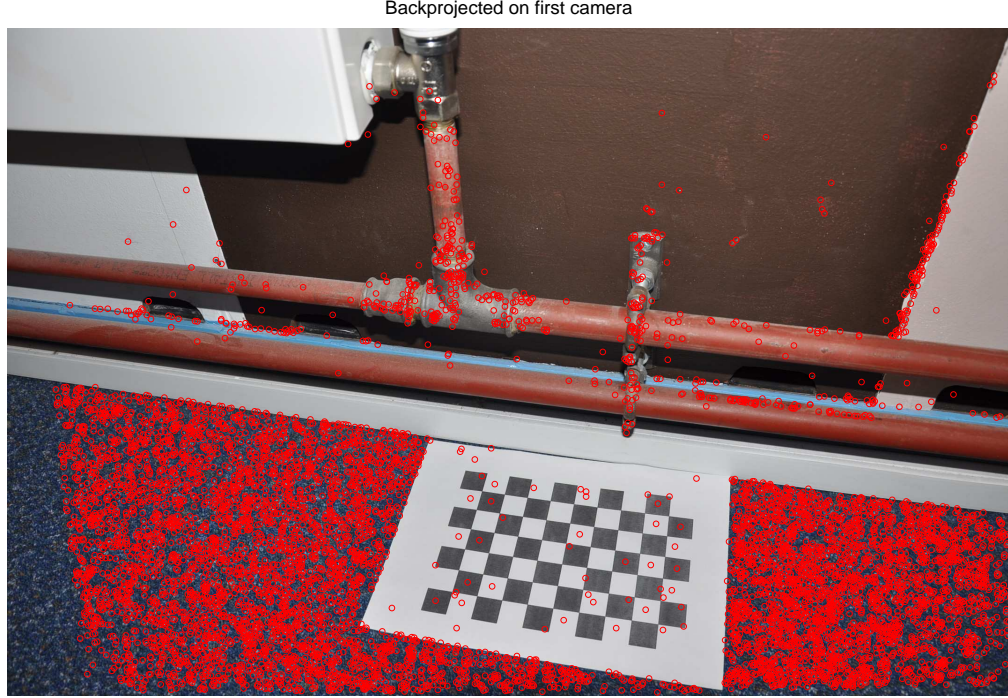
Figure 3.11: Backprojection of 3D cloud onto image 1

real world measurement. The quantitative error (relative or absolute measurement) is provided in Table 3.1 and 3.2. The mean error (in mm) and standard deviation for pyramid dataset are 2.0813 and 1.2704 mm units respectively. In this case, the minimum, maximum and rms value are 0.3000, 4.4900 and 2.4176 respectively. The mean error (in mm) and standard deviation for basic blocks dataset are 0.7685 and 0.5375 units respectively. The minimum, maximum and rms value are 0.1781, 1.6380 and 0.9237 respectively. The general discussion is provided at the end of this section.

3.4.1.5.2 Dataset: Pipes. The actual and estimated distance for grid 7,10 is provided in Table 3.3. We got accuracy of maximum 99.7% for $(P4, P5)$ and minimum 92.9% for $(P6, P7)$. The mean error (in mm) and standard deviation for the points selected in pipe as shown in Fig. 3.15 are 1.2722 and 1.3802 mm units respectively. The minimum, maximum and rms value are 0.1850, 3.8207
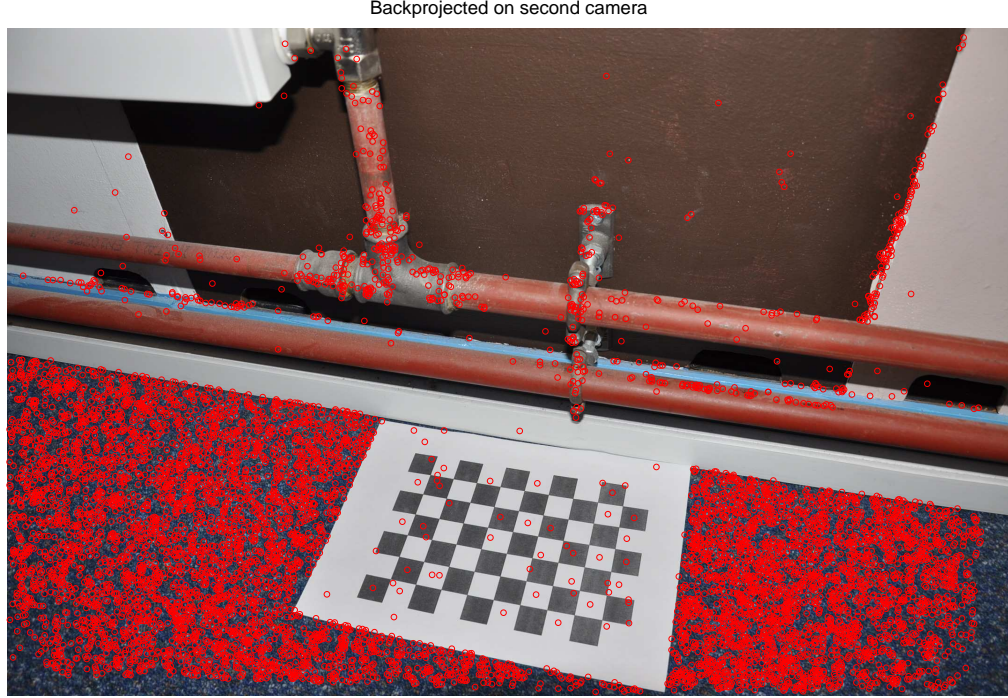
Figure 3.12: Backprojection of 3D cloud onto image 2

and 1.7905 respectively.

3.4.1.5.3 Dataset: compressor engine. The distances between 3D coordinates of various selected points (Fig. 3.16) are summarized in Table 3.4. The mean error (in mm) and standard deviation are 2.1683 and 1.4403 mm units respectively. The minimum, maximum and rms value are 0.1271, 3.9252 and 2.5455 respectively.

3.4.1.5.4 Dataset:Foyer. The distance between set of points selected in the foyer dataset (Fig. 3.17) is shown in Table 3.5. The mean error (in mm) and standard deviation are 1.7451 and 1.2856 mm units respectively. The minimum, maximum and rms value are 0.1072, 3.9760 and 2.1493 respectively.
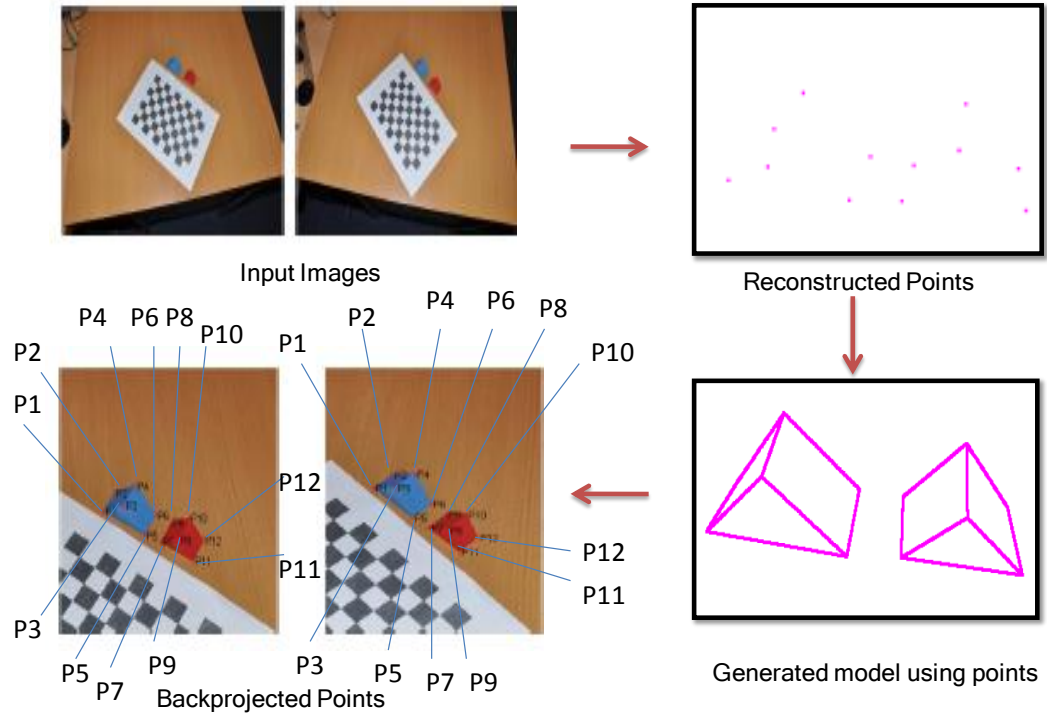
Figure 3.13: 3D points, model, backprojected points of pyramid

3.4.1.5.5 Discussion The overall results of the 3DMT dataset categories such as pyramids, basic blocks, pipes, compressor and foyer are summarized in Table 3.6 and mean error (excluding compressor) is 1.4 mm whereas mean error including compressor is 1.6 mm. The error is within the maximum and minimum range of 3.4 - 0.19 mm (excluding compressor) and 3.6 - 0.18 mm (including compressor). The error comparison graph is shown in Fig. 3.21.

It is observed that the measurement is usually not acceptable for any point combination whose backprojection error (BPE) is higher. For example, backprojection error as shown in Fig. 3.18 for foyer, any measurement associated with point 7 has high error since backprojection error is relatively large and at peak value (7pixels). The backprojection error of the pyramid as shown in Fig.3.19 is relatively lesser within 0.8 pixels. In this case, the user is advised to a) use different combination of stereo images captured from the environment to ensure

Ipts	D1	D2	MEr	Ipts	D1	D2	MEr
P1,P2	30	28.9	1.1000	P3,P5	40	42.69	2.6900
P3,P4	30	30.694	0.6940	P4,P6	40	42.9	2.9000
P5,P6	30	29.14	0.8600	P7,P9	30	31.92	1.9200
P7,P8	30	29.70	0.3000	P8,P10	30	33.94	3.9400
P9,P10	30	28.98	1.0200	P9,P11	30	30.71	0.7100
P11,P12	30	31.48	1.4800	P10,P12	30	33.09	3.0900
P1,P3	40	43.4	3.4000	P1,P5	57	59.18	2.1800
P2,P4	40	42.527	2.5270	P7,P11	40	44.49	4.4900
Average error							**2.0813 mm**

Table 3.1: 3D measurement accuracy of pyramid dataset Ipts- Inspection points, (D1-actual distance(mm), D2- estimated distance(mm), MEr- measurement error)
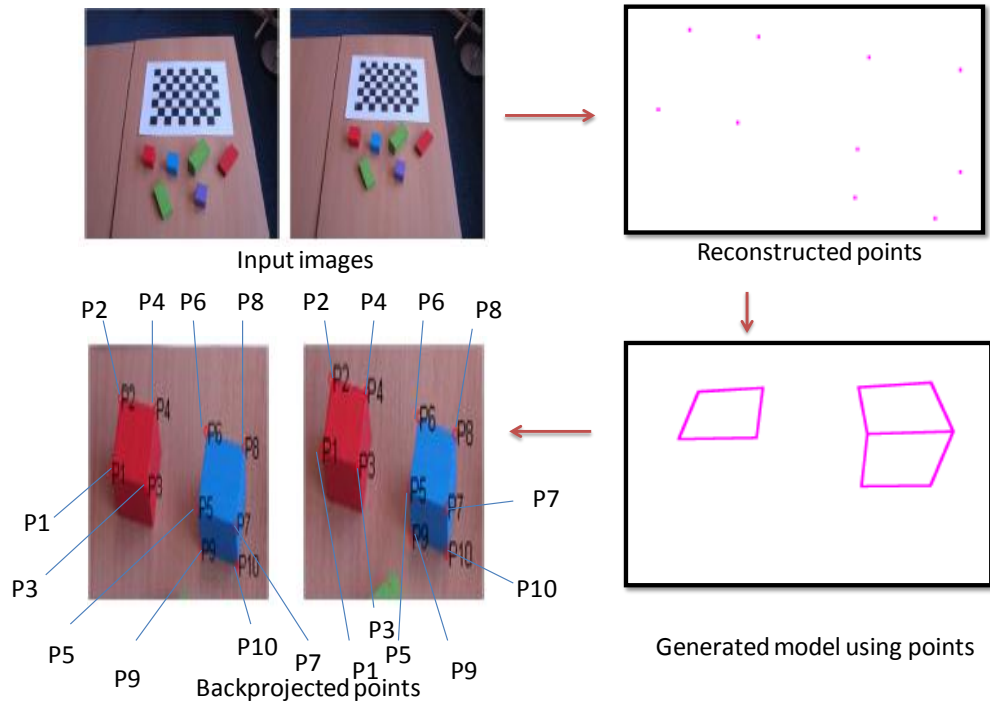
Figure 3.14: 3D points, model, backprojected points of basic blocks

reliability and high accuracy is met, b) take multiview images focusing the object of interest and the calibration pattern for better calibration. Also, the set of

65

Ipts	D1	D2	MEr	Ipts	D1	D2	MEr
P1,P2	30	30.1781	0.1781	P9,P10	30	29.3603	0.6397
P3,P4	30	28.9182	1.0818	P5,P6	30	29.7002	0.2998
P2,P4	30	28.7959	1.2041	P7,P8	30	29.2842	0.7158
P1,P3	30	28.3620	1.6380	P5,P9	30	29.4887	0.5113
P6,P8	30	30.2433	0.2433	P7,P10	30	31.6162	1.6162
P5,P7	30	29.6750	0.3250	P5,P6	30	29.7002	0.2998
Average error							**0.7685 mm**

Table 3.2: 3D measurement accuracy of basic blocks dataset Ipts- Inspection points, (D1-actual distance(mm), D2- estimated distance(mm), MEr- measurement error)
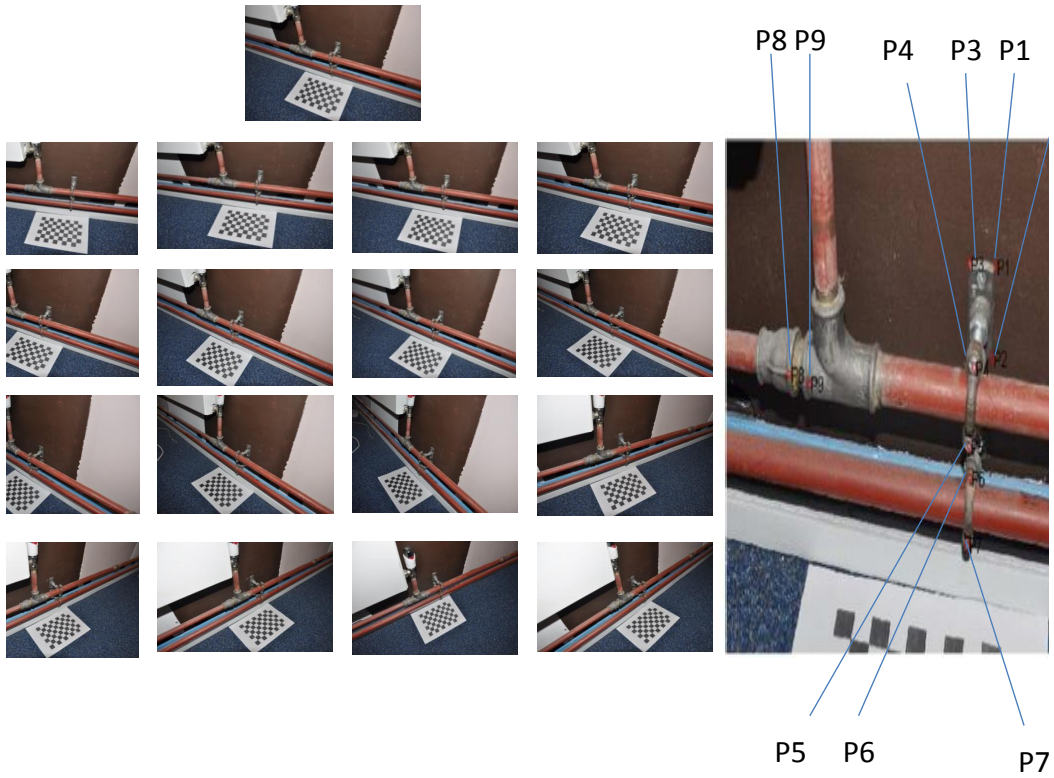
Figure 3.15: Pipes dataset and points selected on images

guidelines for taking multiview images for 3D measurement are summarized.

1. Use the calibration pattern which has board width = 7 and height = 10

66

Ipts	D1	D2	MEr
P1,P2	64 ± 0.5	65.3780	1.3780
P1,P3	28 ± 0.5	29.5734	1.5734
P4,P5	56 ± 0.5	55.8150	0.1850
P5,P6	28 ± 0.5	28.3370	0.3370
P6,P7	54 ± 0.5	50.1793	3.8207
P8,P9	22 ± 0.5	22.3390	0.3390
Average error			**1.2722 mm**

Table 3.3: Actual distance and distance comparison between points of interest (Pipe) Ipts- Inspection points, D1-actual distance(mm), D2- estimated distance(mm), MEr- measurement error
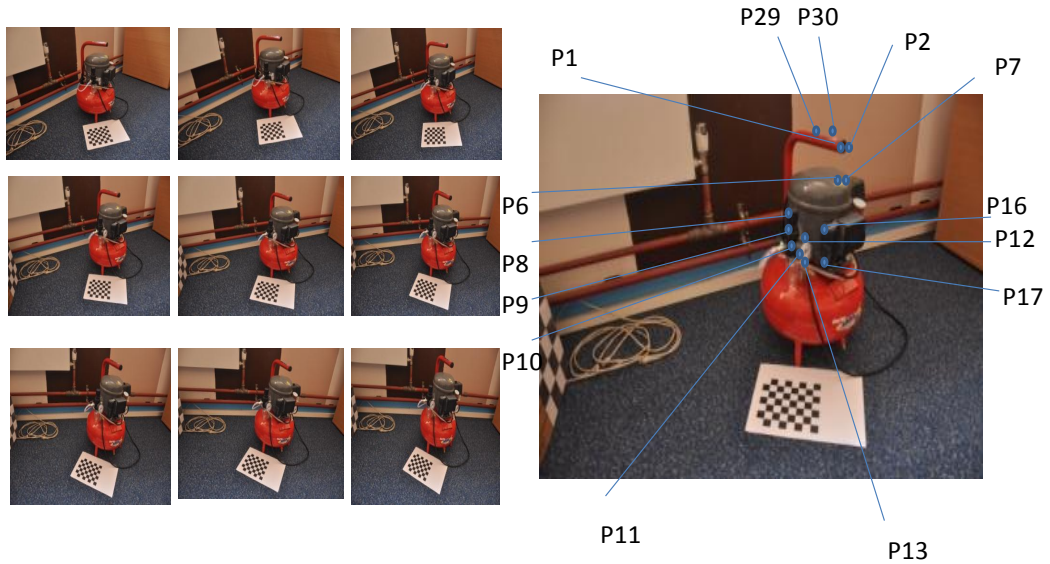
Figure 3.16: Compressor dataset and points selected on images

and dimensions of square 23 respectively.

2. All the squares must be clearly visible (unoccluded).

3. The chessboard must be plane

4. Take photos with camera positions as shown in Fig.3.20

Ipts	D1	D2	MEr
P1,P2	25 ±0.5	26.2795	1.2795
P6,P7	20 ±0.5	23.651	3.651
P8,P9	39 ±0.5	39.8643	0.8643
P10,P11	19 ±0.5	16.2592	2.7408
P12,P13	39 ±0.5	42.9252	3.9252
P16,P17	80 ±0.5	80.1271	0.1271
P29,P30	96 ±0.5	98.59	2.5900
Average error	96 ±0.5	98.59	**2.1683 mm**

Table 3.4: 3D measurement accuracy of compressor dataset Ipts- Inspection points, (D1-actual distance(mm), D2- estimated distance(mm), MEr- measurement error)
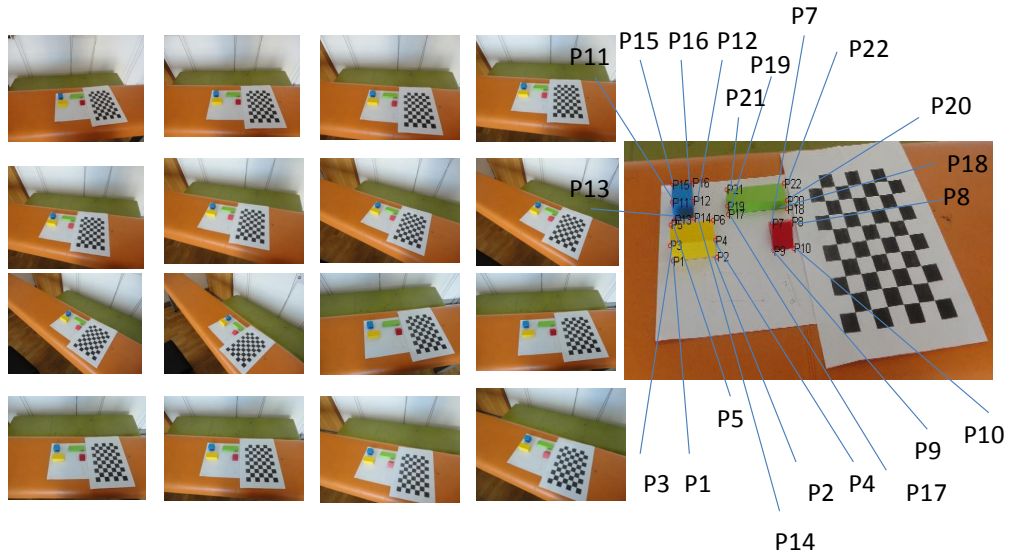
Figure 3.17: Foyer dataset and points selected on images

5. Use a tripod

6. The tilt angle is constant (for example 45 deg)

Also, as shown in Fig. 3.22 to Fig. 3.25, FPSS is much more accurate and

Ipts	D1	D2	MEr	Ipts	D1	D2	MEr
P1,P2	60	61.2416	1.2416	P12,P14	30	28.5565	1.4435
P1,P3	30	29.8928	0.1072	P12,P16	30	30.1227	0.1227
P2,P4	30	26.3269	3.6731	P15,P16	30	31.9356	1.9356
P4,P6	30	29.0957	0.9043	P17,P18	90	93.1355	3.1355
P3,P4	60	61.2520	1.2520	P17,P19	15	16.3776	1.3776
P5,P6	60	62.2860	2.2860	P18,P20	15	14.7329	0.2671
P7,P8	30	33.9760	3.9760	P19,P20	90	93.5801	3.5801
P8,P10	30	32.4982	2.4982	P19,P21	29	28.8006	0.1994
P11,P12	30	31.2943	1.2943	P20,P22	29	29.7181	0.7181
P11,P13	30	30.4805	0.4805	P21,P22	90	93.5472	3.5472
P11,P15	30	32.6068	2.6068				
Average error							**1.7451 mm**

Table 3.5: 3D measurement accuracy of foyer dataset Ipts- Inspection points, (D1-actual distance(mm), D2- estimated distance(mm), MEr- measurement error)

Dataset	Max(e)	RMS(e)	Min(e)	$\mu(e)$	$\sigma(e)$
Pyramids	4.4900	2.4176	0.3000	2.0813	1.2704
Basic blocks	1.6380	0.9237	0.1781	0.7685	0.5375
Pipes	3.8207	1.7905	0.1850	1.2722	1.3802
Compressor	3.9252	2.5455	0.1271	2.1683	1.4403
Foyer	3.9760	2.14931	0.1072	1.7451	1.2856
Average	**3.5700**	**1.9653**	**0.1795**	**1.6071**	**1.1828**

Table 3.6: 3D measurement accuracy overall error comparison

close to ground truth measurements compared to just PSS.

3.4.2 Distance measurement between 3D points and planar surface

Given a 3D point (M_X, M_Y, M_Z) and a plane represented by parameters in normal form, the distance can be calculated using

$$dist = f(M_X, M_Y, M_Z, P_\theta, P_\phi, P_\rho) \tag{3.26}$$
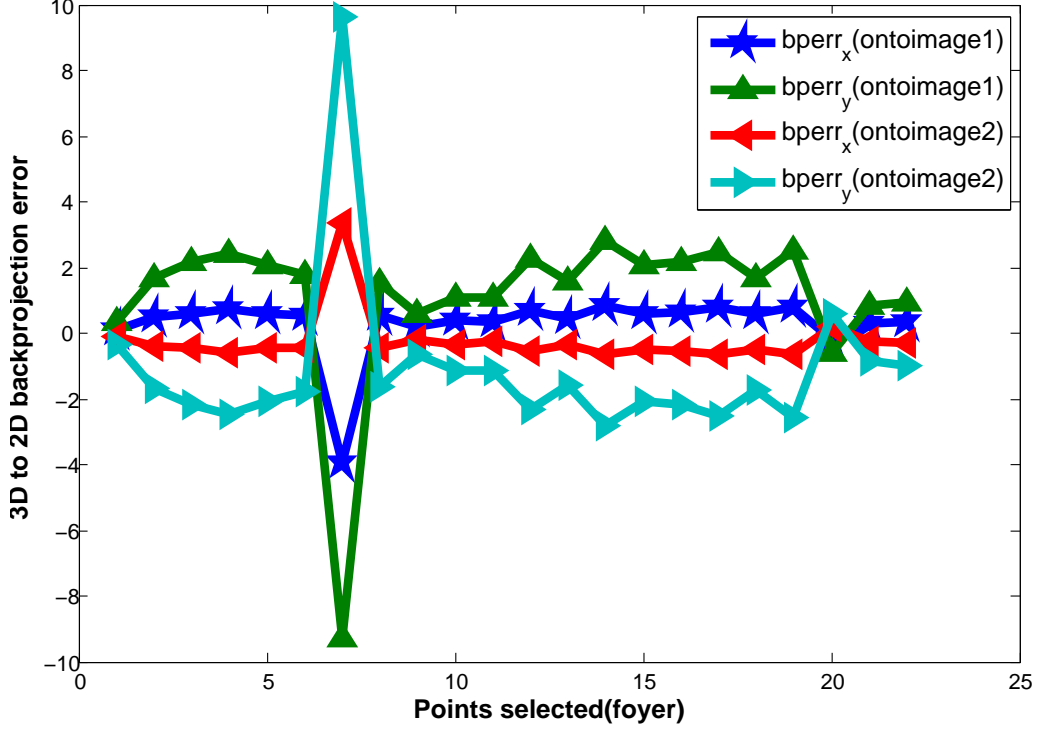
Figure 3.18: Backprojection error of sample datasets (foyer)

$$dist(P, M_X, M_Y, M_Z) = cosP_\theta cosP_\phi M_X + sinP_\theta cosP_\phi M_Y + sinP_\phi M_Z - P_\rho \quad (3.27)$$

where P_θ, P_ϕ and P_ρ are the parameters of the plane normal passing through the origin [refer Fig. 3.26].

As stated by Borrmann et al. (2010), plane extraction or plane fitting is the problem of modeling a given 3D point cloud as a set of planes that ideally explain every data point [19]. Plane can be detected using techniques such as RANSAC (RANdom SAmple and Consensus), 3DHT (3D Hough transform) with 3D cloud obtained using automatic correspondence. The details regarding RANSAC is provided in this Chapter whereas 3DHT is provided in Annexure II.
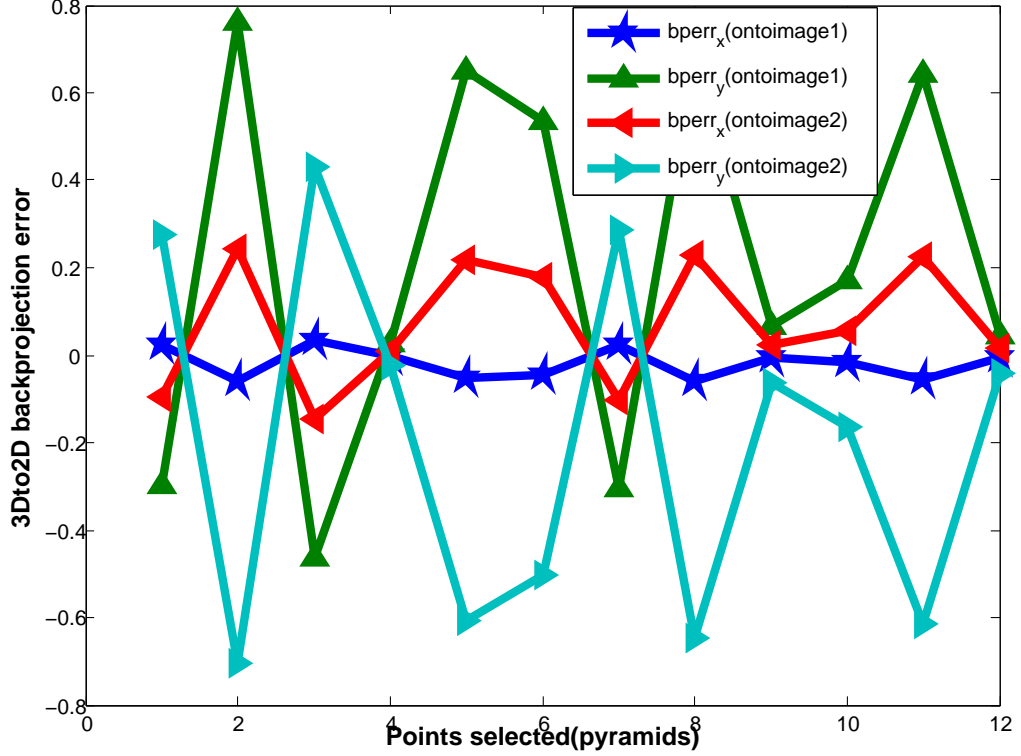
Figure 3.19: Backprojection error of sample datasets (pyramids)

In order to obtain point cloud, we used SIFT correspondence between stereo information of two images and triangulated to 3D. The aim is to detect plane with this cloud and estimate distance between the plane and chosen 3D point. In fact, plane can be detected just using 3 non-collinear points. The latter method requires user to select 3 non-collinear points in stereo in order to obtain the plane.

3.4.2.1 Planar surface detection from noncollinear points (method1)

A plane can be defined just using 3 non-collinear points. As shown in Fig. 3.27, user is required to select 3 non-collinear points in the stereo images in order to obtain the plane. The system of equations $(P_a x + P_b y + P_c z + P_d = 0)$ with the selected points can be solved using Cramer's rule. Let three noncollinear points of 3D data be $Mx1, My1, Mz1, Mx2, My2, Mz2$ and $Mx3, My3, Mz3$. Then the

71
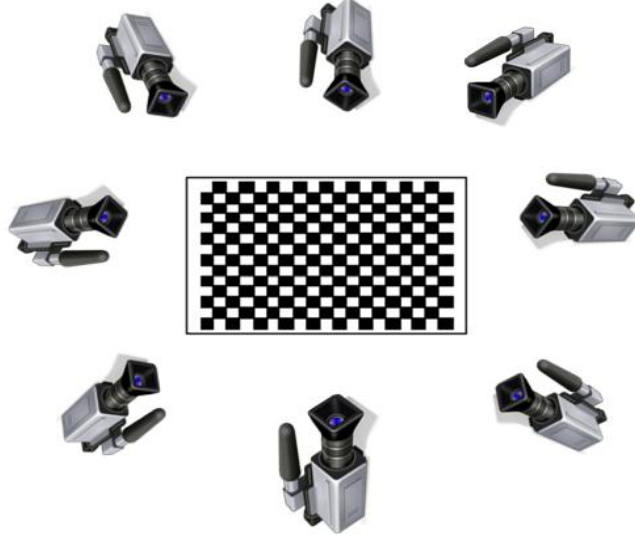
Figure 3.20: Multiview camera positions

parameters of the plane can be obtained as

$$P_\theta = tan^{-1}(P_b/P_a) \qquad (3.28)$$

$$P_\phi = tan^{-1}(-cos(P_\theta)./P_a) \qquad (3.29)$$

$$P_\rho = P_c sin(P_\phi) \qquad (3.30)$$

where P_a, P_b, P_c are calculated as follows.

$$P_a = (\frac{-d}{D}) \begin{vmatrix} 1 & My1 & Mz1 \\ 1 & My2 & Mz2 \\ 1 & My3 & Mz3 \end{vmatrix} \qquad (3.31)$$
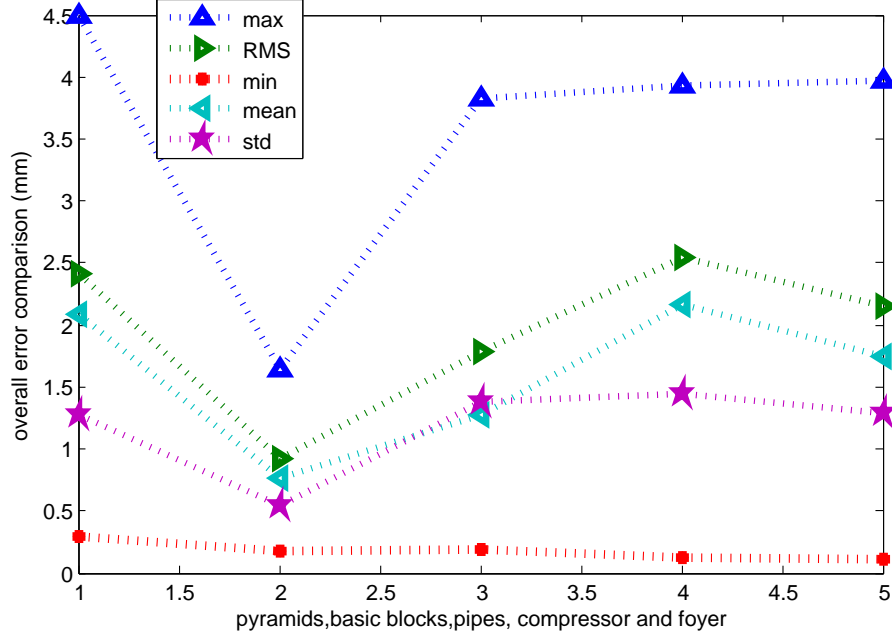
Figure 3.21: Measurement error comparison graph : all datasets

$$P_b = (\frac{-d}{D}) \begin{vmatrix} Mx1 & 1 & Mz1 \\ Mx2 & 1 & Mz2 \\ Mx3 & 1 & Mz3 \end{vmatrix} \tag{3.32}$$

$$P_c = (\frac{-d}{D}) \begin{vmatrix} Mx1 & My1 & 1 \\ Mx2 & My2 & 1 \\ Mx3 & My3 & 1 \end{vmatrix} \tag{3.33}$$

where D is determinant of the 3D data and $d = 2$

3.4.2.1.1 Dataset:Pipes The 3D coordinates obtained from three selected points and the plane parameters are provided in Table 3.7 and Table 3.8 respectively. The obtained plane is shown in Fig. 3.28. The distance between selected point and manual plane is shown in Table 3.9.
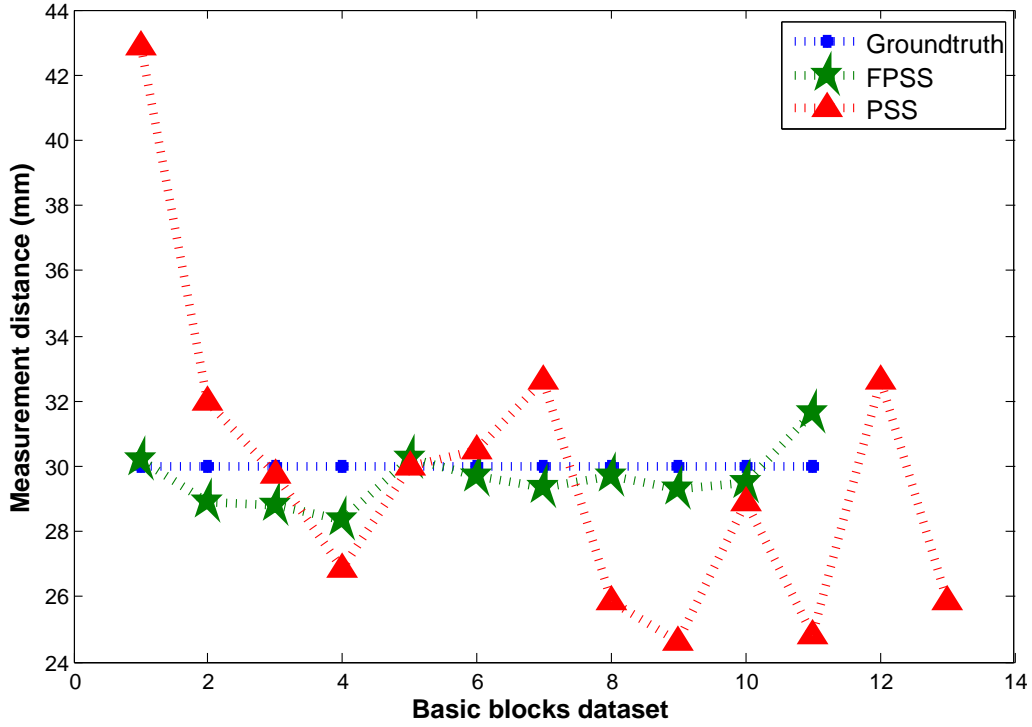
Figure 3.22: Comparison of FPSS and PSS for 3DMT dataset: basic blocks

Points	3D Coordinates
(P1)	(-23.0522,139.108,-3.03038)
(P2)	(-21.7688,-23.7566,-0.202109)
(P3)	(184.81,137.566,-0.372388)

Table 3.7: 3D coordinates of selected 3 points (pipe)

3.4.2.2 Planar surface detection from automatic correspondence using RANSAC (method2)

The RANSAC algorithm is a non-deterministic algorithm that is used to estimate the parameters of a certain model (plane) starting from a set of data. RANSAC is composed of two steps: Hypothesize and test framework.
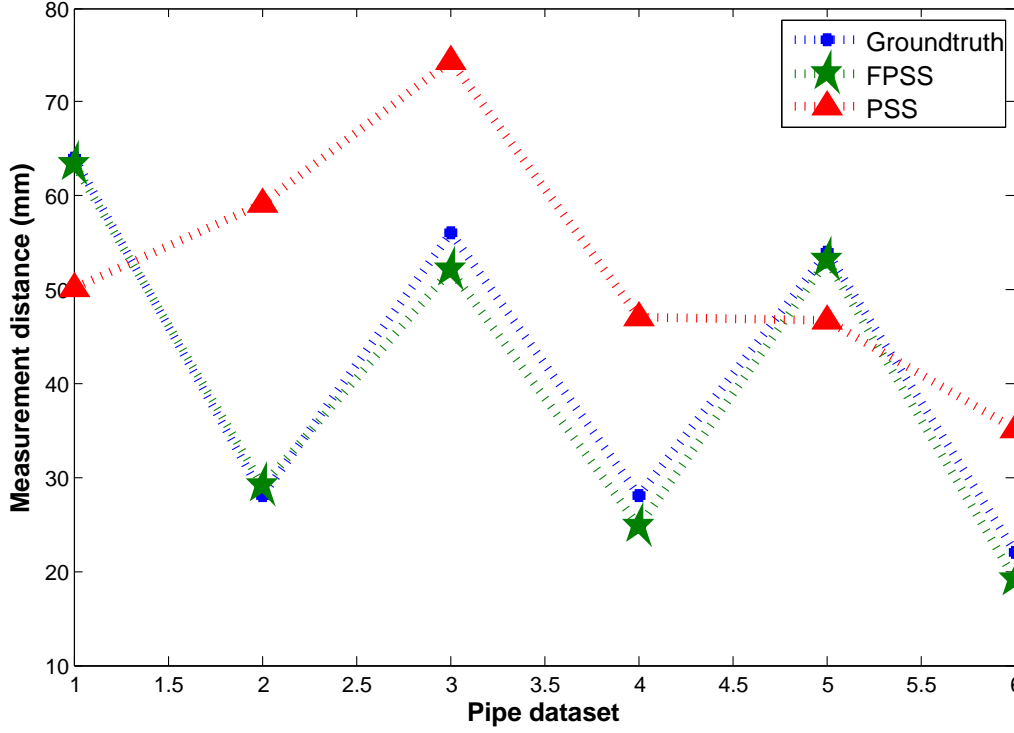
Figure 3.23: Comparison of FPSS and PSS for 3DMT dataset: pipe

Plane Parameters	Values
P_a	-0.0752
P_b	0.1026
P_c	5.9397
P_ρ	5.8923
P_ϕ	1.4443
P_θ	-0.9381

Table 3.8: Plane parameter values for plane from 3 points (pipe)

3.4.2.2.1 Hypothesize: First minimal sample sets (MSSs) are randomly se-
lected from the input dataset and the model parameters are computed using only
the elements of the MSS [92]. The cardinality of the MSS is the smallest suffi-
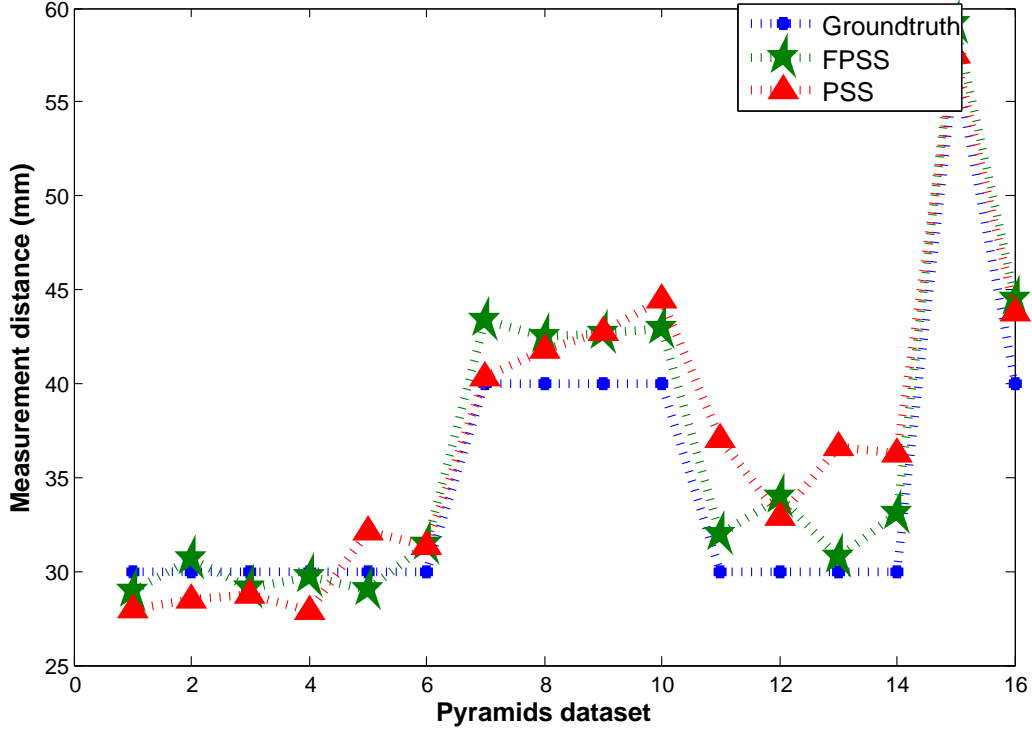
Figure 3.24: Comparison of FPSS and PSS for 3DMT dataset: pyramids

cient to determine the model parameters (as opposed to other approaches, such as least squares, where the parameters are estimated using all the data available, possibly with appropriate weights).

3.4.2.2.2   Test: In the second step, RANSAC checks which elements of the entire dataset are consistent with the model instantiated with the parameters estimated in the first step. The set of such elements is called consensus set (CS). RANSAC terminates when the probability of finding a better ranked CS drops below a certain threshold. In the original formulation the ranking of the CS was its cardinality [92](i.e. CSs that contain more elements are ranked better than CSs that contain fewer elements). Let 3D point cloud (point list) be a matrix of three
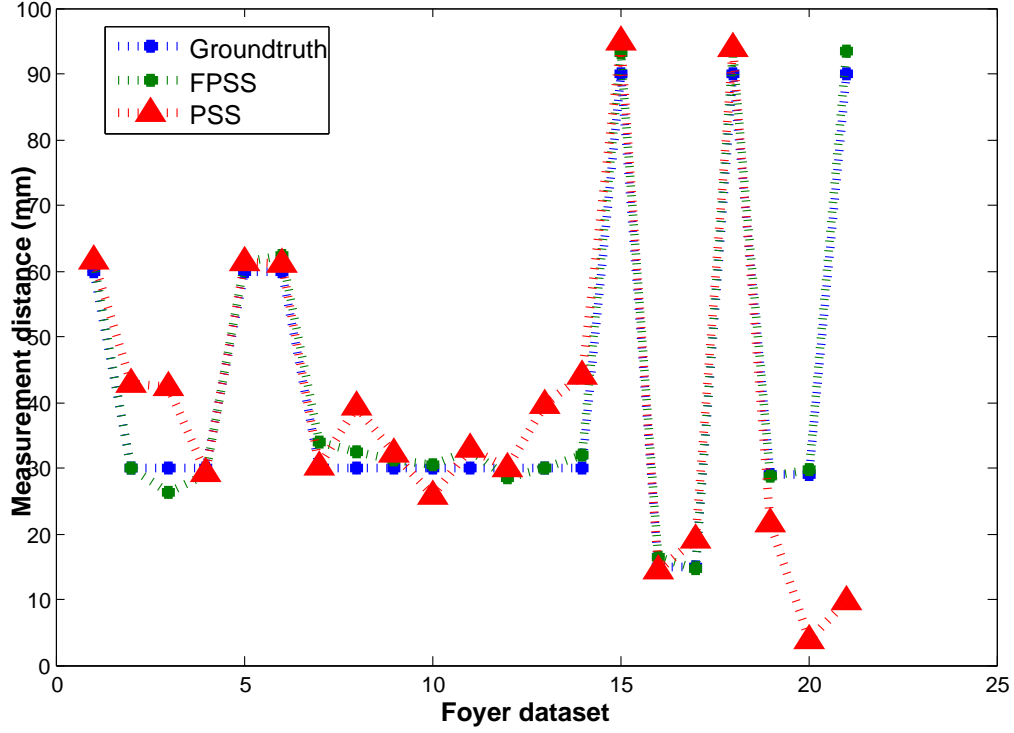
Figure 3.25: Comparison of FPSS and PSS for 3DMT dataset: foyer

coordinate columns X, Y and Z; As stated by Tarsha-kurdi and Grussenmeyer 2007, RANSAC algorithm searches the best plane among the 3D points based on best support criterion [77]. Initially, it selects randomly three points and it calculates the parameters of the corresponding plane. Then it detects all points of the original cloud belonging to the calculated plane, according to a given threshold. The tolerance threshold of distance t between the chosen plane and the other points is used to decide whether a point is an inlier or not. The value of t is related to the altimetric accuracy of the point cloud [77]. Altimetric accuracy refers to how well the data are reconstructed at different altitudes compared to the real world. The altimetric discrepancies would be usually negligible along flat areas compared to sloppy areas [23]. Afterwards, RANSAC repeats these
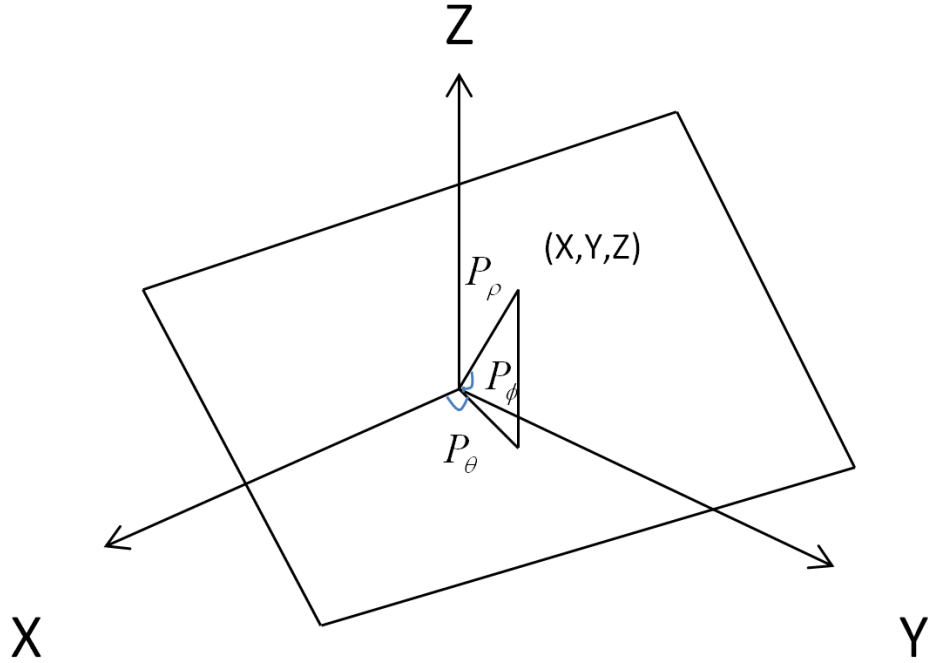
Figure 3.26: Representation of parameters of plane in its normal form

procedures N times; in each one, it compares the obtained results with the last saved one. If the new result is better in terms of the best support criterion [77], then it replaces the saved result by the new one. This plane is obtained by a least squares fit to all the points that were considered to be inliers. The parameters that need to be tuned properly [69] are

1. maximum probable number of points belonging to the same plane and

2. minimum probability of finding at least one good set of observations in N trials. They lie usually between 0.90 and 0.99.

The pseudo code for the RANSAC plane detection algorithm [77] is given in Algorithm. 2.

Figure 3.27: Non collinear points selected in pipe
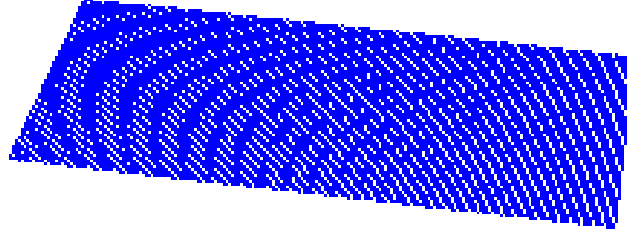
Figure 3.28: Plane from three non collinear points

Let P_a, P_b, P_c, P_d represent the array of plane coefficients obtained by RANSAC method where

$$P_a X + P_b Y + P_c Z + P_d = 0 \tag{3.34}$$

Let X and Y arrays represent grid of points determined by size of the square and step size. The Z array that needs be estimated can be formed as

$$z_{arr} = A x_{arr} + B y_{arr} + C; \tag{3.35}$$

Algorithm 1 pseudo code for the RANSAC plane detection algorithm [77]

1: **while** i $\leq$ N **do**
2:    k = generaterandom(3, 3DpointsArray)
3:    pl = pts2plane(k)
4:    dis = dist2plane (pl, 3DpointsArray)
5:    s = find(abs(dis) <= t)
6:    st = Standarddeviation (s)
7:    if (length(s) > bestSupport or (length(s) = bestSupport and st < bestStd)) then
8:    bestSupport = length (s)
9:    bestPlane = pl; bestStd = st
10:   endif
11:   $i = i + 1$
12: **end while**

where A, B, C are the parameters used for grid generation for display of planar surface specifically z level. x and y are obtained with size of the square and step size.

$$A = -\frac{P_a}{P_c}; B = -\frac{P_b}{P_c}; C = -\frac{P_d}{P_c} \qquad (3.36)$$

Then P_θ, P_ϕ, P_ρ can be found using Eqs. 3.29, 3.30 and 3.31.

The parameters A, B and C can also be determined as follows.

$$A = -cos(P_\theta(\pi/180))cos(P_\phi(\pi/180))/sin(P_\phi(\pi/180)) \qquad (3.37)$$

$$B = -sin(P_\theta(\pi/180))cos(P_\phi(\pi/180))/sin(P_\phi(\pi/180)) \qquad (3.38)$$

$$C = P_\rho sin(P_\phi(\pi/180)) \qquad (3.39)$$

Now, the distance between any given 3D point and plane can be obtained using the following Eq.(4.39).

$$dist(P, X, Y, Z) = cosP_\theta cosP_\phi X + sinP_\theta cosP_\phi Y + sinP_\phi Z - P_\rho \qquad (3.40)$$

where P_θ, P_ϕ and P_ρ are the parameters of the plane normal passing through the origin [77].

Alternatively, it can also be estimated as

$$dist = \frac{|P_a M_x + P_b M_y + P_c M_z + P_d|}{\sqrt{P_a^2 + P_b^2 + P_c^2}} \tag{3.41}$$

The sample point cloud fitted using RANSAC method is shown in Fig. 3.29.
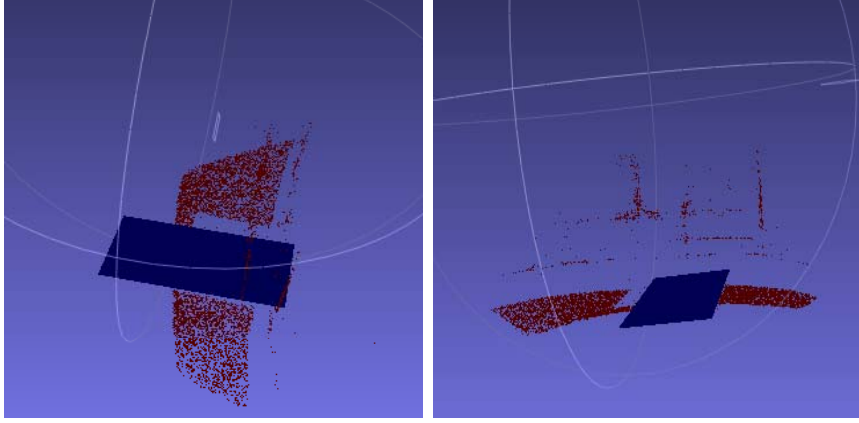
Figure 3.29: Fitted plane: RANSAC (number of vertices: 2011092)

3.4.2.3 Experimental result: distance between datum points and planar surface model

The distances between selected points (refer Fig. 3.30) and plane is provided in Table 3.9. It can be observed from the graphs shown in Fig. 3.31 that in some cases the error is less for both method 1 (1.2248%) and method 2 (0.8938%) respectively. It is worthy to note that method1 is computationally less expensive than RANSAC method as it requires just selection of non collinear points. Further, the mean error in this example is 1.6483 mm for method 1 which is lesser than that of method 2 (2.3214).
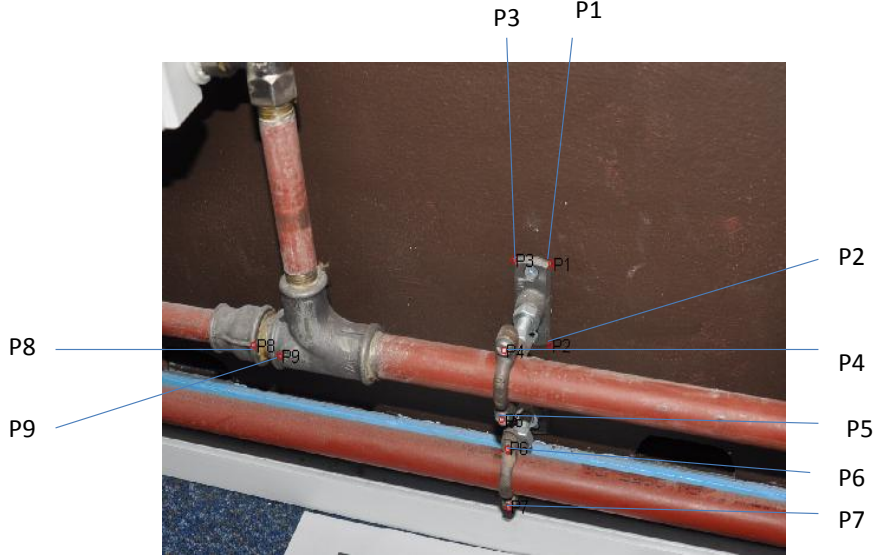
Figure 3.30: Selected point in pipe dataset for estimating distance with reference to plane

3.4.3 Angular analysis between user defined planes

Given the parameters of plane $(P_a, P_b, P_c, P_\rho, P_\phi, P_\theta)$ the angle between planes can be calculated (refer Table 3.11). Let π_1 and π_2 be two planes in the three dimensional euclidean space $\Re$.

The angle θ between these planes is defined by means of the normal vectors n_1 and n_2 of π_1 and π_2 through the relationship:

$$cos\theta = |\frac{< n_1, n_2 >}{||n1||||n_2||}| \tag{3.42}$$

In this example, we consider the angle between planes (both horizontal and vertical) with horizontal (plane ground truth) obtained from non collinear point selection of the calibration pattern as shown in Fig. 3.32. The planar parameters (refer Table. 3.10) are obtained as discussed in Sect. 3.4.2. The mean error between planes is 0.0328 radians (1.8 degree).

Point	D1	D2(M1)	MEr	Err(%)	D2(M2)	MEr	Er (%)
P1	233 ±0.5	231.4464	1.5536	0.6668	232.1580	0.8420	0.3614
P2	173 ±0.5	168.1276	4.8724	2.8164	168.8310	4.1690	2.4098
P3	233 ±0.5	231.0840	1.9160	0.8223	231.4620	1.5380	0.6601
P4	227 ±0.5	226.5170	0.4830	0.2128	226.2930	0.7070	0.3115
P5	175 ±0.5	174.6055	0.3945	0.2254	174.3440	0.6560	0.3749
P6	151 ±0.5	150.1623	0.8377	0.5548	149.9530	1.0470	0.6934
P7	96 ±0.5	96.9993	0.9993	1.0409	96.7866	0.7866	0.8194
P8	211 ±0.5	218.3710	7.3710	3.4934	215.8830	4.8830	2.3142
P9	207 ±0.5	209.4647	2.4647	1.1907	207.206	0.2060	0.0995
Average error			2.3214	**1.2248** %		1.6483	**0.8938** %

Table 3.9: Distance of selected points in pipe with reference to plane (mm) , MEr- measurement error, D1 - actual distance from plane, D2 - distance from plane, M1-RANSAC, M2-Manual

Planar parameters	Horizontal(GT)$\pi1$	Vertical $\pi2$	Horizontal$\pi3$
a	0.0277	-0.1120	0.0079
b	0.1751	0.0040	0.0114
c	-6.7889	-0.0024	-0.2002
ρ	6.6846	-0.0024	0.2002
ϕ	-1.3953	1.4592	-1.5568
θ	1.4139	-0.0361	0.9614

Table 3.10: Estimated planar parameters of the planes, GT-Groundtruth

Planes	GT(rad)	Angle(rad)	E1(rad)	E2(rad)
HP(GT) vs. VP(assembly)	1.570796327	1.552580325	0.0182	0.0016
HP(GT) vs. HP(assembly)	0	0.047458993	0.047458993	-
Average error	-	-	0.0328(1.8 deg)	-

Table 3.11: Angle between planes in assembly (HP-Horizontal plane,VP-Vertical plane, E1-Absolute error, E2-Relative error, GT-Groundtruth rad-radians, deg-degrees)
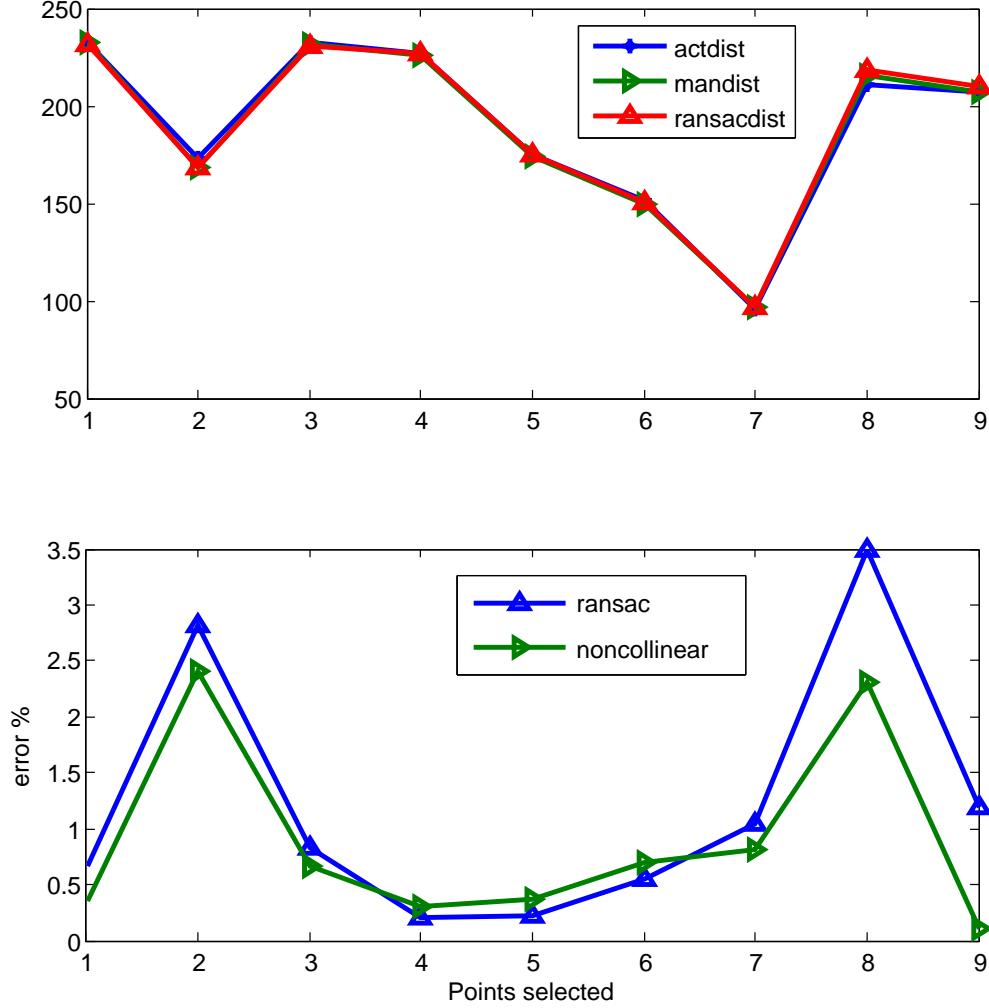
Figure 3.31: (a)Selected points vs. planar distance comparison (b)Selected points vs. percentage error

3.4.4 Angular analysis between edges

Given three 3D points (for example $P2$, $P4$, $P6$) (Mx_1, My_1, Mz_1), (Mx_2, My_2, Mz_2) and (Mx_3, My_3, Mz_3), three edges can be defined between the vertices. The vector for each of these lines are given as follows:

$$v_1 = (Mx_2 - Mx_1)i + (My_2 - My_1)j + (Mz_2 - Mz_1)k \qquad (3.43)$$
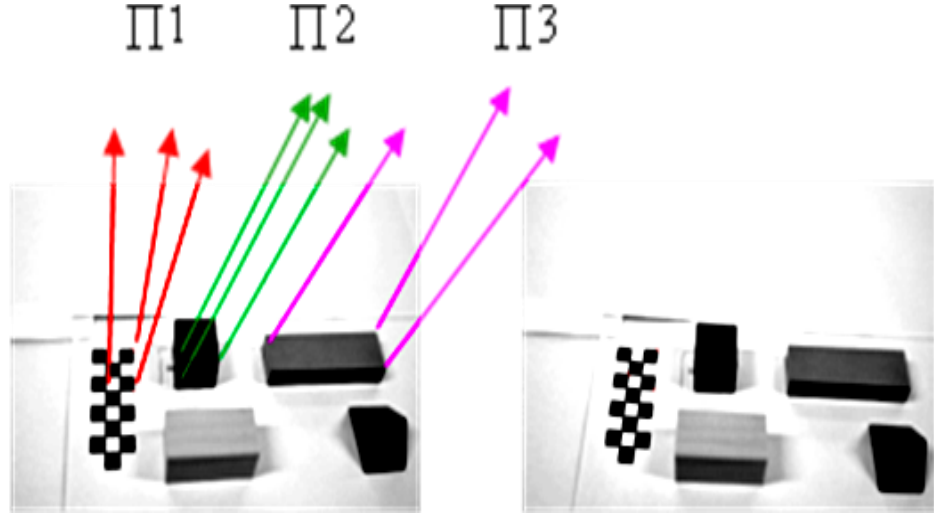
Figure 3.32: Planes selected in stereo views: $\phi 1$- horizontal plane (ground truth), $\phi 2$ - vertical plane of discrepancy blue component (assembly) $\phi 3$ - horizontal plane (green rectangular prism in assembly)
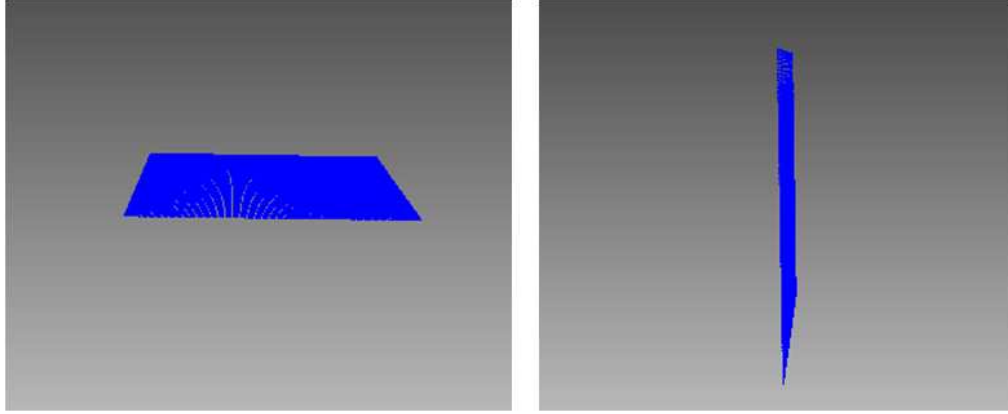
Figure 3.33: Horizontal and vertical planes

$$v_2 = (Mx_2 - Mx_3)i + (My_2 - My_3)j + (Mz_2 - Mz_3)k \qquad (3.44)$$

$$v_3 = (Mx_1 - Mx_3)i + (My_1 - My_3)j + (Mz_1 - Mz_3)k \qquad (3.45)$$

The angle between any vectors $V1 = aMxi + aMyj + aMzk, V2 = bMxi + bMyj + bMzk$, and $V3 = cMxi + cMyj + cMzk$ can then be determined as

$$\theta(V1, V2) = \cos^{-1} \frac{(a_{Mx}b_{Mx} + a_{My}b_{My} + a_{Mz}b_{Mz})}{\sqrt{(a_{Mx}^2 + a_{My}^2 + a_{Mz}^2)}\sqrt{(b_{Mx}^2 + b_{My}^2 + b_{Mz}^2)}} \qquad (3.46)$$

The angle between various edges of discrepancy component (blue cube) shown in Fig. 3.34 are listed in Table 3.12. The mean absolute error for calculated angle is 0.0256 radians(1.4 degrees) and the relative error is 0.0163 radians(0.9 degrees).

Edges	GTA(rad)	MA(rad)	E1(rad)	E2(rad)
$< P2, P1, P3 >$	1.570796327	1.543696599	0.0271	0.0173
$< P3, P1, P5 >$	1.570796327	1.553735733	0.0171	0.0109
$< P1, P3, P4 >$	1.570796327	1.553397139	0.0174	0.0111
$< P3, P4, P2 >$	1.570796327	1.555144214	0.0157	0.0100
$< P1, P5, P6 >$	1.570796327	1.621705836	0.0509	0.0324
Average error			0.0256(1.4 deg)	0.0163(0.9 deg)

Table 3.12: Angle between edges in assembly (E1-Absolute error, E2-Relative error, GTA-Groundtruthangle, MA-MeasuredAngle, rad-radians, deg-degrees)
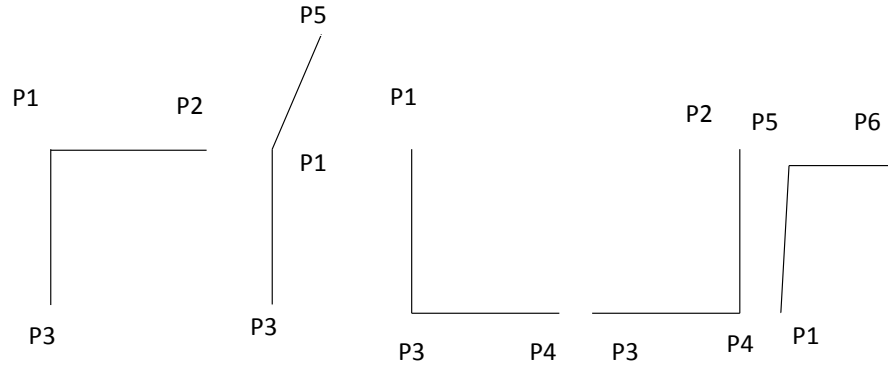
Figure 3.34: Selected edges in assembly for angular analysis

3.5 Usability study

Ten participants took part in usability study of the measurement system. The system was introduced to the users and they rated using the questionaire as provided below.

1. How do you rate the system in terms of easy to use? 0 1 2 3 4 5

2. How do you rate the system providing measurement output functionalities? 0 1 2 3 4 5

3. How do you rate the system handling images of various environment? 0 1 2 3 4 5

4. How do you rate the system in terms of accuracy in 3D measurement? 0 1 2 3 4 5

5. How well does the user input selection process work? 0 1 2 3 4 5

6. How well does the system work on less textured objects? 0 1 2 3 4 5

7. How will you rate the experience with over all system? 0 1 2 3 4 5

Q/U	U1	U2	U3	U4	U5	U6	U7	U8	U9	U10	Mean Score%
Q1	3	4	3	5	4	4	5	3	3	2	72
Q2	4	5	4	4	5	4	5	5	4	4	88
Q3	4	5	5	5	5	4	5	5	5	3	92
Q4	4	5	4	4	5	5	5	5	5	4	92
Q5	3	5	3	5	4	4	5	4	5	3	82
Q6	4	5	4	5	5	4	5	5	4	3	88
Q7	4	5	3	5	5	4	5	4	4	3	84

Table 3.13: Usability study, (U-User, Q-Question)

The usability test rates and mean score are summarized in Table. 3.13. Firstly, the users are satisfied with a mean score of 92% that system can handle images of various environments and can provide high accuracy 3D measurement. Secondly, the users rated 88% for the system that it can handle less textured objects and provide output in both GUI and XML format. The overall experience with the system and input selection is given a score of 84% and 82% respectively. Finally, 72% score was provided for system ease of use since the subjects think that some level of knowledge in the calibration and the stereo view selection process is required.

3.6   Conclusions

In this Chapter, a new 3D measurement framework is proposed for checking a system installation and show whether it is within some predefined tolerance using a camera based calibrated stereo image capture technique with human interaction to select matching disparity points from the multi-view images of the same scene. This enables performing linear measurements that are used to show that two named points that are constrained by some requirement are within tolerance. The distance between multiple identified points or between the identified points and a reference plane defined by a set of identified points, are the types

of measurements that are of interest and are possible. We described the system flow, plus validate the technique via a number of experimental data sets. A measurement tool has been implemented with unique features such as the automated calibration utility, the zoom in option (to help selecting points accurately) and optimal combination of point based disparity and Local Invariant Feature Extraction (LIFE) based disparity which complements each other. These two techniques enabled the measurement between any combination of two points, as well as the distance between any points and the perpendicular distance to a defined plane. The distance between any datum points of interest could be estimated with or without reference to the planar surface model. The tool used multiple views of a scene captured using a basic digital camera and requires the inclusion of a calibration grid within the scene. The application allowed for a user to select a specified number of consistent points/features on the multiple views, which allows the application to indicate the distance between every permutation of pairs of these points and also to give a measure of off-set from a selected reference plane, also defined by three of the selected points. The framework provided two distinct functionalities. The first functionality was where a user provided the points of interest with which distance constraints need to be checked. The second functionality was to get a reasonable 3D cloud using automatic correspondence to facilitate planar surface detection. At this level, we performed outlier removal based on back projection to ensure only correct 3D points are retained. A reference plane was detected using a non-deterministic algorithm, such as RANSAC. We thus obtained the 3D distance with respect to the reference planar surface to any user points of interest. In summary, the distance between any datum points is found and also the distance between any chosen point and planar surface is estimated in a given installation environment at mm level accuracy. Beyond the state of the art, we designed and implemented a 3D measurement tool which can provide accurate measurements between any given datum points of interest at the mm level in an installation environment with or without reference to the planar surface model. The idea of using image based stereoscopic measurement and the optimal combination of point based and automatic disparity for measurement in 3D space with respect to the planar surface reference is used in the proposed framework. A system has been designed and implemented that addresses the

problem of accurate 3D measurement of any given environment in a relatively quicker manner. We describe in next Chapter 4, how this measurement framework is used as a bottom up approach for discrepancy checking and analysis task in the industrial production audit.

Chapter 4

3D PACT (Production Audit Compare Tool) for discrepancy checking

4.1 Introduction

This Chapter describes a new model matching solution which comprises the following capabilities.

1. A digital reconstruction of the fabricated product by using multiple captured images to reposition parts according to the actual model.

2. The projection onto the 3D digital reconstruction of the safety related installation constraints, respecting the original intent of the constraints that are defined in the digital mock-up.

3. Identification of the differences between the 3D reconstruction of the actual product and the design time digital mock up of the product

4. Identification of the differences/non conformances that have a relevance to safety driven installation requirements with reference to the original safety requirement intent.

Steps '1', '2', '3' and '4' together give the safety engineer the ability to overlay a digital reconstruction that should be as true to the fabricated product as possible so that they can see how the product conforms or doesn't conform to the safety driven installation requirements. The work has produced a concept demonstrator that will be further developed in future work to address accuracy, workflow and process efficiency. The hypothesis statement of this Chapter is H2: Use semantic knowledge of the model (intelligence from DMU model) as key information for discrepancy checking (DMU model generation, point cloud segmentation such as color, connected component analysis, pose estimation using datum and object etc.)

4.2 Proposed approach and framework description

The schematic framework as shown in Fig. 4.1 contains key components such as pose estimation using datum, generation of geometry primitives using semantic metadata knowledge, 3D segmentation using knowledge and noise removal, 3D feature extraction and classification, model alignment, discrepancy checking etc. The prior knowledge that we have beforehand regarding the product design is in CATIA XML format (please refer Fig. 4.2). Semantic refers to the tags or label available in XML file of CATIA model. The tag represents the components present in an assembly environment. In the example scenario, we have 5 nodes. The shape types are calibration, cube, rectangular prism, and triangular prism (refer Sect. 4.2.2 for more details). Specifically, the information such as component label, size, width, height, color, orientation (rotation, translation) are used for 3D point cloud processing (for more information refer Sect. 5.2.1 of Chapter 5). We have implemented an utility using XML tool box that can parse the XML data in GUI.

4.2.1 3D structure recovery from multi-view images

Furukawa and Ponce (2010) [37] has proposed a multiview stereopsis method for reconstruction which constitutes three key components such as : a) patch based
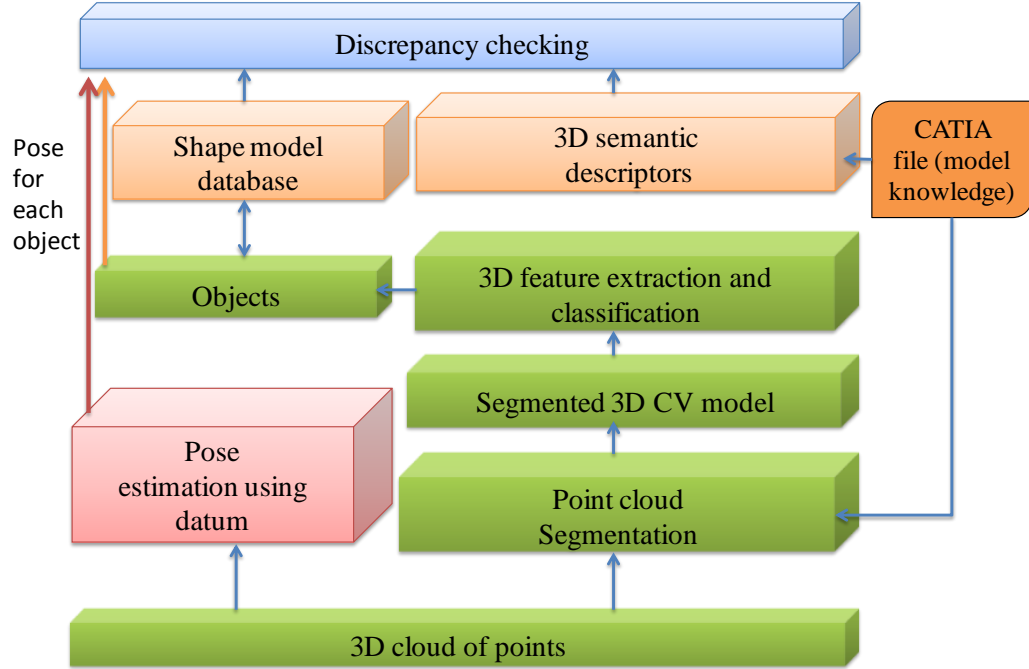
Figure 4.1: MISSA knowledge based 3D model matching framework

MVS (multi-view stereo) algorithm that reconstructs a set of oriented points, b) conversion of the patches into a polygonal mesh model and c) polygonal-mesh based MVS algorithm for mesh refinement. In PMVS2, camera parameters are estimated using automatic correspondences from multi-view images. However, both accuracy and completeness of the 3D structure can be ideally achieved by using calibration pattern especially with objects that lack texture. This is because most of the objects present in the industrial installations are not well textured. 3D for textured objects (house dataset with 18 images) and non textured objects (pipes dataset with 13 images) using PMVS2 without and with calibration pattern are shown in Fig. 4.3. It can be seen that though the results of 3D for the textured portion of the house dataset is visually promising, it can be still improved using PMVS2 with calibration. Further, for pipe dataset, 3D obtained using PMVS2 with calibration is significantly better. Also, as another example, 3D cloud for compressor for multiview images shown in Fig. 4.5 is provided in Fig. 4.6.

```xml
<?xml version="1.0" encoding="utf-8" ?>
<shapes unit="mm">
  <shape type="CALIBRATION" name="c5" kind="component">
    <dimensions x="8" y="3" z="10" />
    <color r="0.0" g="0.0" b="0.0" />
    <transparency value="0.0" />
    <translationPt x="10" y="-40" z="0" />
    <XVector x="1.0000" y="0.0000" z="0.0000" />
    <YVector x="0.0000" y="1.0000" z="0.0000" />
    <ZVector x="0.0000" y="0.0000" z="0.0000" />
  </shape>
  <shape type="CUBE" name="c1" kind="component">
    <dimensions x="30" y="0" z="0" />
    <color r="0.0" g="0.0" b="1.0" />
    <transparency value="0.0" />
    <translationPt x="25" y="135" z="60" />
    <XVector x="1" y="0" z="0" />
    <YVector x="0" y="1" z="0" />
    <ZVector x="0" y="0" z="1" />
  </shape>
  <shape type="RECTANGULARPRISM" name="c2" kind="component">
    <dimensions x="30" y="90" z="15" />
    <color r="0.0" g="1.0" b="0.0" />
    <transparency value="0.0" />
    <translationPt x="25" y="135" z="7.5" />
    <XVector x="1.0000" y="0.0000" z="0.0000" />
    <YVector x="0.0000" y="1.0000" z="0.0000" />
    <ZVector x="0.0000" y="0.0000" z="1.0000" />
  </shape>
  <shape type="RECTANGULARPRISM" name="c3" kind="component">
    <dimensions x="30" y="60" z="30" />
    <color r="1.0" g="1.0" b="0.0" />
    <transparency value="0.0" />
    <translationPt x="25" y="150" z="30" />
    <XVector x="1" y="0" z="0" />
    <YVector x="0" y="1" z="0" />
    <ZVector x="0" y="0" z="1" />
  </shape>
  <shape type="TRIANGULARPRISM" name="c4" kind="component">
    <dimensions x="43" y="30" z="21" />
    <color r="1.0" g="0.0" b="0.0" />
    <transparency value="0.0" />
    <translationPt x="25" y="135" z="85" />
    <XVector x="1" y="0" z="0" />
    <YVector x="0" y="1" z="0" />
    <ZVector x="0" y="0" z="1" />
  </shape>
</shapes>
```

Figure 4.2: Model XML file (from automated reasoning system)

PMVS2   **PMVS2 with calibration**

N = 43744   N = 63896

House dataset (18 images)

N = 32726   N = 109502

Pipe dataset (13 images)
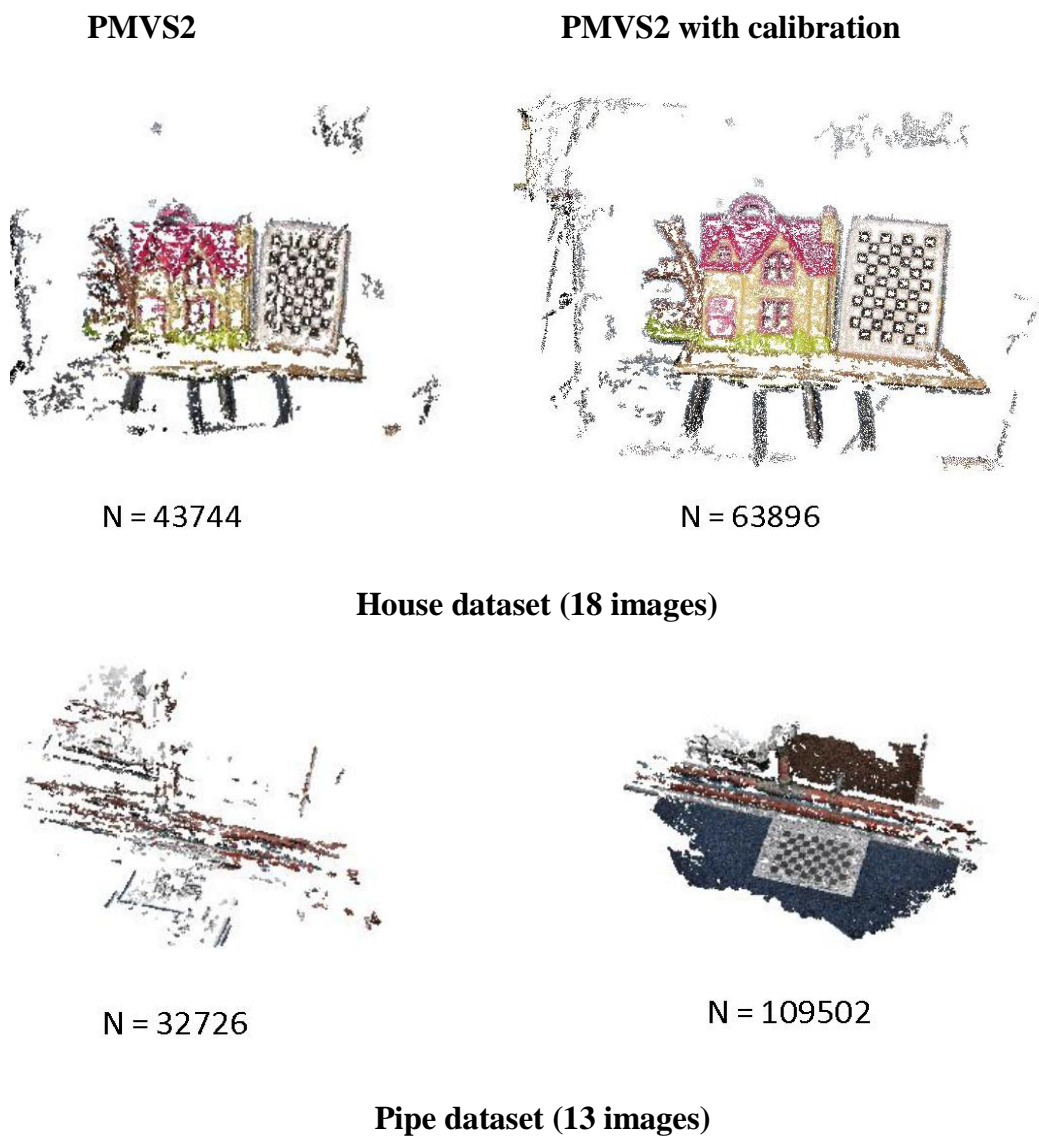
Figure 4.3: 3D cloud using PMVS2 without and with calibration
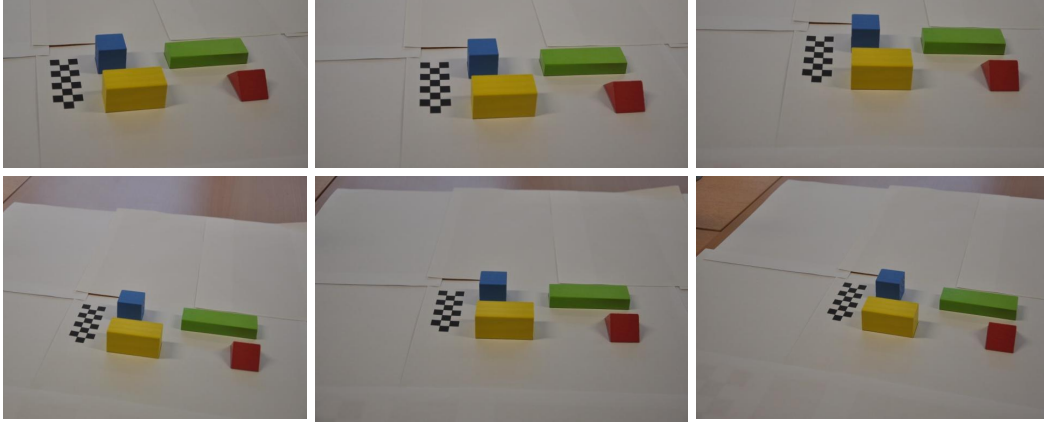
Figure 4.4: Multi-view images
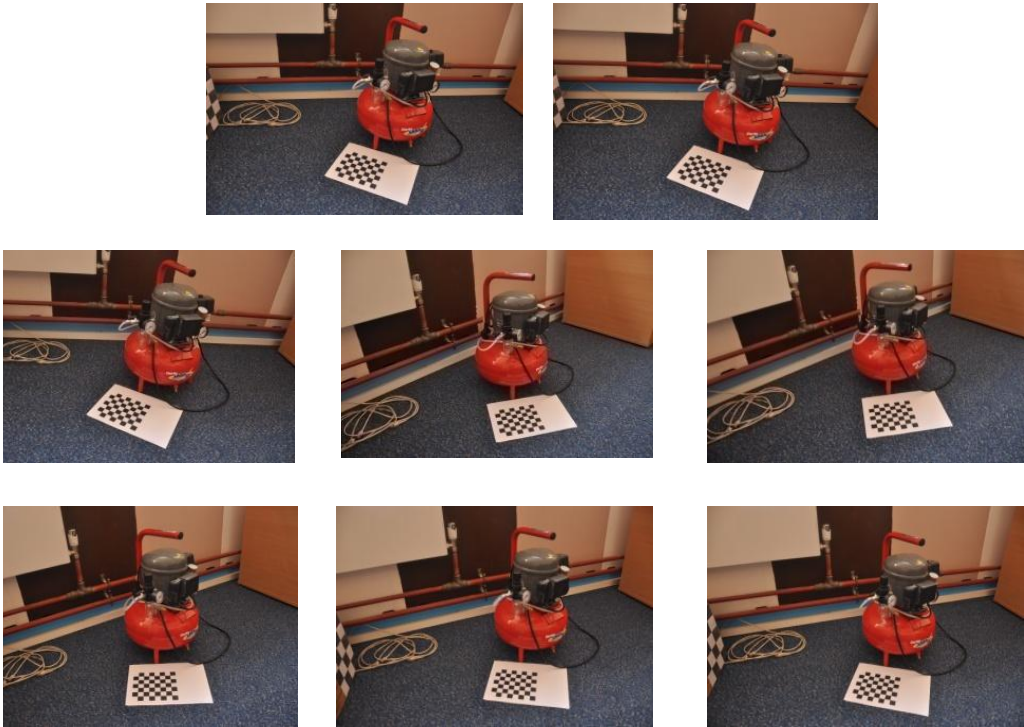
Figure 4.5: Sample images of compressor dataset

4.2.2 Initial environment and datum setup / generation of geometry shapes database for training and classification

In order to perform model matching, we require an assembly environment that needs to be tested whether the setup is in line with geometry model. We gen-

Figure 4.6: 3D cloud of compressor dataset (number of points = 55779)

erated a setup of multiple objects for example, 4 geometric primitive shapes in a controlled environment. Since we know the exact geometry and position of various parts, we can use this for model alignment tasks. The multiview images of this setup environment in the real world are shown in Fig. 4.4. The dataset is available online at [1]. The objective of having this kind of setup is to test the abilities of the system and as proof of concept for model matching methodology. We generated a database of 1400 3D geometric shapes such as cubes, rectangular and triangular prisms of scaling sizes, different orientations along x, y and z axis. The rotation is performed on each individual axis of 5 degrees from 1 to 360 (refer Fig. 4.7 for generated sample shapes). Unlike Georgel et al. (2007), instead of anchor plates [38], we use calibration pattern as datum reference information. Initially, we use calibration pattern as part of prior geometric knowledge. Instead of using a calibration pattern, an object can be defined as datum.

4.2.3 Pose estimation using datum

Let M be the reconstructed CV model of the datum shape and M' represent the geometry model shape. For example, calibration as shown in Fig. 4.8 is used as datum to recover the pose information. The task is to minimize the difference between cloud of points and find the best alignment of M with M' to
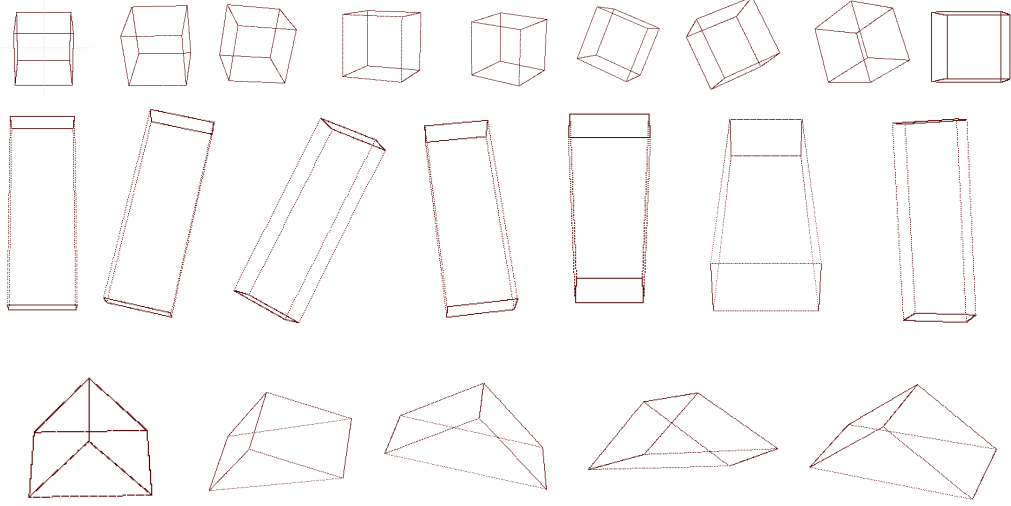
Figure 4.7: Sample 3D geometry shapes: cube, rectangular prism, triangular prism rotated along x, y and z axis

obtain the pose information. The points are associated with nearest neighbor criteria and transformation parameters are estimated using a mean square cost function. With the estimated parameters, points are transformed. $\{m1_i\}$ and $\{m2_i\}$ represent the point sets of models.

$M = \{m1_i\}_{i=1}^{N1}$ and $M' = \{m2_i\}_{i=1}^{N2}$

This problem can be formulated based on least square (LS) criterion as follows.

$$\min_{R,T,j\epsilon\{1,2...N2\}} \sum_{i=1}^{N_1} \|(Rm1_i + T) - m2_j\|_2^2 \qquad (4.1)$$

where $R^T R = I_m$ and $|R| = 1$, R and T are rotation and translation parameters.

The two main steps of ICP algorithm are as follows.

The correspondence between two point sets M and M' based on $(p-1)^{th}$ rigid transformation is achieved as

$$c_p(i) = \underset{j\epsilon\{1,2...N2\}}{\arg\min} \|(R_{p-1}m1_i + T_{p-1}) - m2_j\|_2^2 \qquad (4.2)$$

The rotation and translation parameters are obtained by minimizing the
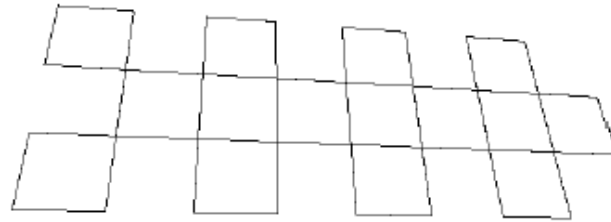
(a)   Geometry model shape (N = 4800)
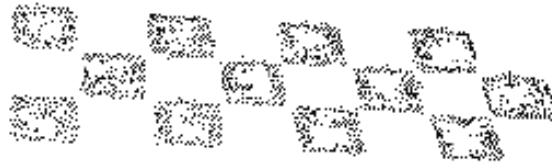
(b)   Reconstructed CV model (N = 1920)
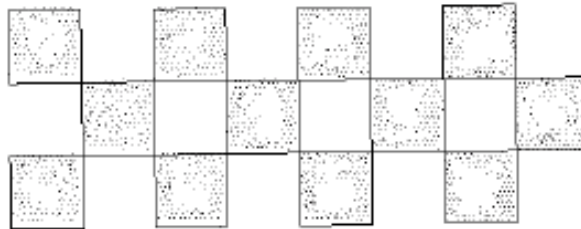
(c)   Aligned models (N = 6720)

Figure 4.8: Geometry model shape, reconstructed CV model and model alignment for pose estimation

squared distance

$$(R_k, T_k) = \underset{R^T R = I_m, det(R) = 1, T}{\arg\min} (\sum_{i=1}^{N1} \left\| (Rm1_i + T - m2_{c_p(i)} \right\|_2^2) \qquad (4.3)$$

The obtained R and T transform the CV model to the geometry model. In order to transform from geometry to CV model, the parameters such as R' and $-T'$ can be used.

The ICP convergence error and the number of iterations for datum (calibration in this example) and other shapes are plotted in Fig. 4.9.
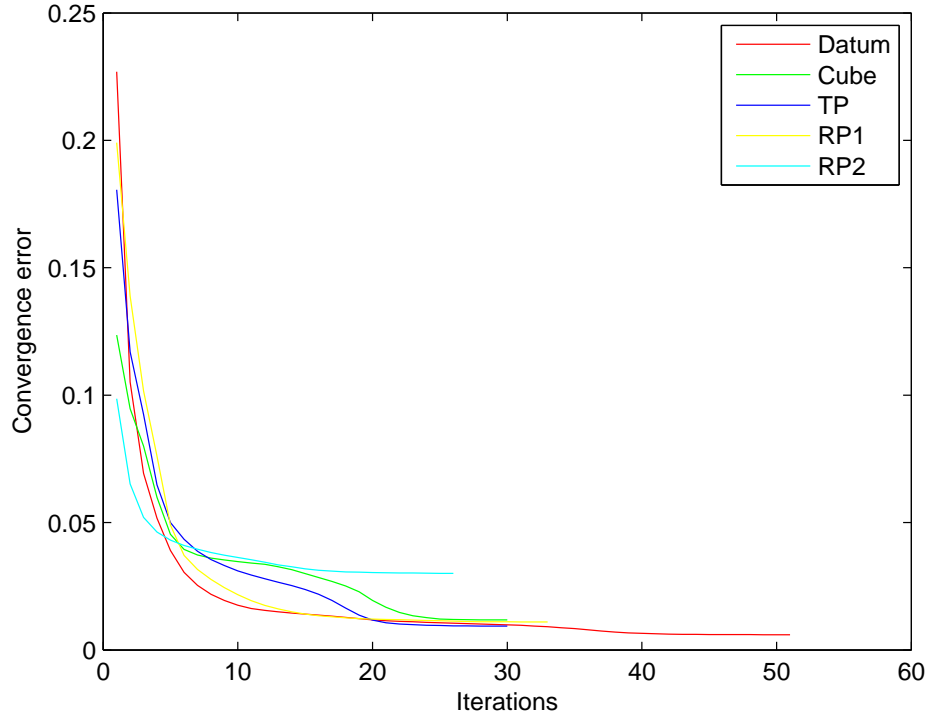
Figure 4.9: Iteration vs. convergence error (ICP) [Datum-calibration pattern, TP-triangular prism, RP1,2-rectangular prism (green and yellow)]

4.2.4 Generation of geometry knowledge primitive

The semantic information is used for building the part primitives with the exact object component. For example, a geometry primitive pyramid generated is shown in Fig. 4.10. The primitives thus formed for shapes such as cube, triangular and rectangular prism can be seen in Fig. 4.11. 3D cloud is segmented using metadata knowledge where information such as color, location and shape class labels etc., are available. The 3D cloud of points needs to be compared and fitted with DMU (digital mock-up) model shape primitives thereby enabling further discrepancy analysis. Model based matching is based on fact that the whole object is a transformation (projection) of a preconceived model.
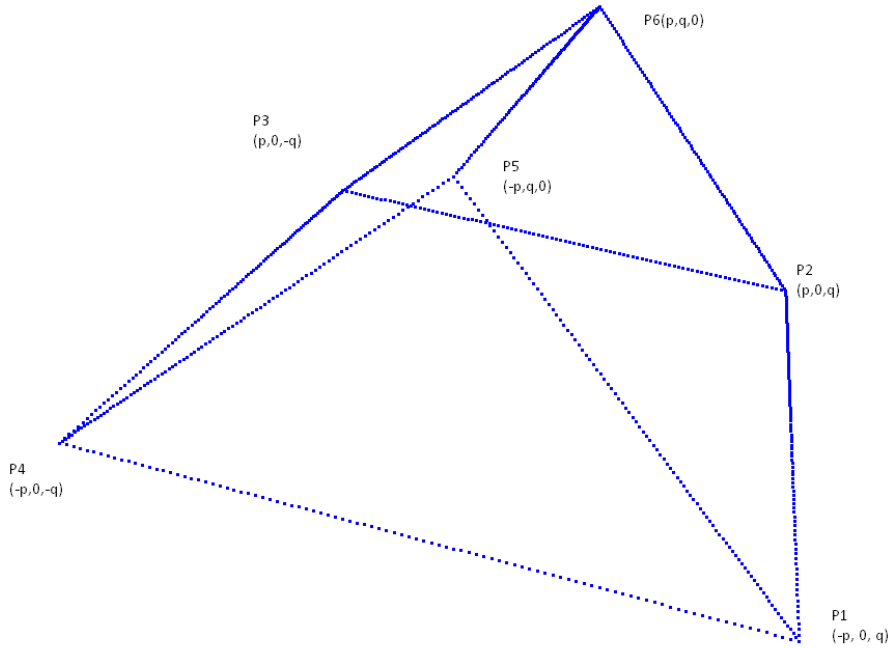
Figure 4.10: Sample geometry primitive generation

CL	S,W,H	R	T	C
CUBE	30 0 0	1 0 0 0 1 0 0 0 1	25 25 15	0 0 255
RECTANGULARPRISM	30 90 15	1 0 0 0 1 0 0 0 1	25 135 7.5	0 255 0
RECTANGULARPRISM	30 60 30	1 0 0 0 1 0 0 0 1	85 40 15	255 255 0
TRIANGULARPRISM	43 30 21	1 0 0 0 1 0 0 0 1	77.75 165 10.5	255 0 0

Table 4.1: Geometry representation CL-Class Label,S-Size,W-Width,H-Height,R-Rotation matrix,T-Translation vector, C-Color(RGB)
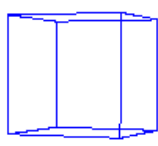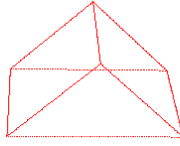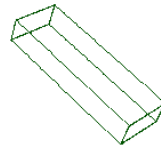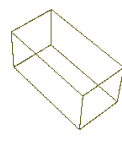
N = 1208　　　N =906　　　N =1208　　　N = 1208

Figure 4.11: Three geometry shapes

4.2.5　Point cloud processing (3D segmentation using knowledge and noise removal)

As discussed in previous sections and shown in Table 4.1, since we know the semantic description regarding each of the objects such as color, position, class label we use it as key information to aid segmentation. For example in this setup, each object has distinct color information which is highly useful in segmenting those objects. The rule based color segmentation based on constraints is provided in Algorithm.2. In the algorithm, ind_R, ind_G, ind_B denote an index array of 3D cloud corresponding to R,G,B respectively. rh, rl, gh, gl, bh, bl denote the higher and lower limit of color values as shown in Table 4.2. Further, the location information can be used to localize the search space of the model within the vicinity. The pseudo code for rule based color segmentation is provided.

Thus, from the reconstructed 3D cloud, individual objects can be segmented

Algorithm 2 pseudo code for 3D color based segmentation

Input:3D point cloud array $d = (X, Y, Z)$
Output: Segmented cloud array based on color

1: for $i = 1 : size(d, 1)$
2: if $d(i, ind_R) <=$ rh & $d(i, ind_R) >= rl$&...
3: $d(i, ind_G) <=$ gh & $d(i, ind_G) >= gl$&...
4: $d(i, ind_B) <=$ bh & $d(i, ind_B) >= bl$
5: extract $d(i, :)$
6: end
7: end

Color	RH	RL	GH	GL	BH	BL
Red	255	102	90	0	100	0
Green	90	0	255	100	170	0
Blue	90	0	75	0	255	0
Orange	255	153	173.4	76.5	76.5	0
Yellow	230	100	200	100	45	0
Indigo	173.4	0	255	153	255	153
Violet	255	153	173.4	0	255	153
Black	50	0	50	0	50	0

Table 4.2: Table of color ranges used for segmentation

based on CATIA knowledge. If the segmented cloud is noisy, further processing is done using connected component analysis. The segmented cloud based on color information and connected component analysis is shown in Fig. 4.12. Any 3D set of points that are not separated by boundary is connected. The set of connected components partition the 3D cloud into segments. Connected component analysis is based on union-find technique and label equivalence relationship. The method finds distance between each 3D point index in the point cloud and assign the smallest label to the minimum distance set according to a pre-defined threshold (in this case 0.5 cm). If both of them have labels, then the minimum of two labels is assigned to the other. The algorithm for connected component analysis is provided in Algorithm.3.

As an example, as shown in Fig. 4.12, by performing connected component

---

Algorithm 3 pseudo code for connected component analysis

Input: Noisy 3D point cloud, N, threshold $m = 0.5$**cm**
Output: Segmented cloud array based on connected components

 1: find the distance between each point
 2: Create list for labels of the points a vector of size N. Label all to -1
 3: set label counter $= 0$
 4: for loop
 5: find minimum distance in matrix. indexes i and j. if min dist $>$ m, stop
 6: check the labels of both
 7: if they have no labels, set their labels to label counter and increment label counter.
 8: if one of them have a label, then set the other one with the same label.
 9: if they both have labels, then choose the label with smaller label. set the other to the smaller label.
10: replace the other label with the smaller one in label vector
11: go to 5
12: end

---

analysis, the cloud can be segmented reliably.

From segmented objects, discriminative features such as shape D2, A3, DIR, and spherical moments can be used for classification. Upon the classification of objects, the model alignment can be done using ICP.

4.2.6   3D shape feature extraction

Initially, we used just D2 shape feature extraction from 3D data to demonstrate the classification purpose that can complement CATIA knowledge for model matching. Corresponding to each label (a label represents a common visual characteristic shared between a set of pixels) of the shape models, 3D descriptors such as D2 shape histogram is used in this work. D2 distribution [63] for a set of points P can be calculated as

$$D2(d) = \frac{|p, q \epsilon P s.t \, \|p - q\| = d|}{|P|^2} \tag{4.4}$$

The value of D2 distribution at d is the number of point pairs whose pair-
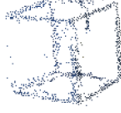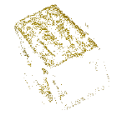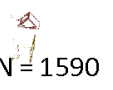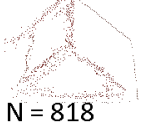
Figure 4.12: Point cloud segmentation (color and connected component analysis), outliers visible in the cloud of points shown in 2nd column.

wise distance is d. As machine contains curvature information, shape histogram information would be useful [44].

4.2.7 Training and 3D classification

The shape models such as cube, rectangular prism and triangular prism are used in the database repository. The mean of all data shape histograms from the model of each class is shown in Fig. 4.14. The shape histogram of sample segmented

N = 1065

N = 1448

N = 3198

N = 818
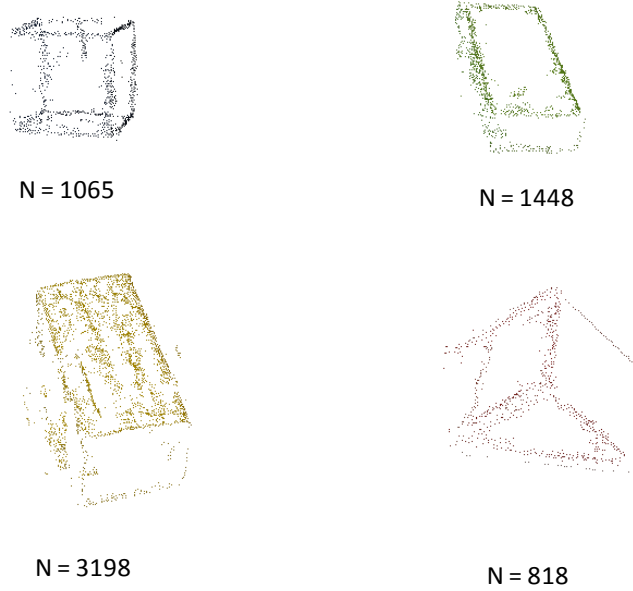
Figure 4.13: Segmented cloud

data of each class is provided in Fig. 4.15. We train and test the classifier using nearest neighborhood method and SVM. The input to the classifier is 3D features extracted from the segmented cloud for identifying the corresponding class labels.

4.2.7.1 k-NN nearest neighbourhood classifier (non parametric method)

This is a method for classifying objects based on closest training examples in the feature space. If $k = 1$, then the object is simply assigned to the class of its nearest neighbor. Given a training set and a distance defined in the attribute space, the basic k-NN rule consists in searching for the k nearest neighbors of an attribute vector. The estimated class probabilities are proportional to the number of C_j class among k nearest neighbors (with $1 < j < n$ and n is the number of classes in the training set), then the chosen j corresponds to the class which has the maximum probability. The value of k must be chosen to minimize
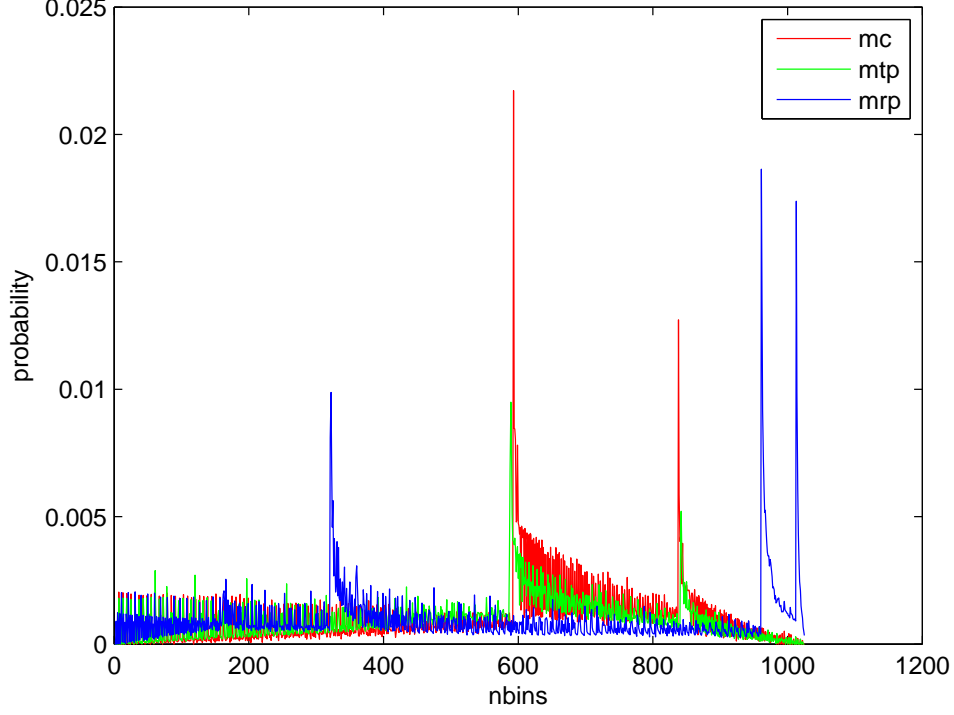
Figure 4.14: Probability vs. number of bins of model data (samples = 512) [mc-model cube, mtp-model triangularprism, mrp-model rectangular prism]

the expectation of test error. To classify a new point, k nearest points from the training dataset is identified and assigned new point to class having large number of representatives among this set. The training vectors include $D2$ values of 1348 objects in a multidimensional feature space, each with a class label. During classification, an unlabeled vector so called query is classified by assigning the label which is more frequent among the k training samples nearest to that query point. $L1$ is used to determine the distance between the query Q and model M.

$$L1(M, Q) = \sum_{i=1}^{n} |Q_i - \mu(M_i)| \qquad (4.5)$$

where Q is the segmented cloud and M is the model.
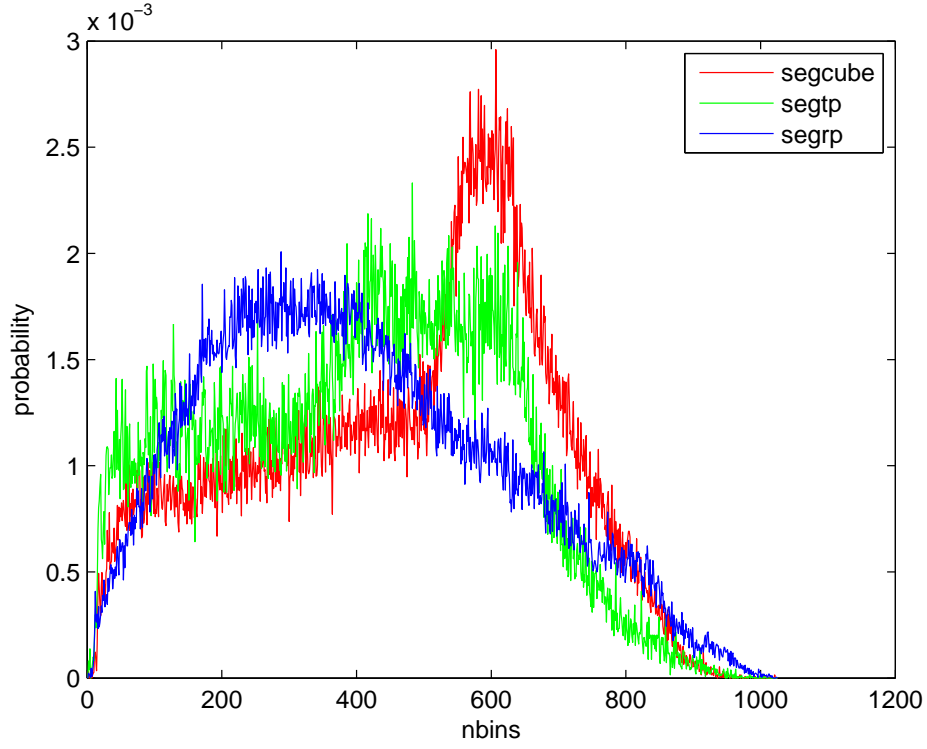
Figure 4.15: Probability vs. number of bins of segmented data (samples = 512) [segcube-segmented cube, segtp-segmented triangular prism, segrp-segmented rectangular prism]

Class	Class A	Class B	Class C
Class A	**0.0665**	0.6243	1.0239
Class B	0.6250	**0.0770**	0.8960
Class C	1.0260	0.8978	**0.0925**

Table 4.3: Average distance between objects and class features

Class	Class A	Class B	Class C
Class A	**0.8165**	0.5427	0.8130
Class B	0.9833	**0.7119**	0.7928
Class C	0.9437	0.7129	**0.7008**

Table 4.4: Average distance between segmented objects and class features

4.2.7.2 Support vector machine (SVM) classifier:

Given samples $x_{1...l}$, SVM finds a linear classifier that satisfies

$$y_i(w.x_i + b) - 1 \geq 0 \qquad (4.6)$$

with margin width $2/\|w\|^2$. Minimizing $\|w\|^2$ in the formulation of the classifier maximizes the margin width and forms the quadratic programming problem

$$L_P = \frac{1}{2} - \sum_{i=1}^{l} \|w\|^2 - \alpha_i y_i(x_i.w + b) - 1 \geq 0 \qquad (4.7)$$

This is equivalent to maximization of dual of L_P

$$L_D = \sum_{i=1}^{l} \alpha_i - \frac{1}{2} \sum_{i,j}^{l} \alpha_i \alpha_j y_i y_j x_i.x_j \qquad (4.8)$$

In order to generalize SVM for nonlinear cases, training examples can be projected to higher dimensional space by $\phi(x)$ for linear separation. Since L_D depends on dot product between x_i and x_j, this can be substituted by kernel function $K(x_i, x_j)$ that computes $\phi(x_i).\phi(x_j)$ rather than directly computing in higher dimensional space.

4.2.7.3 Experimental results for classification:

A) With regard to NN classifier, the average distance between objects and class features from the whole database of objects is summarized in Table.4.3. The correct class being identified can be determined by minimum value in the diagonal elements. Each of the classes A, B, C has a minimum mean value of the cluster as 0.0665, 0.0770 and 0.0925 respectively. Secondly, the average distance between segmented objects and class features is provided in Table 4.4. NN classifier is based on $L1$ distance metric. k-NN classifier is used to predict labels of shape type. The probability values should be ranging from 0 to 1. The values close to 1 represent the the label that best represents the class. Training phase is trivial. Initially, training example is stored with its label. In order to make a prediction,

distance is calculated to training example. Then k closest training examples are retained where $k \geq 1$. Finally, a label that is most common among these examples is chosen. The label is prediction of this test example. The nearest neighbor is a supervised algorithm with function of type $(X \times Y)^n \times X \to Y$. n is the cardinality of the training set. A distance function has type $X \times X \to R$. There are two important factors to consider a)k, b) distance function. It can be observed that each class can be clearly identified except a false detection case where triangle prism is identified with a value of 0.5427 (refer Table 4.4).

B) We used LIBSVM classifier [25] to train the model using generated geometry models individually for three shape classes.

Index	Shape
$1 - 316$	Cube
$317 - 832$	RP
$833 - 1348$	TP

Table 4.5: Train data details

As shown in Table 4.5, among 1348 shapes, first 316 represent cube models. For defining each of the classifier, we trained data of corresponding indexes as train data and labels as 1 while others are set as 0. For test data, we tried different combination to test the prediction performance. For example while testing the cube classifier with same data, we got 100%. For the data from say 200 to 330 we got performance of 89% since the data comes from another class and 0% for data completely different ($317 - 1348$). The classification accuracy is dependent on the training data. We tested with two sets of segmented cubes, rp and tp. Using classifier, the detection of segmented data for cube and tp fall into the same category since both are similar due to its incomplete representation. This is because our segmented models are not dense rather it represents the boundary skeleton of shapes. Though semantic information from geometry model alone suffices to provide good results (refer 4.16) as reported in this paper, the idea of having classifier in this framework is to complement and cross check the verification process. Also, usually classifier works fine if training data and discriminative feature are robust enough. Initially, we used D2 features and in the future, we intend to

increase the performance by testing with more discriminative features/optimal combination of multiple features and robust classifiers.

4.2.8 Model alignment of CV and DMU Model
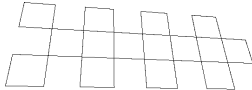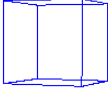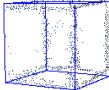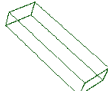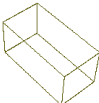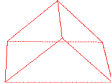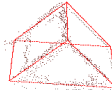
	Geometry primitive	Segmented CV model	Aligned model
Calibration	N = 4800	N = 1920	N = 6720
Blue cube	N = 1208	N = 1065	N = 2273
Rectangular prism (green)	N = 1208	N = 1448	N = 2656
Rectangular prism (yellow)	N = 1208	N = 3198	N = 4406
Triangular prism (red)	N = 906	N = 818	N = 1724

Figure 4.16: Geometry model shape, segmented CV model and model alignment

Let M be the segmented CV model shape and M' represent the geometry model shape. The aim is to obtain pose from each of the segmented point cloud and corresponding geometry shape for model fitting. This information together with pose recovered as in Sect. 4.2.3 is used for discrepancy checking. $\{m1_i\}$ and $\{m2_i\}$ represent the point sets of models. $M = \{m1_i\}_{i=1}^{N1}$ and $M' = \{m2_i\}_{i=1}^{N2}$ The task is to find the best alignment of M with M'. This can be formulated based on least square criterion as follows.

$$min_{R,T,j\epsilon\{1,2...N2\}} \sum_{i=1}^{N_1} \|(Rm1_i + T) - m2_j\|_2^2 \tag{4.9}$$

111

where $R^T R = I_m$ where R and T are rotation and translation parameters. The correspondence between two point sets M and M' based on $(p-1)^{th}$ rigid transformation is achieved as

$$c_p(i) = argmin_{j \epsilon \{1,2...N2\}} \|(R_{p-1}m1_i + T_{p-1}) - m2_j\|_2^2 \qquad (4.10)$$

The rotation and translation parameters are obtained by minimizing the squared distance

$$(R_k, T_k) = argmin_{R^T R = I_m, det(R)=1, T}(\sum_{i=1}^{N1} \|(Rm1_i + T - m2_{c_p(i)}\|_2^2) \qquad (4.11)$$

The obtained R and T transform the CV model to the geometry model. In order to transform from geometry to CV model, the parameters such as R' and $-T'$ can be used. The geometry model shape, segmented CV model and fitted model is shown in Fig. 4.16.

4.2.8.0.1   Case:single objects   The 3D model alignment fitting of the single objects such as cubes and pyramids in the assembly setup are provided in Fig. 4.17 and Fig. 4.18 respectively.

4.2.8.0.2   Case: multiple objects   The 3D model fitting results of the multiple objects (four shapes) in an assembly setup is provided in Fig. 4.19.

4.2.9   Discrepancy checking and analysis

The discrepancy result of the original sample installation (set A) and (set B) used for experimentation is shown in Fig. 4.20 and Fig. 4.21. We use single pose recovered from the whole system (using datum as discussed in Sect. 4.2.3) and pose for each of the objects in a system (as defined in Sect. 4.2.8) so that the entire system is available. The model and CV cloud aligned using the pose information is shown. It can be clearly seen from Fig. 4.21 that the shift of 1 cm in the cube component (blue color) of the sample assembly setup is visible. This deviation from the model would be useful for automatic verification analysis. The difference between the center of the objects aligned by ICP and aligned by
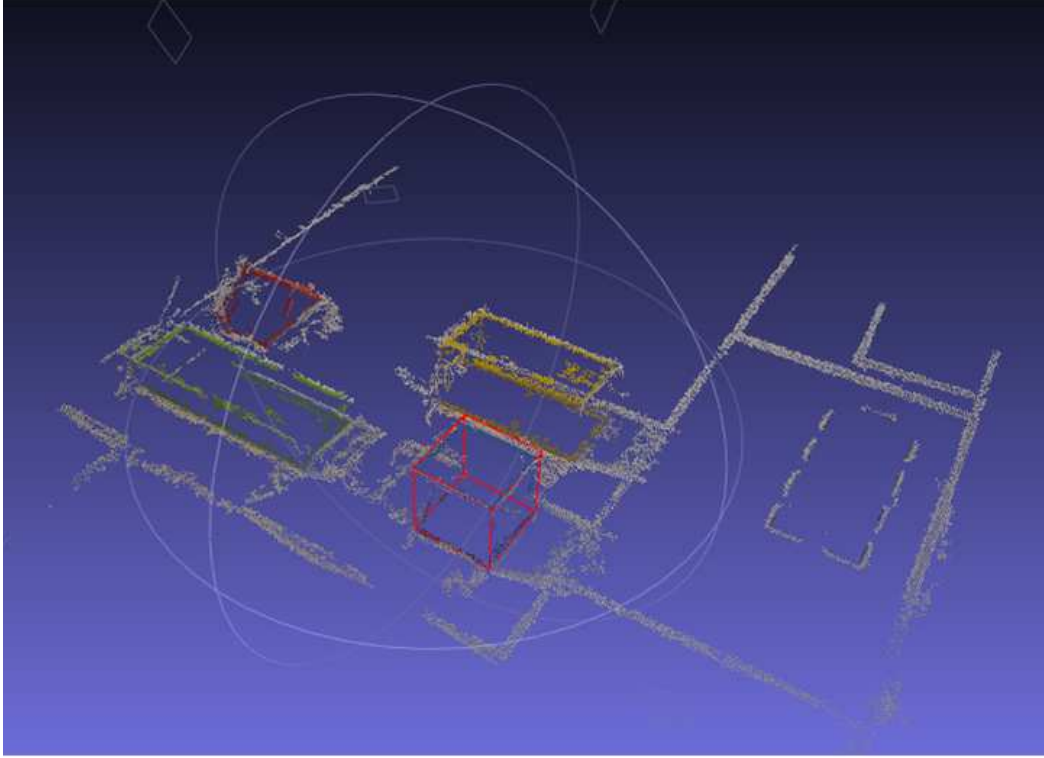
Figure 4.17: Model alignment in 3D space : single object (cube)

calibration location for both normal (set A) and blue shifted (set B) cloud is provided in Table 4.6 and Table 4.7 respectively.

Set A			
Class	**X**	**Y**	**Z**
Cube	**0.0597**	**0.0165**	**0.0997**
Rectangular Prism1	0.0061	0.0315	0.5877
Rectangular Prism2	0.6549	0.2610	0.0746
Triangular Prism	0.2232	0.1043	0.8892

Table 4.6: Difference between centre of objects aligned by ICP and calibration location (setA)

Quantitatively, we can infer and analyze that the distance between center of objects aligned by calibration for set A is 0.11 cm for cube and 0.966 cm. This is obtained using norm of the XYZ difference of the corresponding object (in this
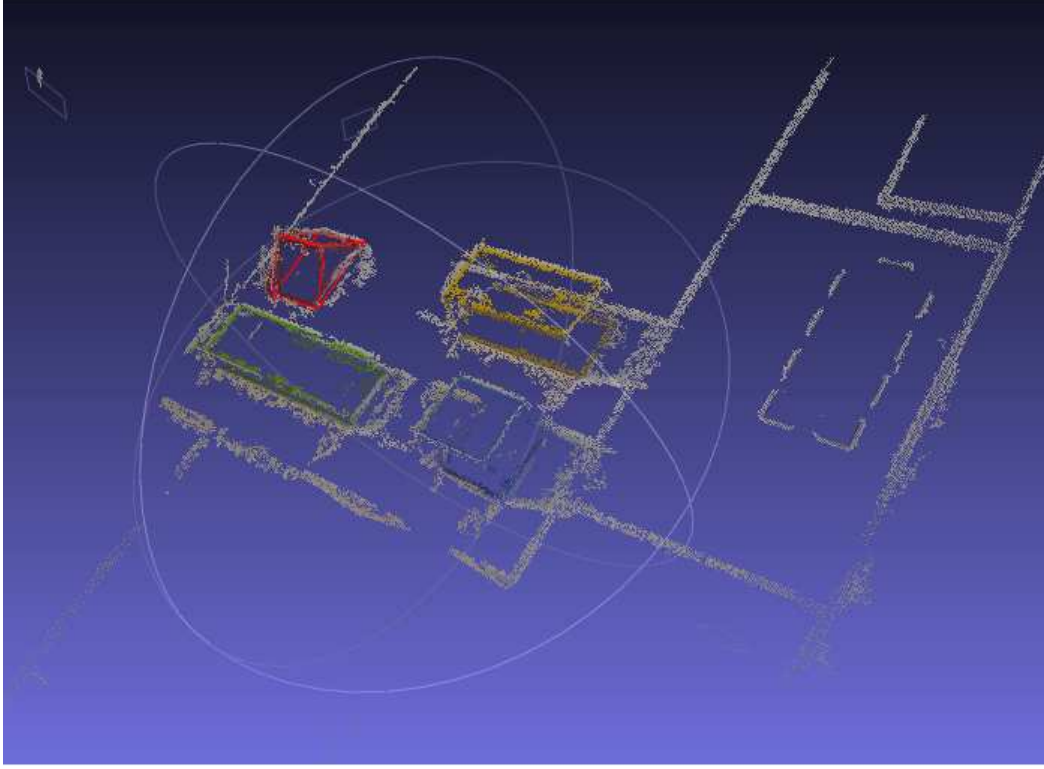
Figure 4.18: Model alignment : single object (pyramid)

Set B			
Class	**X**	**Y**	**Z**
Cube	**0.0663**	**0.9556**	**0.1275**
Rectangular Prism1	0.3401	0.0435	0.4443
Rectangular Prism2	0.1190	0.2941	0.0225
Triangular Prism	0.4288	0.0945	0.7117

Table 4.7: Difference between centre of objects aligned by ICP and calibration location (setB)

example cube). This means there is a drift of 0.966 cm for blue shifted cube with a centre as a reference. This value ideally reflects the discrepancy (magnitude). The relative discrepancy obtained using the pipeline is shown in Fig. 4.22. At this point, we propose to use a bottom up approach to get very precise discrepancy magnitude using 3D SLR FPSS technique discussed in the previous Chapter 3.
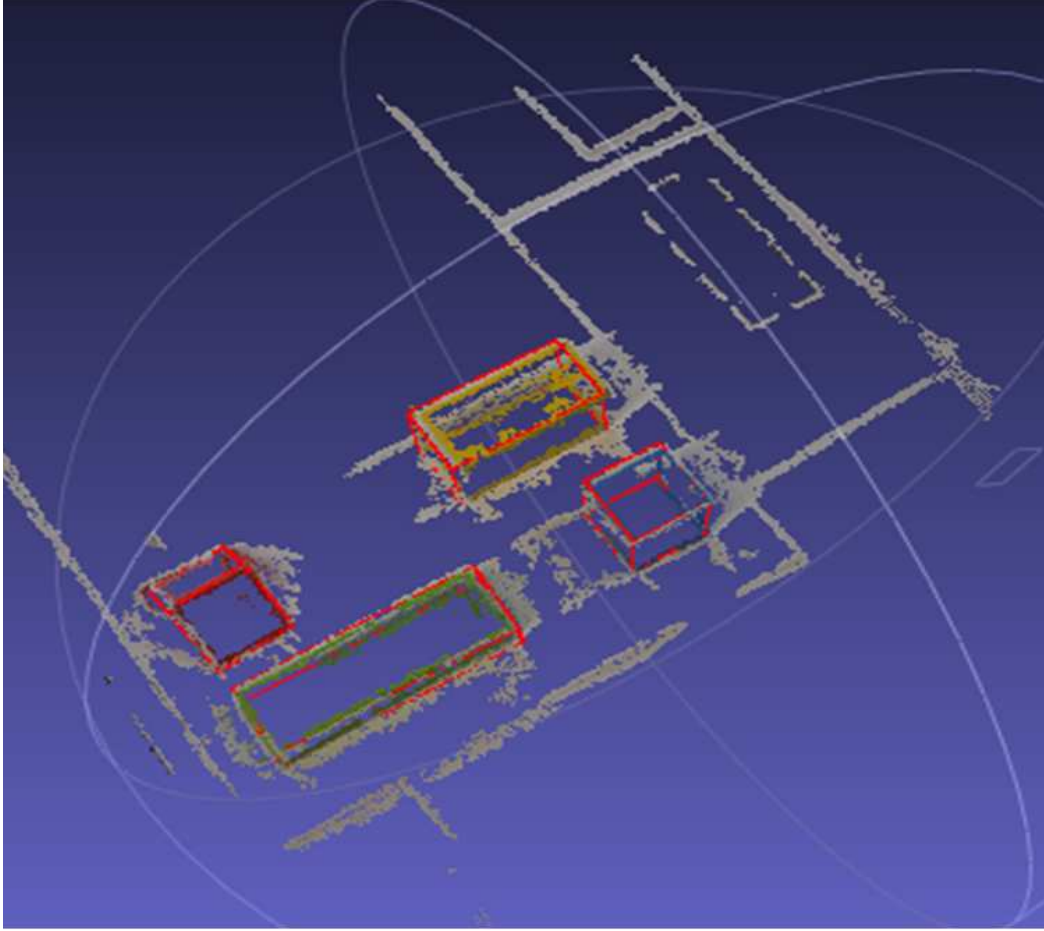
Figure 4.19: Multiple objects alignment in assembly setup

4.3 Discrepancy magnitude and distance analysis using 3D SLR stereo triangulation (FPSS) of an assembly

Traditionally, 3D stereo triangulation [47] is used to estimate the 3D points with the correspondence obtained using the correspondences and camera projection matrices. Recently, Smisek et al 2011 [72] has made a study and found that SLR stereo outperforms kinect in measurement followed by SR4000. The mean error of kinect is 2.39 mm whereas SLR stereo is 1.57 with their data and analysis. Most of
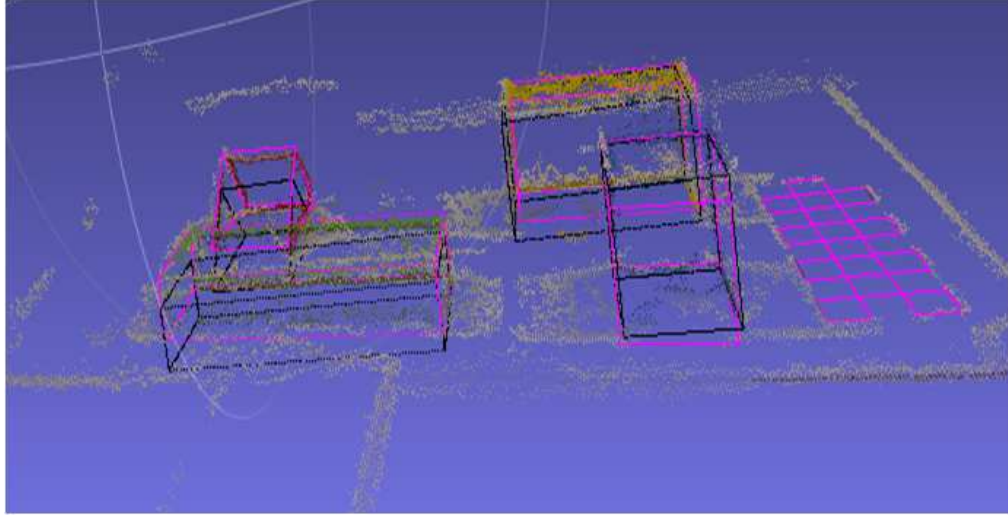
Figure 4.20: Alignment of cloud (set A)
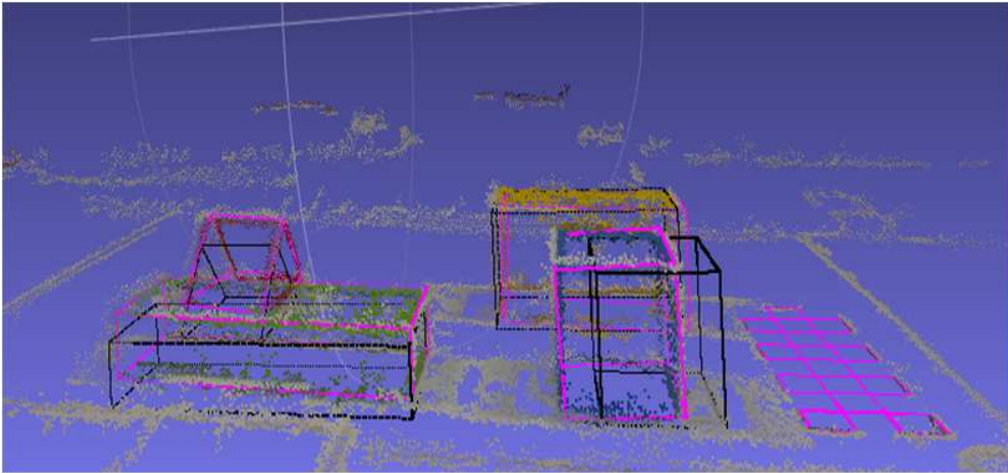
Figure 4.21: Alignment of cloud (set B)

the industrial components such as pipes, wires are not well textured which makes feature detection and matching cumbersome for image processing techniques and sometimes only sparse matches are available. Likewise dark or highly polished surfaces of the subject matter make laser based ranging techniques very noisy.

3D cloud of installation environment obtained using kinect sensor is shown in Fig. 4.23. Also, as mentioned in the previous Chapter, we need to achieve
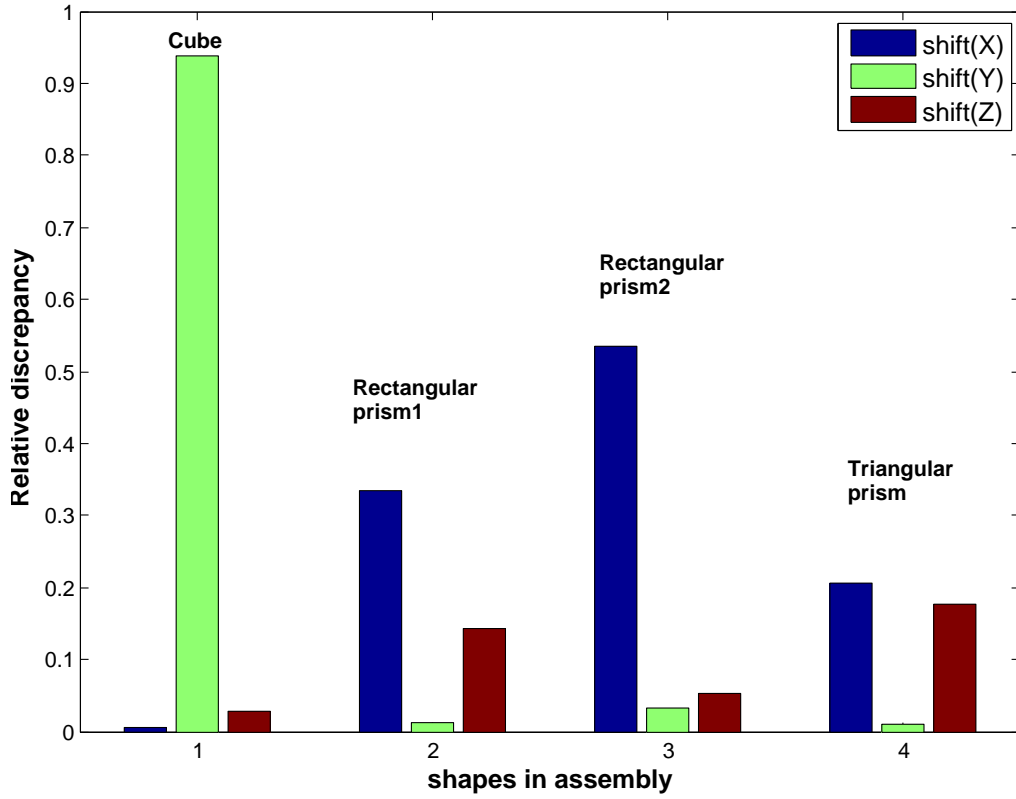
Figure 4.22: Relative discrepancy between the components
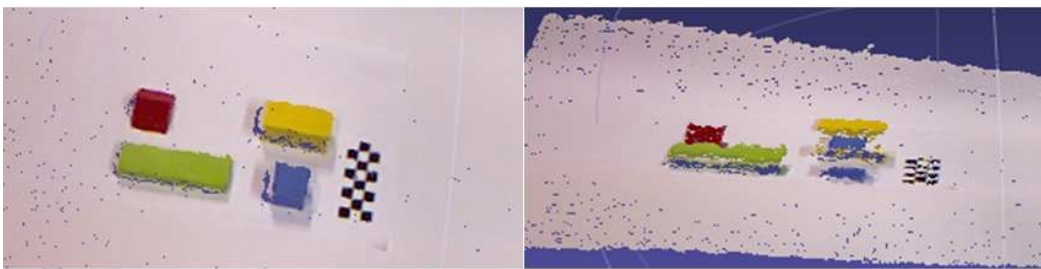
Figure 4.23: 3D cloud of installation environment using kinect sensor

mm accuracy for production audit purposes. The approach that was developed in this work phase was based on image processing techniques that are assisted by user intervention to identify common features where they are not immediately apparent due to the lack of texture. We propose to use focused point based SLR
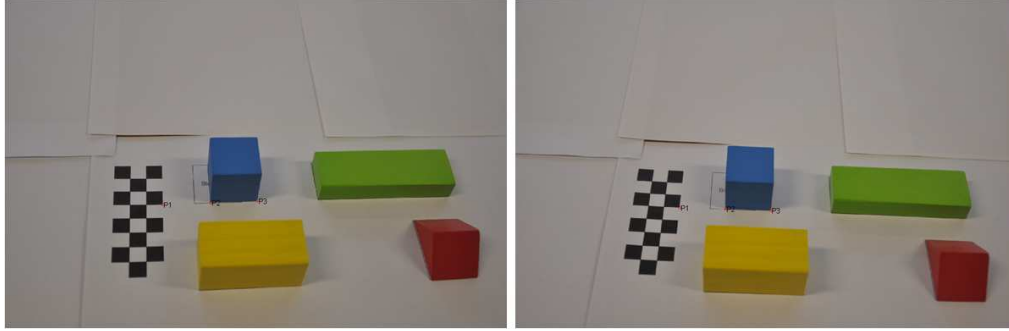
Figure 4.24: Stereo views used for measurement analysis of discrepancy component of fault assembly (Set B)

Assembly	M1	M2	M3	M4	M5	M6
Faulty	20 mm	30 mm	30.076 mm	10.076mm	0.076	0.0025
Normal	20 mm	20 mm	21.485 mm	1.485 mm	1.485	0.0743

Table 4.8: Discrepancy measurement using SLR stereo FPSS, M1-constraint knowledge, M2-ground truth datum point reference [P1], blue component corner [P2] , M3-measured datum point reference [P1], blue component corner [P2], M4-discrepancy, M5-measurement error, M6- relative error

stereo (FPSS) where a safety test engineer selects inspection points of interest with which distance measurement check need to be accomplished. With the camera views, a linear method is used to find the best reconstructed 3D point. Once the blue cube component has been identified to have discrepancy, precise measurement can be made using FPSS. The stereo views used for measuring discrepancy (faulty assembly) is shown in Fig. 4.24.

As shown in Table 4.8, for faulty assembly setup, the measurement between datum(calibration) point reference [P1] and blue component corner [P2] is 30.076 mm which shows the discrepancy of 10.076mm with the measurement error of 0.076 mm. Similarly for the normal assembly, the measurement between datum(calibration) point reference [P1] and blue component corner [P2] is 21.485 mm which shows the discrepancy of 1.485 mm with measurement error of 1.485 mm.

4.4 Conclusions

In this Chapter, we presented a new model comparison that uses input from digital camera and semantic metadata knowledge available from geometry models which can be used for verification tasks. Traditional 3D model matching approaches perform model matching between any given cloud of points. The key achievement in our work is: the usage of CATIA model information to perform model alignment and discrepancy checking. The semantic information XML file output from the automated reasoning system MATHSAT3D which solves installation optimization such that the components obey safety constraints is used. Discrepancy checking is obtained by point cloud processing (segmentation) and pose estimation of the predefined datum and individual objects present in the scene. Each of the components in the 3DCV model is segmented based on the prior knowledge available for the model. Segmentation based on the color knowledge information is performed on the 3D point cloud and if there are any outliers present in the segmented cloud, then connected component analysis approach is used for error removal. For each of the segmented components in 3DCV model, object pose is estimated by matching with the corresponding component in DMU model. The datum (reference) is used for estimating the pose information. Datum could be either calibration pattern, or anchor plates or any well-defined object. Using the pose, the components in real world space are projected to model space. The components in model space can be transformed to real world space vice versa also. By displaying together the real world space model and the transformed DMU model, the discrepancy result is shown. A system has been developed that performs discrepancy checking between 3D vision model of the actual installation to the original digital mock-up, with a focus on the safety driven installation constraints, such as segregation, proximity and orientation. The tool takes a set of images and uses various image processing techniques to create a 3D digital reconstruction of the objects within the set of images. The digital reconstruction is in the form of a cloud of points. The application takes the digital mock-up of the scene and converts it into a similarly defined cloud of points. The two clouds are aligned to each other such that the selected datum, an equipment or calibration chart, has the lowest positioning error, based on feature extrac-

tion and comparison techniques. The rest of the equipment that are recognized are then aligned to the digital reconstruction. The offsets for these equipments are reported as positioning errors that can be used to check for acceptability to the installation tolerances derived from zonal safety installation constraints. The measurement error of the tool is estimated by comparing the measures performed on the calibration grid against the actual dimensions of the grid. As a proof of concept and installation case study, a CV model assembly with four components with one of the components shifted by 1 cm is created whereas the DMU model has all of the components which are in normal position. The discrepancy of the component is demonstrated and the magnitude is quantitatively obtained by finding the norm of the XYZ difference of the corresponding object. Once the object having discrepancy is identified, precise magnitude can be estimated using stereo FPSLR. Ideally, the framework has been a proof of concept for safety analysis and verification and tested in a controlled environment data set for model matching. 3D object structures with respect to other objects position in the scene can be extracted. In future, experiments would be conducted in real industry setup.

Chapter 5

Linking installation constraints to production audit

Discrepancy checking can be performed to check whether there is a deviation in the component placement of a plant assembly using datum (reference) information. The link between the installation optimization constraints and the existing PACT (production audit compare tool) [82] [80] [81] need to be realized. Safety related constraints are used for installation optimization in production audit process. It is possible to perform comparisons between a CATIA product and a 3D computer vision model in the PACT and to measure the error but an automated check that shows that the constraints are respected in the 3D computer vision model of the fabricated product need to be achieved. This chapter addresses this problem using AIV contains, intersect, disjoint, trajectory testing methods on the 3D point cloud model of a sample assembly setup. The hypothesis statement of this Chapter is H3: AIV contains, intersect, disjoint, trajectory test can be performed on the point cloud model to check whether the installation satisfies constraints.

5.1   Introduction

Production audit refers to the process of verification of a plant assembly in an industrial environment against the set of pre-defined safety conditions or instal-

lation constraints. Plant Assembly refers to the components in an installation environment. Installation constraints refer to different danger levels, different components fall within the volume etc. The components should satisfy certain constraints in order to pass the safety testing and auditing/analysis process. In order to facilitate the automation of production audit, in the works [82] [80], discrepancy checking is performed to check whether there is a deviation in an assembly component placement using datum information. For example, we created two use case scenarios that resemble plant assembly setup: a) set A with components analogous to DMU model setup and b) set B with one of the components (blue cube) deviated by 1 cm from its original position.

The proposed algorithm and PACT [82] [80] was able to identify the discrepancy and test results were reported. The model we use is not just a geometry model representation (CATIA format) but it is subjected to a process where installation optimization is performed to check whether the model satisfies certain installation constraints criterion [21]. The optimization process is required to obtain a more accurate model that evolves over time. This model knowledge available in an XML file format is used for DMU generation. In this Chapter, we discuss how safety related constraints are used for production audit process and whether the 3D point cloud model respects these constraints. The contribution of this chapter is the fact that AIV, trajectory, contains, intersect and disjoint test can be performed on the point cloud model to check whether the installation satisfies constraints. The problem of conflict or collision between two equipments and the problem of containment by the correct AIV or the relevant equipment of an idealization are problems that have been solved by FBK [21]. The problem that we address in this research work is.

1. Does the 3D point cloud model of an assembly environment raise a conflict or collision?

2. Is the 3D point cloud model contained by the correct AIV?

3. Is the 3D point cloud model components are in trajectory path?

Up to our knowledge, there is no work available in the literature so far that checks the 3D point cloud model against the model which evolved after installation

optimization process and verify the set of installation constraints such as AIV contains, intersect, disjoint, trajectory are met with final 3D point cloud assembly. The Chapter is organized as follows. In the first Sect. 5.1, we have introduced production audit. The significance of AIV, trajectory testing is provided in Sect. 5.1.1. Sect. 5.2 provides the framework description. DMU, AIV and trajectory model creation is discussed in Sect. 5.2.1. The details regarding pose estimation and model alignment is provided in Sect. 5.2.2. In Sect. 5.2.3, methodology and results for AIV contains, intersect and disjoint are discussed. Trajectory testing is provided in Sect. 5.2.4.

5.1.1 Significance of AIV, trajectory related safety related constraints

There may be some components which are not in an acceptable zone as it exceeds the bounding planes of AIV. There may be cases where the component looks from one angle seems that it obligues the safety constraint but actually it doesn't. In most cases where the physical model deviate from the DMU model confirmed by the existence of LD constraint, the components need to be re-positioned spatially such that any possibility of a high danger condition is alleviated at an early level since aggregation of these LD constraints can cause high danger. At any point of time as the assembly is progressing, installations have to be regularly monitored/examined to make sure that a) either components respect the installation constraints or b) taken remedy measures to alleviate the hazardous levels such as ones by possible bursts. If both components say C1 and C2 are placed in the same trajectory path T, these components should be of a nature such that the burst of one doesn't affect the other component placement. If there are too many components in an assembly without respecting the model assembly design, this could be hazardous and should be fixed.

5.2 Methodology description

The proposed production audit pipeline shown in Fig.5.2 is based on two key inputs: a) 3D point cloud structure recovery from the images and b) DMU model
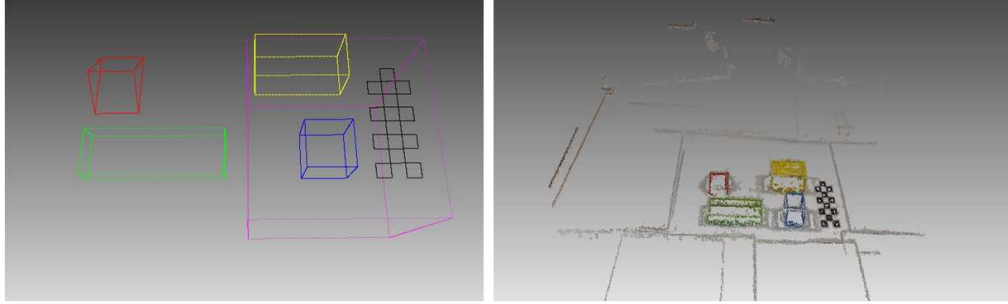
Figure 5.1: DMU and real model of the assembly setup

generation using XML file [82] [80]. The point cloud is assimilated from the environment and comparison is done to the DMU, AIV, trajectory model based on model knowledge.
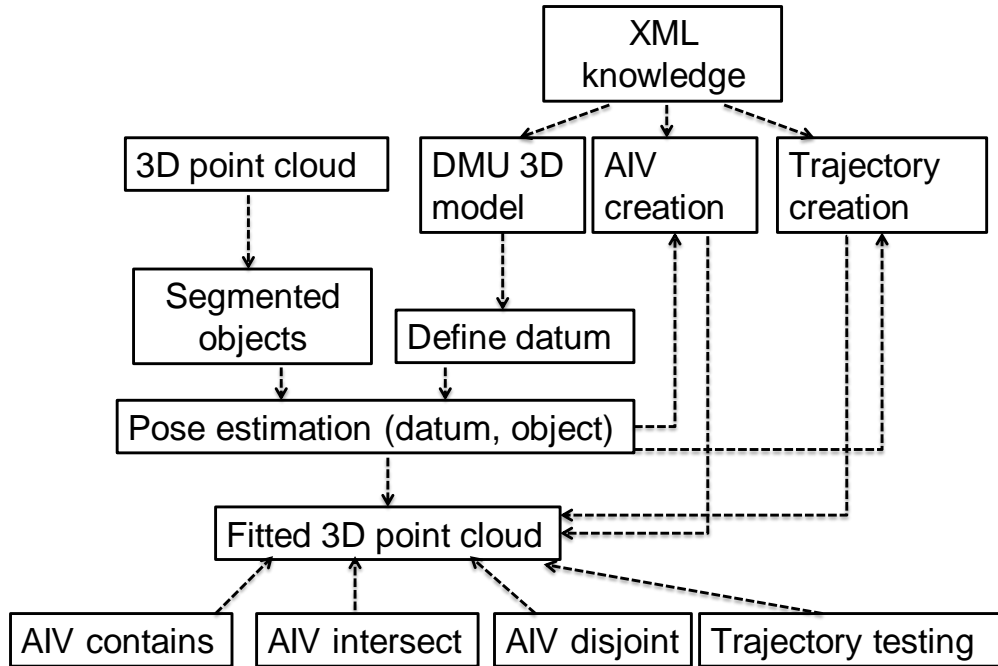
Figure 5.2: Proposed framework pipeline overview

5.2.1 DMU, AIV, Trajectory model generation based on model knowledge

DMU model is generated using semantic knowledge information available from the system design model. Constraints XML file represents the new configuration of components that satisfies the constraints high danger, low danger, non-safety related constraints (the volume containing the components) and the components cannot overlap etc. AIV is modeled based on library parts knowledge which is defined as in XML format as provided in Fig. 5.3. In this example, "dimension" represent width, height and depth parameter of the object. "TranslationPt" represent the translation vector and "XVector, YVector, ZVector" represent the rotation matrix, color represents the RGB value of the component. "kind" represents whether the shape type is component, volume or trajectory. In this example component with name c6 is of a kind volume (AIV), c7 is trajectory whereas other components with name c1-c5 are of component types. The library parts are created using the DMU knowledge. Then the library part is converted into a set of vertices that are then used to draw edges and then planes. Then the edges and planes are used to check that the idealization lies within the AIV and does not conflict etc. i.e.

1. Use the 3D point cloud to check that it is contained within the AIV.

2. Use the 3D point cloud to show that there is no conflict or collision

5.2.2 Model alignment of the assembly components (datum vs. object fitted)

The collision detection analysis can be done on the point cloud but alignment needs to be done before this constraint checking. Model alignment is a problem of registering 3D shapes and involves estimating the pose information that can facilitate discrepancy checking and constraint testing. The aim is to perform collision tests with fitted 3D point cloud based on model alignment rather directly on the 3D point cloud. Both datum (reference based: eg. Calibration) and object fitted (individual component such as c1-c4 in XML file) are considered. The pose

```xml
<?xml version="1.0" encoding="utf-8" ?>
<shapes unit="mm">
  <shape type="CUBE" name="c6" kind="volume">
      <dimensions x="100" y="0" z="0" />
      <color r="1.0" g="0.0" b="1.0" />
      <transparency value="0.7" />
      <translationPt x="25" y="25" z="15" />
      <XVector x="1" y="0" z="0" />
      <YVector x="0" y="1" z="0" />
      <ZVector x="0" y="0" z="1" />
  </shape>
  <shape type="CUBE" name="c1" kind="component">
      <dimensions x="30" y="0" z="0" />
      <color r="0.0" g="0.0" b="1.0" />
      <transparency value="0.0" />
      <translationPt x="25" y="25" z="15" />
      <XVector x="1" y="0" z="0" />
      <YVector x="0" y="1" z="0" />
      <ZVector x="0" y="0" z="1" />
  </shape>
  <shape type="RECTANGULARPRISM" name="c2" kind="component">
      <dimensions x="30" y="90" z="15" />
      <color r="0.0" g="1.0" b="0.0" />
      <transparency value="0.0" />
      <translationPt x="25" y="135" z="7.5" />
      <XVector x="1.0000" y="0.0000" z="0.0000" />
      <YVector x="0.0000" y="1.0000" z="0.0000" />
      <ZVector x="0.0000" y="0.0000" z="1.0000" />
  </shape>
  <shape type="TRIANGULARPRISM" name="c4" kind="component">
      <dimensions x="43" y="30" z="21" />
      <color r="1.0" g="0.0" b="0.0" />
      <transparency value="0.0" />
      <translationPt x="77.75" y="165" z="10.5" />
      <XVector x="1" y="0" z="0" />
      <YVector x="0" y="1" z="0" />
      <ZVector x="0" y="0" z="1" />
  </shape>
  <shape type="FRAGMENTTRAJECTORY1" name="c7" kind="trajectory">
      <dimensions x="70" y="650" z="15" />
      <color r="1.0" g="1.0" b="0.0" />
      <transparency value="0.0" />
      <translationPt x="100" y="40" z="10.5" />
      <XVector x="1" y="0" z="0" />
      <YVector x="0" y="1" z="0" />
      <ZVector x="0" y="0" z="1" />
  </shape>
</shapes>
```

Figure 5.3: Sample model XML file with AIV, trajectory

for the environment is determined using both datum fitted (DF) and object fitted

(OF) as follows.

$$P^{OF}_{i_{i=1..S_n}} = PE(S^{CVM}_{i_{i=1..S_n}}, S^{GM}_{i_{i=1..S_n}}) \tag{5.1}$$

$$P^{DF}_{i_{i=1..S_n}} = PE(D^{CVM}, D^{GM}) \tag{5.2}$$

Where CVM is the Computer vision model of the segmented object of a setup and GM is the corresponding geometry model. PE (Pose estimation) is the 3D model alignment function which recovers the transformation matrix that minimizes the difference between CVM and GM.

$$(R_k, T_k) = \operatorname*{arg\,min}_{R^T R = I_m, det(R) = 1, T} \left(\sum_{i=1}^{N1} \left\| (Rm1_i + T - m2_{c_p(i)} \right\|_2^2 \right) \tag{5.3}$$

where

$$c_p(i) = \operatorname*{arg\,min}_{j \epsilon \{1,2...N2\}} \left\| (R_{p-1} m1_i + T_{p-1}) - m2_j \right\|_2^2 \tag{5.4}$$

N1 and N2 are the number of 3D points in CVM and GM respectively. FE is the fitting error between two models. Fitting error (FE) is the absolute difference between distance (AIVcentre, component) in model space and distance(AIVcentre,component) in fitted(translated) space [2 cases, both datum and object fitted]. Pose for object fitted is the function of segmented objects in the computer vision reconstructed model and the corresponding segmented objects in the geometry model space. Object fitted is data transformed according to transformation matrix recovered from segmented object and corresponding model object. Datum fitted is data transformed according to transformation matrix recovered from datum (calibration information) calib cloud and corresponding model.

The point cloud along with AIV before and after alignment is shown in Fig. 5.4 and Fig. 5.5.
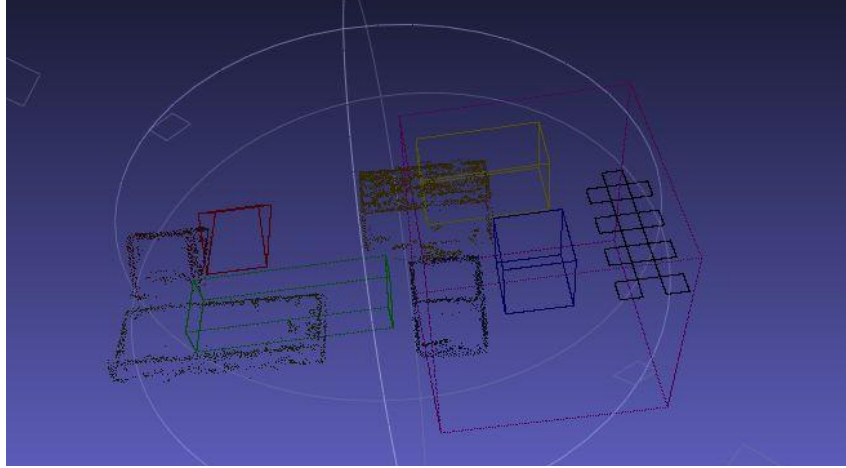
Figure 5.4: 3D Point cloud with DMU, AIV (before alignment)
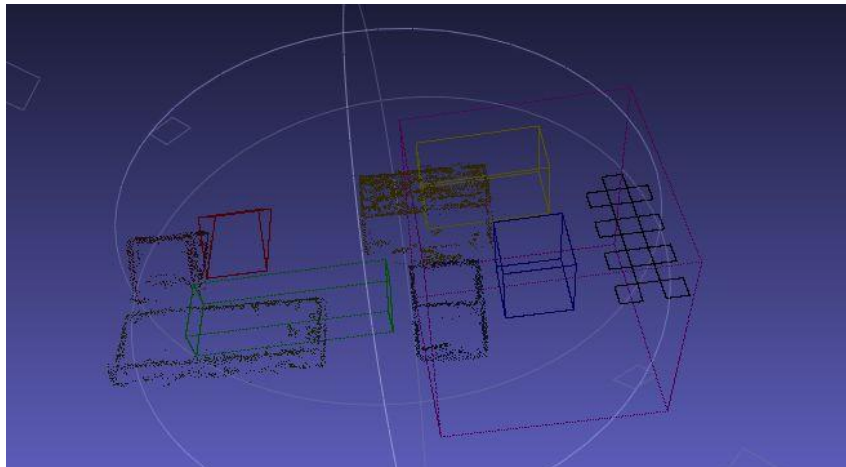
Figure 5.5: 3D Point cloud model with the DMU, AIV (after alignment)

5.2.3 AIV Contains, Intersect and Disjoint Test

5.2.3.1 Contains test(AIV)

Given the individual components of an assembly setup and the acceptable installation volume in an installation environment, it is necessary to automatically understand the spatial relationship between components such that it does not interfere with trajectory fragments and are within the Acceptable Installation Volume (AIVs). AIV should be designed such that the components are in safe

zone. Either the component placement is safe or not is adaptive and always subjective to that particular point of time as each component placement is dependent on neighboring component nature. So in order to ensure the overall safety of the system, the checking of individual component is vital but not enough. AIVs and trajectory sort of checks need to be done.

5.2.3.2 Collision detection using boundary check

In general, the components with AIV can be spatially categorized under the following cases

1. components fully inside the other component,

2. components that touch partially.

3. components that do not interfere with one another

The collision algorithm 4 has been able to provide just whether object collides or not (case 1) however, it is not able to identify two components with two cases (case 2 and 3).

Algorithm 4 Finding contains relationship(AIV, Assembly component)

Input: AIV, 3D segmented point cloud model
Output: Component state

1: Collision (AIV, assembly component) = {0, if minx1 > minx2
2: maxx1 < minx2
3: miny1 > maxy2
4: maxy1 < miny2
5: minz1 > maxz2
6: maxz1 < minz2
7: 1 else }

The case 1 is under contains relationship whereas case 2 and 3 should be observing intersect and disjoint relationship. In the next section, we discuss about how to identify the spatial relationships such as intersect and disjoint for the components with AIV.

5.2.3.3 Intersect and Disjoint test (AIV)

The environment is analyzed for understanding spatial relationship between corners of each component of the assembly (both object and datum fitted) with AIV. The distance between AIVcentre and corner of the components of segmented point cloud is estimated. The center of AIV (pink cube : transformed space) is 4.4363, 54.1817, 15.5334 whereas for AIV(pink cube: model space) is 25.0000, 25.0000, 15.0000. The graph shown in Fig. 5.6, 5.7, 5.8, 5.9 displays the distance between AIV center and components (object, datum fitted), object (model space). It can be observed that the only blue cube has distance less than half of the edge length of AIV (50) and hence satisfying contain constraint.

Dist	Co1	Co2	Co3	Co4	Co5	Co6	Co7	Co8
AIV,Cube	40.9290	41.3181	58.5327	58.8055	47.1621	21.7569	46.8216	21.0085
AIV,RP1	130.7490	131.6674	50.4778	52.8111	39.4359	126.8955	36.2517	125.9424
AIV,RP2	97.9383	98.1016	106.2719	106.4224	80.5724	69.2105	80.3735	68.9789
AIV,TP	135.6910	158.3072	136.9532	110.0322	120.7722	145.7224	NA	NA

Table 5.1: Distance between AIV centre and components (object, datum fitted), object(modelspace) Co-corner

---

Algorithm 5 Finding spatial relationship(AIV, Assembly component)

Input: AIV, 3D segmented point cloud model
Output: Component state

1: For all components in assembly
2: if distance(AIV,allcorners) < edgelength(AIV)/2
3: then componentstate = 'CONTAINS';
4: elseif   min(distance(AIV,allcorners))<   edgelength(AIV)/2
   max(distance(AIV,allcorners)) > edgelength(AIV)
5: then componentstate = 'INTERSECTS';
6: elseif   min(distance(AIV,allcorners))>   edgelength(AIV)/2
   max(distance(AIV,allcorners)) > edgelength(AIV)
7: then componentstate = 'DISJOINT';
8: end

5.2.3.3.1 Proposed algorithm for finding spatial relationship whether the components in assembly are contained, intersect, disjoint The methodology is tested for both datum fitted and object fitted. The algorithm 5 is able to identify whether the components in assembly are contained, intersect and disjoint. Since the components are basic shapes and AIV is also well defined, the method works. However, there is a need for robust collision detection method for handling complex shapes. We would discuss plane plane intersection and octree collision detection technique in next Section.
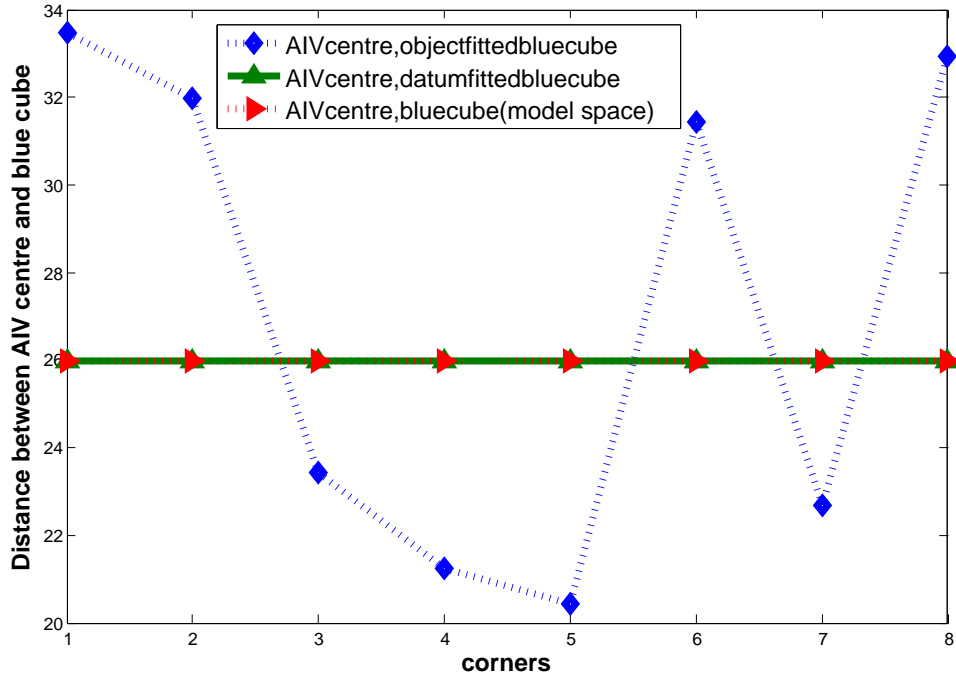
Figure 5.6: Distance between AIV centre and blue cube component(object, datum fitted), object(modelspace)

5.2.4 Trajectory test

The aim of trajectory test is to check how far the components have deviated from the trajectory path. A trajectory is an obelisk like structure which can be
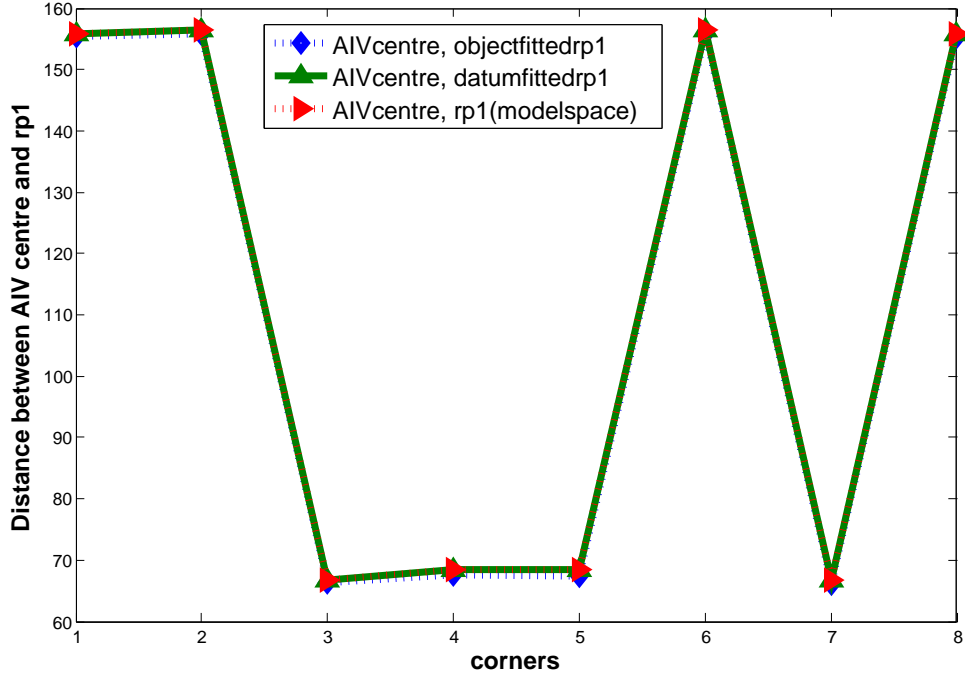
Figure 5.7: Distance between AIV centre and rectangular prism 1 component(object, datum fitted), object(modelspace)

used to find the spatial compatibility of the components. It also resembles an elongated rectangular prism which has a triangular prism like shape at the end. This test is necessary since there might be a situation where engine bursts and component placement is affected in the assembly. The purpose of the test is to assist installation optimization. The rules that are formulated in the constraints as shown in Fig. 5.12 need to be checked. The safety result(HD-High Danger, LD-Low Danger) and the corresponding possible cause condition are provided in the left and right side respectively. This means that the components should not be placed in the trajectory path that has particular risk possibility either low or high. The purpose is to alleviate the high danger conditions while minimizing the low danger conditions.
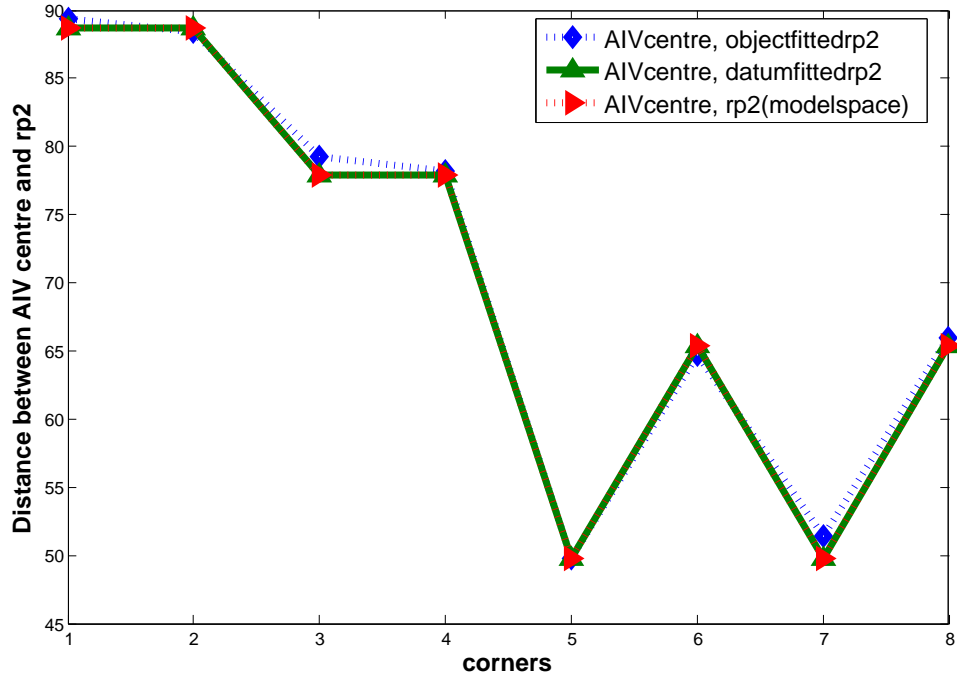
Figure 5.8: Distance between AIV centre and rectangular prism 2 component(object, datum fitted), object(modelspace)

5.2.4.0.2 Use case The trajectory is modeled using the DMU model XML knowledge and the pose information recovered from the transformation between a DMU and the 3D point cloud model (refer Fig. 5.11).
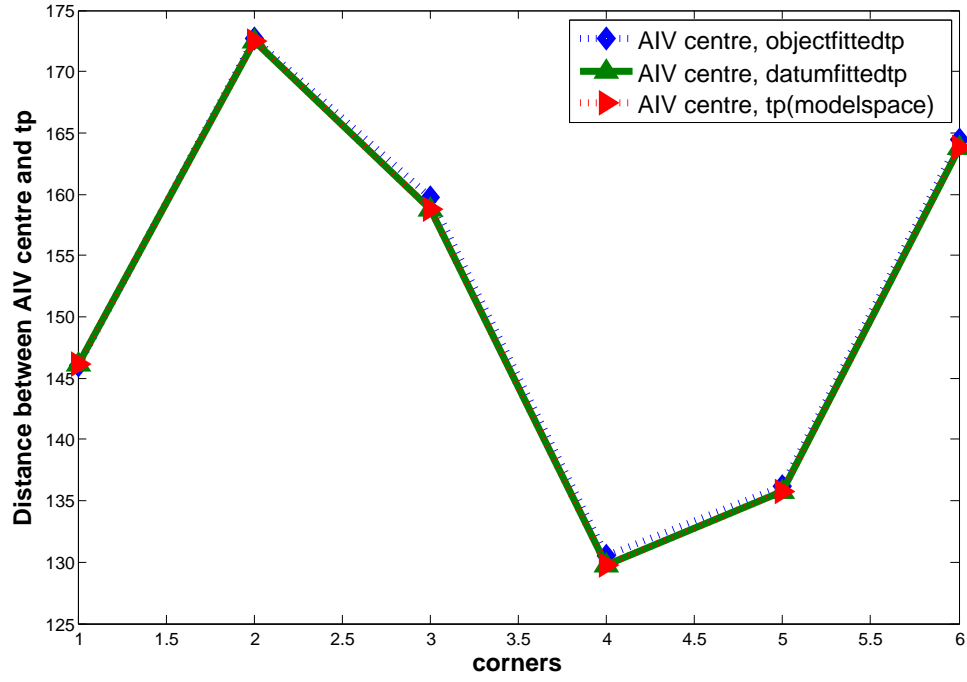
Figure 5.9: Distance between AIV centre and triangular prism component(object, datum fitted), object(modelspace)
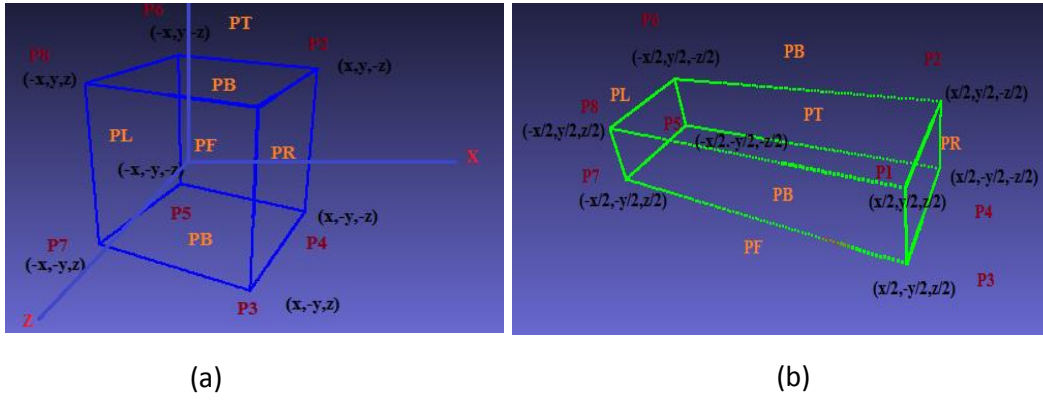
Figure 5.10: Planes in sample assembly components (a-cube, b- rectangular prism)

AIV,Assembly component	Contains	Intersect	Disjoint
AIV, BLUE CUBE	Yes	No	No
AIV, RECTANGULARPRISM (GREEN)	No	No	Yes
AIV, RECTANGULAR PRISM (YELLOW)	No	Yes	No
AIV, TRIANGULAR PRISM (RED)	No	No	Yes

Table 5.2: AIV, Assembly component spatial relationship
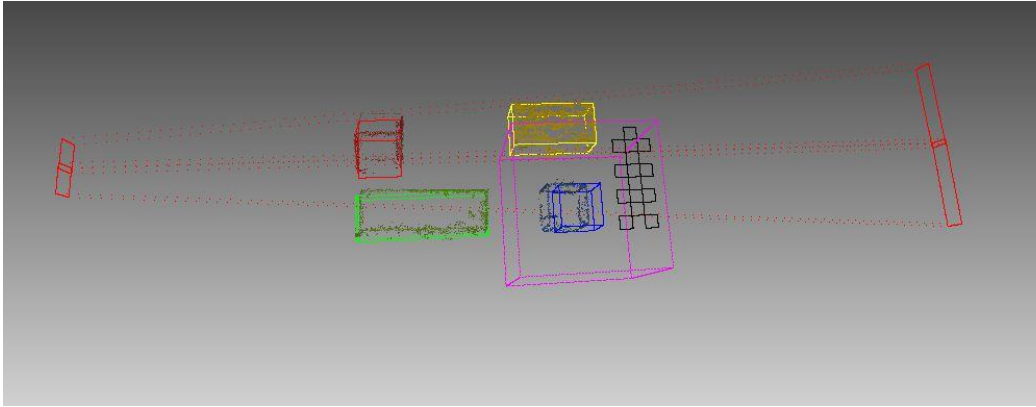
Figure 5.11: Trajectory and the fitted 3D point cloud

```
DEFINE high_danger1: BOOLEAN : = ((intersect(traj1,final(comp1)) and
Intersect(traj1, final(comp2))) or (intersect(traj1, final(comp1)) and
Intersect(traj1, final(comp3))) or (intersect(traj1, final(comp2)) and
Intersect(traj1, final(comp3))))
DEFINE high_danger2: BOOLEAN : =  ((intersect(traj2,final(comp1)) and
Intersect(traj2, final(comp2))) or (intersect(traj2, final(comp1)) and
Intersect(traj2, final(comp3))) or (intersect(traj2, final(comp2)) and
Intersect(traj2, final(comp3))))
DEFINE high_danger3: BOOLEAN : =  ((intersect(traj3,final(comp1)) and
Intersect(traj3, final(comp2))) or (intersect(traj3, final(comp1)) and
Intersect(traj3, final(comp3))) or (intersect(traj3, final(comp2)) and
Intersect(traj3, final(comp3))))
Formula (high_danger1 = false) and (high_danger2 = false) And (high_danger3 = false)
DEFINE low_danger1: BOOLEAN : =  (intersect(traj1, final(comp1)) or
intersect(traj1, final(comp2)) or intersect(traj1, final(comp3)))
DEFINE low_danger2: BOOLEAN : = (intersect(traj2, final(comp1)) or
intersect(traj2, final(comp2)) or intersect(traj2, final(comp3)))
DEFINE low_danger3: BOOLEAN : = (intersect(traj3, final(comp1)) or
intersect(traj3, final(comp2)) or intersect(traj3, final(comp3)))
PROBLEM ADJUST MINIMIZE countTrue (low_danger1, low_danger2, low_danger3)
```

Figure 5.12: Constraints for checking the trajectory

135

5.2.4.1 Plane plane intersection

A simplistic approach is to define plane for the assembly components in 3D space and check the intersection with other planes i.e checking 6 planes of trajectory intersecting with any of the 23 planes of the components?. C1 Cube 6 planes (plane top, plane bottom, plane left, plane right, plane front, plane back) C2 RP 6 planes (plane top, plane bottom, plane left, plane right, plane front, plane back) C3 RP 6 planes (plane top, plane bottom, plane left, plane right, plane front, plane back) C4 TP 5 planes (slant left, slant right, plane front, plane back, plane bottom) Volume V1 6 planes (plane top, plane bottom, plane left, plane right, plane front, plane back). Given the normal vectors to the planes and any point on the plane, the relationship between planes can be identified. If the dot product between N1 and V is 0, then plane 1 and plane 2 coincide else it is disjoint. Where V is the vector between a point that belongs to plane 1 and point that belongs to plane 2. The plane collide iff there exists a line when they intersect. If the planes are not parallel, they should intersect in a line. Plane plane intersection is not optimal solution for complex shapes and further only plane front and back trajectories can be used to check collision.

5.2.4.2 Octrees collision detection

Octrees discretize the input data and are used to partition 3D points recursively into eight octants (refer Fig.5.13). The node stores center of the space the node represents. The advantage of the octree is that only required surface levels to be tested can be considered. The point cloud is projected from object space to planar space using octree approach and then perform the collision detection in this space. In this example, we use the node that has the maximum number of grids at a given level. The translated object is calculated based on the pose recovered using datum.
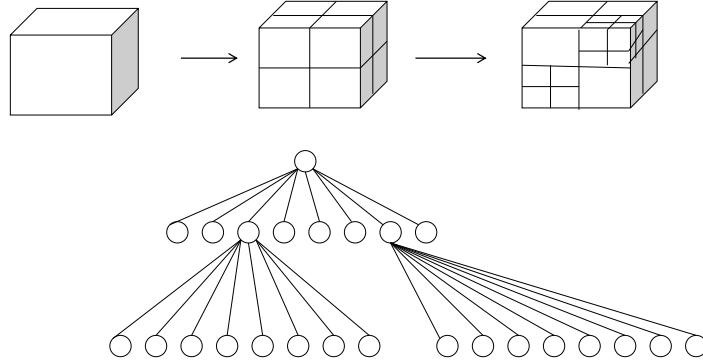
Figure 5.13: Octree representation

Algorithm 6 Obtaining octree group center (pseudo code)

1: Estimate pose between CV model and geometry model

2: transobjectlabel= transformpoints(pose, modelpoints)

3: tx3 = transobjectlabel$(:, 1)$

4: ty3 = transobjectlabel$(:, 2)$

5: tz3 = transobjectlabel$(:, 3)$

6: tOctreelabel= Octree(tx3, ty3, tz3, MinCubeSide)

For example, the transformation matrix in the sample assembly setup is

$$MCalib = \begin{bmatrix} 0.9999 & -0.0125 & -0.0007 & -20.2380 \\ 0.0125 & 0.9999 & -0.0076 & 28.9850 \\ 0.0008 & 0.0076 & 1.0000 & 0.3247 \\ 0 & 0 & 0 & 1.0000 \end{bmatrix}$$

The minimum cube side is 2. For each of the components, the number of groups considered for the discretization is provided in Table .5.3. There are child, groupcentre and cube length for each of the octree representation.

S is the structure level of octree.

Algorithm 7 Obtaining octree group center (pseudo code)

1: for $I = 1$:Ngroups

2: $x(I, :)$ =tOctreelabel$(1, S)$. group$(1, I)$.groupcenter

3: end

Component	NGroups
Traj1	1:249
Traj2	1:269
Bluecube	1:60
RP1	1:170
RP2	1:96
TP	1:63

Table 5.3: Groups for octree discretization

Here label represents either the trajectory, components or volume (AIV) , x is the array that contains 3D point center of the nodes. It can be observed from Table 5.5 and 5.4 that HD1, LD1, LD2 is true for the defined scenario. For installation optimization, LD1 and LD2 should be minimized by relocating the components such that the rules C1 is not in trajectory 1 , C2 is not in trajectory 1, C3 is not in trajectory 2 are adhered. The high danger condition as shown in constraint T1,C1+T1,C2 should be false. This can be achieved since by minimizing the low danger condition LD1, high danger is automatically alleviated (since LD1 is a subset of HD1 with aggregated conditions based on and operation).

Intersect	Traj1	Traj2
Bluecube	1	0
RP1	1	0
RP2	1	1
TP	1	1

Table 5.4: Intersect status of the components (trajectory and point cloud)

Safety condition	HD1	HD2	LD1	LD2
T,C(IR)	(T1,C1 and T1,C2) or (T1,C1 and T1, C3) or (T1, C2) and (T1, C3)	(T2,C1 and T2,C2) or (T2,C1 and T1,C3) or (T2, C2 and T2, C3)	(T1, C1 or T1, C2 or T1, C3)	(T2, C1 or T2, C2 or T2, C3)

Table 5.5: Trajectory component and danger conditions, IR-intersect relation
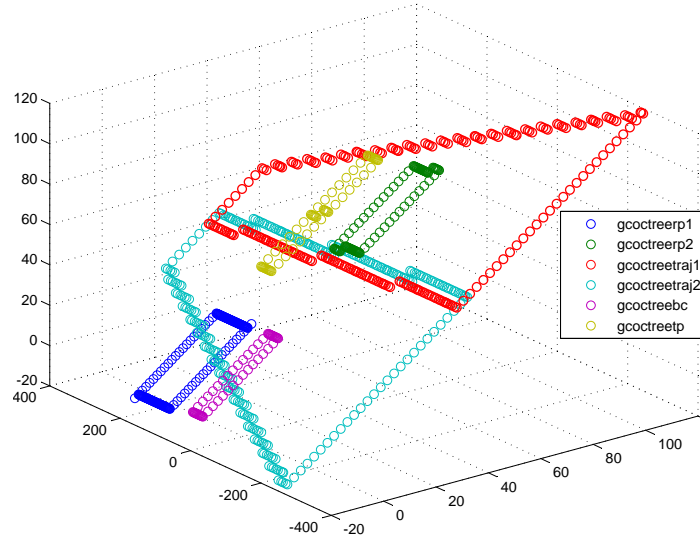
Figure 5.14: Scatter plot of group centers (octree) of assembly setup (gc-group center)

5.3 Conclusions

In this Chapter, we described how installation constraints are useful and the various means of testing on the point cloud model to check these constraints during the production audit process. AIV contains, intersect and trajectory testing has been discussed. AIV testing is performed based on the necessity to make measurement within areas where the assembly or DMU is not complete or where the nature of the material used in the product means that accurate measurements are not possible but rather the part needs to be shown to lie within an acceptable installation volume, eg. wire harness attachment points are precisely installed to within a tight tolerance though the actual wire harness are installed compared to a much more approximate tolerance. Trajectory testing is performed to check how far the components have deviated from the trajectory path since there might be a situation where engine bursts and component placement is affected in the

assembly.

The various boundary based, spatial constraint based, plane plane intersection and octree collision detection based algorithms were discussed. A novel algorithm is proposed which uses the pose recovered from the PCAT module and perform collision detection between AIVs, trajectory and the installation components. The usefulness of AIV and trajectory concept to check the point cloud installation to assist the production audit process has been shown.

Chapter 6

Conclusions

The aim of this chapter is to summarize what have been discussed in this thesis. A description of what has been achieved in this research work, benefit and possible direction for facilitating the safety engineer to use the tool has been outlined.

In Chapter 1, basic concepts of the production audit process and prevailing limitations were discussed. The scope for improvement and a list of contributions that had been accomplished during this research work were provided. The work was about the 3D vision assisted production safety audit process that takes measurements of the fabricated sections of an installation from captured images and uses model knowledge to compare the measurements to the required constraints defined within the specification model of the installation in the Digital Mock-Up. The background and existing techniques were described in Chapter 2. The background of the industrial environment and the application concept development process was discussed. The list of identified requirements that need to be considered while designing the application and challenges had been summarized. We provided the literature review made on 3D model based vision related to the production audit. The survey was presented in two key fields related to this research such as user guided measurement and testing, discrepancy checking of installations. The techniques such as discrepancy checking using augmented reality and ToF cameras were discussed. Few basics of 3D reconstruction were described. In Chapter 3, a proposed framework for 3D PAMT (Production audit measurement tool) that uses input from a digital camera used for the verification tasks was presented. A 3D based measurement system was demonstrated with capa-

bilities that aims to assist the safety personnel and verify whether the following segregation constraints are duly respected a) distance between datum points of interest, b) distance between points with respect to planar surface. The advantage of optimally using the combination of both point and automatic disparity coupled with planar surface detection was demonstrated.

A new cost-effective and robust framework for 3D PACT (Production audit compare tool) that uses input from a digital camera and semantic metadata knowledge available from geometry models which can be used for verification tasks was described in Chapter 4. The discrepancy checking and analysis of CV model with DMU was carried out to confirm whether the installation requirements are met. Furthermore, 3D feature extraction and classification had also been studied to complement the verification tasks. Ideally, the framework acted as proof of concept for safety analysis and verification and had been tested with a controlled environment data set for model matching. 3D object structures with respect to another object's position in the scene can be extracted.

In Chapter 5, how safety installation constraints fit within the production audit process had been described. It was also demonstrated that how the installation optimization constraints file were linked to the ACAT (Advanced constraint analysis tool) and to perform an automated check that the installation constraints were respected.

In Annexure I, the description of user manuals for the proposed production audit software such as 3D PAMT, 3D PACT were provided. Further, an additional software for constraint viewing has been described.

In Annexure II, 3DHT (3D Hough Transform) technique for planar surface detection is discussed.

Finally, in Annexure III, a new depth based segmentation technique GrabcutD, an improvement to existing Grabcut (graph cut based) segmentation method was proposed. The goal is to extract pixel accurate object silhouettes from the multiple views of an object that can then be used to generate 3D convex hulls for the objects. Conventional Grabcut relies only on colour information to achieve segmentation. However, in stereo or multiview analysis, there is an additional information that could be also used to improve segmentation. Clearly, depth based approaches bear the potential discriminative power of ascertaining whether the

object is nearer of farer. We show the usefulness of the approach when stereo information is available and evaluate it using the standard datasets against state of the art result.

6.1 Impact, Benefit of the proposed system

This research proposed a potential capability that could be used to both audit the product during final assembly against safety installation constraints and may be used within the line maintenance environment to check if in-service modifications or damage invalidates an installation constraint or safety assumption regarding an installation. The workflow that has been identified seems to fit the needs of both the engineers in both use scenarios. The measurement system has been tested by a MISSA project partner on an industrial aircraft dataset and results are quite promising. The user experience needs to be polished but the main ideas needed behind the user interface have been identified and are mainly present in the developed application. The key success of the proposed system is underlined by low cost budget and processing time compared to existing methods in the market which are much expensive and time consuming. Moreover, the end users typically need less knowledge expertise to handle system which process the selected portion of the environment at a particular instant of time that need to be analyzed. It is anticipated that this type of cost and time driven solutions would ideally have a tangible impact over a period of time in both small and large scale industrial applications that require 3D measurements and discrepancy checking. Further, the various techniques in proposed system can also be potentially deployed in other applications that are based on rigid and non deformable objects. More details regarding run time analysis of the system are provided in Section. 6.1.1.

6.1.1 Algorithm run time analysis

A system with Intel (R) core 2 Duo CPU E8500 at 3.16 GHz, 2GB RAM and Matlab/C++ based implementation is used for the experiments. The run time results for various algorithms are reported in Table. 6.1. In the PAMT mea-

Algorithm	Average runtime(sec)
PAMT measurement (C++,Matlab)	< 60
PAMT planar, angular analysis (Matlab)	< 300
PACT discrepancy checking (C++,Matlab) (excluding 3D reconstruction)	< 240

Table 6.1: Algorithm run time

surement process, excluding calibration, from the experiments conducted on few trials of data algorithm computes in less than 60 Sec. The distance with reference to planar surface angular analysis is achieved < 300 Sec. In order to speed up the calibration process, C++ environment is preferred. This has to be done initially just once for each new environment. Similarly, in the PACT discrepancy checking process, 3D reconstruction using structure from motion in C++ is computationally expensive. The processing time is around 3 to 4 hours depending on the number of multiview images. All the other processes (combination of both Matlab and C++ implementation) are completed in less than 4 minutes for the defined use case scenario. In the PACT algorithm, C++ is used for outlier removal using connected component analysis.

6.2 Progress achieved beyond the state of the art

The following section discusses the achieved advancement beyond the state of the art. Work was carried out in two fields. The first was related to 3D Measurement and Testing and the second was related to 3D Model Matching.

6.2.1 3D measurement progress achieved beyond the state of the art

Beyond the state of the art, we designed and implemented a 3D measurement tool which can provide accurate measurements between any given datum points of

interest at the mm level in an installation environment with or without reference to the planar surface model. The idea of using image based stereoscopic measurement and the optimal combination of point based and automatic disparity for measurement in 3D space with respect to the planar surface reference is used in the proposed framework. A system that addresses the problem of accurate 3D measurement of any given environment in a relatively quicker manner. The distance between any datum points of interest can be estimated with or without reference to planar surface model. The tool uses multiple views of a scene captured using a basic digital camera and requires the inclusion of a calibration grid within the scene. The application allows for a user to select a specified number of consistent points/features on the multiple views, which allows the application to indicate the distance between every permutation of pairs of these points and also to give a measure of off-set from a selected reference plane, also defined by three of the selected points.

6.2.2 3D model matching progress achieved beyond the state of the art

Traditional 3D model matching approaches perform model matching between any given cloud of points. The key achievement in our work is: the usage of CATIA model information to perform model alignment and discrepancy checking. This is obtained by point cloud processing (segmentation) and pose estimation of any predefined datum and individual objects present in the scene. A system has been developed that performs discrepancy checking between 3D vision model of the actual installation to the original digital mock-up, with a focus on the safety driven installation constraints, such as segregation, proximity and orientation. The tool takes a set of images and uses various image processing techniques to create a 3D digital reconstruction of the objects within the set of images. The digital reconstruction is in the form of a cloud of points. The application takes the digital mock-up of the scene and converts it into a similarly defined cloud of points. The two clouds are aligned to each other such that the selected datum, an equipment or calibration chart, has the lowest positioning error, based on

feature extraction and comparison techniques. The rest of the equipment that are recognized are then aligned to the digital reconstruction. The offsets for these equipments are reported as positioning errors that can be used to check for acceptability to the installation tolerances derived from zonal safety installation constraints. The measurement error of the tool is estimated by comparing the measures performed on the calibration grid against the actual dimensions of the grid.

The work has achieved the main objective of defining an application to address production audit of the fabricated product from a safety perspective and developing and testing a prototype. Not all the requirements have been achieved that are needed for an industrial instance of such an application.

Chapter 7

Annexure I : Production Audit Software

7.1 MISSA3DAudit PAMT GUI tool details

7.1.0.0.1 Compiling instructions: Run Click on MISSA3DAudit.exe to run the system. Initially, command prompt will be shown and will take some time to open the GUI. Note : If you get dll errors, then install MCRInstaller.exe which is enclosed in package.

7.1.1 GUI tool details

GUI Tool comprises three main panels in the tool. A) Display Panel, B) Control Panel and C) Result Panel.

7.1.1.0.2 Load images User selects the production audit images (here after called images) using this button as shown below.
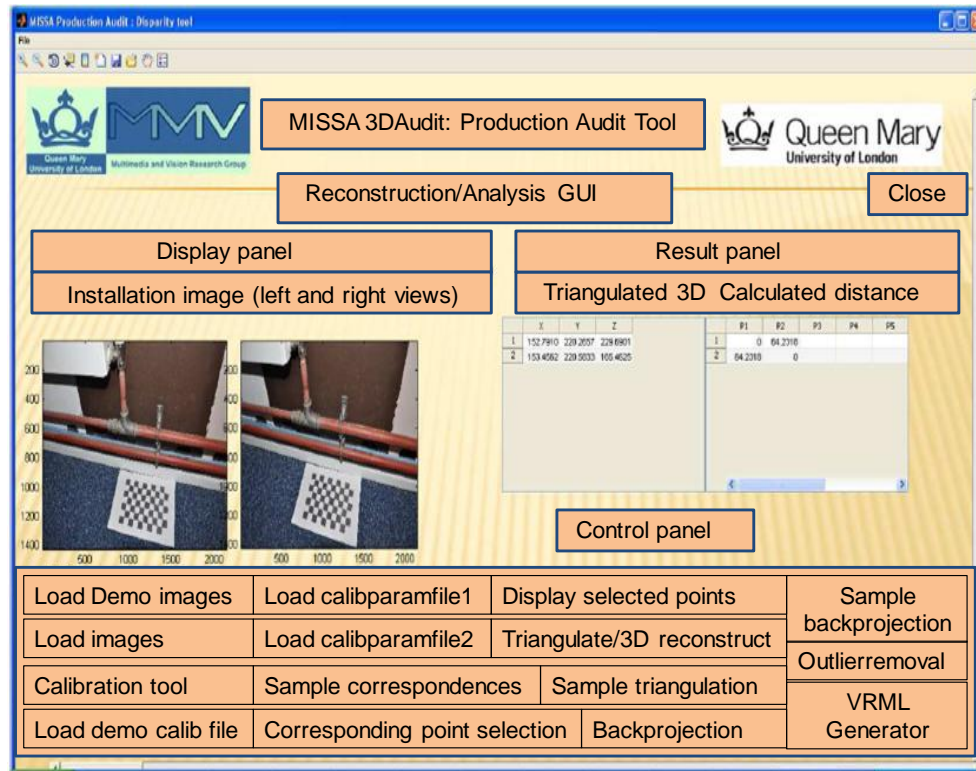
Figure 7.1: MISSA 3D audit GUI system
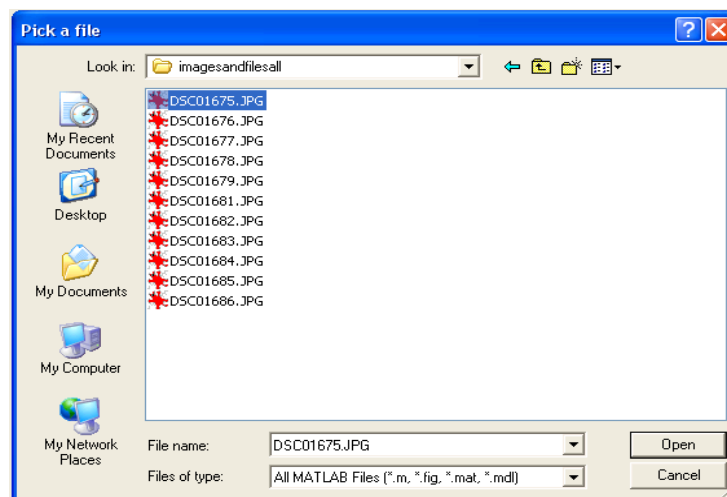
Figure 7.2: Loading images

7.1.1.0.3 Calibration tool The images need to be captured using the calibration pattern attached in the package and placed inside the folder "ImagesandCalibfiles". For further detailed instructions, refer Calibration Instructions Manual. This utility will generate .clb calibration file for all the images.

7.1.1.0.4 LoadCalibParamfile1 Here user selects .clb calibration file corresponding to first image. The calibration file contains intrinsic and extrinsic camera parameters.

7.1.1.0.5 LoadCalibParamfile2 Here user selects .clb calibration file corresponding to second image. The calibration file contains intrinsic and extrinsic camera parameters.

7.1.1.0.6 Corresponding Points selection User will be given choice how many points he needs to select for the inspection and safety analysis process. While selecting points, first user will be shown left image where he will select the point coarsely. Then, the system will show the point location on left image that is zoomed so that user can select the point location very accurately (refer Fig. 7.3). Similarly, user will be shown right image where he will select the point coarsely. Then the point location on right image will be zoomed so that user can select the point location very accurately. Note: Before running corresponding points selection button, close any displayed Message Dialog box figures. Upon successful loading of corresponding points, following message dialog box will be displayed.

7.1.1.0.7 Display selected points This will show the selected points with tags like P1, P2..etc as shown in Fig. 7.4.

7.1.1.0.8 Triangulate/3D Reconstruct Utilizing correspondence points and camera parameters, this button reconstructs 3D points. This also finds the 3D distances between individual points. Both 3D point locations and distances are fed to result panel.
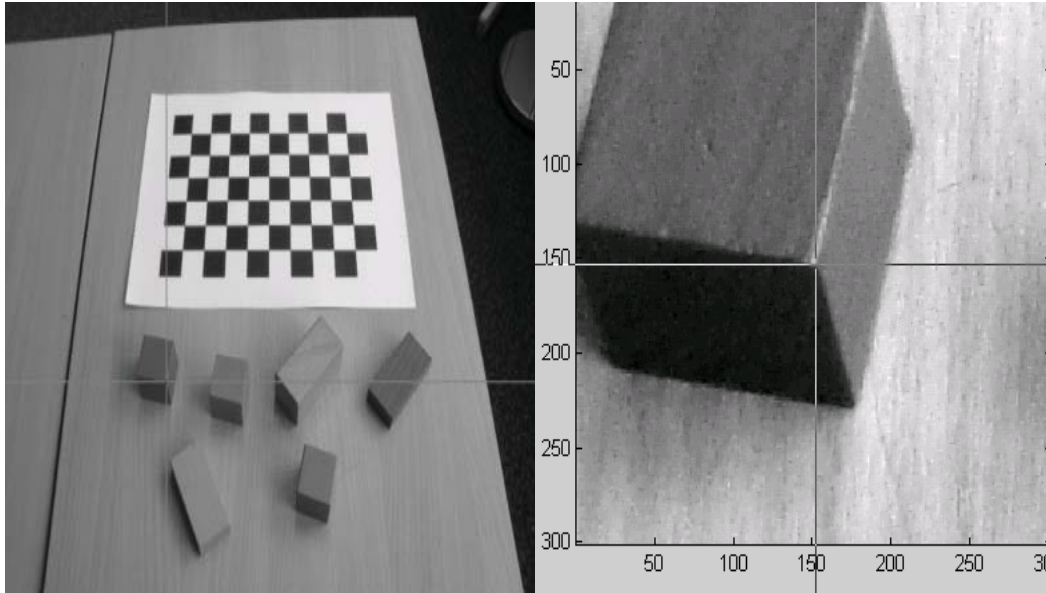
Figure 7.3: Selection of point(coarsely) and (accurate)
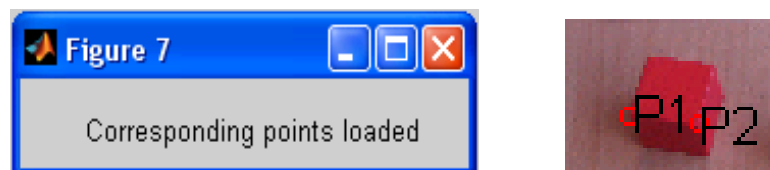
Figure 7.4: (a) Message dialog box and (b) displaying selected points

7.1.1.0.9 Backprojection Using the known camera parameters, calculated 3D points are backprojected to each of the camera for testing purpose.

7.1.1.0.10 VRML generator VRML generator converts calculated 3D points to VRML proprietary format and displays using VRML viewer.

7.1.2 Result panel

This shows reconstructed 3D points and 3D distances as shown in Fig. 7.5.

3D Points   3D distance

	X	Y	Z
1	-26.8594	-118.4836	31.8443
2	-24.2252	-90.9968	34.3163
3	-2.5024	-113.9152	17.0732

	P1	P2	P3	P4	P5
1	0	27.7231	28.8498		
2	27.7231	0	35.9785		
3	28.8498	35.9785	0		

Figure 7.5: Result panel

7.1.3   Sample case

The buttons highlighted in red colour makes this sample case sequence run flow. First load images (DSC01677.JPG, DSC01678.JPG) and then followed by corresponding calibration files (LoadDemoCalibfile button).

7.1.3.0.11   Sample correspondences: This will load the set of correspondences which are already selected. One can visually those points using display points button.

7.1.3.0.12   Sample triangulation   Using the sample correspondences , this button displays reconstructed 3D points and distances.

7.2   MISSA 3D model matcher/discrepancy checking GUI tool details

7.2.0.0.13   Compiling instructions: Run Click on MISSAModelMatcherV2.exe to run the system. Initially, command prompt will be shown and will take some

time to open the GUI. Note : If you get dll errors, then install MCRInstaller.exe which is enclosed in package.

7.2.1 GUI tool details
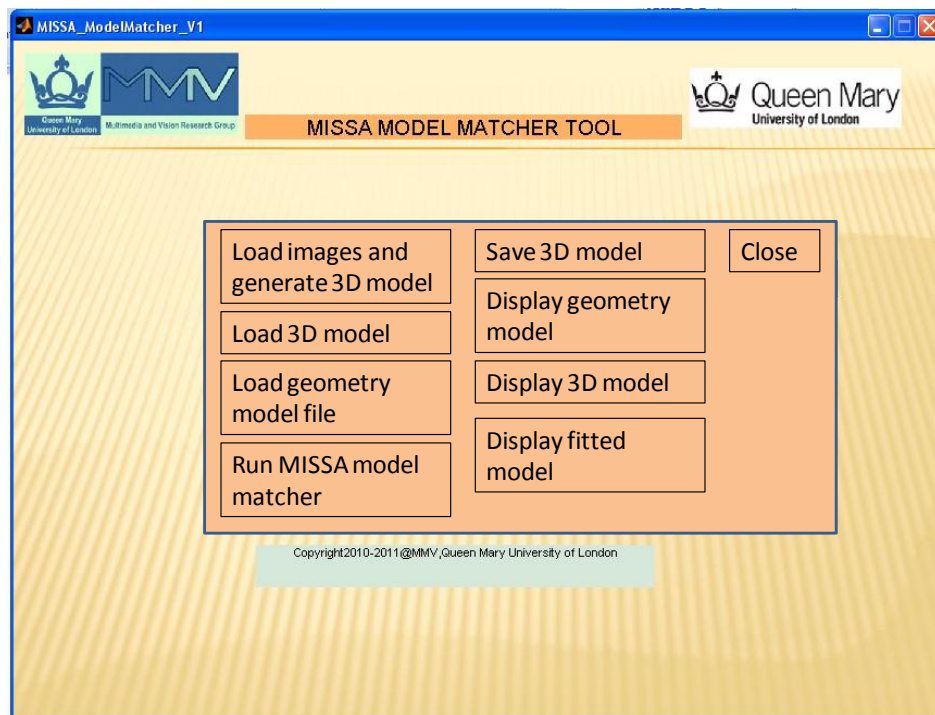
Figure 7.6: Model matcher graphical user interface

7.2.1.0.14 Prerequisites All the captured multiview images need to be in a separate folder (say images). Model file of the environment (text format) is required for matching process (see example: modelcalibrationdatum.txt for more information). Install MeshLab [10], an opensource software in the PC. It is assumed that Software is installed in C:Files MeshLab. Otherwise display buttons will not work.

7.2.1.0.15 Sequence of flow to test   The first sequence can be followed if new set of images are fed to the system for 3D processing.

7.2.1.0.16 Sequence 1   Loadimages and generate 3D Model → Display 3D model → Load geometry model file → Display geometry model → Run MISSA Model Matcher → Display fitted model The second sequence is suitable for reusing the already generated 3D model for further processing.

7.2.1.0.17 Sequence 2   Load 3D model → Display 3D model → Load geometry model → Display geometry model → Run MISSA model matcher → Display fitted model.

7.2.1.0.18 Load images and Generate 3D model (Button1):   Here all the images are loaded and steps including calibration and 3D model generation from images are achieved. This will take some minutes and once the process is done, a display message can be noticed. This will also use whole processor, so do not try to run other applications while it is generating 3D cloud.

7.2.1.0.19 Save 3D model(Button2):   The 3D model generated from the previous step is saved in any required folder for further use.

7.2.1.0.20 Load 3D model (Button 3):   Any model that is saved in the previous step before is loaded into the run environment.

7.2.1.0.21 Display 3D model (Button 4):   Using this button, either generated 3D model from button 1 or button 3 can be visualized using MeshLab Viewer. As it can be seen in GUI, this button is common for viewing 3D from button 1 or from button 3.

7.2.1.0.22 Load geometry model file (Button 5):   The geometry file (or model file) is loaded. In this release, a model format as shown below is used. The last column highlighted in red defines which object is the datum (by setting value as 1). Datum object should be the first object if it is CALIBRATION pattern,

second object if it is different than CALIBRATION pattern. CALIBRATION pattern should always be the first object in the file).

7.2.1.0.23 Display geometry model file (Button 6): Visualization of the model file can be done.

7.2.1.0.24 Run MISSA model matcher (Button 7): Using both the generated 3D model obtained from images and the geometry model, model matching can be achieved.

7.2.1.0.25 Display fitted model (Button 8): The result of fitting 3D model with geometry model can be visualized. In current version release, the discrepancy checking result can be seen in the command line.

7.2.1.0.26 Close (Button 9) The system can be quit.

7.3 Software: 3D viewer utility and constraint viewer

This GUI software is based on QT (Qtopia) framework, OpenGL and C++ implementation. The customized application is built with constraint msat viewing utility on top of the opensource meshviewer.

7.3.0.0.27 Load 3D CV model: Loads and displays 3D cloud from an existing folder.

7.3.0.0.28 Load installation constraints: Load and displays installation constraint msat format, xml format

7.3.0.0.29 Load Discrepancy output model: Load and displays 3D fitted cloud.
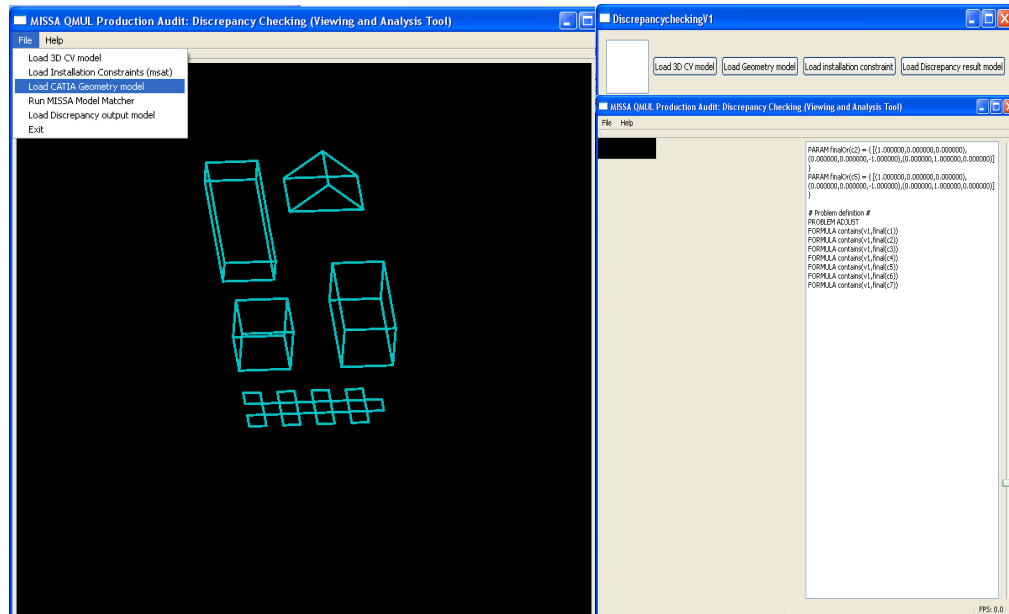
Figure 7.7: 3D viewer utility and constraint viewer

7.3.0.0.30 Load CATIA geometry model: Load and displays 3D geometry model. This is GUI supplementary software and has two way control windows one for viewing the 3D cloud and other for viewing installation constraints and has control bar for accessing 3D cloud views. Note: Multiwindows in same interface makes cloud unzoomable and hence in this version 1 window but 2 way control is designed for both installation constraint viewing and 3D viewing. Acknowledgements: Meshviewer.

Chapter 8

Annexure II: 3D Hough transform

8.1 3D Hough transform

3DHT is used to fit plane to 3D points and it is performed in parametric space Borrmann et al. (2010) [19] Kurdi et al. (2007) [77]. As discussed in Chapter 3, plane can be parameterized by its normal vector n and its distance from the origin. Initially parameters of model (plane) are calculated, and the method looks for the accumulator's bin that the parameters fall into, and increase the value of that bin. Local maxima in accumulator space are identified by finding the bins with the highest values. The peaks can be identified by fixing the threshold and by choosing multiple thresholds, primary and secondary level plane can be extracted. The 3D point cloud (pointlist) which is a matrix of three coordinate columns X, Y and Z; θ, ϕ and ρ axis (discrete intervals). The pseudo code for 3D hough transform is given in algorithm 8.

Dis_{min} and Dis_{max} are the distances between the origin and the two extremities of the cloud points calculated at lines 1 and 2; H is a 3D matrix; θ_{mat}, ϕ_{mat}

Algorithm 8 3DHT algorithm: planar surface detection [77]

1: $X_{min} = \min(X)$; $Y_{min} = \min(Y)$; $Z_{min} = \min(Z)$
2: $X_{max} = \max(X)$; $Y_{max} = \max(Y)$; $Z_{max} = \max(Z)$
3: Calculation of Dis_{min}; Dis_{max}
4: θ = from 0 to 360, step = θ_{step}; n_θ = length(θ)
5: ϕ = from -90 to +90, step = ϕ_{step}; n_ϕ = length(ϕ)
6: $n_{rho} = 2 \times (Dis_{max} - Dis_{min})/\rho_{step}$
7: ρ = from Dis_{min} to Dis_{max}; step = ρ_{step}
8: $\theta_{mat}(n_\phi, n_\theta) = [\theta \quad \theta \quad \theta \quad \theta]'\pi/180$
9: $\phi_{mat}(n_\phi, n_\theta) = [\phi \quad \phi \quad \phi \quad \phi]'\pi/180$
10: $H(n_\theta, n_\phi, n_\rho) = 0$
11: ratio = $(n_\rho - 1)/(\rho(n_\rho) - \rho(1))$
12: for k = 1 to length(X)
13: $\rho_{mat} = cos(\phi_{mat})cos(\theta_{mat})X(k) + ...cos(\phi_{mat})sin(\theta_{mat})Y(k) + sin(\phi_{mat})Z(k)$
14: $\rho_{indix} = round(ratio(\rho_{mat} - \rho(1) + 1))$
15: for i = 1 to n_ϕ
16: for j = 1 to n_θ
17: $H(j, i, \rho_{index}(i, j)) = H(j, i, \rho_{index}(i, j)) + 1$
18: next j ; next i ; next k

and ρ_{mat} are 2D matrices; θ, ϕ and ρ are three lists. Once H is determined as shown in the above algorithm, peak can be detected in the 3D matrix by searching the voxels having maximum values.

As it is shown in Fig. 8.1, the plane obtained using RANSAC and Hough transform is in line with the base of pipe point cloud. However, we used RANSAC since it is faster than other methods due to its iterative concept model.

Plane fitted (RANSAC)
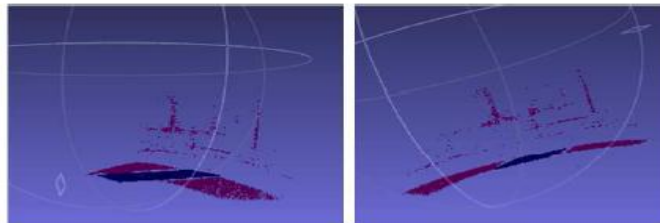
Plane fitted (3D Hough)
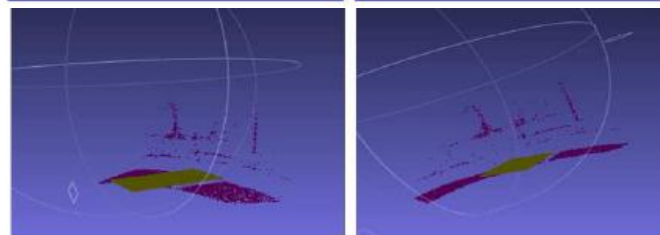
Figure 8.1: Plane fitting using RANSAC, 3DHT

Chapter 9

Annexure III : GRABCUTD : Improved GRABCUT using depth information

Popular state of the art segmentation methods such as GrabCut include a matting technique to calculate the alpha values for boundaries of segmented regions. Conventional GrabCut relies only on color information to achieve segmentation. Recently, there have been attempts to improve GrabCut using motion in video sequences. However, in stereo or multi-view analysis, there is additional information that could be also used to improve segmentation. Clearly, depth based approaches bear the potential discriminative power of ascertaining whether the object is nearer of farer. In this work, we propose and evaluate a GrabCut segmentation technique based on combination of color and depth information. We show the usefulness of the approach when stereo information is available and evaluate it using standard datasets against state of the art results. The hypothesis statement of this Chapter is H4: Using extra depth information in existing graph cut mechanism, segmentation can be improved.

9.1 Introduction

Image segmentation has been very old and active research over several decades. It can be used in silhouette generation which is used in many potential computer vision applications such as 3D reconstruction using visual hull [35], event detection [55] etc. For example, Guillemaut et al. (2009) has used joint robust Graph-cut optimization and reconstruction for high quality free viewpoint video [40]. Izquierdo et al. (2002) has explained the key components that are necessary for an advanced segmentation toolbox [45]. The six different schemes deployed are variance-based detection of uniform regions, real-time histogram-based segmentation, fast nonlinear diffusion, diffusion-based object segmentation, morphology-based object segmentation and object segmentation by contour matching.

Popular state of the art segmentation methods such as GrabCut by Rother et al. (2004) include a matting technique to calculate the alpha values for boundaries of segmented regions [67]. Conventional GrabCut relies only on color information to achieve segmentation. Our goal is to enhance the capability of GrabCut technique using depth information obtained from stereo or multiview analysis. GrabCut is an improved version of GraphCut which uses energy minimization techniques for segmentation [20] [51]. Lazy snapping is another interactive image cut system which is also based on graph cut and boundary refinement.

Recently, there have been few attempts made to improve the existing GrabCut technique. Corrigan et al. (2008), has provided a matting using motion extended GrabCut which works for videos [30].

Han et al. (2009) has extended the GrabCut integrating multiscale nonlinear structure tensor [41]. Chen et al. (2008) has provided improved GrabCut using Gaussian mixture model optimization [26]. Prakash et al. (2008) has provided a combined approach based on both active contour and GrabCut for automatic foreground object segmentation [62].

Other approaches for foreground segmentation utilize extra information by processing two images. Sun et al. (2007) has proposed Flashcut for foreground segmentation based on flash, motion, and colour information [75]. Reinhard et al. (2005) has used depth of field information in which they consider object which

is in focus and other without focus [66].

As stated by Torralba et al. (2002), there exists a strong relationship between structure of the scene and depth [78]. In stereo or multi-view analysis, there is additional information that could be also used to improve segmentation. In fact, during the generation of computer generated 3D movies and animations, depth information is known prior. The depth can also be generated using (Time of Flight) ToF cameras or using stereo vision techniques. Several methods for disparity estimation [17], [84] have been proposed. They can be categorized into local and global stereo methods.

The only approach that uses depth information in foreground segmentation is Zhu et al. (2009), who has provided a methodology for optimized depth inference where information from both depth and stereo images are considered. Thus obtained depth map is subsequently used to enhance matting [90].

The organization of the Chapter is as follows: In Sect. 9.1.1, we will explain GrabCut techniques. In Sect. 9.2.1, depth based segmentation is described. GrabCut using 4 channels is explained in Sect. 9.2.2. In Sect. 9.3, we provide the experimental results and analysis of our framework. Finally, the conclusions and futurework are described in Sect. 9.4.

Figure 9.1: Ballet sequence image [13]

The dataset of ballet sequence is as shown in Fig. 9.1. The disparity map of image can be obtained using many state of the art local or global methods. For example, the depth map of dancer image [91] is as shown in Fig. 9.1 above.

Figure 9.2: Existing method(GrabCut) results for Ballet sequence

9.1.1 Existing GrabCut technique

Existing GrabCut technique by Rother et al. (2004) [67] works as follows: Initial trimap is created by user selecting a rectangle. Background class B is represented by the pixels outside rectangle and outer are unknown which belongs to foreground class A. The corresponding pixels are assigned to each class which is created using Orchard bouman clustering algorithm. The GMMs are thrown away and new GMMs are learned from the pixel sets created in the previous set. The segmentation is estimated using GraphCut which provides tentative classification of pixels belonging to the respective classes. The above process is iterated until convergence. As one can see from the Fig. 9.2, quality of the existing Grab-Cut over the dataset is not satisfactory especially in the case of dancer. The arm and hand portion is totally missing. This problem can be alleviated using depth information along with the available color based segmentation model.

9.2 Proposed Method

As discussed earlier, our proposed method is based on both depth and colour segmentation model. Firstly, we discuss about depth based segmentation in order to show justification for using disparity along with colour information later in our framework.

9.2.1 Depth based segmentation (DBS)

We have already proposed an optimal framework for finding disparity in model based 3D reconstruction [84]. Using any of the available techniques, disparity map can be found. Disparity range information provides details regarding whether the object is nearer or farer.

The algorithm used in generation of disparity maps by Zitnick et al. (2004) consists of three main steps [91]: a) segmentation of image (smooth image using ansiotropic diffusion function), b) find initial disparity distribution (DSD) for each segment where DSD is set of probabilities over all disparities for individual segment in image and c) disparity smoothing using constraints which states that neighbouring segments with similiar color also has same disparity The depth range of interest can be obtained by finding the pixels with specific disparity range interval and subsequently, silhouette can be formed by assigning the pixels inside a region 255 value for highlighting foreground while all others are assigned 0 (background).

For example, in tsukuba image, we can isolate lamp and head separately (refer Fig. 9.3,9.4). However, in Ballet sequence man dataset, the methodology could not generate accurate silhouettes (refer Fig. 9.3,9.1) since too many pixels fall in the same range that makes classification of pixels more tedious. In fact, this problem could be alleviated by searching for pixels in a range within a bounding box.

Considering the various issues that have been discussed so far, we propose a novel framework that includes both color and depth information.
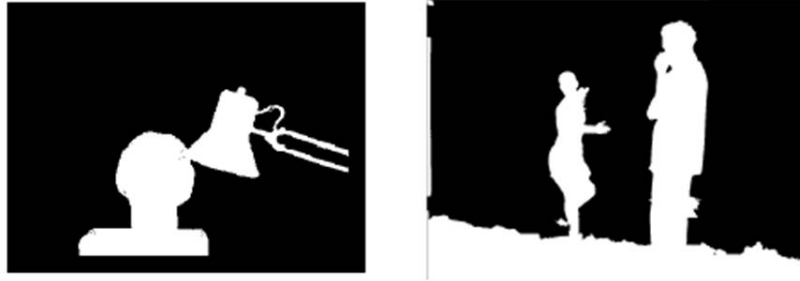
Figure 9.3: Results of silhouette extraction from depth map.
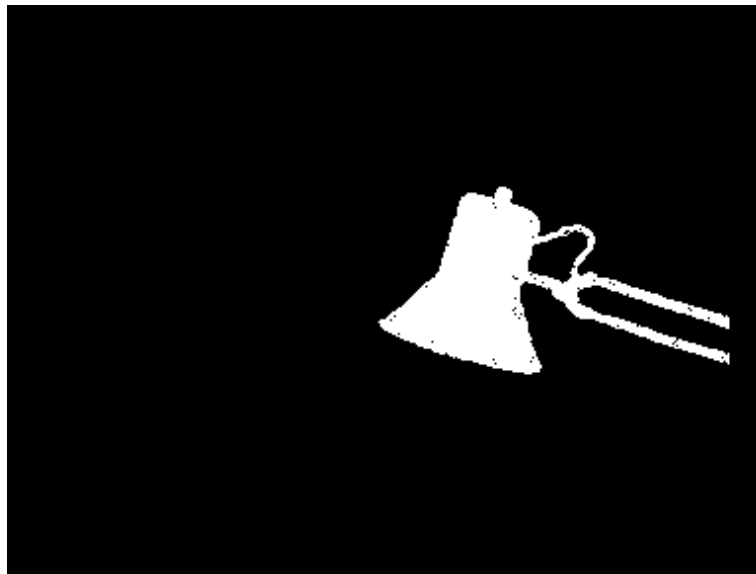
Figure 9.4: Segmentation of lamp in Tsukuba using depth.

Usually, any given image can be represented as 3 channel image *R, G, B* components. RGB values encoded in 24 bits per pixel and are specified using three 8-bit unsigned integers (0 through 255) representing the intensities of red, green, and blue.

We also include the depth level (8 bit) information. In this work, we consider image as 4 channel components including disparity map [0-255]. The values 255 and 0 means nearer and farer respectively. The histogram of R,G,B and D channels of Ballet image is shown in Fig. 9.5.
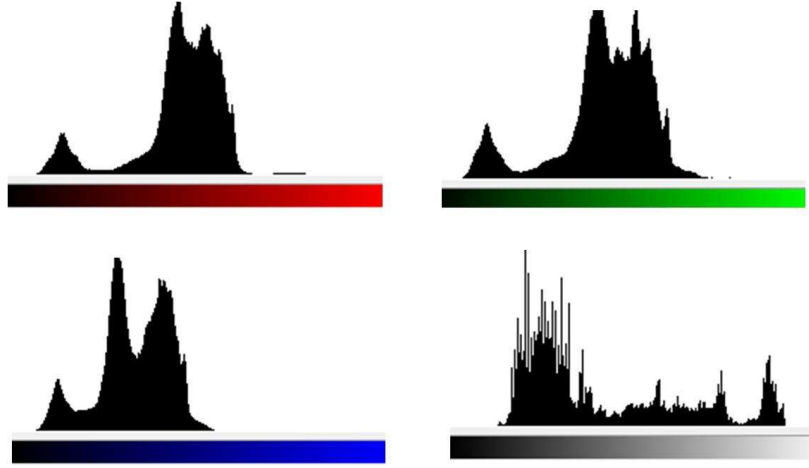
Figure 9.5: Histogram of a)red, b)green, c)blue and d) Depth channels of Ballet image.

9.2.2 GrabcutD: modified GrabCut using 4 channels

GrabCutD works as follows. Initially, user selects a bounding box and the pixels inside and outside rectangle is represented by foreground and background classes respectively. From each trimap selection of the foreground and background, the histograms are formed using 4 channels information ($Red, Green, Blue, Depth$) instead of just color ($Red, Green, Blue$). The gaussian mixture model (GMM) components are assigned to pixels and learned from the 4 channel Image. The energy model is defined based on the foreground and background histograms and the minimum energy represents good segmentation. The segmentation is estimated using graph cut which provides tentative classification of pixels belonging to the respective classes. The above process is iterated until convergence. The

formulation of above mentioned process is briefly described below. Let us consider image as an array $I = (I_1, ..I_n...I_N)$ which includes both R,G,B levels and depth values respectively. The segmentation is array of opacity values $\underline{\alpha} = (\alpha_1, ...\alpha_N)$ at each pixel. 0 for background and 1 for foreground. θ is the parameter which represents foreground and background histogram distribution (histogram model).

$$\underline{\theta} = h(I; \alpha), \alpha = 0, 1 \tag{9.1}$$

Given an image I and model $\underline{\theta}$, the segmentation task is to infer unknown opacity variables $\underline{\alpha}$.

The energy E is defined such that minimum represent good segmentation and it captures coherence in both color space and depth.

$$\underline{\alpha} = \arg\min_{\underline{\alpha}} E(\underline{\alpha}, \underline{\theta}) \tag{9.2}$$

GMM components are a full covariance gaussian mixture with K components ($K = 5$). A vector $k = k_1...k_n...k_N$ is defined and k_n assigns unique GMM component to each pixel either from background or foreground.

The Gibbs energy for segmentation is of the following form

$$E(\underline{\alpha}, k, \underline{\theta}, I) = U(\underline{\alpha}, k, \underline{\theta}, I) + V(\underline{\alpha}, I) \tag{9.3}$$

The data term U which considers both color GMM and depth GMM models evaluates the fit of opacity distribution α to data I. It is defined as follows.

$$U(\underline{\alpha}, k, \underline{\theta}, I) = \sum_n D(\alpha_n, k_n, \underline{\theta}, I_n) \tag{9.4}$$

where

$$D(\alpha_n, k_n, \underline{\theta}, I_n) = -\log p(I_n|, \alpha_n, k_n, \underline{\theta}) - \log \pi(\alpha_n, k_n) \tag{9.5}$$

$p(.)$ is Gaussian probability distribution and $\pi(.)$ refers to mixture weighting coefficient.

The model parameters are represented by

$$\underline{\theta} = \{\pi(\alpha, k), \mu(\alpha, k), \sum(\alpha, k), \alpha = 0, 1, k = 1...K\} \tag{9.6}$$

where π is the weight, μ is the mean and $\sum$ is the covariance of 2K Gaussian components for foreground and background distributions.

Smoothness factor V is defined as follows.

$$V(\alpha, I) = \gamma \sum_{m,n \in C} [\alpha_n \neq \alpha_m] exp - \beta ||I_m - I_n||^2 \tag{9.7}$$

m, n represent the set of neighbouring pixels in C.

In this work, we propose to use scaling function in smoothness factor thereby emphasizing the importance of depth. This can be achieved by using weighted L2 norm

$$||I_m - I_n||_\tau^2 = \sqrt{\sum_{i=1}^{m,n} \tau_i(c(i,m) - c(i,n))^2} \tag{9.8}$$

Where the scaling weightage factor τ and c are as follows.

$$\left\{ \begin{array}{ll} \tau_0 & c_o = r \\ \tau_1 & c_1 = g \\ \tau_2 & c_1 = b \\ \tau_3 & c_1 = d \end{array} \right\} \tag{9.9}$$

The tuning parameter ψ is τ_3.

$$\tau_0 = \tau_1 = \tau_2 = (1 - \psi)/3; \tag{9.10}$$

If ψ values are set to zero and 1, then the model represents GrabCut and GrabCut(depth) respectively.

9.3 Experimental results

We have evaluated the proposed methodology using two publicly available standard datasets a) MSR (ballet dancer) [Fig. 9.1] and b) middlebury dataset (baby, midd1, tsukuba, teddy, art, moebius) [Fig. 9.8].

Figure 9.6: Ballet sequence dancer: (a) GrabCut and (b) GrabCutD (color and depth)

As shown in Fig. 9.6, arm and hand portion of dancer is not identified using GrabCut method (refer a), while our proposed method performed relatively better (refer b). Also, while considering the segmentation of man Fig. 9.7, bottom portion of the image has not been identified using GrabCut while ours is able to segment.

As shown in Fig. 9.8, our algorithm is able to segment the objects better in at most all dataset images. For the first example (Baby dataset), the GrabCut technique (colour) performs poorly since color of map in the background and that of baby significantly coincides. On the other hand, just using depth information, it fails to segment the dancer dataset since the view is close to camera. However, using both depth and colour with tuning parameter value ($\psi = 0.75$) , our algorithm performs better as expected. In order to illustrate the convergence per-

Figure 9.7: Ballet sequence man: (a) GrabCut and (b) GrabCutD (color and depth)
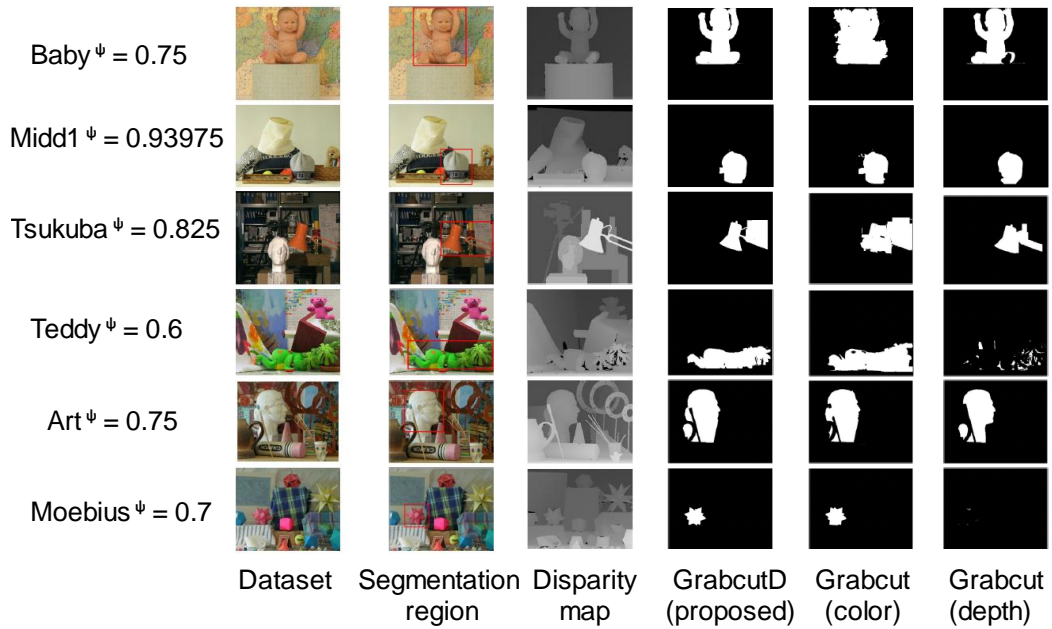
Figure 9.8: Middlebury dataset results comparison using GrabCut (color), proposed GrabCutD (color and depth) and GrabCut (depth)

formance, we show different results of varying the parameters with corresponding silhouette information (Fig. 9.9,9.10).
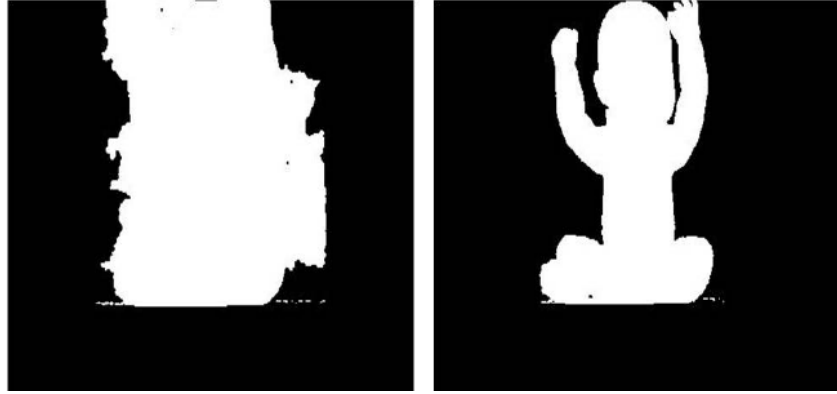
Figure 9.9: Baby ($\psi = 0.25$) and ($\psi = 0.75$).
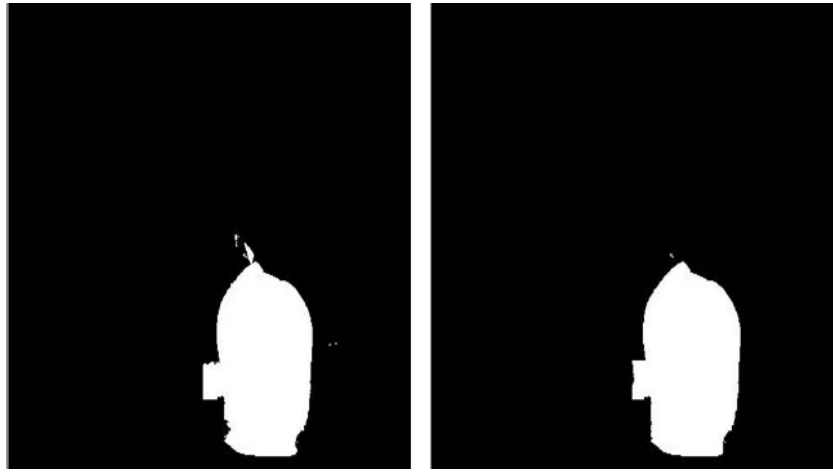
Figure 9.10: Midd1 ($\psi = 0.25$) and ($\psi = 0.9375$).

In the middll1 image, the right side curve portion of the hat is clearly seen in GrabCutD whereas in GrabCut, it is not the case. It can also be inferred that depth alone might not be sufficient for segmentation in some challenging datasets especially if there is no distinct depth information of the object of interest (refer teddy image). For this example, color information is also needed. ($\psi = 0.60$). Further, in moebius image, star like structure on the top portion is not having clear depth which might affect the resulting quality. However, bottom portion of

the image is segmented clearly using our method.
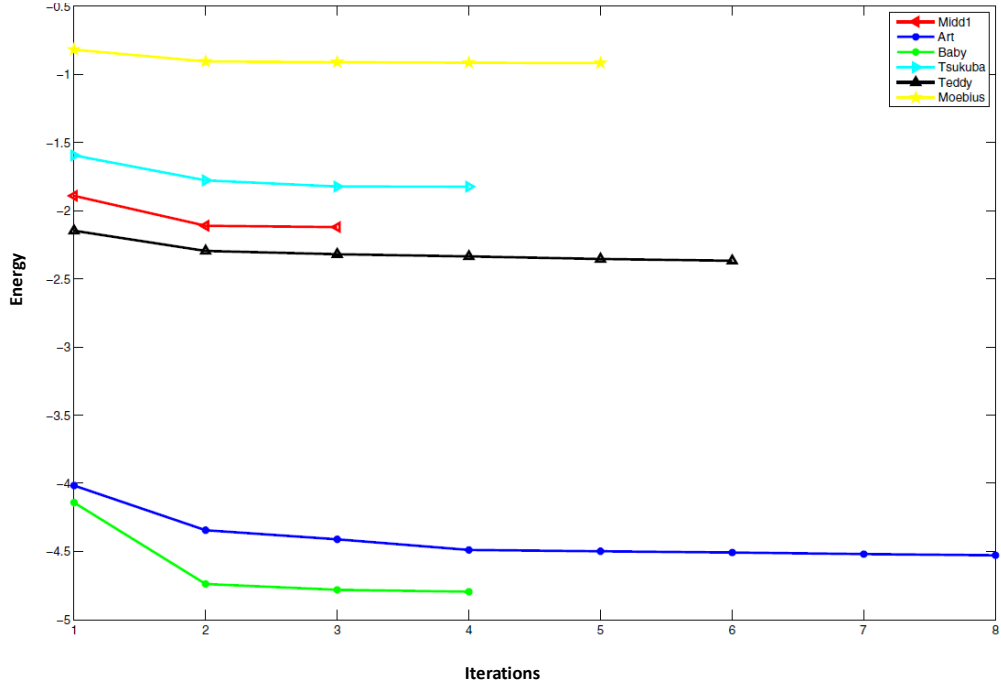
Figure 9.11: Energy vs. Iteration (GrabCutD).

The graph shown in Fig. 9.11 displays the energy convergence of different middlebury datasets using the proposed GrabCutD method. As shown in the graph, teddy took maximum of iterations $n = 8$ to converge to the minimum energy for segmentation while midd1 image converged in $n = 3$ iterations.

9.4 Conclusions

In conclusion, based on the above results, we proved that depth based information will improve the GrabCut technique. In this work, we have proposed a novel method extended with depth information. The limitations of the GrabCut are overcome by integrating depth information. If there are challenging situations like having erroneous depth, it might affect the resulting quality. In order to show usefulness of the approach, we have conducted experiments on different standard datasets. The efficiency of the proposed methodology is clearly justified. As fu-

ture work, we would further investigate to learn the tuning parameter adaptively for any given model based on color and disparity information.

Bibliography

[1] 3dmm dataset. http://mmv.eecs.qmul.ac.uk/~karthike/DiscrepancyDataset/. 97

[2] 3dmt dataset. http://mmv.eecs.qmul.ac.uk/~karthike/MeasurementDataset/. 61

[3] Adobe acrobat. http://help.adobe.com/en_US/Acrobat/9.0/Professional/index.html. 21

[4] Airbus. http://www.airbus.com/. 43, 44, 45, 61

[5] Airbusimagesource. http://free-photo-download.info/free-photo-downloads/aircraft-airbus_a350_picture.html. xi, 5

[6] Dmu. http://www.vc10.net/History/Individual/testspecimen.html. xi, 5, 45

[7] Faro. http://www.faro.com/. 19

[8] Geomagic. http://www.airbus.com/. 19, 26, 39

[9] Kubit. http://www.kubit-software.com/. 19

[10] Meshlab. http://meshlab.sourceforge.net/. 21, 152

[11] Missa more integrated systems safety assessment. http://www.missa-fp7.eu/. xi, 4, 6, 7, 17, 49

[12] Missa more integrated systems safety assessment. `http://insight3d.sourceforge.net/`. 36

[13] Msr3dvideodataset. `http://research.microsoft.com/enus/um/people/sbkang/3dvideodownload`. xiv, 161

[14] Saeaerospacestandard. `http://standards.sae.org/aerospace/safety/standards`. 7

[15] A. Abdelrahim, M. El-Melegy, and A. Farag. Realistic 3d reconstruction of the human teeth using shape from shading with shape priors. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 64 –69, june 2012. 32

[16] M. Attene, B. Falcidieno, and M. Spagnuolo. Hierarchical mesh segmentation based on fitting primitives. *Vis. Comput.*, 22(3):181–193, Mar. 2006. 27

[17] S. T. Barnard and W. B. Thompson. Disparity analysis of images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2(4):333–340, Apr. 1980. 161

[18] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359, June 2008. 19, 37, 57

[19] D. Borrmann, J. Elseberg, K. Lingemann, and A. Nüchter. A Data Structure for the 3D Hough Transform for Plane Detection. In *Proceedings of the 5th IFAC Symposium on Intelligent Autonomous Vehicles (IAV '10)*, Lecce, Italy, September 2010. 70, 156

[20] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222 –1239, nov 2001. 160

[21] M. Bozzano, A. Ferranti, K. Vaiapury, A. Aksay, X. Lin, E. Izquierdo, A. Cavallo, and C. Papadopoulos. Development description report for installation optimisation issue b and development description report for production audit. *MISSA Project D 3.81 Technical document*, 2011. 19, 28, 40, 43, 44, 45, 122

[22] A. Bradski. *Learning OpenCV, [Computer Vision with OpenCV Library ; software that sees].* O'Reilly Media, 1. ed. edition, 2008. Gary Bradski and Adrian Kaehler. 51, 58

[23] F. Bretar and N. Chehata. Terrain modeling from lidar range data in natural landscapes: A predictive and bayesian framework. *Geoscience and Remote Sensing, IEEE Transactions on*, 48(3):1568 –1578, march 2010. 77

[24] R. A. Brooks. Model-based three-dimensional interpretations of two-dimensional images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5:140–150, 1983. 31

[25] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. 110

[26] D. Chen, B. Chen, G. Mamic, C. Fookes, and S. Sridharan. Improved grab-cut segmentation via gmm optimisation. In *Proceedings of the 2008 Digital Image Computing: Techniques and Applications*, pages 39–45, Washington, DC, USA, 2008. IEEE Computer Society. 160

[27] H.-T. Chen, M.-C. Tien, Y.-W. Chen, W.-J. Tsai, and S.-Y. Lee. Physics-based ball tracking and 3d trajectory reconstruction with applications to shooting location estimation in basketball video. *J. Vis. Comun. Image Represent.*, 20(3):204–216, Apr. 2009. 33

[28] Y. Chen, T.-K. Kim, and R. Cipolla. Inferring 3d shapes and deformations from single views. In *Proceedings of the 11th European conference on computer vision conference on Computer vision: Part III*, ECCV'10, pages 300–313, Berlin, Heidelberg, 2010. Springer-Verlag. 38

[29] H. Chunmei, Z. Qiusheng, Y. Wentao, and Z. Jingtang. Research on 3d reconstruction of petrochemical enterprise pipeline and its collision detection based on 3d-scannning technology. In *Proceedings of the 2009 International Forum on Computer Science-Technology and Applications - Volume 01*, IFC-STA '09, pages 354–357, Washington, DC, USA, 2009. IEEE Computer Society. 37

[30] D. Corrigan, S. Robinson, and A. Kokaram. Video matting using motion extended grabcut. In *Visual Media Production (CVMP 2008), 5th European Conference on*, pages 1 –9, nov. 2008. 160

[31] N. Dinesh. A guide to stereo vision and 3d imaging. *NASA Tech Briefs.*, pages 12–16. 21, 22, 23

[32] R. Fabbri and B. Kimia. 3d curve sketch: Flexible curve-based stereo reconstruction and calibration. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1538 –1545, june 2010. 36

[33] O. Faugeras. *Three-dimensional computer vision: a geometric viewpoint.* MIT Press, Cambridge, MA, USA, 1993. 53

[34] D. Fidaleo and G. Medioni. Model-assisted 3d face reconstruction from video. In *Proceedings of the 3rd international conference on Analysis and modeling of faces and gestures*, AMFG'07, pages 124–138, Berlin, Heidelberg, 2007. Springer-Verlag. 38

[35] K. Forbes, A. Voigt, N. Bodika, and I. Group. Visual hulls from single uncalibrated snapshots using two planar mirrors, 2004. xi, 35, 160

[36] D. A. Forsyth and J. Ponce. *Computer Vision: A Modern Approach.* Prentice Hall Professional Technical Reference, 2002. 32

[37] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1362–1376, Aug. 2010. 28, 34, 36, 40, 92

[38] P. Georgel, P. Schroeder, S. Benhimane, S. Hinterstoisser, M. Appel, and N. Navab. An industrial augmented reality solution for discrepancy check. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, ISMAR '07, pages 1–4, Washington, DC, USA, 2007. IEEE Computer Society. 24, 25, 97

[39] M. Grzegorzek, S. Sav, E. Izquierdo, and N. E. O'Connor. Local wavelet features for statistical object classification and localisation. *IEEE Multimedia*, 17(1):56–66, January–March 2010. 30, 32

[40] J.-Y. Guillemaut, J. Kilner, and A. Hilton. Robust graph-cut scene segmentation and reconstruction for free-viewpoint video of complex dynamic scenes. In *ICCV'09*, pages 809–816, 2009. 160

[41] S. Han, W. Tao, D. Wang, X. Tai, and X. Wu. Image segmentation based on grabcut framework integrating multiscale nonlinear structure tensor. *Trans. Img. Proc.*, 18:2289–2302, October 2009. 160

[42] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 1106–, Washington, DC, USA, 1997. IEEE Computer Society. 51

[43] B. Heisele, G. Kim, and A. Meyer. Object recognition with 3d models. In *BMVC*, 2009. 29

[44] C. Y. Ip and W. C. Regli. A 3d object classifier for discriminating manufacturing processes. *Comput. Graph.*, 30(6):903–916, Dec. 2006. 22, 27, 105

[45] E. Izquierdo and M. Ghanbari. Key components for an advanced segmentation system. *IEEE Transactions on Multimedia*, 4(1):97–113, 2002. 160

[46] E. Izquierdo and S. Kruse. Image analysis for 3d modeling, rendering, and virtual view generation. *Computer Vision and Image Understanding*, 71(2):231–253, 1998. 33

[47] E. Izquierdo and J.-R. Ohm. Image-based rendering and 3D modeling: A complete framework. *Signal Processing: Image Communication*, 15(10):817–858, Aug. 2000. 33, 50, 51, 115

[48] S. Kahn, H. Wuest, D. Stricker, and D. W. Fellner. 3d discrepancy check via augmented reality. In *ISMAR*, pages 241–242, 2010. 24, 25

[49] E.-H. Kim, J. Hahn, H. Kim, and B. Lee. Profilometry without phase unwrapping using multi-frequency and four-step phase-shift sinusoidal fringe projection. *Opt. Express*, 17(10):7818–7830, May 2009. 23

[50] C. Li and Y. Zhou. 3d auto reconstruction for street elevation based on line and planar feature. In *The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, pages 460–466, 2010. 36

[51] Y. Li, J. Sun, C.-K. Tang, and H.-Y. Shum. Lazy snapping. *ACM Trans. Graph.*, 23(3):303–308, Aug. 2004. 160

[52] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, Nov. 2004. 19, 37, 57

[53] A. D. Marshall, R. R. Martin, and D. Hutber. Automatic inspection of mechanical parts using geometric models and laser range finder data. *Image Vision Comput.*, 9(6):385–405, 1991. 30

[54] T. J. Mckinley, M. M. McWaters, and V. K. Jain. 3d reconstruction from a stereo pair without the knowledge of intrinsic or extrinsic parameters. In *Proceedings of the Second International Workshop on Digital and Computational Video*, DCV '01, pages 148–, Washington, DC, USA, 2001. IEEE Computer Society. 33

[55] A. Mokhber, C. Achard, and M. Milgram. Recognition of human behavior by space-time silhouette characterization. *Pattern Recogn. Lett.*, 29(1):81–89, Jan. 2008. 160

[56] J. L. Mundy. Object recognition in the geometric era: A retrospective. In *Toward Category-Level Object Recognition'06*, pages 3–28, 2006. xi, 29, 30, 32

[57] S. Nagatomo, J. Hayashi, and S. Hata. Proposal of calibration error-tolerant 3d measurement method using stereo vision for robot vision. In *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, pages 3263 –3268, aug. 2009. 22

[58] L. Nan, L. Ni, L. Haitao, and Y. Chen. A study on 3d measurement technology for apparels accessories. *Networking and Distributed Computing, International Conference on*, 0:261–264, 2010. 21

[59] U. Park and A. K. Jain. 3d model-based face recognition in video. In *ICB'07*, pages 1085–1094, 2007. 32

[60] F. Perez Nava and J. Luke. Simultaneous estimation of super-resolved depth and all-in-focus images from a plenoptic camera. In *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video, 2009*, pages 1–4, 2009. 37

[61] W. A. Perkins. A model-based vision system for industrial parts. *IEEE Trans. Comput.*, 27(2):126–143, Feb. 1978. 30

[62] S. Prakash, R. Abhilash, and D. Sukhendu. SnakeCut : an Integrated Approach Based on Active Contour and GrabCut for Automatic Foreground Object Segmentation (2008), 2008-04-03 [Surya Prakash, R. Abhilash, Sukhendu Das]. *ELCVIA : Electronic Letters on Computer Vision and Image Analysis*, 6(3), 2007. 160

[63] B. C. R. Osada, T. Funkhouser and D. Dobkin. Shape distributions. *ACM Transactions on Graphics*, 21(4):807–832, 2002. 104

[64] T. Rabbani and F. van den Heuvel. 3d industrial reconstruction by fitting csg models to a combination of images and point clouds. In *XXth ISPRS Congress: Proceedings of Commission V*, pages 7–12, 2004. 47

[65] T. Rabbani and F. van den Heuvel. Efficient hough transform for automatic detection of cylinders in point clouds. In *Workshop Laser scanning 2005*, pages xx–yy, 2005. 27, 32

[66] E. Reinhard and E. A. Khan. Depth-of-field-based alpha-matte extraction. In *Proceedings of the 2nd symposium on Applied perception in graphics and visualization*, APGV '05, pages 95–102, New York, NY, USA, 2005. ACM. 161

[67] C. Rother, V. Kolmogorov, and A. Blake. GrabCut: interactive foreground extraction using iterated graph cuts. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 309–314, 2004. 160, 162

[68] A. Saxena, S. H. Chung, and A. Y. Ng. 3-d depth reconstruction from a single still image. *International Journal of Computer Vision (IJCV*, 76:2007, 2007. 38

[69] R. Schnabel, R. Wahl, and R. Klein. Efficient ransac for point-cloud shape detection. *Comput. Graph. Forum*, pages 214–226, 2007. 78

[70] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*, CVPR '06, pages 519–528, Washington, DC, USA, 2006. IEEE Computer Society. 34, 35, 36, 37

[71] E. S. Shijagurumayum and F.-C. Chang. Preliminary survey of multiview synthesis technology. In *Proceedings of the 2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, IIH-MSP '10, pages 163–166, Washington, DC, USA, 2010. IEEE Computer Society. 37

[72] J. Smisek, M. Jancosek, and T. Pajdla. 3d with kinect. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1154 –1160, nov. 2011. 115

[73] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph.*, 25(3):835–846, July 2006. 38

[74] N. Snavely, S. M. Seitz, and R. Szeliski. Modeling the world from internet photo collections. *Int. J. Comput. Vision*, 80(2):189–210, Nov. 2008. 38

[75] J. Sun, S. Kang, Z. Xu, X. Tang, and H. Shum. Flash cut: Foreground extraction with flash and no-flash image pairs. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1 –8, june 2007. 160

[76] G. Takahashi and R. Matsuoka. Accuracy of measurement using a pair of stereo images acquired by finepix real 3d w1 without controls. In *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume XXXVIII, 2010. 56

[77] F. Tarsha-kurdi and P. Grussenmeyer. Hough-transform and extended ransac algorithms for automatic detection of 3d building roof planes from lidar data. In *Part 3/W52*, 2007. 77, 78, 80, 81, 156, 157

[78] A. Torralba and A. Oliva. Depth estimation from image structure. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(9):1226–1238, Sept. 2002. 161

[79] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. 3(4):323–344, 1987. 51

[80] K. Vaiapury, A. Aksay, X. Lin, and E. Izquierdo. Model based 3d vision and analysis for production audit purposes. *Infocommunications Journal.*, 3(3):1–8, 2011. 121, 122, 124

[81] K. Vaiapury, A. Aksay, X. Lin, E. Izquierdo, and C. Papadopoulos. A new cost effective 3d measurement audit and model comparison system for verification tasks. *Multidimensional Systems and Signal Processing*, pages 1–47. 10.1007/s11045-012-0200-9. 121

[82] K. Vaiapury, A. Aksay, X. Lin, E. Izquierdo, and C. Papadopoulos. A vision based audit method and tool that compares a systems installation on a production aircraft to the original digital mock-up. *SAE Int. J. Aerosp.*, 4(2):880–892, 2011. 121, 122, 124

[83] K. Vaiapury, P. Atrey, M. Kankanhalli, and K. Ramakrishnan. Non-identical duplicate video detection using the sift method. In *Visual Information Engineering, 2006. VIE 2006. IET International Conference on*, pages 537–542, 2006. 37

[84] K. Vaiapury and E. Izquierdo. An o-fdp framework in 3d model based reconstruction. In *Proceedings of the 2010 12th International Asia-Pacific Web Conference*, APWEB '10, pages 424–429, Washington, DC, USA, 2010. IEEE Computer Society. 161, 163

[85] G. Vogiatzis and C. Hernández. Practical 3D Reconstruction Based on Photometric Stereo. In R. Cipolla, S. Battiato, and G. M. Farinella, editors,

Computer Vision, volume 285, chapter 12, pages 313–345. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. 34

[86] S. Webel, M. Becker, D. Stricker, and H. Wuest. Identifying differences between cad and physical mock-ups using ar. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, ISMAR '07, pages 1–2, Washington, DC, USA, 2007. IEEE Computer Society. 1

[87] Y. S. Z. Li and C. Wang. Real time complex 3d measurement. In *International Conference on Computer Modeling and Simulation)*, 2009. 23

[88] Z. Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11):1330 – 1334, nov 2000. 51

[89] F. Zhao. Machine recognition as representation and search. Technical report, Cambridge, MA, USA, 1989. 29

[90] J. Zhu, M. Liao, R. Yang, and Z. Pan. Joint depth and alpha matte optimization via fusion of stereo and time-of-flight sensor. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 453 –460, june 2009. 161

[91] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High-quality video view interpolation using a layered representation. *ACM Trans. Graph.*, 23(3):600–608, Aug. 2004. 162, 163

[92] M. Zuliani. Ransac for dummies. Nov. 2008. 75, 76