Electronic Thesis and Dissertation Repository

6-22-2012 12:00 AM

Consistent Density Scanning and Information Extraction From Point Clouds of Building Interiors

Kuldeep Kumar Sareen, *The University of Western Ontario*

Supervisor: Prof. George K. Knopf, *The University of Western Ontario*
A thesis submitted in partial fulfillment of the requirements for the Doctor of Philosophy degree in Mechanical and Materials Engineering
© Kuldeep Kumar Sareen 2012

Follow this and additional works at: https://ir.lib.uwo.ca/etd

Recommended Citation

Sareen, Kuldeep Kumar, "Consistent Density Scanning and Information Extraction From Point Clouds of Building Interiors" (2012). *Electronic Thesis and Dissertation Repository*. 608.
https://ir.lib.uwo.ca/etd/608

CONSISTENT DENSITY SCANNING AND INFORMATION EXTRACTION FROM POINT CLOUDS OF BUILDING INTERIORS

Spine title: Consistent Density Scanning and Information Extraction from Point Clouds of Building Interiors

Thesis Format: Monograph

by

Kuldeep Kumar <u>Sareen</u>

Graduate Program in Mechanical and Materials Engineering

A thesis submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

The School of Graduate and Postdoctoral Studies

Western University

London, Ontario, Canada

June, 2012

WESTERN UNIVERSITY
School of Graduate and Postdoctoral Studies

CERTIFICATE OF EXAMINATION

Supervisor

_____
Dr. George Karl Knopf

Co-Supervisor

_____
 Dr. Robert Canas

Supervisory Committee

_____

Examiners

_____
Dr. Ralph O. Buchal

_____
Dr. O. Remus Tutunea-Fatan

_____
Dr. Hanif M. Ladak

_____
Dr. Ling Guan

The thesis by

Kuldeep Kumar Sareen

entitled:

Consistent Density Scanning and Information Extraction from Point Clouds of Building Interiors

is accepted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Date_____ _____
Chair of the Thesis Examination Board

ii

ABSTRACT

Over the last decade, 3D range scanning systems have improved considerably enabling the designers to capture large and complex domains such as building interiors. The captured *point cloud* is processed to extract specific Building Information Models, where the main research challenge is to simultaneously handle huge and cohesive point clouds representing multiple objects, occluded features and vast geometric diversity. These domain characteristics increase the data complexities and thus make it difficult to extract accurate information models from the captured point clouds.

The research work presented in this thesis improves the information extraction pipeline with the development of novel algorithms for consistent density scanning and information extraction automation for building interiors. A restricted density-based, scan planning methodology computes the number of scans to cover large linear domains while ensuring desired data density and reducing rigorous post-processing of data sets.

The research work further develops effective algorithms to transform the captured data into information models in terms of domain features (layouts), meaningful data clusters (segmented data) and specific shape attributes (occluded boundaries) having better practical utility. Initially, a direct point-based simplification and layout extraction algorithm is presented that can handle the cohesive point clouds by adaptive simplification and an accurate layout extraction approach without generating an intermediate model.

Further, three information extraction algorithms are presented that transforms point clouds into meaningful clusters. The novelty of these algorithms lies in the fact that they work directly on point clouds by exploiting their inherent characteristic. First a rapid data clustering algorithm is presented to quickly identify objects in the scanned scene using a robust hue, saturation and value (H S V) color model for better scene understanding.

A hierarchical clustering algorithm is developed to handle the vast geometric diversity ranging from planar walls to complex freeform objects. The shape adaptive parameters help to segment planar as well as complex interiors whereas combining color and geometry based segmentation criterion improves clustering reliability and identifies

unique clusters from geometrically similar regions. Finally, a progressive scan line based, side-ratio constraint algorithm is presented to identify occluded boundary data points by investigating their spatial discontinuity.

ACKNOWLEDGEMENTS

To my Son, Arhaan, for giving me the strength to work through the tough phases in completing this work.

NOMENCLATURE

x y z	=	Spatial coordinates of the captured point
r g b	=	Color parameter of the captured point
b	=	Blue color parameter of the captured point
α	=	Horizontal scanning resolution
λ	=	Vertical scanning resolution
PAF	=	Planar alignment factor
$\angle_{H\ max}$	=	Maximum horizontal angular spans
$\angle_{V\ max}$	=	Maximum vertical angular spans
N	=	Number of scanned points in horizontal direction
M	=	Number of scanned points in vertical direction
D	=	Minimum scanner distance from the wall
d_n	=	Data density distance at n^{th} point
θ	=	Incident angle
L	=	Domain length
d_{alw}	=	Allowable data density distance
$n\alpha$	=	Restricted angular span
L_n	=	Consistent scanning range
ξ	=	Fraction factor for feature extraction
f	=	Fraction value for limited scanner distance (D)
θ_l	=	Restricted angular span
m_d	=	Multiple value for minimum data density distance
x_1	=	Minimum data density distance
X_n	=	Linear scanning range
O_l	=	Overlap distance between consecutive scans
n_{scans}	=	Number of scans to cover the range
d_{nc}	=	Measured captured data density
d_{nc4}	=	Measured captured data density with four neighboring points
d_{nc8}	=	Measured captured data density with eight neighboring points
L_n^{max}	=	Maximum scanner range
L_n^{min}	=	Minimum scanner range
$\hat{n}$	=	Simplification direction vector
N_p	=	Number of simplifying parallel planes

$\Psi_{threshold.}$	=	Average angular deviation threshold
t_v	=	Vicinity region thickness
H	=	Point cloud height
r	=	Reduction ratio
t	=	Slice thickness
Ψ_{avg}	=	Average angular deviation value
N_s	=	Point cloud set in each simplification slice
p_i	=	i^{th} point in the point cloud
q_j	=	j^{th} neighboring point around a point p_i
d_{ij}	=	Neighborhood distance vector (q_j-p_i)
$\Psi_{i,j}$	=	Angular deviation of distance vector ($d_{i,j}$) from normal vector $\hat{n}$
t_v^{max}	=	Maximum vicinity region thickness
μ	=	Division factor for maximum vicinity region thickness
p_p	=	Corresponding projection point for a point 'p' in space
m_p	=	Number of linear segments for intersection investigation
f_l	=	Focal length of the digital camera
O_c	=	Optical centre of shape capturing system
O_p	=	Perspective centre of shape capturing system
$x_a\,y_b$	=	Projected point coordinates in the optical plane
u, v	=	Parametric parameters in two orthogonal directions
h s v	=	Hue, Saturation and Value of a data point (p)
q	=	Number of nearest points for plane approximation
Δh	=	Hue deviation
S_r	=	Point data saturation parameter
FD_{Avg}	=	Fixed distance average of neighboring points
FDN	=	Number of points in the fixed distance neighborhood
β	=	Multiplication factor for computing FDNs
K_r	=	Number of nearest points for computing PAF
a_{rp}	=	Average angular residual value
d_f	=	Fixed neighborhood distance
AADD	=	Adaptive average density distance
DistF	=	Multiplication factor used with AADD to compute d_f
$n_x\,n_y\,n_z$	=	Direction normal vector at spatial point p(x y z)

θ_{avg}^i = Average angular deviation of p_i from normal vector $\hat{n}$

$\theta_{avg}^{threshold}$ = Permissible threshold of angular deviation

h^{thres} = Permissible threshold of hue deviation

$\lambda_{normVector}^{thres}$ = Permissible threshold of normal vector

$SR_r^{i,j}$ = Side ratio at the $(i,j)^{th}$ point in M x N points grid along rows

$SR_c^{i,j}$ = Side ratio at the $(i,j)^{th}$ point in M x N points grid along columns

$SR_l^{i,j}$ = Side ratio at the $(i,j)^{th}$ point in M x N points grid in lateral directions

$SR_{(i,j)}$ = Limiting value of the side ratios at $(i,j)^{th}$ point

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1   INTRODUCTION

1.1   Background and Motivation

The digitization of existing objects and complex 3D environments has emerged as an active research field in virtual reality having diverse applications ranging from anatomical reconstruction, cartography, cultural artefact modeling, digital archaeology, infrastructural renewal, and 'as built-as design' evaluations. This reverse engineering process can be divided into two main stages (1) acquisition of spatial geometry and (2) extraction of vital information in terms of key features or complete digital models. The effectiveness of the overall process can be improved by identifying, eliminating or reducing the inaccuracies of the shape capturing process in the first stage and then developing customized post-processing and shape modeling tools for extracting specific geometric information in the second stage.

Three dimensional (3D) range scanning technology is an efficient and fast method of capturing accurate spatial geometry as a set of 3D data points from the visible surfaces of the scanned scene called a *'point cloud'*. These range scanning systems have been significantly improved over the last decade and can now capture surface geometries from surfaces lying few hundred meters away, in a single scan position (Callieri, Cignoni, Dellepiane *et al.*, 2009). These scanners can be used to capture geometries from very large scenes such as industrial facilities, historical sites and occupied building interiors. However, the capturing accuracy of these scanners is restricted by the traditional scanning errors arising from sensor noise and varying surface characteristics. In addition to these errors, other domain specific inaccuracies also transpire in the *point cloud* of large building interiors due to restricted scanner accessibility, presence of multiple objects, irregular surfaces geometries, occluded features and overlapping regions with varying surface complexities. Such a scanned scene generates a cohesive point cloud with varying data density, missing surface geometries and multiple region discontinuities. Although present day scanners are quite efficient in minimizing the traditional scanning errors arising from scanner mechanisms (Alonso, Rubio, Martin *et al.*, 2011; Li and Mitchell, 1995; Lichti, 2007; Zhuang and Roth, 1995), surface properties (Bucksch, Lindenbergh and Ree, 2007; Křemen, Koska and Pospíši, 2006) and ambient conditions (Lichti, Gordon and Tipdecho, 2005; Pfeifer, Dorninger, Haring *et al.*,

2007), handling domain specific geometries and complex feature characteristics of large scan scenes is still an open research field.

The geometric information extraction process on the other hand, is highly application dependent. Most published research has mainly focused on reverse engineering of shapes (Curless, 2000; Hoppe, DeRose, Duchamp *et al.*, 1992; Várady, Martin and Cox, 1997), which often rely on adaptive post-processing tools (Breunig, Kriegel, Ng *et al.*, 2000; Davis, Marschner, M.Garr *et al.*, 2002; Knorr, Ng and Tucakov, 2000; Liepa, 2003; Richter, 2009; Sotoodeh, 2006; Turk and Levoy, 1994; Weyrich, Pauly, Heinzle *et al.*, 2004) for compensating the scanning inaccuracies while extracting approximate geometric features, developing fairly accurate model(s) and generating representative virtual environments. Most of these existing techniques (Curless, 2000; Várady, Martin and Cox, 1997) are demonstrated on single, isolated objects, where small inaccuracies are compensated through local approximation. Few other approaches (Alharthy and Berthel, 2004; Elberink and Vosselman, 2009; Pfeifer, Dorninger, Haring *et al.*, 2007; Sampath and Shan, 2007) involving large data sets are developed for building exteriors and structured *point clouds* captured by airborne scanners, where the scanned geometry is approximated through simplified geometries. Further, these simplified approaches normally focus on large planar structures in the scanned scene alone and omits intricate geometric details of other, relatively small objects.

However, accurate information extraction from point clouds of building interiors is quite challenging task as it has to compensate unstructured point data having varying data density arising due to the stationary position of the scanner and domain's dimensional characteristics. Further, dissimilar surfaces ranging from planar walls to complex objects lying on the floor have to be differentiated for desired feature extraction. The *point cloud* representing multiple, and, comparable sized objects in the scanned scene, have to be processed, so that individual tools for layout extraction, data decimation, object identification, feature extraction, pattern recognitions, and virtual modeling can be developed.

The biggest bottleneck in extracting desired information is the characteristics of the *point cloud*, which is an uninformative set of spatial data points representing the multiple objects and discontinuous surfaces in the scanned scene. This shape capturing process cannot differentiate between the spatial complexity, feature geometry, location and orientation of objects. Thus, the data is often processed using additional geometric parameters in order to

get the desired information. e.g. geometric modeling through polygonal surfaces (Dey, 2007; Edelsbrunner and Mücke, 1994; Hoppe, DeRose, Duchamp *et al.*, 1992), visualizing point clouds in graphics applications (Adamson and Alexa, 2003; Alexa, Behr, Cohen-Or *et al.*, 2003; Fabio, 2003), data simplification for fast data processing (Alharthy and Bethel, 2002; Araújo and Jorge, 2005; Nooruddin and Turk, 2003; Sampath and Shan, 2007) and segmentation and simplification for improving computational efficiency for individual surface extractions (Boulaassal, Landes, Grussenmeyer *et al.*, 2007; Brenner, 2005; Pu and Vosselman, 2006), modeling (Kim and Shan, 2011; Zhou and Neumann, 2008) and virtual scene creation (Lerma, Navarro, Cabrelles *et al.*, 2011; Lu, Shi and Zhu, 2008).

The effectiveness of all these post-processes not only relies on the accuracy of the initially captured data, but also on the post-processing steps and their suitability for a specific application e.g. hole filling strategies of single isolated objects cannot be extended to building interiors as the multiple regions comprising the hole edges might belong to different objects. Further, a geometric model from such a cumulative point cloud is not of much use except for pure visual scene understanding. Hence, customized tools need to be devised to compensate for domain specific scanning errors and identify and extract application specific features such as building layouts, key surface features, and segmented objects. Later, modeling approaches can be used to generate individual models from these features and segmented data sets representing single isolated objects, if required.

1.2   Problem Formulation

The work of this thesis focuses on the domain-specific limitations of 3D range scanning and subsequent information extraction processes involving occupied building interiors. As mentioned above, it is essential to identify and reduce the domain specific scanning inconsistencies and develop specific, information extraction tools, instead of generating unified geometric models from cohesive point clouds. Further, as the size of the captured data is normally huge, it is also desirable that these methods can be directly implemented on points, so that the desired information can be extracted without an explicit reconstruction of an intermediate surface model.
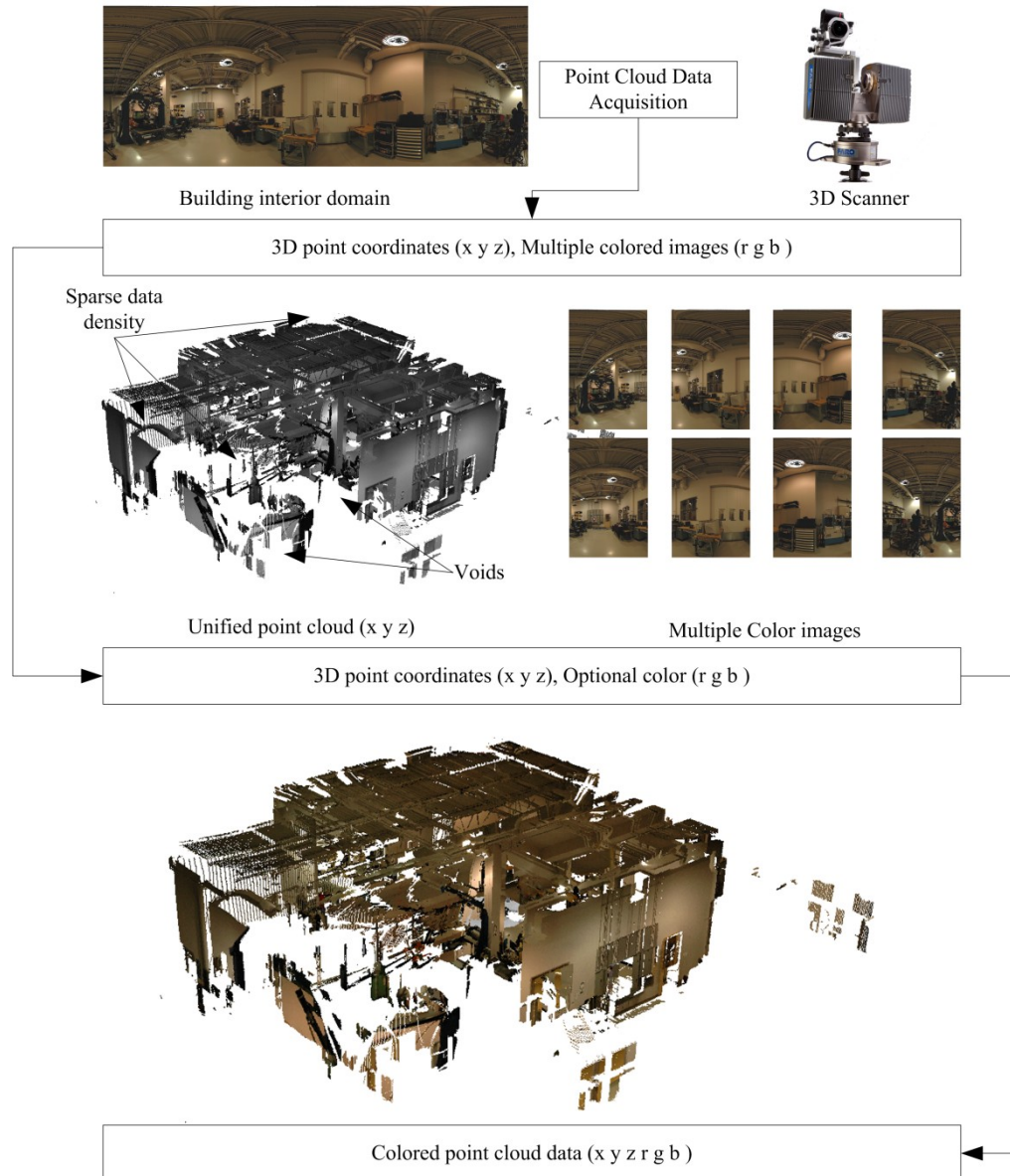
Figure 1.1: Basic scanning process of a building interior.

Figure 1.1 shows the basic process of 3D scanning of building interiors. A scanner captures the cohesive data sets from multiple objects in the scanned scene and generates unified point cloud. The geometry is primarily captured in terms of 3D point coordinates (x y z). In addition to the spatial coordinates, color (r g b) can also be captured using a scanner mounted digital camera, that can capture multiple synchronized colored images, from which a panoramic image can be created depicting the unified colored scene. Finally the color

parameters (r g b) from the mapped pixels from this panoramic image are merged with the corresponding point coordinates (x y z) resulting in colored point cloud (x y z r g b).

A uniform density data set can be captured if the distance between the scanner and the domain's surfaces remains constant. A moving scanner can maintain this condition to some extent; however, their availability and subsequent registration inaccuracies limit their use. Recently, one such scanner has been proposed (Chen, Kua, Shum *et al.*, 2010; Liu, Carlberg, Chen *et al.*, 2010), which is still in development phase. Thus, a stationary scanner is mainly used to capture large domains and generate point clouds with data inconsistencies. This approach requires effective scanning methodologies and robust data processing algorithms to compensate the data inconsistencies captured in the point clouds.

There are three distinct scanning inconsistencies that are observed in the data set representing building interiors: (1) *non-uniform data density*, (2) *region voids* and (3) *data cohesiveness* arising from stationary scanning of dimensionally incoherent domain having multiple objects and overlapping geometries.

The non-uniformity in the captured data arises due to the varying domain distance from the stationary scanner having fixed angular resolution. Here, the *data density* can be defined in terms of number of captured points per unit area. However, this definition largely depends on the surface orientations, which is often not known at the scanning stage. Thus, the *data density* is implicitly defined as the local distance between neighboring data points. The density can be represented as the inverse of Euclidean distance between two neighboring data points. As a stationary scanner captures the surface geometries using a fixed angular scanning resolution, a uniform data-density point cloud can only be obtained from a domain, whose surfaces lie at a comparable distance from the scanner. However, that is often not the case, and the captured data exhibits large variation in the data density based on the surface distance from the scanner. Such inconsistent data densities are especially observed in domains having large dimensional disparity (large length/width ratio) such as narrow pathways, long hallways and interiors with limited domain accessibility as shown in Figure 1.2, where the far-away regions are often captured with very sparse data or low data density as shown in the scanned data of a long hall-way in Figure 1.3.
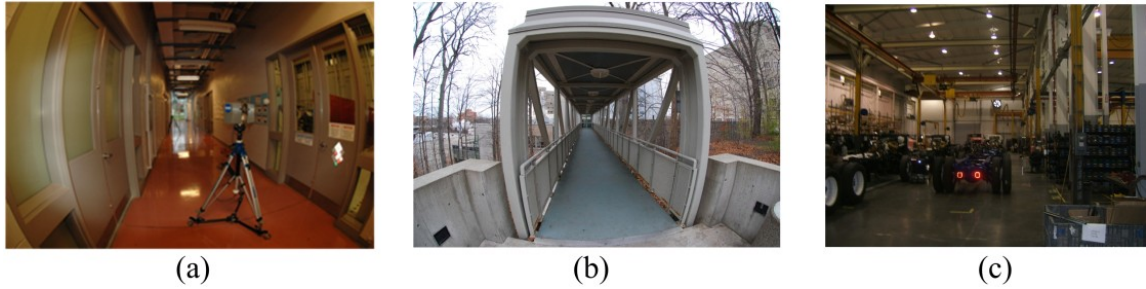
Figure 1.2: Narrow interior pathways and restricted accessibility scanning scenarios (a) Interior corridors (b) Slender over-bridge architecture, and, (c) Interior domains with restricted accessibility.
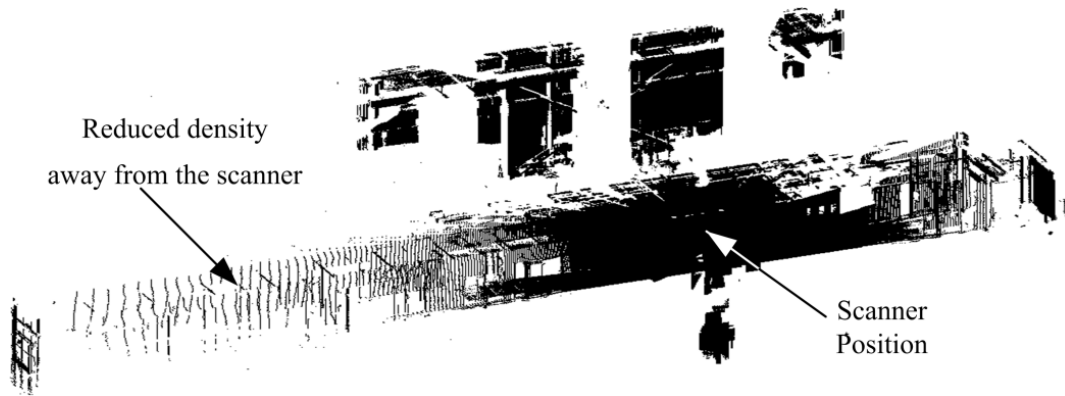
Figure 1.3: Captured spatial data from an interior building pathway.

The region voids are generated due to the presence of multiple objects and overlapping geometries that partially restricts the surface accessibility in the scanned scene. The scanning process fails to capture these occluded regions and generates corresponding data voids, which are often surrounded by discontinuous boundary of multiple objects. Such voids cannot be approximated from the surrounding data set as in traditional hole filling algorithms (Barequet and Kumar, 1997; Borodin, Novotni and Klein, 2002; Wu, Wang and Han, 2008) of isolated objects. Further, it becomes even more difficult to differentiate these voids if the occluded regions forming these voids are lying very close to each other and have similar surface geometric properties.

In additions to these voids, the unified scanning process also combines the point clouds from multiple objects into a single, cohesive point cloud. So, specific information extraction pertaining to prominent geometric features, individual objects and diversified surfaces becomes a complex task as it requires the identification of a section of the point cloud

representing the desired feature to be extracted. This cohesive point cloud represents geometrically diverse surfaces ranging from large planar regions to intricate interior objects and single coherent approach cannot handle such shape diversity and geometric intricacy. Further, there are instances, where the adjacent scanned regions have similar geometries, but they belong to different objects. In this case, additional geometric parameters, computed from local data are also not able to form a reliable differentiating parameter to extract individual features.

An effective scan planning methodology can play a significant role towards minimizing these scanning issues. However, in the absence of any specific guidelines, the shape capturing process largely depends on a scan planner's skill alone.

Even highly accurately captured data require effective information extraction algorithms to covert these uninformative point clouds into a usable format. However, the first step of any information extraction process is defining the term *information*, which varies considerably based on the applications in which the point cloud is to be used. The *information* may be defined as specific feature geometries, or key dimensional parameters. It may be defined as primitive shape approximation for simple shape representation, decimated data sets for improving computational efficiency, segmented data sets of individual objects or surfaces for creating geometric models or virtual scenes. Thus, individual *information extraction* processes are to be developed that can process the cohesive and uninformative data accordingly for the desired outcome.

For many interior scanning applications such as layout extraction, object identification, feature extraction and pattern recognitions, a cumulative model of the cohesive point cloud is not helpful except in pure graphic applications (Adamson and Alexa, 2003; Kua, Corso and Zakhor, 2012; Polleyfeys, Koch, Vergauwen *et al.*, 1999), where the missing regions are very roughly filled. Some other applications (Budroni and Böhm, 2009; Kua, Corso and Zakhor, 2012; Shih and Hu, 2007) extract simplified models through simple planar fitting and piecewise merging of simplified surfaces. Such models are good for simple representation of scanned scenes and are quite effective for visualizing and communicating the captured data; however, their potential in extracting individual geometric properties is far less formulaic, which limits their usability in vast variety of applications involving complex objects. As a complete geometric model is not required most of the times, it is essential to

generate specific information extraction tools that can convert the captured data into useful features.

1.3   Research Objectives and Scope

The objective of this thesis is to develop a framework for reverse engineering of building interior environments, where systematic scan planning and subsequent information extraction methodologies are developed. This framework will minimize the domain-specific scanning errors through effective scanning methodology and enable the designer to extract specific geometric information through the proposed tools. Further, as the captured data from large building interiors is often very large, it is desirable that the proposed post-processing and information extraction tools are robust and are applicable directly on the point clouds. In general, this thesis focuses on the two important stages of the overall reverse engineering process of building interiors: shape capture and information extraction.

The first stage is dedicated to the development of a systematic scanning methodology that encompasses the domain's dimensional characteristics and the restricted scanner's accessibility and generates a *consistent density* point cloud. The proposed scan planning methodology provides a mathematical tool that defines the limited domain that can be scanned with desired data density using a stationary scanner. Thus, the number of scans required to cover the whole domain can be computed and the desired data density condition can be maintained.

The second stage extracts specific information from the captured data. The specific requirements of the reverse engineering process involving building interiors are analyzed and the desirable features are identified. This study focuses on few specific features namely layout extraction and data segmentation and develops corresponding feature extraction tools to make the point cloud informative and usable.

The layout extraction approach exploits the planarity of the data sets representing relatively flat regions such as floor, ceiling and walls using a *planar alignment factor* (PAF). This approach achieved data reduction by omitting the interior point clouds and uses only the planar (wall, floor, or ceiling) point cloud that defines the building blueprint. The approach is extended to formulate a data segmentation algorithm, where PAF is used to subdivide the cohesive point clouds into its individual segments using two hierarchical steps and simple

(planar) and intricate (interior) data clusters are extracted sequentially. The proposed hierarchal segmentation approach exploits the spatial conformity, local surface geometry and color information to form a reliable segmentation strategy. These proposed approaches work directly on point clouds and no explicit surface reconstruction is involved. A side-ratio constraint algorithm is also presented to identify discontinuous boundaries.

The proposed work will expand the existing literature with a consistent density scan planning methodology and three information extraction tools. The resultant dense point clouds can be directly used for visualization and accurate feature extraction. The direct point-based, layout extraction methodology can be used to extract blueprints, area and volume calculations, and path planning. Further, the individual segmented data obtained through the proposed color-assisted, geometry driven methodology can be used for various applications such as pattern recognition, object identification, floor layout planning, individual geometric modeling and virtual reality environment reconstruction. Finally, the occluded boundaries can be used to refine extracted data clusters and improve scene understanding.

1.4   Outline of the Thesis

The literature review on 3D range scanning of building interiors, scanning inconsistencies, scan planning, layout extraction and data segmentations approaches is discussed in Chapter 2. This chapter also summarizes the common information extraction approaches involving large point clouds data sets and it differentiates them from building interiors applications. It presents the objectives of range scanning of building interiors along with specific requirements and the reason for choosing the specific features, for which the methodologies and tools are developed in this thesis.

Chapter 3 introduces a scan planning methodology for a typical building interior environment. The proposed approach establishes the need of uniform density point clouds and defines parameters for defining the data density. It discusses the effect of domain's dimensional parameters and the scanner's position on the captured data density and devises a robust scan planning methodology for generating consistent density point clouds. It also enlists a set of guidelines for scanning an interior building such that the occluded regions are minimised. The efficacy of the proposed approach is presented on diverse data sets representing long pathways, narrow exterior facets and occupied building interiors.

The following chapters exhibit three information extraction algorithms. The layout extraction algorithm, data segmentation algorithm and discontinuous boundary detection algorithm are presented in chapters 4, 5 and 6, respectively. All of these algorithms do not require the explicit reconstruction of intermediate surfaces and instead uses inherent characteristics of the point cloud such as spatial coherence, locally computed geometric parameters and color consistency evaluation.

The layout extraction algorithm (chapter 4) identifies planar regions defining the floor plan by retaining the corresponding points using a planar alignment factor. The process omits the complex interior point cloud and the sliced wall data set is used to generate section layouts. Subsequently, the inherent color characteristics are exploited along with the planar alignment factor, spatial coherences and locally computed adaptive geometries to segment the cohesive point cloud using a color based and subsequently shape based hierarchical approach (chapter 5). The shape based hierarchy manages to effectively segment the point cloud from complex as well as simple planar regions. Lastly, a discontinuous boundary detection algorithm (chapter 6) is presented that helps in identifying the occluded region boundaries in the point cloud representing multiple overlapping objects.

Chapter 7 takes a closer look at the proposed methodologies and demonstrated algorithms in terms of the probable applications. It demonstrates the usability of the resultant data in applications such as visualization for virtual reality, building conformance verification for as built as design, floor layout extraction for surveying, geometric parameter computation, navigation path planning and data segmentation for pattern recognition, object identification, and individual geometric modeling.

Chapter 8 summarizes the scan planning methodology and information extraction algorithms. It also provides the limitations of the proposed system, and recommendations for improvement and future work.

CHAPTER 2   LITERATURE SURVEY

2.1   Introduction

During scanning of building interiors, the quality of the captured data largely depends on adaptability of the *3D scanning sub-system* to the dimensional, positional and topological characteristics of the domain and its objects. A scanning *sub-system* controls the scanning parameters according to the domain characteristics in such a way that accurate data are captured. The state of this captured data ascertains the level of information that has been captured and can be extracted in subsequent data interpretation stages. Thus, at this stage, it is essential to optimize the *3D scanning sub-system* by removing, controlling and minimizing scanning inconsistencies as much as possible so that the captured data are the true representation of the scanned domain. Subsequently, the *information extraction* sub-system uses data interpretation tools that transform the captured point cloud into desired geometric and spatial features. The development of these domain-specific data interpretation tools requires deep understanding of the domain characteristics in terms of its explicit needs, key features and subsequent applicability of the extracted information.

This chapter discusses the related research work involved in various aspects of a these two *sub-systems* of the reverse engineering process involving building interiors. To better understand the idea behind the development of proposed techniques and specific contribution of this thesis, it is essential to review previous techniques used and their relative merits and demerits. Much of the work in range scanning and shape reconstruction process has been demonstrated on single, isolated objects or exterior building facets having no or manageable surface occlusions. Very few methods have used interior buildings as application domains and have limited their focus on manual scan planning, data simplification, shape approximation and scene visualization of the unified captured data. Most of these techniques rely on the development of accurate post-processing tools to get rid of the scanning limitation, instead of optimizing the scanning parameters. Interestingly, processed data for one application often makes it useless for another, thus, the post-processing must focus on data and information retention while extracting useful information for a given application.

Finally, most of these techniques use an intermediate surface model and are not directly applicable on point clouds, which is much better approach for large point clouds due to their large size.

2.2   Shape Acquisition

The spatial geometry may be acquired by a variety of techniques based on the capturing accuracy, scanning time and capturing mechanism of the scanner and the geometric characteristics of the domains. A detailed shape acquisition pipeline is compiled by Bernardini and Rusmeier (Bernardini and Rushmeier, 2002). Complex 3D shapes are commonly captured using *contact* or *non-contact* type acquisition systems. A *contact-type* system (e.g. a coordinate measuring machine (CMM)) uses a contact probe to capture surface details with high accuracy (Li and Gu, 2004). However, the size restrictions of the acquisition hardware, necessary physical contact of the probe with the scanning objects and significant scanning time limit its applicability to the small objects only.

The *non-contact* shape acquisition systems (Li and Gu, 2004; Várady, Martin and Cox, 1997) on the other hand, can rapidly capture accurate surface details of very complex objects due to its fast speed. Moreover, no physical contact with the scanning object makes it especially suitable for historical artefacts and delicate objects. This technique emits some sort of light or radiation and detects its reflection from the surfaces in the scanned scene to capture their shape in terms of a set of spatial data points called a *point cloud*.

Triangulation based scanners shine a laser on the subject and capture its relative position on the object surface in the camera's field of view (Mayer, 1999). The actual position of the surface point in space is determined by triangulating the position of the emitter, camera, and the laser's position in camera's field of view. However, the essential condition is that the surface must be visible in the camera's point of view at all times that requires the laser and camera to be quite close the object surface, making it impractical to scan large objects. Such triangulation-based range finders can scan isolated objects within the limited range and, are not suitable for scanning large structures and complex scenes.

White light scanning methods (Ailing, Chunhui, Zhuangde *et al.*, 2008; Fleischer, Windecker and Tiziani, 2001) also work on triangulation principle but use white light fringes projected on the object being scanned. These fringes form contours on the object, which are

captured by two cameras from different viewing directions. This type of scanning method requires placing the object in converging field of view of the two cameras, which restricts the size of the object that can be scanned. Thus, large structures such as building interiors can't be captured with this scanner. Their suitability for large structures is also limited due to their lengthy setup time, controlled lighting requirements and structured measurement environments.

Time-of-flight based range scanners on the other hand, can capture geometries by timing the round-trip of a light pulse and thus can captures far-away geometries up to few hundred meters. The accuracy of these captured data is governed by the precision with which the time can be recorded. Other large scale laser scanners measure the surface position by comparing the phase shifts between the reference and the reflected signals of the modulated light. Modern day time-of-flight scanners can compensate for data distortion due to small amounts of vibrations (Blais, Picard and Godin, 2004) and are capable of capturing 10,000~100,000 points every second using system of rotating mirrors. Thus, very large domains can be captured within seconds or minutes based on the required resolution. The fast acquisition speed has helped the 3D scanning process to expand its application to large sites for reverse engineering, infrastructure renewal, as-built drawing generation of large structures, industrial plants, monuments, documentation of historical and cultural sites, site modeling, feature extraction, quantity surveying, freeway redesigning, creating geographic information system (GIS) maps, and virtual reality modeling etc.

Lately, high quality digital photogrammetry (Guarnieri, Vettore and Remondino, 2004) has also emerged as a viable option to generate 3D models from multiple digital images captured from different directions. It has been widely demonstrated on satellite, aerial and terrestrial images for applications involving cultural heritage (El-Hakim, Gonzo, Voltolini *et al.*, 2007; Gruen, Remondino and Zhang, 2004), urban planning and forensic investigations (Gonzalez-Aguilera and Gomez-Lahoz, 2009). In all these applications, visualization is more important than geometric accuracy. Photogrammetry is an effective option in capturing large 3D domain if multiple images can be taken from far-off distance covering the whole domain in each shot for effective merging, which can be easily done for building exteriors (Boehler and Marbs, 2004). Alternatively, it can be used to capture specific objects such as statues and small intricate shapes through multiple images taken from converging directions. In case of building interiors, the approach can work for generating approximate visual models, but will

require a large number of images which is limited by time or location constraints. Moreover, the process of converting 2D images to 3D data sets requires typical photogrammetric pipelines based on sensor calibration, feature characteristics, image orientation and surface measurement techniques and it does not straightaway generate 3D data set unlike time-of-flight and phase shift scanners.

Several other methods can also extract 3D shapes through extraction of iso-surfaces from CAT scans (Rocchini, Cignoni, Montani *et al.*, 1999; Sun, Tian, Wu *et al.*, 2010), shape carving of video streams (Polleyfeys, Koch, Vergauwen *et al.*, 1999) and shape extractions from passive silhouettes (Zheng, 1994), however, their capturing accuracies and modeling approaches limits their applicability to large domains such as building interiors.

In conclusion, the non-contact based range scanners working on the time-of-flight or phase-shift principles (Callieri, Cignoni, Dellepiane *et al.*, 2009; Schuon, Theobalt, Davis *et al.*, 2008) exhibit fast shape acquisition, higher geometric precision and direct generation of 3D point clouds and hence, are commonly used in a typical shape acquisition pipeline of building interiors.

This diverse spectrum of applications uses the captured data in their own unique way according to their explicit set of requirements. Unfortunately, the immense point clouds acquired by these non-contact range scanners include random scanning errors arising from sensor accuracy and the object's surface properties and other systematic scanning errors, originating from domain-specific parameters (domain size, presence of multiple objects, relative location of objects and scanner). The random errors introduce noise, erroneous outliers, missing regions, and sampling inconsistencies in the captured data sets. These errors can be minimized with improved hardware, but small amount of such errors are ubiquitous in the captured data. Additionally, systematic errors are responsible for generating inconsistent data density, large data voids and occluded boundaries due to domain-specific characteristics such as multiple overlapping objects and limited scanner positions.

These errors restrict the designer's ability to extract useful information from the *point cloud*. Thus, it is essential to anticipate these systematic errors and to understand the domain-specific requirements so that the scanning process is planned to minimize the scanning inconsistencies and the captured data can facilitate accurate information extraction from *point clouds* and not through approximation.

2.3   Scanning Inconsistencies

In the context of interior building scanning applications, a stationary scanner is often used that projects a laser beam onto an angled rotating mirror, which reflects the beam to the scanned scene. The whole scanning unit also rotates around the vertical axis and thus captures everything in its whole 360° in its horizontal plane and about 320° in the vertical plane due to the position of the scanner. A stationary scanner often captures the spatial geometry using a horizontal scanning resolution (α), which is fixed for the whole horizontal domain. Thus, as the region is located away from the scanner, the distance between two consecutive points reduces and only sparse data density can be captured in this region. This condition is highly prevalent in interior domains having large dimensional disparity. The accuracy is lost due to two primary reasons (1) varying density that decreases as the distance of the surfaces from the scanner increases and (2) increased span of the missing data due to occluded objects at large angular obliquity. e.g. in Figure 2.1, a stationary scanner is shown that captures a *point cloud* from multiple objects in the scanned scene, as can be seen, with the fixed horizontal scanning resolution (α), the spacing between the captured data points increases away from the scanner. Further, the size of missing wall geometries (A' and B') also increases for the same sized overlapping objects (A and B), lying at varying distances from the scanner.
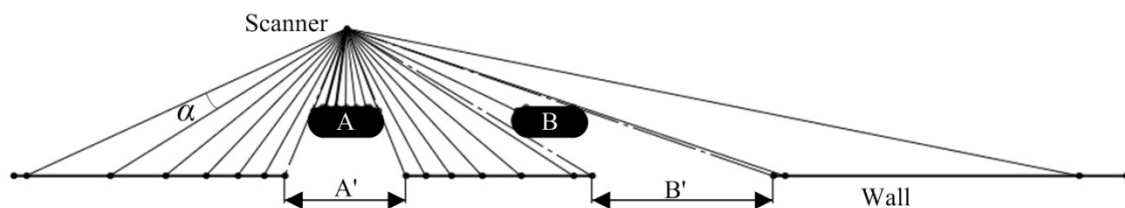
Figure 2.1:   Scanning scenario of multiple objects with a single stationary scanner (top view).

Such scanning inconsistencies are often dealt with in the post-processing stage, where the erroneous outliers are eliminated, missing data is approximated and inconsistent density is interpolated as compiled in the following subsections.

2.3.1 Varying data density

A density variation in the data set is the result of erroneous *outliers* or *under-sampled data* due to the scanning mechanism of a stationary scanner. An *outlier* is defined as an "*observation that deviates so much from other observations so as to arouse suspicion that it was generated by a different mechanism*" (Hawkins, 1980). It often represents false data point in Cartesian space due to overlapped signal from multiple surfaces. In case of large, occupied building interiors with multiple objects in the scanned scenes, *outliers* are often recorded along the edges of occluded objects along the scanning rays (Kua, Corso and Zakhor, 2012) due to partially reflected signals from the foreground and background objects. The *under-sampling* on the other hand is a systematic error of a stationary scanner, due to which the far-away surfaces are captured with lower data density and can be controlled with effective scan planning.

Outliers are generally identified and removed during post processing stage using distance (Knorr, Ng and Tucakov, 2000) or density (Breunig, Kriegel, Ng *et al.*, 2000; Sotoodeh, 2006) characteristics of the neighboring data points or are extracted as by-product of data clustering process (Jain, Murty and Flynn, 1999; Rousseeuw and Leroy, 1987; Sotoodeh, 2007).

Knorr (Knorr, Ng and Tucakov, 2000) used a distance-based evaluation approach to identify a point as an outlier if at least a fraction '*b*' of the data set '*P*' is further away than a pre-defined distance '*r*'. This method is dependent upon the selection of these parameters and demands higher computational cost ($O(n^3)$) for large point clouds. The density based approaches (Breunig, Kriegel, Ng *et al.*, 2000; Sotoodeh, 2006) on the other hand, identify outliers based on user defined *MinPts* around the point. A point with a drastically different value of *MinPts* than its surroundings is more likely to be an outlier. However, the selection of *MinPts* is a complex task for point cloud data set with vastly varying data densities, where a real surface point may be identified as outlier due to the sparse data density in that region. Alternatively, the clustering approaches (Filin, 2002; Jain, Murty and Flynn, 1999; Rousseeuw and Leroy, 1987; Sotoodeh, 2007) use segmentation to group clusters having similar characteristics, and identify *outliers* as points that are not part of any cluster. A. K. Jain has recently compiled an extensive report (Jain, 2010) an overview of clustering, discussed major challenges of designing clustering algorithms and enlisted emerging research

trends in this field including semi-supervised clustering, ensemble clustering, simultaneous feature selection clustering and large scale data clustering.

However, the performance of all these post-processed outlier detection methodologies is largely dependent on the initial characteristics of the scanned data and the local variation of point density affects the accuracy of the extracted outliers. Thus, an effective scanning methodology needs to be adopted that generates an initial point set with consistent data density, which minimizes the need of subsequent post-processing strategies or increase their reliability.

In contrast, under-sampled data sets or voids represent regions with low point cloud density or missing data, respectively. Under-sampled data sets often lead to inaccuracies in extracting the details of individual features. Most existing post processing techniques (Barequet and Kumar, 1997; Carr, Fright and Beatson, 1997; Curless and Levoy, 1996; Davis, Marschner, M.Garr *et al.*, 2002; Liepa, 2003; Nooruddin and Turk, 2003; Weyrich, Pauly, Heinzle *et al.*, 2004) focus on void filling strategies and compensate for under-sampling by interpolating the rough surrounding surface to increase data density and fill the missing gaps with synthetic data points and do not work directly on the point cloud data. For large objects and spaces, the regions with missing data may become significant because huge occlusions can exist between neighboring structures or discontinuous surfaces. The traditional void filling approaches are based on triangular meshes (Barequet and Kumar, 1997; Davis, Marschner, M.Garr *et al.*, 2002; Nooruddin and Turk, 2003) or volumetric implicit surfaces (Carr, Fright and Beatson, 1997; Curless and Levoy, 1996; Liepa, 2003; Weyrich, Pauly, Heinzle *et al.*, 2004) and extending these approaches to very large data sets with multiple holes and density variations often leads to topological issues due to unknown shape semantics and unguided neighborhood connectivity of disjointed regions. Moreover, it is computationally very expensive to use these techniques on large volume scanned data (10^6 to 10^8 data points) representing building interiors.

Some recent researchers (Chen, Kua, Shum *et al.*, 2010; Liu, Carlberg, Chen *et al.*, 2010; Ohno, Kawahara and Tadokoro, 2009) have proposed innovative scanning mechanisms that reduces the distance dependency of the surfaces from the scanner. Prof. Zokhor's research group (Chen, Kua, Shum *et al.*, 2010; Kua, Corso and Zakhor, 2012; Liu, Carlberg, Chen *et al.*, 2010) at University of California, Berkeley is developing a back pack scanner that can generate the consistent density data sets from building interiors as this

scanner captures the surrounding data from nearest surfaces while sitting on the back of a moving person. This back pack scanner is focused mainly on visualization applications, where the simplified geometry is superimposed with their corresponding images. Hence, geometric details of interior space are not captured accurately. Moreover, the Global Position System (GPS) locations used to register multiple scans are to be mapped to known coordinates and require robust and accurate geometric accuracies that the present GPS systems lack, especially in interior environments. Another researcher group (Ohno, Kawahara and Tadokoro, 2009) has proposed a scanner that controls the scanning parameters based on the initial guess of the shapes to capture dense point clouds from complex objects. It controls the angle and the angular velocity of the pan-tilt mechanism and thereby captured higher data density from complex regions. This time consuming process requires multiple scanning of the domain, recursively computing the captured density and re-scanning the regions with enhanced data density. The process might not work in environments, where the domain can't be isolated for such a long time. Both these scanners are at their initial development stages and are not commercially available. Therefore, the domains are mainly captured using stationary scanners, which necessitates an effective scanning methodology that captures maximum surface geometry and minimizes subsequent data approximations must be devised in order to ensure shape capturing accuracy.

A consistent data density is essential to support direct point based modeling schemes (Alexa, Behr, Cohen-Or *et al.*, 2001; Linsen, 2001) for representing scanned surfaces in both modeling (Alexa, Behr, Cohen-Or *et al.*, 2003; Alexa, Behr, Cohen-Or *et al.*, 2001; Fleishman, Cohen-Or, Alexa *et al.*, 2003; Guennebaud and Gross, 2007) and computer graphic applications (Adamson and Alexa, 2003; Amenta and Kil, 2004; Duranleau and Poulin, 2006; Kalaiah and Vashney, 2003). A direct point based visualization approach is very helpful in accessing the quality and relevance of the captured data without generating intermediate surfaces, which are normally not required in scanning scenario of building structures involving multiple objects with varying feature sizes.

2.3.2 Missing data

Missing data are a common error in all 3D scanning applications. These are normally encountered due to the inconsistent surface properties. However, in case of occupied building

interiors, the size of the missing data is large and it often corresponds to the occluded geometries of multiple objects and their relative locations from the scanner.

Most applications involving isolated objects approximate the missing data through post-processing using hole filling strategies, which are primarily based on surface approximation of intermediate triangular meshes (Barequet and Kumar, 1997; Davis, Marschner, M.Garr *et al.*, 2002; Nooruddin and Turk, 2003) or volumetric implicit surfaces (Carr, Fright and Beatson, 1997; Curless and Levoy, 1996; Weyrich, Pauly, Heinzle *et al.*, 2004). Generating an intermediate meshed model from very large and unified point clouds is not viable, and is not required, as it is essential to segregate individual data sets representing multiple objects in order to extract desired geometric features or individual surface models. Multiple range scans can be registered (Bornaz, Lingua and Rinaudo, 2003; Hu, Zha and Zhang, 2006) to capture missing regions and large domains. However, it requires meticulous scan planning that depends on the planner's skill level and no explicit guidelines are available that can assist in scanning large interior domains.

Thus, a systematic shape acquisition pipeline must be devised that captures maximum surface geometry and minimizes subsequent data approximations in order to extract accurate shape information in various down-stream applications. An explicit set of scanning guidelines will assist the designer to plan an effective scanning strategy that generates point clouds with desired set of data characteristics in terms of point cloud density and missing occlusions.

2.4   Post-processing and Information Extraction

Post-processing operations adapt the captured point clouds to the application at hand for specific information extraction. Based on the application, these operations may aim at data simplification, feature extraction, data segmentation, geometric model reconstruction, or virtual reality modeling. Each post-processing operation treats the captured data in its own unique way in order to extract the desired output from the point cloud, e.g. visualization applications simplify the captured data to minimize the computation cost for fast display, feature extraction applications focus on individual attributes, pattern recognition applications aims to identify individual objects through segmentation, robotic path-planning applications do proximity analysis to find unrestricted pathways, and virtual reality modeling applications

attempts to reconstruct the scanned environments. It is essential to identify the application specific needs in order to devise post-processing methodologies for that specific application. The following sub-section describes some of these needs and their current research status for applications involving building interior.

2.4.1   Point cloud simplification

The shape acquisition mechanism of a typical range scanner cannot differentiate surface complexities of geometrically diverse objects lying at similar distances from the scanner. Such equidistant objects are captured with comparable data density. The scanner's parameters are often set in such a way to capture high density data sets to ensure that sufficient surface geometry from complex shapes is recorded. This leads to over-sampled data points representing simple surfaces such as planes. The high density data points for such simple surfaces are to be simplified to make it usable for most successive applications.

Most simplification algorithms rely on some sort of initial shape assessment to devise a reduction methodology. These simplification methodologies attempt to retain this initial approximated shape during data reduction process. Schroeder *et al*. (Schroeder, Zarge and Lorensen, 1992) simplified meshes by geometric and topological characterization of polygonal models locally. This approach removes vertices based on their deviation from the local average plane. Vertex removal and re-triangulation can generate topologically inconsistent models. Hoppe *et al*. (Hoppe, DeRose, Duchamp *et al.*, 1992) used a complex energy minimization approach that minimizes the signed geometric distance of the generated surface from the point cloud to create a topologically optimized triangulated model that works well for uniform and noise free data. Day *et al*. (Dey, Giesen, Leekha *et al.*, 2001) presented a point cloud simplification method with a user-controlled density guarantee which detects redundancy in the input point cloud with the help of the *Cocone* algorithm and local feature size concepts (Amenta, Choi, Dey *et al.*, 2000). The approach is quite effective, but only uses local curvatures as a driving factor for data decimation. Pierre and others (Pierre, Giuliana, Craig *et al.*, 2007) have compiled a detailed survey on the re-meshing of surfaces for graphical applications. In general, polygonal simplification requires an initial triangulated surface model to start the decimation process, which is a complex and error-prone process of generating a void free triangulated model from point clouds with large voids due to occlusions.

An adaptive triangulation (Akkouche and Galin, 2001; Araújo and Jorge, 2005) of implicit surfaces can extract a decimated, smooth and water tight triangulated model by optimizing the sizes of the triangles. However, such a model approximates the underlying implicit surface model only and not the original point cloud data set. This method is also computationally expensive, especially for large spatial data such as building scans.

Direct point cloud simplification, on the other hand, attempts to reduce the number of data points without generating any intermediate surface. Lee *et al*. (Lee, Woo and Suk, 2001) presented a normal deviation driven, octree-based spatial subdivision strategy to extract a representative 3D data point from each cell for data simplification. However, it requires a structured and noise free point cloud for simplification. Another height decision method (Chang and Chang, 2002) was used to remove points from the non-contour part of the data set with a limited applicability for non-complex contoured surfaces. Pauly and others (Pauly, Gross and Kobbalt, 2002; Pauly, Kobbalt and Gross, 2004) modified the mesh-based simplification theory to achieve point-based simplification for point-based graphics and multi-resolution surface modeling using the moving least square (MLS) method and iterative simplification process. This method re-computes the MLS surface after each point removal and is computationally expensive. Song (Song and Feng, 2008) proposed an approach of progressively removing the non-boundary points. This algorithm is also computationally expensive and cannot be extended to objects that have no distinct boundaries. The applicability of direct point cloud simplification method is advantageous but it requires complex, recursive computation, which is often expensive.

Most simplification approaches on large data sets have been demonstrated on exterior Airborne data set (ADS) where the data decimation is achieved as a by-product of the local surface fitting (Alharthy and Berthel, 2004; Boulaassal, Landes, Grussenmeyer *et al.*, 2007) or user-defined feature extraction process (Pu and Vosselman, 2006; 2007; Vosselman, Gorte, Sithole *et al.*, 2004). In the first approach, the scanned data sets are approximated by multiple local surfaces with relatively simple geometry such as planes and the points with very low deviations from these locally fitted planes are removed. This approach has been extensively used in airborne data sets (Rottensteiner, 2003; Rottensteiner and Briese, 2002; Sampath and Shan, 2010; Verma, Kumar and Hsu, 2006; Vosselman and Dijkman, 2001; Zhou and Neumann, 2008) where multiple regions such as grounds, walls and roofs are simplified and are approximated through local plane fitting. The second set of algorithms

simplify a point cloud by removing the non-feature data points while retaining the data points representing a specific feature such as roofs (Vosselman, Gorte, Sithole *et al.*, 2004), walls, doors (Pu and Vosselman, 2006) and windows (Pu and Vosselman, 2007). The later approach focuses on the segmented data regions representing the desired feature and simplifies the data set by removing non-feature data points. However, as the primary objective of these processes is surface modeling and not explicit data simplification, thus, it often over-simplifies the scanned geometry, especially the regions with complex geometries.

A few researchers (Bahmutov, Popescu and Mudure, 2006; Barnea, Filin and Alchanatis, 2007; Budroni and Böhm, 2009; Wang and Luebke, 2003) have demonstrated the simplification strategies on triangulated models, which are generated from interior point cloud data. A simplified triangulation model (Wang and Luebke, 2003)was used that integrates the scanned data using local normal and confidence-level semantics along with weighted color and texture coordinates. Such a cumulative triangulated model can enhance the overall visualization of the scanned scene, but it cannot differentiate geometric diversity of individual objects in the scene. Another similar method (Bahmutov, Popescu and Mudure, 2006) had demonstrated a rapid simplification methodology for large interiors using simple shape semantics and registered textures, e.g. an approximated rectangular cuboids with mapped textures can be used to represent long pathways in building interiors. This approach generates visually appealing results with limited geometric accuracy. A plane sweep algorithm based method (Budroni and Böhm, 2009) recognizes the planar structures of a room in the vertical and horizontal directions, sequentially. Thus, points representing walls, ceiling and floor are segmented and are approximated by simplified planar data sets. The result generated simplified surface models of piece-wise planar data sets.

The simplification of large, complex domains requires an adaptive approach that does not over simplify point clouds and retains multiple features in the scanned scene. Most simplification methods demonstrated on building interiors rely on identification of planar regions in the point clouds and tend to over-simplify the scanned geometries and thus are not suitable for occupied building interiors which include both planar (walls, floor and ceiling) and complex (freeform objects) geometries, simultaneously.

2.4.2   Feature extraction

The point cloud data set of occupied building interiors represents unified point clouds from multiple objects that involve wide range of features. It is essential to transform this raw data into some usable format through specific feature extraction or segmentation. Such post-processing of data is governed by characteristics of the scanned scene and the subsequent usability of the captured data.

Specific feature extraction approaches often rely on a priori knowledge of scanned features and segregation of corresponding point clouds. A semantic knowledge management system (Duan, Cruz and Nicolle, 2010) was proposed to classify the building elements for architectural reconstruction based on their geometry (points, planes) and building elements (walls, floors, ceilings, door and windows). Rusu *et al*. (Rusu, Marton, Blodow *et al.*, 2008) presented a similar object/feature classification in a typical household (kitchen) environment. They also generated methodologies to extract and model these specific features from the corresponding point cloud data sets for virtual reality reconstruction. Similar context-based methodologies are being developed by Dr. Huber's group at the Robotic Institute of Carnegie Mellon University (Huber, Akinci, Tang *et al.*, 2010; Oliver and Huber, 2010; Xiong and Huber, 2010), which aims to generate semantic 3D models of indoor environments. The methodology requires a priori knowledge of the building features and shape semantics. However, the type of feature to be extracted depends largely on the application of the extracted data and can be anything from floor plan layouts, doors, windows, wall features, and interior objects of the 3D-scanning environments.

Automated floor plan extraction has been extensively demonstrated on 2D aerial images of 3D building exteriors. An approximate 3D shape extraction approach has been demonstrated by combining the CCD images and sparse laser scanning samples (Hongjian and Shiqiang, 2006). It uses the Laplacian sharpening approach that processes the CCD images with Laplacian filter to extract 2D edges and determines the building height from sparse aerial data points. The approach can only extract primitive shapes of building exteriors. Another improved approach of building feature extraction from terrestrial laser data (Pu and Vosselman, 2006) uses the segmentation approach with different feature constraints based on the data segments, which may not be suitable for domains with no predefined constraints. A geotechnical monitoring framework (Su, Hashash and Liu, 2006) was

presented for an urban excavation application. This paper describes the planning, execution, and data processing phases of collecting accurate construction information with an unprecedented level of detail on the as-built site conditions. The usability of 3D range images is extended to interior design applications in a feasibility study (Shih and Hu, 2007). In this paper, the captured range images are mainly utilized for visual inspection only and a feasibility study is proposed for applications involving design modeling, production drawing, construction monitoring and follow-up interior maintenance and management. However, the feasibility study needs to be demonstrated on real applications that require the development of robust geometric data extractions strategies by solving issues such as identifications of occluded geometries and voids, establishing individual feature correspondence and segmentation of cohesive data.

Interior floor plan has been a well studied field in robotics navigation research. In which case, a robot-mounted laser scanner is often used to generate floor plans. This presents an additional challenge of addressing robot localization. Moreover, the extracted layouts are not significantly accurate to be used as a building layout. Recently, floor plans are extracted from 3D models created from reverse engineering of point clouds. Most of these algorithms rely on planar region approximation and region growth approach (Hähnel, Burgard and Thrun, 2003), random sample consensus algorithm (Schnabel, Wahl and Klein, 2007), traditional Hough transforms (Tarsha-Kurdi, Landes and Grussenmeyer, 2007) and plan-sweep search algorithms (Budroni and Böhm, 2009; Johnston and Zakhor, 2008). Hähnel's approach (Hähnel, Burgard and Thrun, 2003) identifies planar regions by approximating a plane through random set of points by minimizing the sum of the squared distances between the points and the plane, whereas the plane normal is given by the eigenvector. Subsequently, neighboring planes are merged together into large local planes. The random sample consensus (RANSAC) paradigm (Schnabel, Wahl and Klein, 2007) is used for fitting parametric model to the data. A candidate plane is fitted through three points and testing the remaining points to compute its planar score. The plane is identified based on a threshold score and its converged trials. Improved Hough Transforms (Vosselman, Gorte, Sithole *et al.*, 2004) use surface normal to speed up the planar surface detection process, where the plane parameters are mapped to the individual points and their normals in space, thereby reducing the risk of detecting false data planes. The plane sweep algorithm (Budroni and Böhm, 2009) identifies planar regions by sweeping a predetermined plane and computing a

distance threshold of points from the sweeping plane. Most of these methods solve the floor plan problems as a sub-part of a more complex problem, which is often computationally expensive due to the huge size of the point cloud.

A relatively simple layout extraction approach works on the partial scanned data captured from a single horizontal slice of the domain. A piece-wise linear model is fitted to this data set and layouts are extracted using traditional iterative point fitting (Duda, Hart and Stork, 2001), split and merge (Pavlidis and Horowitz, 1974) and other similar, computationally inexpensive approaches (Nguyen, Gächter and Martinelli, 2007). However, these approaches are unable to successfully fill the gaps created during scanning due to occlusions. Thus, it will be useful to generate direct point based approaches and extract layouts from sliced data sets from 3D point cloud data sets.

2.4.3   Point cloud segmentation

One of the most important aspects of automatic and semi-automatic generation of 3D and building information models (BIM) from point clouds is the fragmentation of the data into manageable clusters. The convenience of segmenting such small clusters helps in reducing the computational cost and simplifying the scanned scene besides extracting specific geometric information. Such segmentation is even more crucial for cases representing large point clouds of building interiors.

Such large point clouds are often segmented by approximating the identified regions into planar surfaces. These surface classifications have been extensively demonstrated on LiDAR data for building model extraction (Elberink and Vosselman, 2009; Forlani, Nardinocchi, Scaioni *et al.*, 2006; Kim and Shan, 2011), point cloud registration (Behan, 2000), specific pattern or feature extraction (Alharthy and Bethel, 2002), terrain segmentation (Arefi and Hahn, 2005), and structure recognition (Vosselman, Gorte, Sithole *et al.*, 2004). Most of these algorithms focus on model reconstruction from LiDAR data in two main steps: (1) separating ground and non-ground points (Zhang *et al*., 2003; Shan and Sampath, 2005) to simplify the point cloud and (2) processing non-ground points to determine planar roof segments (Rottensteiner, 2003; Tarsha-Kurdi, Landes and Grussenmeyer, 2007; Vosselman, Gorte, Sithole *et al.*, 2004). During this reconstruction process, the data are segmented as an essential by-product and not the main objective of this virtual model reconstruction process.

The segmentation-based simplification of exterior point clouds lies in the fact that the unimportant points (ground points) are equidistant from the airborne scanner and thus their depth analysis often forms the basis of a reliable classification strategy. Moreover, such segmentation strategies operate under the assumption that the scanned surfaces are unobstructed. In an indoor environment, the surfaces are often obstructed by multiple objects in the scanned scenes. Additionally, the varying distance of these surfaces from the scanner generated non-uniform data sets, which increases the segmentation complexities.

Relatively few segmentation methods have been applied successfully to range scans of building interiors. A hybrid approach (Wolfart, Sequeira, Ng *et al.*, 1999) was proposed for surface segmentation of the triangulated interiors through data re-sampling, local surface fitting, and discontinuous edge detection to solve the problem of segmenting building interiors. Although the technique produced satisfactory results for large planar regions such as walls and cabinets, the algorithm could not accurately cluster points that lie on small objects with freeform shapes due to inadequate seed points. To address this problem, Rabbani (Rabbani, Heuvel and Vosselman, 2006) used normal residuals to identify reliable seed points in the cloud and a normal deviation-based smoothness parameter for region growing. The method was able to successfully extract unique clusters from a scanned industrial environment. The computed geometric parameters were, however, not highly reliable and prone to significant errors in overlapping regions which lead to data over-segmentation. In contrast, a plane-sweep algorithm (Budroni and Böhm, 2009) was proposed for segmenting 3D point clouds into planar clusters. The relative advantage of this approach is that it works directly on sampled points instead of tessellated surface-based approaches (Rabbani, Heuvel and Vosselman, 2006; Wolfart, Sequeira, Ng *et al.*, 1999). Although effective, the technique omits all non-planar data points and is an approximation rather than the segmentation technique. Therefore, it cannot directly be used to segment building interiors that contain desks, chairs and personal items.

Many researchers have approximated the scanned data into its semantics features, such as walls, windows, or doors using planar primitives. Stamos (Stamos, Gene, Wolberg *et al.*, 2006) used an adaptive segmentation approach to identify such planar regions and model them with planes whereas the non-planar regions were modeled by meshed surfaces. Such an adaptive approach can handle planar and complex regions, alike and reduces modeling complexity considerably. However, it first generates the tessellated model and then simplifies

it to generate the adaptive model. Similarly, a prior knowledge regarding the feature sizes, positions, and topologies is beneficial in recognizing specific elements (doors, windows, roof and walls) in a point cloud (Pu and Vosselman, 2009; Vosselman, Gorte, Sithole *et al.*, 2004), which can help in differentiating primitive shapes, such as planes, cylinders and spheres by means of non interactive methods to detect clusters. Other strategies segment planes using the RANSAC algorithm (Boulaassal, Landes, Grussenmeyer *et al.*, 2007). Although these approaches have been demonstrated on exterior facets but the methodology is applicable to interior environments as well and need further investigations.

In building interior environments, in addition to structural elements, the space is often occupied by furniture and other objects, which create large occlusions during scanning. These interior objects influence the detection of major structural elements such as walls, floor and ceiling. (Dell'Acqua and Fisher, 2002) showed a technique to reconstruct the shape of planar surfaces behind arbitrary occluding surfaces by combining small planar regions with similar geometric properties. An edge based planar reconstruction methodology (Castellani, Livatino and Fisher, 2002) was also proposed by combining occluded edges of the interior objects. Wang and Oliveira (Wang and Oliveira, 2007) proposed a smooth surface reconstruction methodology that was able to fill small occlusions in relatively complex areas. Recently, a labelling and merging algorithm (Oliver and Huber, 2011) was presented, in which the regions are identified, labelled and are grouped together to create hole free planar regions such as walls. The approaches have been demonstrated on relatively less cluttered data sets involving small objects. Moreover, such methods can extract only primitive surfaces from the point clouds.

Complex surface extraction methods from range data of large building interiors often exploit known shape or size semantics to generate cumulative building models. An automatic data segmentation and geometric feature extraction algorithm (Huang and Menq, 2001) was presented. Wang and Luebke (Wang and Luebke, 2003) generated triangulated models of the scanned data by using normal and confidence-level semantics along with weighted color and texture coordinates. A cumulative triangulated model can enhance the overall visualization of the scanned scene, but it does not permit surface segmentation and reconstruction of individual objects in the scene. Bahmutov *et al* (Bahmutov, Popescu and Mudure, 2006) demonstrated a rapid building reconstruction method for large structures by using shape semantics such as rectangular cuboids for pathways walls. The super-positioning of

registered color information on this sparse depth map can generates photorealistic models with over-simplified geometries (Grazzini, Prasad and Dillard, 2010; Xu, Ye and Fan, 2002; Zhang, Yang, Liu *et al.*, 2008). These over-simplifications often lose critical geometric details. It is, therefore, difficult to use a single algorithm that simultaneously handles simple planar shapes and complex, organically shaped objects. Furthermore, most geometry based segmentation methods require intermediate surface generation (Wolfart, Sequeira, Ng *et al.*, 1999) or geometric parameter computation (Rabbani, Heuvel and Vosselman, 2006) which are often not reliable if the scanned scene contains multiple overlapping objects.

Color based segmentation methodologies (Cheng, Jiang, Sun *et al.*, 2001; Debevec, Taylor and Malik, 1996; Yang, Kim, Toh *et al.*, 2010; Zhang, Yang, Liu *et al.*, 2008) have been extensively used for images. With the availability of colored point clouds from modern day 3D-scanners, these approaches can be exploited to segment colored point clouds as well.

A color-based registration methodology (Xu, Ye and Fan, 2002) was demonstrated for isolated 3D-objects, where the objects were scanned and the color parameters were used to register multiple views. Wang and Luebke (Wang and Luebke, 2003) presented an integrated methodology that combines color and 3D distance fields. This approach generates a color spatial model by combining 3D Euclidean distances and 2D range images. Although they used the colored range data to generate cohesive interior scenes, the mapped color information can be exploited to form an effective segmentation strategy. Zhan, Liang and Xiao (Zhan, Liang and Xiao, 2009)proposed such a color-based segmentation strategy (color based segmentation) for clustering the scanned data of a heritage site and reiterated the importance of using color as the segmentation parameter, especially for areas with similar geometric properties. However, color cannot be used as a sole differentiating parameter for formulating robust segmentation strategies. This is partly due to the fact that the physical properties of the scanned surfaces and irregular lighting condition generate varying shades of single colors. Recently, an illumination-invariant color space (Yang, Kim, Toh *et al.*, 2010) is proposed which can be effectively used to formulate robust color-based segmentation strategies. Further, combining the geometric and color based parameters can facilitate effective segmentation of colored point clouds

2.5   Concluding Remarks

Over the last decade, laser sensors have become more adaptable to be used in capturing large, complex structures of building interiors. It has emerged as an accurate and very fast means of capturing large complex geometries. However, extracting desired information from this cohesive point cloud is a cumbersome task, especially when the point cloud represents occupied building interiors with multiple overlapping objects. Specific tools are required to convert the point clouds into information models to improve its usability. The literature survey focuses on core issues of this shape capturing and information extraction process and identifies key parameters fulfilling the post-processing expectations of the point cloud representing building interiors.

This chapter provided a detailed discussion on the scanning techniques for capturing spatial geometry from point clouds. All aspects of the 3D scanning process were considered to identify parameters responsible for accurate acquisition of interior environments. The conditions leading to inconsistent densities and voids are discussed. Current scanners and their shape acquisition mechanisms were explored in terms of their scanning resolution, capturing speed, dimensional accuracies and measurement limitations were explored and literature addressing these issues was compiled.

The main scanning inconsistencies considered were variation in data density and missing data points from occluded regions, which are very critical in the case of building interiors. The variation in data density is an implicit problem of modern day stationary scanners capturing large domains where as the missing data mainly corresponds to occluded geometries in occupied building interiors. These non-uniformity and voids in the data sets are often improved by post processing the captured data through shape approximation and thus the resultant shape often is over-simplified. Moreover, such approximation also fails in the absence of a reliable up-sampling model for significantly sparse data set and large voids. Very few methods have presented preliminary studies on redesigning scanning mechanisms for capturing consistent data density. Nonetheless, stationary scanners are still largely used to capture large domains. Excessive post-processing corresponding to inconsistent density and data voids can be avoided by using registered data from multiple scans, but no specific scanning guidelines are available. Thus it is imperative that an effective scanning

methodology is essential to minimize the scanning inconsistencies at the initial stage that will minimize subsequent post-processing during information extraction stage.

Even an accurately captured point cloud represents a set of spatial data points with no direct usability. An effective set of tools are required to extract desired geometric, spatial and dimensional features from the point clouds. Defining these features is a complex task as they are highly application dependent and the information extraction tools cannot be standardized. However, all such tools need to have two main desirable characteristics: (1) its direct applicability on point clouds to avoid costly shape reconstruction cost for extracting preliminary features and (2) its application-specific simplification/segmentation strategy to handle large point clouds.

The interior building layout extraction algorithms described in the literature often obtain the layout from the reconstructed surface model of the captured data, which is computationally very expensive for large data sets and is of little use for occupied building interiors due to reconstruction inaccuracies. Thus, there is a need to have efficient simplification and layout extraction algorithm which can efficiently tackle cohesive data sets of occupied building interiors. The layout extraction should be applicable on original point clouds and should compensate data representing interior objects and occluded geometries.

It is essential to segment the point clouds to improve scanned scene clarity pertaining to individual objects/regions of scanned environment. Traditional geometry based algorithms are effective in segmenting individual surface data clusters of an isolated object only. Such approaches are not effective in scanning environments involving multiple objects with similar geometric properties and overlapping regions. Thus, additional differentiating parameters are to be identified to improve the robustness and level of segmentation.

The next chapter introduces an efficient scan planning methodology for capturing large building interiors with consistent data density and minimum information loss. Such data can be used to extract accurate information during post-processing of the scanned data.

CHAPTER 3   CONSISTENT DENSITY SCANNING

3.1   Introduction

Range scanning of complex, three-dimensional objects has been used in a variety of engineering and scientific applications (Várady, Martin and Cox, 1997). Commercial range scanners capture the shapes by acquiring spatial data points from visible surfaces in the scanned scene. It is essential to capture sufficient data points to represent an accurate geometry of the scanned surface, feature or an object. The usability of this captured data is governed by the quality of the point cloud acquired during the range scanning which can be defined in terms of captured data density and its consistency over the whole scanned domain. The captured data density obtained from small, isolated objects is often very consistent due to the coherent characteristics of the scanned domain and fairly uniform distance between the scanner and the object being scanned.

However, when a 3D scanner captures large domains involving diverse features, multiple objects and incoherent geometries, the captured data density is not consistent. The variation in the data density occurs due to the scanner's limitations and domain's physical parameters such as vast geometric diversity, presence of occluded regions and surface properties. It is difficult to control the inconsistencies arising from surface properties and positions of the objects in the scanned scene. However, the properties of the captured data can be controlled to some extent by compensating the scanner's limitations with effective scan planning. A uniform density data set can only be captured if the distance between the scanner and the domain's surfaces remains constant. Thus, a moving scanner is an ideal option, however, such scanners are still in development stage and the data is captured largely with stationary scanners with fixed angular scanning resolutions. Moreover, limited number of stationary scanner positions and varying surface distances from the scanner contribute towards scanning inconsistencies that can only be controlled through effective scan planning. The data density is reduced over the surfaces lying away from the scanner and thus, essential shape details may be lost. Due to this disparity in data density, most post-processing approach cannot be equally effective over the whole scanned domain, especially in regions with sparse point clouds. This density variation is not given that much importance in existing literature as most algorithms aspire to extract only primitive shapes from the point clouds, for

which, the sparse data does suffice. However, such inconsistent density point clouds are not sufficient in scanning scenario involving large domains with complex feature capturing capability.

3.2   Need of Consistent Data Sets

The consistent data density is essential to capture and visualize accurate scanned geometries. The dense data set facilitate precise reconstruction of captured features to accurately model the virtual reality environments. Such reconstruction also provides a fast way of evaluating the underlying structural form of the scanned data using point-based visualization techniques (Pauly, Keiser, Kobbelt *et al.*, 2003) that exploit both coordinate data and the corresponding normal vectors. However, accurate normal vectors and the geometric features can only be generated if the density of acquired data points does not vary significantly across the entire scanned domain. A uniform distribution or consistent density of data points is almost impossible in a single scan of large objects and spaces that have restricted accessibility, unavoidable occlusions, discontinuous surfaces, and distant regions that produce spurious and under-sampled data.

A consistent data density is also required for approximating the missing data arising from object occlusions and restricted space accessibility. Availability of sufficient data points around the occluded regions ensures that the local shape is accurately reconstructed and can be extrapolated to approximate the missing regions. Such region approximation ensures that the gap is filled with accurate shapes and is not over-simplified.

Problems arising from inconsistent data density and data under-sampling are commonly addressed using post processing tools (Barequet and Kumar, 1997; Carr, Fright and Beatson, 1997; Curless and Levoy, 1996; Davis, Marschner, M.Garr *et al.*, 2002; Liepa, 2003; Nooruddin and Turk, 2003; Weyrich, Pauly, Heinzle *et al.*, 2004) that attempt to interpolate the surrounding data points in order to approximate missing data points or sparse data regions. These techniques are not always effective or computationally efficient for large domains due to the volume of missing data, existence of overlapping objects, and large variations of feature sizes.

A viable approach of minimizing the post-processing of data sets is to control the captured data density at the shape acquisition stage through an effective scan planning

methodology. However, an experienced scan planner is needed to plan and implement such a methodology and no systematic approach is available that can help in planning the scanning methodology for generating fairly consistent point clouds. A desirable method must associate the domain's dimensional parameters to the scanner's settings to compute the optimum scan capturing settings. Such a process will ensure that the captured data density is fairly consistent and desired shape features can be extracted from the point cloud. In this work, a systematic scan planning methodology, applicable to large building interiors is introduced that produces consistent point cloud data with user defined data density. The proposed method determines the partial range that a stationary scanner can capture ensuring the data density and computes the total number of range scans required to cover the whole domain within the desired "limiting density value".

3.3   Range Scanning of Building Interiors

A typical scanning scenario involving large buildings, structures, and enclosed environments uses a stationary, rotating range sensor to capture shape and geometry. The sensor consists of a laser projection system and photo detectors that record either the spatial displacement or the time-of-flight of the reflected light. If N and M are the number of rays emitted in horizontal and vertical planes, respectively, then the corresponding angles between two consecutive rays, defined as angular scanning resolutions (α and λ) are defined as follows:

$$\alpha = \frac{\angle_{H\max}}{N-1}; \qquad \lambda = \frac{\angle_{V\max}}{M-1} \tag{3.1}$$

where $\angle_{H\max}$ and $\angle_{V\max}$ are the maximum angular spans that the scanner can traverse in the horizontal and vertical directions, respectively. The data points from the visible surfaces are captured at this fixed angular resolution through the laser rays and thus, the geometric parameters of the scanned domains governs the quality of the captured data.

3.3.1   Domain shapes and data characteristics

The quality of the captured point cloud can be symbolically measured through the local variation of the captured data density from a given surface, which is primarily dependent on

its distance from the scanner due to fixed angular scanning resolutions. Hypothetically, each stationary scanner can capture a spherical domain with uniform distribution of data points due to the fixed distance of its surfaces from the scanner. Such a spherical domain can be termed as an *ideal domain* of a stationary scanner (Figure 3.1(a)). A consistent data set can also be captured from the cylindrical domain, when a small variation in the captured point cloud is permitted along the vertical span (Figure 3.1(b)).

(a)   (b)
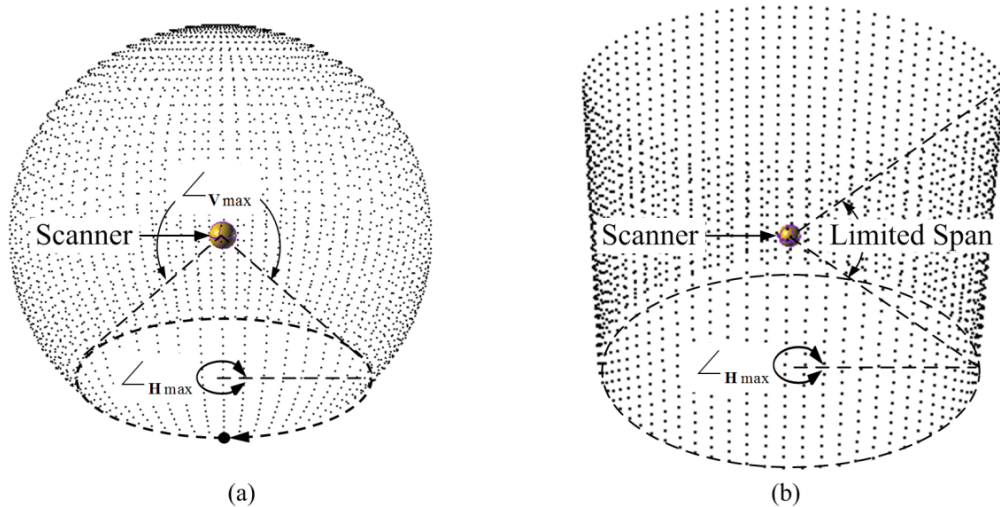
Figure 3.1:   Ideal domains for a stationary scanner for capturing consistent data density (a) Uniform data sets from a spherical domain, and (b) Low density variation in cylindrical domain.

In most practical cases, the actual world domains will differ drastically from these *ideal domains* and thus uniform density data sets are not feasible. Instead, the quality of the captured data is measured in terms of a limiting value of the captured data density called "consistent density". The data density is said to be consistent if its variation in the captured data set is less than a predefined value. The captured data density at a given location in the scanned scene is mainly dependent on its distance from the scanner. A relatively consistent data density can be captured from a domain, where the surfaces lie at a comparable distance from the scanner such as a square room or a large hall having comparable dimensional sides (length/width ≈ 1). However, linear spaces (Figure 3.2) such as long corridors and hallways with very large length/width ratio are severely affected with density variation problems arising due to vastly increasing distances of its surfaces from the stationary scanner and are addressed in this study.
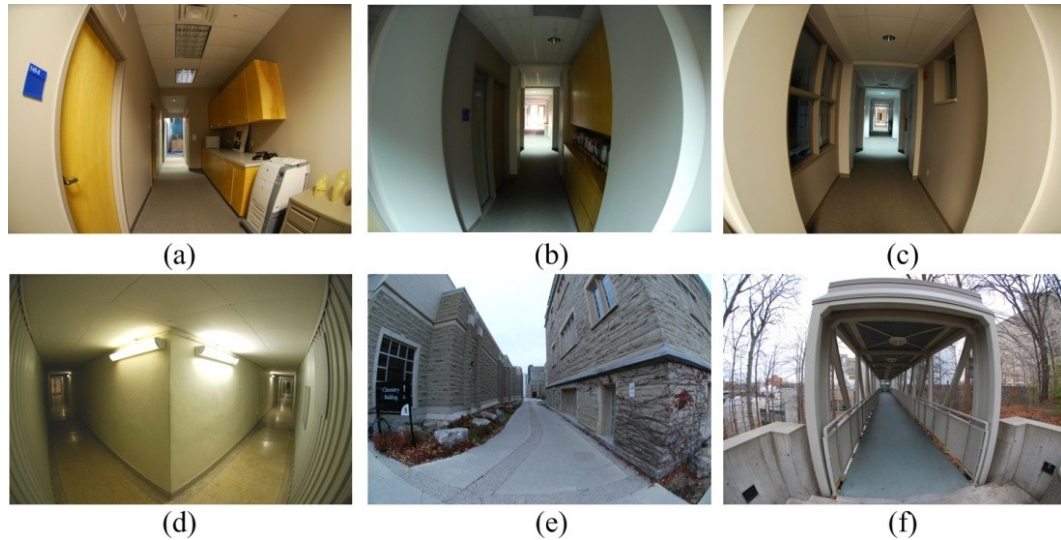
Figure 3.2: Narrow interior pathways and restricted accessibility scanning scenarios (a, b, c) Interior corridors (d) Under-ground tunnels (e) Constricted exterior walkway (f) Slender over-bridge.

In a typical scanning scenario of such large linear spaces, a scanner is placed at a distance (D) from the linear world domain, and geometry is captured using a fixed angular scanning resolution (α) as shown in 2D in Figure 3.3. The spatial distribution of point cloud largely depends on the geometric characteristics of the involved surfaces or features, and hence point cloud density in terms of the number of points per unit area/vol. is not feasible. Instead, data density in the captured point cloud is indirectly defined in terms of fast and measurable parameter called *data density distance* (d_n), which is the distance between two consecutive captured data points along the scan direction. Data density is maximum at the nearest region from the scanner where d_n and incident angle (θ) are minimum. The data density at the nth-point depends on its distance from the scanner. The reduction in data density is appreciable along long, narrow pathways where the high length/width ratio restricts the scanner distance (D) to be low. This low scanner distance is responsible for large date density distances (d_n) at far away regions as the angular scanning resolution (α) is often fixed for a given scanner setting.
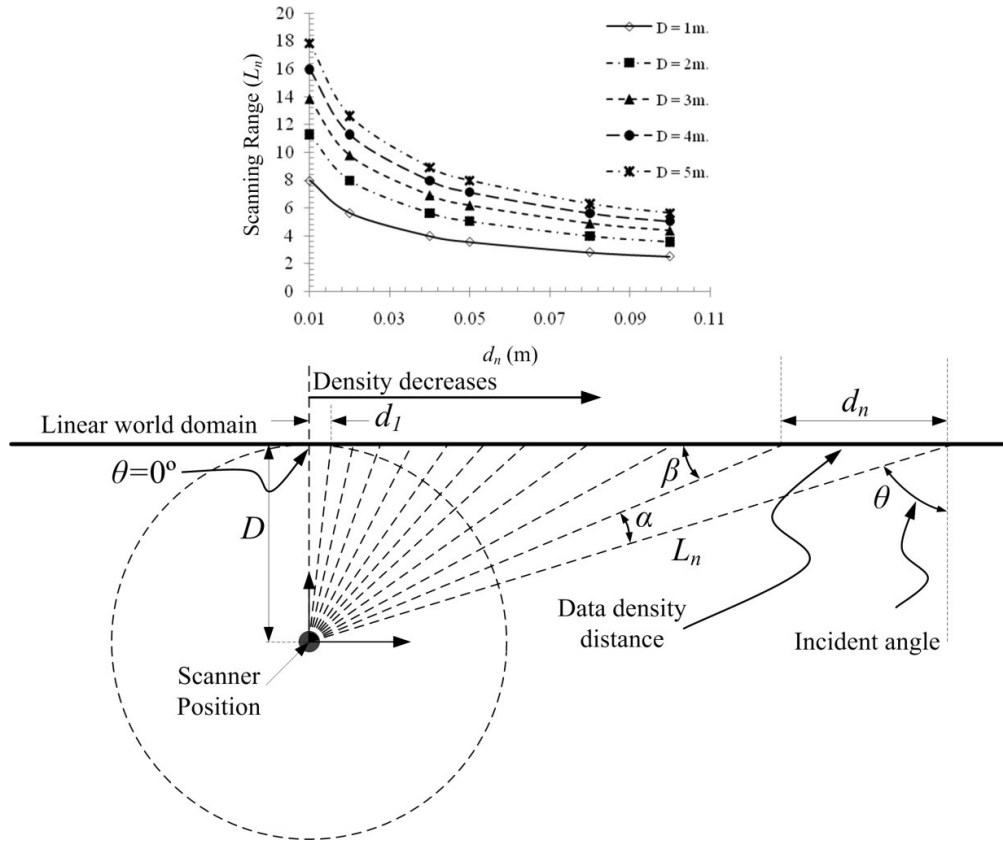
Figure 3.3: Data density distance (d_n) variation in a typical narrow domain scanning scenario using a stationary scanner.

In addition, the further away the measured surface discontinuity is from the stationary scanner, the more likely large spurious data values will exist because of the *mixed pixel problem* (Breunig, Kriegel, Ng *et al.*, 2000). This type of erroneous data is the result of overlapping signals from the front and back surfaces along the silhouette boundaries (Figure 3.4) along the scanning direction. With an increase of incident angle (θ) and data density distance (d_n), the scatter of such *outliers* increases (from **A** to **B**). Ideally, a scanner must use minimum incident angle (θ=0°) to capture the data to minimize the spread of spurious data, which is not practical for a stationary scanner and thus, an effective scan planning methodology must use a limiting value of these scanning parameters to compute the permissible scanning range for consistent scan data density.
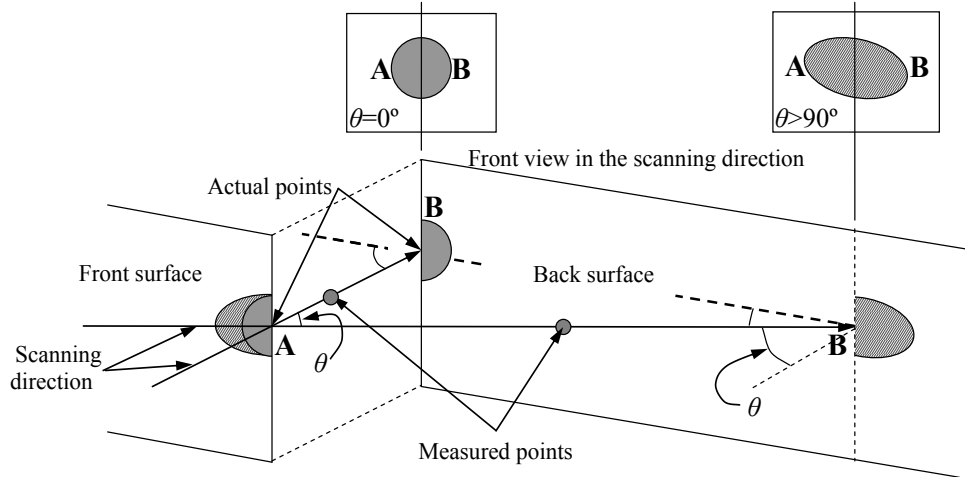
Figure 3.4:   Scanning outliers due to mixed pixel problem at silhouette boundary

3.3.2   Captured data density

The lack of synchronization between the *ideal domain* (spherical) of a stationary scanner and the real world domain leads to non-uniformity in the captured data. As the geometric characteristics of the features involved in different real world domains vary considerably, a highly generalized mathematical description for defining the data density in terms of data density distance (d_n) is not feasible for individual features and diverse applications. A simplified geometric representation of the domain using piecewise integration of multiple planar surfaces is proposed that helps in formulating the mathematical definition of data density. Such simplification is also necessary as the exact dimensional details of domain features are often not available at a pre-scan stage.

The data density distance is computed by extending the scanning rays from its ideal domain to the real world, planar domain and the captured data density distance (d_n) at a location is computed for given scanner specifications (*N, α, scanning time*) and key geometric parameters of the world domain (*D, L, θ*). Analytically, the captured data density distance (d_n) at the nth point along the horizontal scanning line is defined by the eqn. (A.1) in Appendix-A.

$$d_n = \frac{D\sin(\alpha)}{\cos(n\alpha)\cos((n-1)\alpha)} \qquad (3.2)$$

It is to be noted that data density distance (d_n) is defined in terms of a domain parameter (D) and angular resolution (α) only and hence, does not require extensive spatial information of the domain features

3.3.3 Restricted data density

Restrictive data density is defined by the limiting condition of the allowable data density distance (d_{alw}) such that captured data represent coherent data density throughout the domain. The limiting value of this user defined, allowable data density distance (d_{alw}) is governed by the desired data characteristics of the subsequent applications where the captured data is to be used. It can be computed from the smallest feature size for geometric modeling, shape capturing capabilities of the reference elements for accurate registration or sparse data distribution for simplified shape extraction. In most practical cases, the data density requirements vary considerably for different applications and defining an allowable density condition (d_{alw}) is a complex task. Three possible cases are proposed to define the value of this allowable data density distance (d_{alw}) to cover most of the practical cases and corresponding scan planning methodology is outlined in the following paragraphs.

3.4 Consistent Point Cloud Scanning Methodology

Scanning large buildings, structures, and enclosed environments with a single stationary scanner (Figure 3.3) generates a point cloud with varying data density. An efficient scan planning methodology can generate consistent data sets by restricting the variation in data density distance (d_n) within an allowable value (d_{alw}) and the resultant data set is called a consistent point cloud.

In this study, three cases are proposed to define the restricted value of the allowable data density distance (d_{alw}), however, the proposed methodology can work for any user defined value of allowable data density distance (d_{alw}). This restricting condition defines the scanning range of each stationary scanner that can be captured to ensure the desired data density condition. Restricting the captured data density distance (d_n) to a maximum value (d_{alw}) reduces the overall scanning range of a stationary scanner. Although the commercial scanners have large scanning range up to few hundred meters, only a small portion of this range can be captured with desired data density and is termed as *consistent scanning range* (L_n).

A consistent scanning range (L_n) for each stationary scanner is computed analytically by comparing the captured data density distance (d_n) at n^{th} point and the allowable d_{alw} using the process described in the flow-chart (Figure 3.5). The analytical model computes the number of points (n) that can be captured from the linear domain by checking the density condition, recursively. The numbers of points determines the permissible angular span ($n\alpha$) which is used to compute the corresponding *consistent scanning range* (L_n) of a stationary scanner.

Multiple scanning ranges are captured by positioning the scanner at equidistance locations computed using the proposed scan planning methodology and then these individually captured data sets are registered to generate overall point cloud, which is consistent in terms of its data density variations.
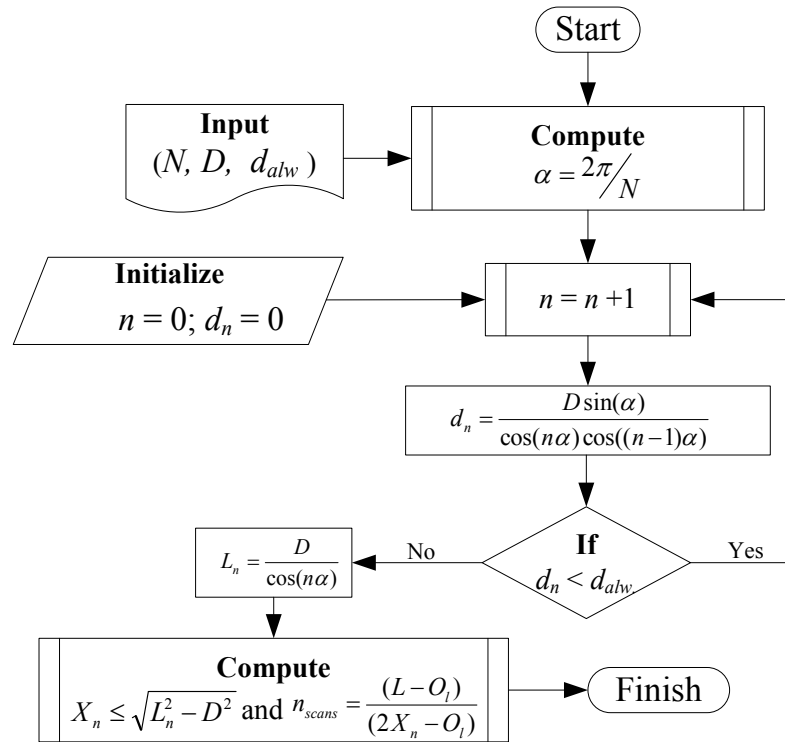
Figure 3.5: Flowchart of the consistent scanning process.

3.4.1 Defining allowable data density distance

Depending upon the type of geometry being captured and the physical scanner parameters, one or more of the following three cases can be used to define the allowable data density

distance (d_{alw}) and compute its corresponding consistent scanning range (L_n) as discussed below.

Case I: : d_{alw} = constant = ξ• feature size

Scanning applications with specific shape capturing objectives define the allowable density as a discrete value based on the size of the desired feature (d_{alw} = constant = ξ• feature size). In this case, a suitable fraction value (ξ) is selected in such a way that sufficient data points are captured from the desired feature to represent its true surface geometry. A low value of ξ captures more number of points from feature that help in recreating its true geometries. The relationship between a consistent scanning range (L_n) and the other scanner parameters for this d_{alw} condition is defined by Eqn. (A.2) in Appendix-A as follows:

$$L_n^2 = \frac{d_{alw} D}{sin\alpha} = \frac{d_{alw} D}{\alpha} = d_{alw} D \frac{N}{2\pi} \qquad \text{where } (sin\alpha \approx \alpha \text{ for small } \alpha) \qquad (3.3)$$

where the permissible scanning range can be computed for a given value of D and N and an allowable value of d_{alw}. The quadratic relationship between L_n and D for a constant value of d_{alw} and N indicates that the a consistent scanning range (L_n) reduces rapidly for small scanner distance (D). The consistent scanning range (L_n) can be increased by using a higher resolution scan (large N) for narrow spaces, which is limited by the available scanning time.

Case II: d_{alw} = f D

This case is suitable for applications where the desired data density is not known and the accessibility to the domain surfaces is very restrictive i.e. the scanner distance D can only have very small values or the domain may have multiple occluded surfaces. The 3D scanning with such small scanner distance (D) generates data sets where the data density diminishes at a much faster rate and only a small range can be captured with given data density condition (Figure 3.3). This limited scanning range for narrow spaces and multiple surface occlusions limit the accurate selection of reference elements (contrast circles or spheres) due to the sparse data density in far off regions. The incident angle (θ) increases rapidly and thus makes it difficult to capture and use the reference elements for accurate registration due to their

distorted projected shapes (contrast circle) and sparse data density (reference spheres). Therefore, the allowable data density condition (d_{alw}) is restricted by the shape capturing capability of these reference elements. The sharp reduction of data density in this case is due to the low D value and thus the restrictive data density condition has to be a function of scanner distance D as given in Eqn. 4.

$$d_{alw} = f D \qquad (3.4)$$

where the value of fraction ($f \leq 1$) is dependent on the permissible variation of data density around the scanner and is computed from the restricted angular span ($n\alpha \leq \theta_l$) for a specific position of the scanner. An optimum value ensures that the projected geometry of the reference elements is clearly identifiable in the scanned scene for effective registration of scans. The fraction value is defined as

$$f = \frac{\sin(\alpha)}{\cos(n\alpha)\cos((n-1)\alpha)} \qquad (3.5)$$

where $n\alpha \leq \theta_l$ and it has been seen in the experiments that the reference elements of diameter of 0.15m lying within a restricted incident angle ($\theta_l \leq 55°$) at angular scanning resolution of $\alpha = 0.072°$ are accurately captured and generate good results during the data registration from the multiple scans. The maximum consistent scanning range L_n with this density condition is computed as follows:

$$L_n^2 = fD^2 \frac{N}{2\pi} \qquad (3.6)$$

The scanning range can be computed by solving Eqns. 3.5 and 3.6 simultaneously with the restricted incident angle θ_l. The scanning range computed in this way ensures that each scan data sets include selectable reference elements, which are identifiable from multiple scans for data registration. The selection of fraction value (f) is dependent on the size of the reference elements and their accessibility in the captured data.

Case III: $d_{alw} = m_d d_1$

This restrictive data density condition ensures that the variation in the captured data is coherent for given hardware parameters. This condition is applicable when the time available for range scanning is limited and the captured data set is used to extract basic shape information such as layout and cross-section. Some such application scenarios include busy pathway, high traffic tunnel and subways. In this case, it is essential to control the overall variation in data density and thus, the permissible data density distance (d_{alw}) is defined as a multiple of minimum data density distance (d_1) permitted by the scanning time constraint i.e. $d_{alw} = m_d x_1 = m_d d_1$; where $m_d \geq 1$). This option is used where sufficient density variation is permitted and a multiple value (m_d) can quickly provide the least data density distance captured at the far end. A large multiple value of about 50-75 can help in capturing major dimensional distances and approximated layouts. The consistent scanning range (L_n) for this density condition is computed from the relationship defined in Eqn. (A.4) of Appendix-A:

$$L_n^2 = \frac{m_d D^2}{\cos\alpha} \tag{3.7}$$

The consistent scanning range is mainly dependent upon the scanner's distance D and multiple factor (m_d) for a specific number of points N. However, the range does not change appreciably with the change in the number of points because the denominator, $\cos(\alpha)$, is almost constant (≈ 1) for a given span of scanning resolution angle α (0.009° to 0.09°).

3.4.2 Computing permissible scanning range

The consistent scanning range (L_n) is selected based on the specific requirements of the application. The whole world domain can be scanned using multiple scans and registered with sufficient overlap (minimum 3-4 common reference elements for consecutive scans) as shown in Figure 3.6. The captured data within the consistent scanning range (L_n) is retained from each scanner in the registered set to ensure that the overall data density characteristics are achieved.
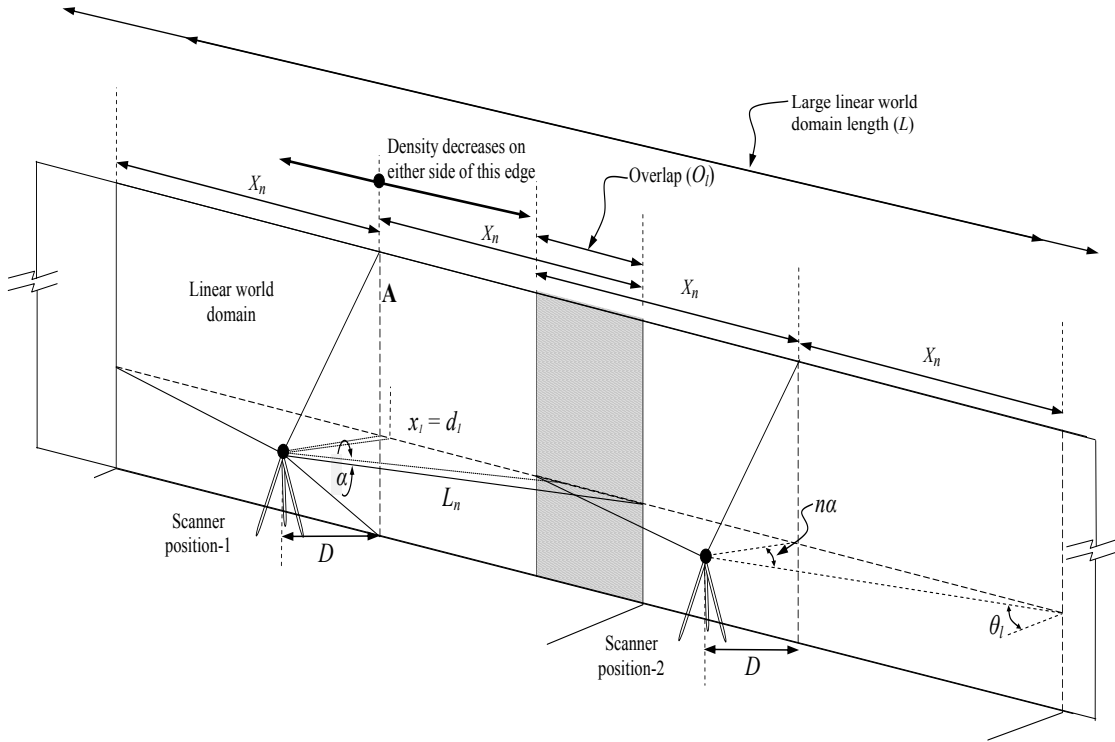
Figure 3.6:   Digitizing large linear domain with multiple stationary scanners.

Linear scanning range (X_n) defines the permissible domain length along the scanned surfaces that is captured with the consistent data density condition. This range (X_n) is computed from the consistent scanning range (L_n) and the corresponding scanner distance (D) as shown in Eqn. 3.8. Once X_n is computed, the total number of scans (n_{scans}) required to capture the linear world domain (L) can be computed as below:

$$X_n = \sqrt{L_n^2 - D^2} \text{ and } n_{scans} = \frac{(L - O_l)}{(2X_n - O_l)} \tag{3.8}$$

where O_l represents the domain overlap between two consecutive scanning ranges which can be determined based on the position and capturing accuracy of the reference elements. The above methodology is used to scan linear world domains with narrow pathways or restricted bottlenecks in the following section to demonstrate its effectiveness.

3.4.3　Captured data density evaluation

The proposed scanning methodology attempts to generate consistent density scan data sets from slender domains with large length to width ratio. However, it is important to evaluate the effectiveness of the captured data set in terms of the actual data density. The actual captured data density distance (d_{nc}), from the real linear domain is measured and compared with its corresponding theoretical value (d_n) to prove the quantitative effectiveness of the proposed methodology.

The data density distance, captured at the nth point is measured as the average distance of its nearest spatial points. The captured data set is expected to form a rectangular grid of 3D coordinate points on the scanned surface, where each data point is surrounded by 8 spatial points as shown in Figure 3.7. Thus, the captured data density distance can be computed from 4 or 8 nearest points of this rectangular grid. The corresponding captured data density distances (d_{nc4} and d_{nc8}) at few sampled data points along the scanned domain are measured using N-nearest point search and compared with the expected data density distances at those points computed analytically.
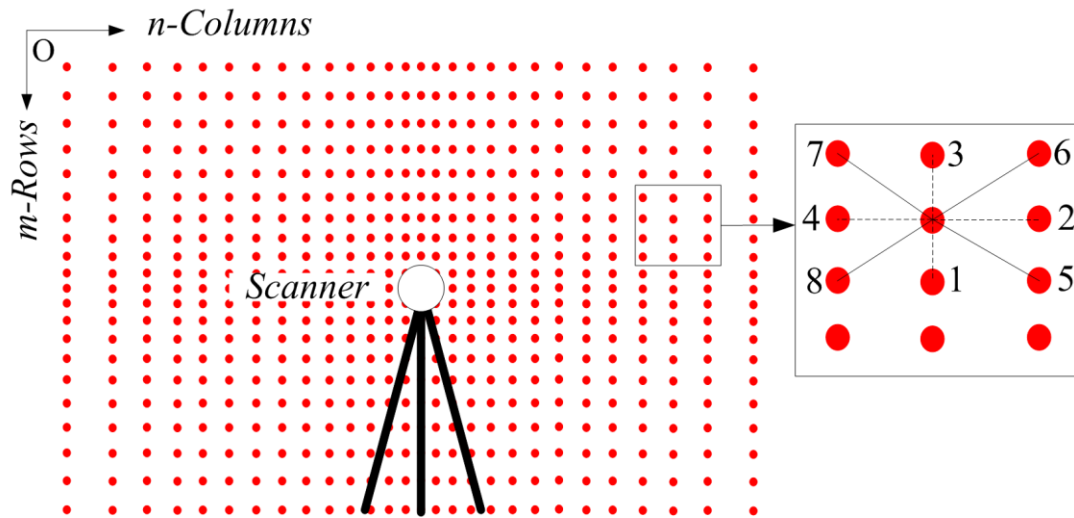
Figure 3.7:　Expected point cloud distribution over a linear domain using stationary scanner.

3.5   Scanning Results and Discussions

The proposed scanning methodology is first demonstrated on a 46m long interior hallway. This test environment largely contains long, relatively flat walls, an open floor, ceiling pipes, and doors. The domain is captured with multiple scans at various desired data density conditions using a stationary range scanner FARO-LS880 (specifications in Table 3.1). Only one side of this long corridor is shown in the results for clear visualization of captured shape. The methodology is demonstrated on large building corridors (50m $\times$ 70m) with very narrow corridors (D < 3m), an industrial unit (75m $\times$ 45m) with restricted pathways, and an exterior facet (35m long) with slender walkway.

Table 3.1: Technical specifications of FARO® laser scanner (LS 880).

Parameters	Values
Range (L_n^{max})	76.7 m
Minimum Range (L_n^{min})	0.6 m
Linear Error at 25m. (84% reflectivity)	3 mm
Max. Resolution (points)	40000 x 17224
Measurement Rate	120000 pixel/sec
Angular Scanning Resolution (α)	0.009°-0.09°
Scanning Time (max. to min.)	1:49:12 to 0:01:05
Laser Power	10.5 mW
Wave Length	785 nm
Vertical Angular Span ($\angle_{V\max}$)	320°
Horizontal Angular Span ($\angle_{H\max}$)	360°

In the first case, Figure 3.8 (a) shows a fish-eye view of the scanned space and its corresponding spatial data set (Figure 3.8(b)) captured with a single stationary scanner.

The scanner captures a dense point data set from the closest region on the domain (region **A**) with clear, identifiable features. The acquired density of data is clearly reduced as the regions become distant from the scanner. The regions in Figure 3.8(b) exhibit the captured data with typical scanning characteristics such as missing data due to surface

occlusions or reflective properties. Consequently, surface details are lost in these regions (regions **B** and **C**), especially due to occlusions from the vertical hanging flags on the left side of the domain (region **C**). Thus, a single scan cannot effectively capture such narrow, long corridors and multiple scanning is required.

(a) The scanned space

(b) The scanned point cloud data
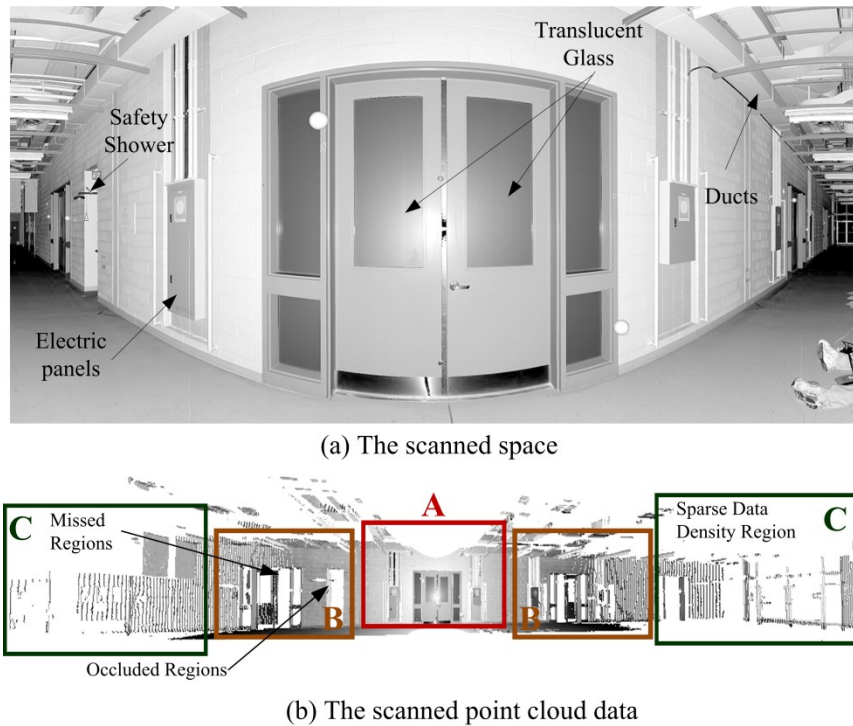
Figure 3.8:   (a) The scanned domain elevation and its corresponding (b) Scan data with a single stationary scanner.

3.5.1   Multiple scan planning

The proposed consistent density-based multiple scan plan methodology is used to capture this long, narrow corridor. A restrictive density value (d_{alw}) is used to compute the permissible scanning range (L_n or X_n), which, in turn, is used to compute the number of scans to capture the whole domain. The final scanning parameters, including the allowable density conditions, partial scanning ranges, number of scans and the expected data density distances are compiled in Table 3.2. It is to be noted that the data densities are determined using three possible cases, however, the first case is used to validate the models and then other two cases are demonstrated on different domain applications in Section 3.5.4.

Table 3.2: Experimental parameters for multiple scanning with allowable data density distances.

Case I: d_{alw} = Const						Case II: d_{alw} = fD						Case III: d_{alw} = $m_d d_1$ = $m_d x_1$					
d_{alw} (cm)	Scanning Parameters	X_n (m)	n_{scans}	Expected d_n(cm)		f	Scanning Parameters	X_n (m)	n_{scans}	Expected d_n(cm)		m_d	Scanning Parameters	X_n (m)	n_{scans}	Expected d_n(cm)	
--	$n\alpha$ = 83°	23	1	Min.	0.355	--	$n\alpha$ = 83°	23	1	Min.	0.355	--	$n\alpha$ = 83.01°	23	1	Min.	0.355
	n = 1153			Max.	23.685		n = 1153			Max.	23.685		n = 1170			Max.	23.685
11.5	$n\alpha$ = 79.35°	15	2	Min.	0.355	0.018	$n\alpha$ = 65°	5.8	6	Min.	0.355	100	$n\alpha$ = 84.26°	27	1	Min.	0.355
	n = 1102			Max.	11.41		n = 902			Max.	5.07		n = 1158			Max.	35.55
5.3	$n\alpha$ = 72.6°	9	3	Min.	0.355	0.007	$n\alpha$ = 55°	3.8	9	Min.	0.355	75	$n\alpha$ = 83.37°	23	1	Min.	0.355
	n = 1008			Max.	4.95		n = 764			Max.	1.91		n = 1136			Max.	26.625
3	$n\alpha$ = 68.05°	7	5	Min.	0.355	0.004	$n\alpha$ = 45°	2.7	10	Min.	0.355	50	$n\alpha$ = 81.82°	19	2	Min.	0.355
	n = 945			Max.	2.93		n = 625			Max.	1.037		n = 1090			Max.	17.75
2	$n\alpha$ = 54.81°	4	8	Min.	0.355	0.0025	$n\alpha$ = 35°	1.9	19	Min.	0.355	25	$n\alpha$ = 78.46°	13	3	Min.	0.355
	n = 731			Max.	1.996		n = 486			Max.	0.6827		n = 994			Max.	8.875

The feature based allowable density condition (Case I) is used to demonstrate the steps of the proposed methodology. Eight spheres (S_{L1}-S_{L8}, S_{R1}-S_{R8}) and twelve contrast circles (C_{L1}-C_{L12}, C_{R1}-C_{R12}) of 0.150m and 0.200m diameters respectively are used as reference elements on each wall of the corridor for subsequent registration as shown in Figure 3.9.

(a) Top view of scanned domain and location of reference spheres on both walls

(b) Front view of scanned domain and location of reference circles

S_{Li} – i^{th} Sphere on left wall; S_{Ri} – i^{th} Sphere on right wall; C_{Li} – i^{th} Circle on left wall C_{Ri} – i^{th} Circle on right wall
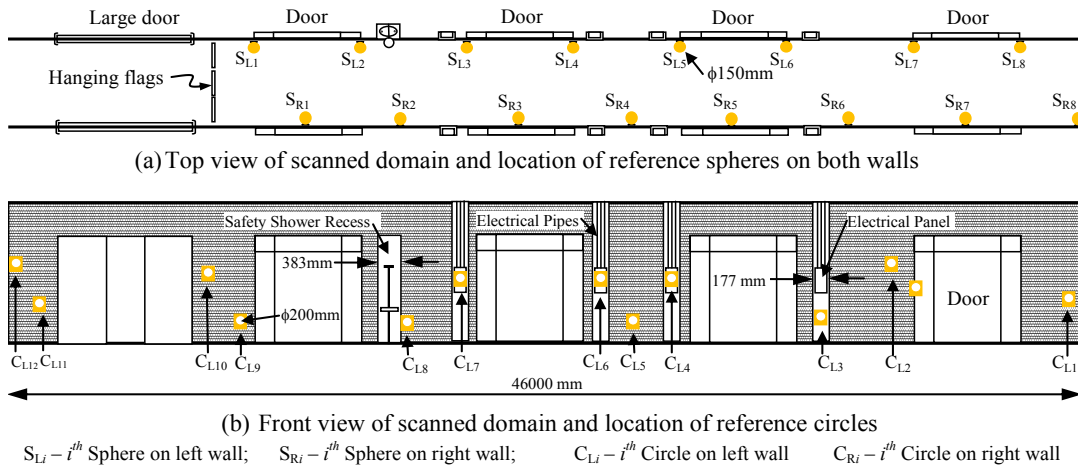
Figure 3.9: The locations of the reference elements along the scanned scene

Although the contrast circles are shown only on one side of the wall, similar contrast circles and spheres are added on the other wall. The scanner distance D = 2.82m is used due to the limited availability of corridor space. The domain is scanned with different values of d_{alw} and the corresponding parameters and the results are presented in the following section.

3.5.2 Scan results of narrow pathway

The narrow pathway (Figure 3.8(a)) is scanned using multi-scan methodology and each scanned data set is pre-processed to remove the outliers in an effort to minimize the cumulative scanning errors. These multiple scanned data sets are then merged and data away from cut-off regions is removed and consistent scanning ranges are retained. For the first case ($d_{alw}=const.$), the hallway is scanned with d_{alw}=11.5 cm (ξ=0.3, feature size=38.3cm) using two scans. Each scanner position captures the whole world domain (Figure 3.10(a) and (b)) with large density variations, which are registered using the common reference elements (S_{L3}, S_{L4}, S_{L5}, S_{R3}, S_{R4}, S_{R5}, C_{L4}, C_{L6}, C_{R4}) from both the walls as shown in Figure 3.10(c).

Although this combined data set exhibits a restricted data density distance of 11.5cm at the far-away regions, it is not sufficient to accurately capture the reference elements and surface geometry in those regions. However, the nearby reference elements (S_{L3}, S_{L4}, S_{L5}, S_{R3}, S_{R4}, S_{R5}, C_{L4}, C_{L6}, C_{R4}) from both the walls are sufficient to register the data accurately. The same world domain is scanned with d_{alw} = 5.3cm (ξ=0.3, feature size = 17.7cm) and 3cm (ξ=0.3, feature size = 10cm) to capture electric cabinets and pipes using 3 and 5 scans, respectively, as shown in Figure 3.11(a) and (b).

It is evident from Figure 3.11(a) that the feature visualization and shape capturing capability improves with an increase in the number of scans. However, the geometric details of some of the intricate features in the scene such as the door frame recesses, electrical panels and pipes are missing. The lost data corresponds to the feature sizes smaller than the permissible scan density distance (d_{alw}). These minute details in the scanned scene are captured by further reducing the allowable scanning data density distance (d_{alw}) to 3cm and scanning the whole domain with five scans as shown in Figure 3.11(b). Only few regions are not captured effectively due to the typical surface characteristics of the individual features such as translucent glass panes, which permit the scanning rays to pass partially through it and cannot be captured. Other sources of error correspond to occluded regions and materials having poor reflective properties. Accurately registered, dense scanned data from multiple partial scans can capture greater surface details for better visualization and subsequent geometric extraction of small features such as safety shower recess and electric panels (Figure 3.11 (b)).

(a) Captured point cloud data with 1st scanner position
and identified references (S_{L3}, S_{L4}, S_{L5}, S_{L6}, S_{L7}, S_{R3}, S_{R4}, S_{R5}, S_{R6}, S_{R7}, C_{L3}, C_{L4}, C_{L5}, C_{L6}, C_{R2}, C_{R3}, C_{R4})

(b) Captured point cloud data with 2nd scanner position
and identified references (S_{L2}, S_{L3}, S_{L4}, S_{L5}, S_{R1}, S_{R2}, S_{R3}, S_{R4}, S_{R5}, C_{L4}, C_{L5}, C_{L6}, C_{L7}, C_{L8}, C_{R4}, C_{R5})

(c) Registered point cloud data using both the scans
using common reference elements (S_{L3}, S_{L4}, S_{L5}, S_{R3}, S_{R4}, S_{R5}, C_{L4}, C_{L6}, C_{R4})

Figure 3.10: Multiple scan registration for reducing overall data density variation.

(a)

(b)
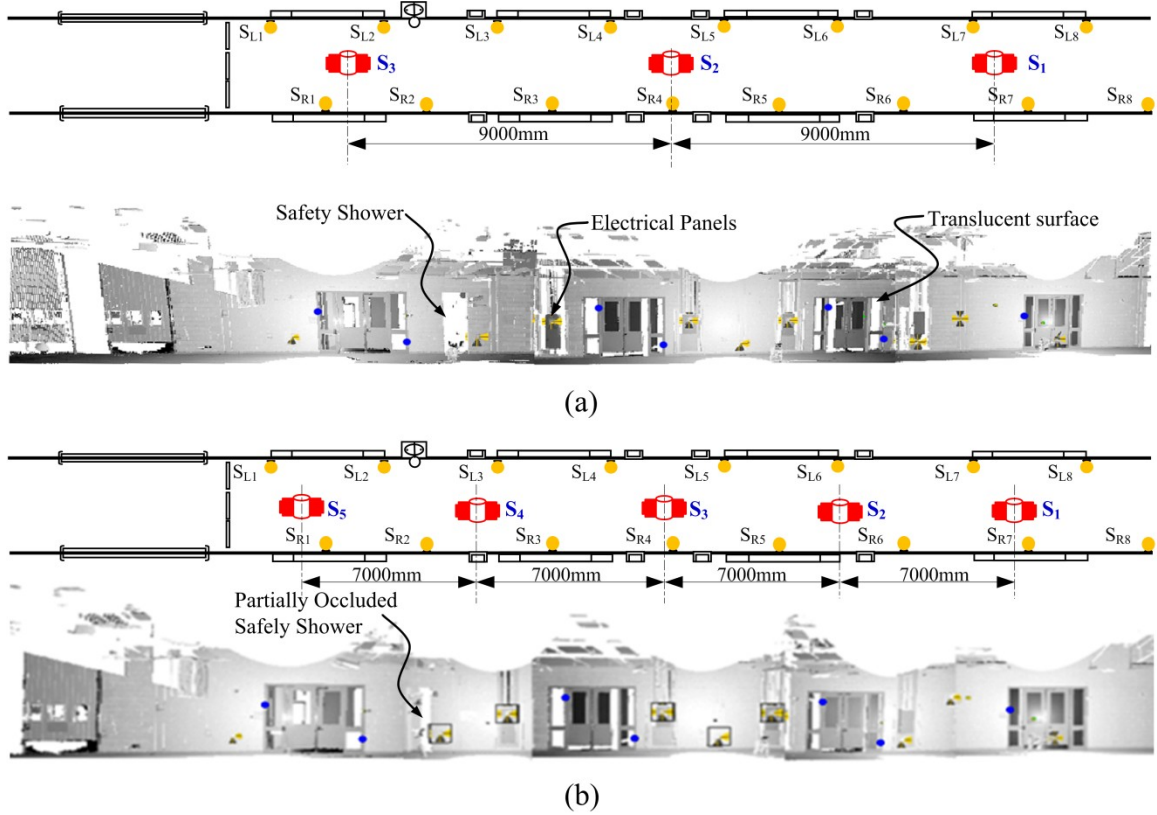
Figure 3.11: The registered scanned data set from multiple scans with (a) d_{alw} =5.3cm. (3 scanning position) and (b) d_{alw} = 3cm. (5 scanning position).

Furthermore, merging several scans with inadequate or poorly captured reference elements leads to registration inaccuracies. The consistent scanned data in Figure 3.12 (a) is obtained by registering eight very small scans with each having its scan range of about 4m. The surface visibility of the registered scan has deteriorated from the previous illustration and the features cannot be clearly identified due to added secondary noise. For example, the depth recesses of the shower and electrical panels are filled with noisy data set, which makes it difficult to visualize these features. When a minimum of three reference elements cannot be captured by consecutive scans, a better approach to generate high density data sets is to increase the scanning resolution instead of increasing the number of scans. Another important observation is that the shape capturing capability of the occluded regions is mainly dependent on the location of the scanner. Such hidden regions can be effectively captured by placing the scanner close to the occluded region. Thus, the same domain can be scanned with 5 scanner

positions and the safety shower data can be captured with additional scanner position in front of the safety shower and merged with the original scanned data as shown in Figure 3.12 (b).

(a)

(b)
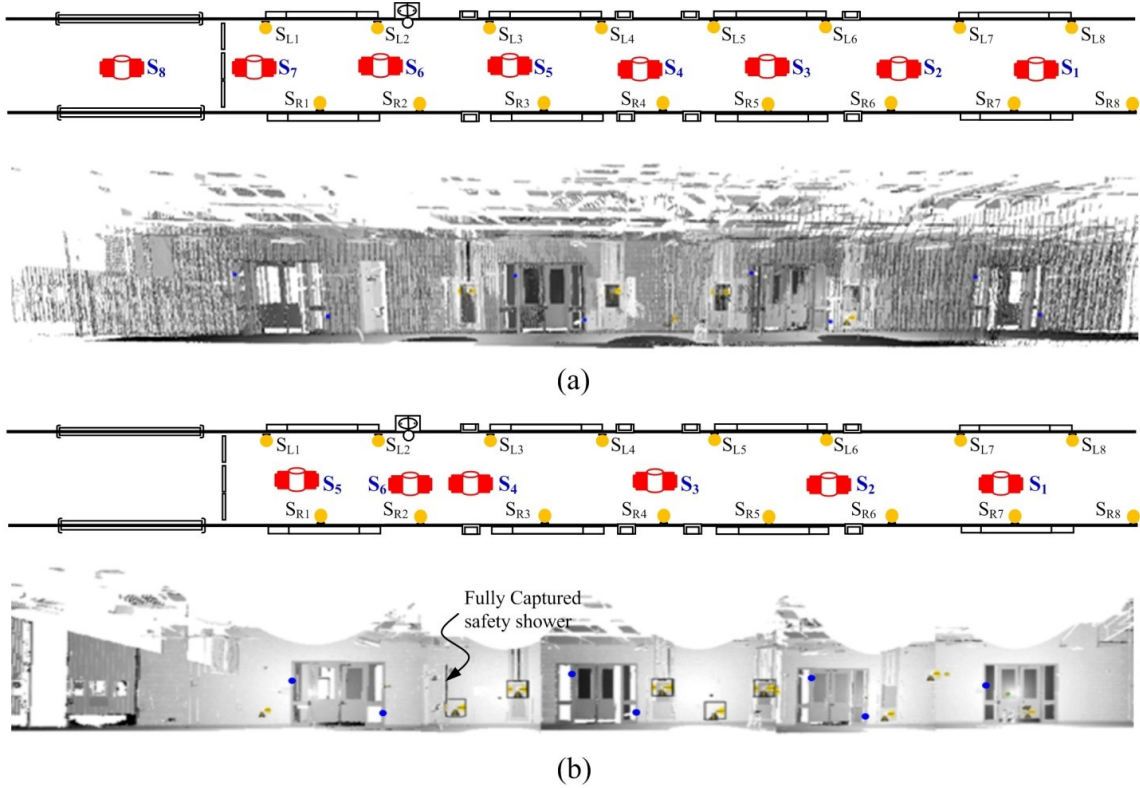
Figure 3.12: The registered scanned data from multiple scans with (a) d_{alw} = 2cm (8 scanning positions) and (b) d_{alw} =3cm. (5+1 scans for capturing desired feature (safety shower).

3.5.3   Scan data density evaluation

The actual captured data density distance (d_{nc}) is measured and compared with the theoretical value (d_n) to access the quantitative evaluation of the proposed methodology. The data density distance, captured at the nth point is measured as the average distance of its nearest spatial points. The captured data set is expected to form a rectangular grid of 3D coordinate points on the scanned surface, in which, each data point is surrounded by 8 spatial points. Thus, the captured data density distance can be computed from 4 or 8 nearest points of this rectangular grid. The corresponding captured data density distances (d_{nc4} and d_{nc8}) at few sampled data points along the scanned domain are measured using N-nearest point search. These average densities distances (d_{nc4} and d_{nc8}) are plotted against the expected theoretical

value (d_n) for all the four scanning scenarios presented in Section 3.5.2 as shown in Figure 3.13.
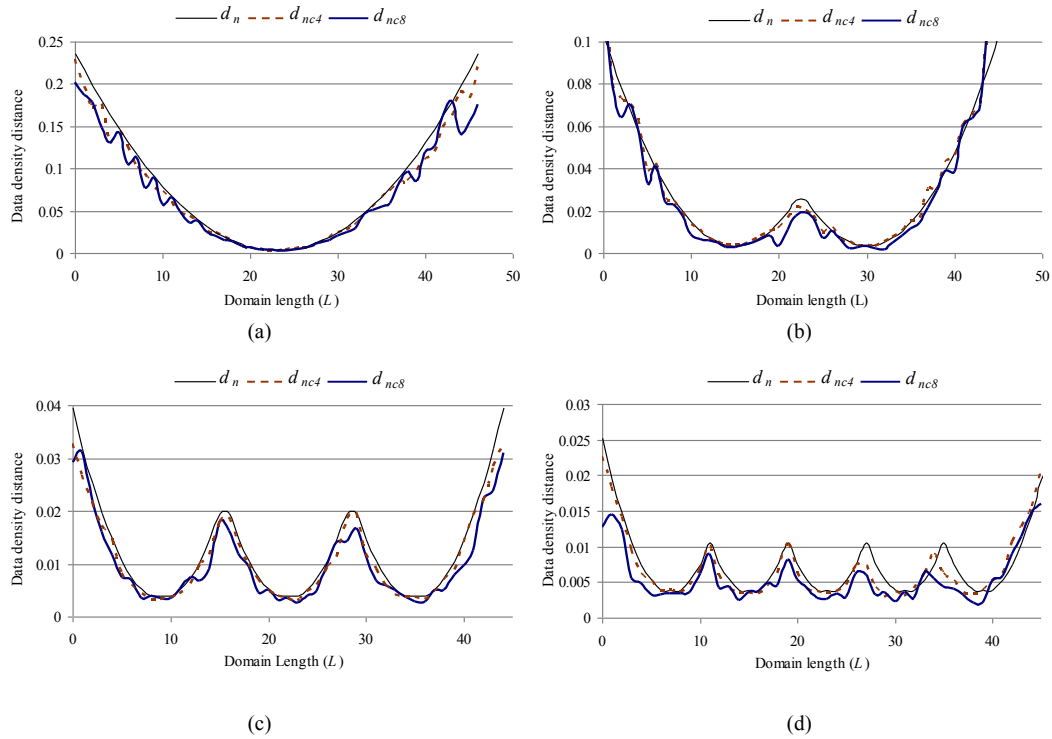
Figure 3.13: A comparison of captured data density distances (d_{nc4} and d_{nc8}) with the theoretical (d_n) value along the scanning domain (L) using (a) one (b) two (c) three and (d) five scans.

It can be seen from these graphs (Figure 3.13(a-d)) that the captured data density distances (d_{nc4} and d_{nc8}) comply well with the theoretical values (d_n), especially in the scanner's vicinity. This is due to the fact that this region is densely captured and is least affected by the typical scanning errors with small incident angle. Due to lesser overlap, the registration noise in this region is also minimal (Figure 3.13(b)). The small variations in the captured and theoretical values are due to the secondary data noise introduced during scan registration and the surface characteristics of that region. The data density variations at far off distances from the scanner are relatively large due to large incident angle in those areas. The overall density variation in the captured data reduces with the increase in the number of partial scans (Figure 3.13(c) and (d)). The data density distance (d_{nc}) in the overlapped region is normally less than its corresponding theoretical value (d_n), since the captured density in this region corresponds to the points from two consecutive scan data sets, thus a greater number of points per unit area are available and that decreases the average data density

distance. In general, the captured data density distance conforms well to its theoretical value and thus, prove the effectiveness of the proposed methodology in achieving a desired value.

3.5.4 Real world domain scanning examples

The proposed methodology is implemented on the corridors of a large building structure (50m × 70m), interiors of an industrial domain (75m × 45m) with very tiny pathways and a 35m long exterior facet of a building as shown in Figure 3.14.

<div align="center">(a)   (b)   (c)</div>
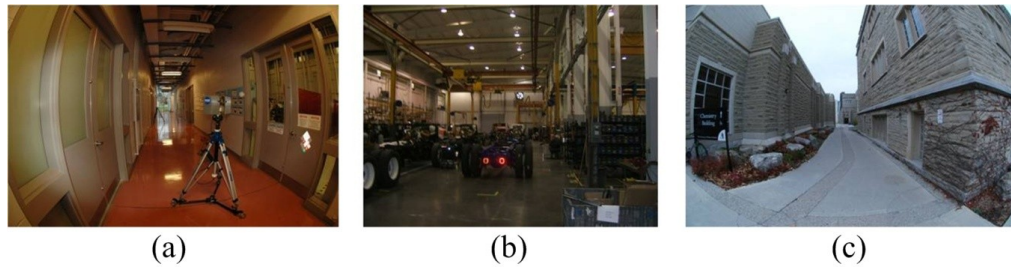
Figure 3.14: Real world domains including (a) Narrow interior corridors (b) Industrial domain (c) Exterior facet.

The scanning complexity of the large interior pathways is due to its narrow domain with large length/width ratio and varying feature sizes in the captured data. An effective scan plan is governed by the subsequent use of the captured data. An application may require to capture specific feature size (Case-I: $d_{alw} = \xi \bullet$ *feature size*), merge multiple scans accurately (Case-II: $d_{alw} = fD$) or extract basic shape outline (Case-III: $d_{alw} = m_d x_1$). The challenge is not only to place scanner positions at its computed locations, adjust for the presence of multiple objects in the scanned scene but also to identify common reference elements from different scanning positions for effective registrations and thus asks for limited angular scanning span. The third example of scanning an exterior facet tests the proposed approach for applications with complex features and evaluates its registration accuracy.

The building corridors are scanned with $d_{alw} = 5.3$cm where $X_n = 9$m, which required 14 scans to cover the total length of about 220 m of these corridors. Thirty small reference spheres (diameter = 0.150m), 40 contrast circles and 4 extra large reference spheres (diameter = 0.3m at intersections) were used to register these multiple scans to capture the data with desired data density. Figure 3.15(a) shows the top view of the raw registered data set from all these 14 scans. In Figure 3.15(b), the same data set is shown after removing the floor data

points to improve the visual clarity. It is evident from these figures that a highly dense data set can be generated with consistent data density throughout the domain using the proposed methodology. The same domain is scanned with the $m_d = 75$(Case III), with four scans. The registered results of raw data are compiled in Figure 3.16(a) and refined results are presented in Figure 3.16(b). In this case, this data set is not very dense, but is good enough for extracting basic geometric information such as section layouts with minimum scanning and subsequent data processing.

(a)   (b)
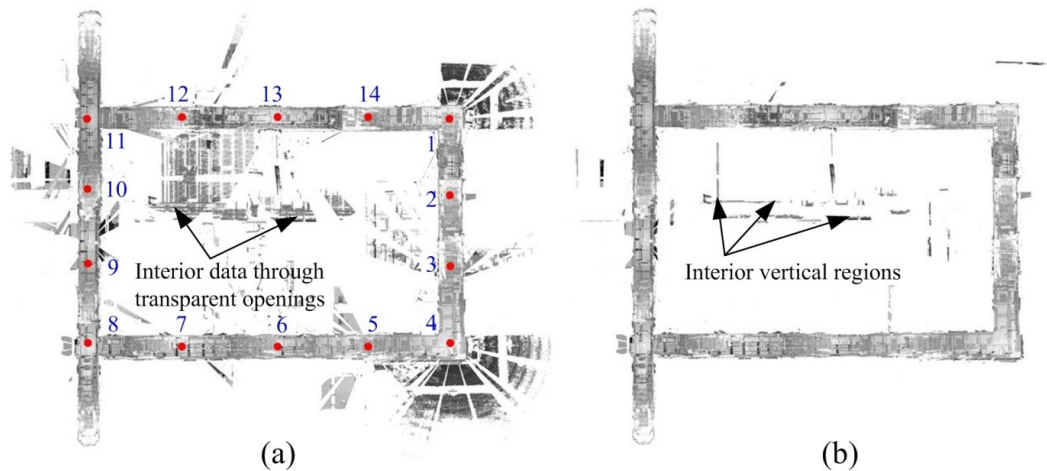
Figure 3.15: Narrow interior corridors scanning results (top view) of large building using 14 scans for d_{alw} = 5.3cm. (a) Raw registered data and (b) Registered data without the floor data points.

(a)   (b)
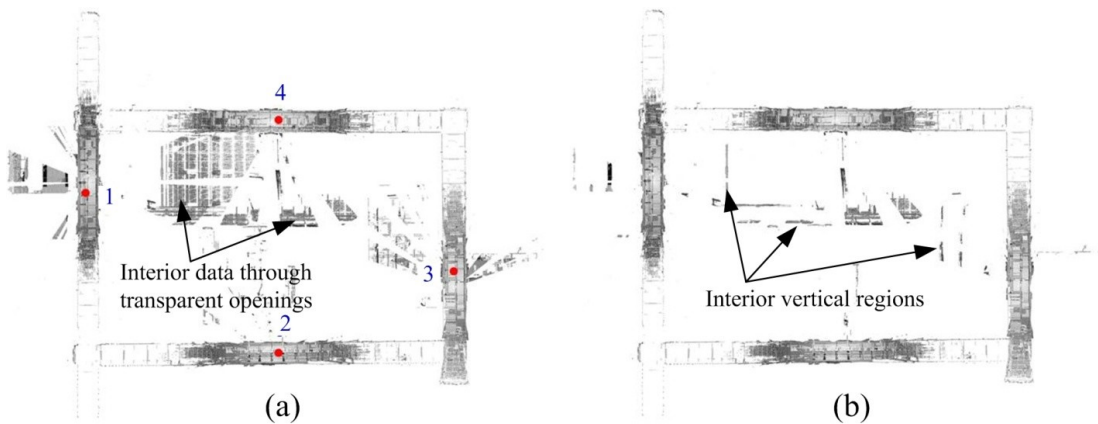
Figure 3.16: Narrow interior corridors scanning results (top view) of large building using 4 scans for d_{alw} = $m_d d_1$ (m_d = 75) (a) Raw registered data and (b) Registered data without the floor data points.

The Case-II ($d_{alw} = fD$ with $f = 0.007$, $n\alpha = 55°$) is used in an industrial domain (75m × 45m) having very tiny pathway accessibility. The total area was covered by scanning 1:5

resolution scans at 37 different positions throughout the domain. The proposed scan planning methodology required 32 scanner positions and additional 5 scans were used to capture occluded regions. Thirteen reference spheres and 52 contrast circles were used in the registration process. Here, a relatively small number of reference elements are used as multiple reference elements are accessible from multiple scanner positions due to open interior environment. The approximate distance between consecutive scanner positions was computed to be about 10m. The scanner was placed at these computed distances, subjected to the space availability. Using a stationary scanner is especially useful in this case as it can be placed at any position, which may be otherwise inaccessible to a mobile scanner. Figure 3.17(a) shows the slice of the scanned data showing the position of the scanners and the Figure 3.17(b) shows the registered data set with a partitioning plane. The data set on the right side of the plane shows the interior registered data, which is obtained by removing the ceiling and floor data points in this region. The noise-free interior data and matching region boundaries in Figure 3.17(b) indicate that accurate registration was achieved from multiple scans. Further, the sliced data (Figure 3.17(a)) shows that accurate layouts can be effectively extracted from the registered scanned data. The scanning issues due to the near side of the objects are not encountered as the minimum distance between the scanner and the scanned surfaces was more than the minimum range (0.6m).

(a)   (b)
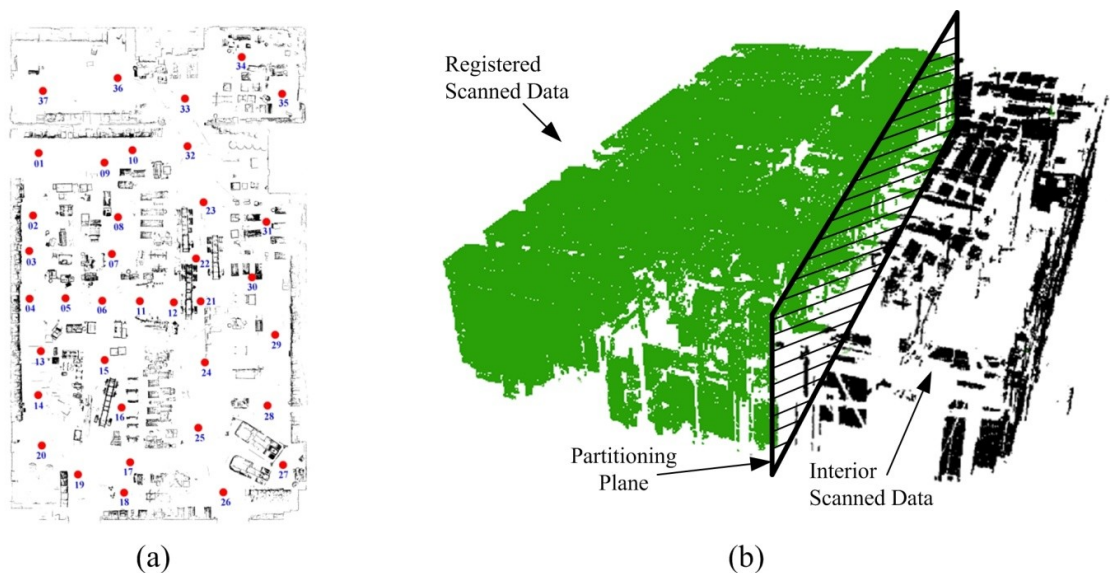
Figure 3.17: Interior industrial building scanning results with 37 scans (a) Location of the scanner in an industrial environment (b) Dense registered data with interior data showing the accuracy of multi-scan registration process.

Finally, the proposed methodology is demonstrated on exterior building facet to demonstrate its effectiveness in capturing surface details. An exterior facet length of about 35m is scanned with d_{alw} = 3cm and D = 2.5m at 1:5 resolution. The whole domain was scanned with 5 distinct positions of the scanner with $X_n \approx$ 7m. The data sets are captured and are registered from these five scanner positions. Specific data sections from the cumulative registered data set are reconstructed as shown in Figure 3.18. It is clear from this Figure 3.18 that a consistent density data set captured by the proposed methodology can be effectively used to reconstruct the geometric shapes of the captured surfaces. In the reconstructed parts from the point clouds, the details of the stone work on the wall, windows geometry and related features are accurately captured and reconstructed. The specific details of small features such as the door switch, stone work on the wall, and bench geometry are also captured effectively. Thus, the proposed methodology is capable of generating consistent density data sets by capturing and registering multiple scans using controlled data density variations. The captured features can be accurately reconstructed, if required.
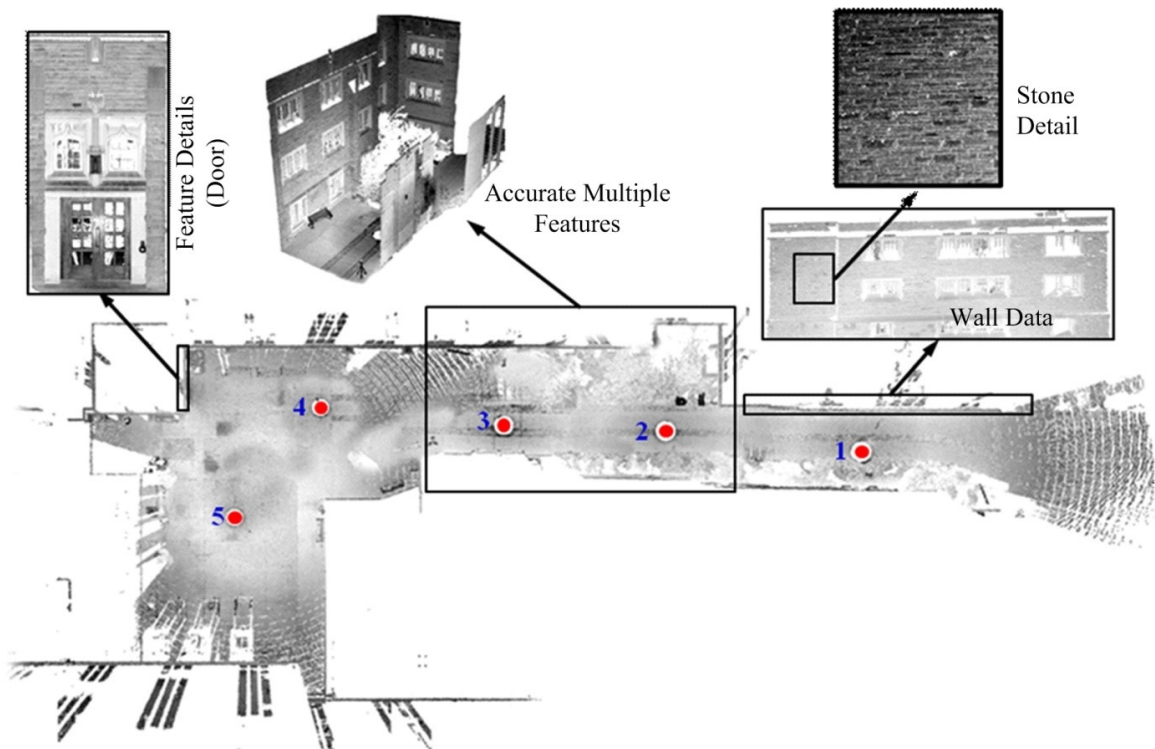
Figure 3.18: Exterior facet scanning results using 5 scans with d_{alw}=3cm.

3.6 Scanning Guidelines

The Following guidelines help in generating consistent point cloud data sets:

1. Place scanner in such a way that domain surfaces are equidistance from the scanner.
2. In case of slender domains, the surfaces in the immediate neighborhood of the scanner are only considered as those will be captured with the scanner.
3. The distance between adjacent scanner positions should conform to its computed value for a given data density condition.
4. It is beneficial to capture a large slender domain using multiple low resolution scans, provided the scans can be registered without introducing any secondary noise.
5. It is beneficial to use higher resolution and time consuming scans only if time is not a limiting factor and the computed scan ranges (L_n) becomes very small for a desired data density condition and introduces secondary noise.
6. The reference spheres should be placed in such a way that these are visible from multiple scanner positions.
7. The occluded regions can be minimized by scanning such a region directly using lowest incident angle by placing the scanner in its nearest vicinity.
8. The outliers and other scanning errors must be removed/minimized before such data is registered.

3.7 Concluding Remarks

The digitization process of a large domain using 3D range scanning is often affected by inconsistent data densities that lead to inaccuracies in shape visualization and specific information extraction. The effect of inconsistent densities is especially critical for applications involving large slender domains and stationary scanners.

In this chapter, factors responsible for inconsistent densities are identified and an effective, restrictive density-based scan planning methodology to capture accurate shape geometry from large real-world domains is presented. Three cases are proposed to select the restricted data density based on the desired feature size, scan registration constraints, and overall density distribution. The restricted density is used to compute the consistent scanning range for each scanner position. A dense point cloud is extracted by registering the scanned data and retaining the partial scanning ranges from multiple scans. An analytical

methodology is presented to formulate the mathematical relationships for computing the consistent scanning range for each stationary scanner for given density conditions. The effectiveness of the proposed scanning methodology is demonstrated on a long, narrow corridor by generating a consistent density data set, measured quantitatively. A low incident angle (θ_l) is necessary to capture and register multiple scans of narrow spaces with small scanner distance (D). The occluded regions can be effectively captured by placing the scanner in front of these regions and thereby reducing incident angle and corresponding surface occlusions. The scanning flaws corresponding to the surface characteristics can be partially improved by multiple scanning of a given domain. Finally, this systematic scan planning methodology has been demonstrated on diverse real life application domains and effective scanning results have been achieved. The approach ensures that similar data density data sets are captured, even from long linear domains, where density variations pose complex challenges to accurate information extraction.

CHAPTER 4   DATA SIMPLIFICATION AND LAYOUT EXTRACTION

4.1   Introduction

Range scanning of large buildings generates an immense point cloud of spatial data points, which are to be processed to compensate the scanning inconsistencies and generate functional information model by extracting useful information for subsequent applications. Although, it is helpful to minimize scanning inconsistencies at the scanning stage as described in the last chapter, it is still essential to develop specific post-processing tools that can convert the captured data sets into useful format in terms of specific geometries, features and individual data segments.

Application specific needs define the features/geometries that are to be extracted from its point cloud. Reverse engineering focuses on automatic or semi-automatic reconstruction strategies to generate geometric models through segmentation and surface modeling (Várady, Martin and Cox, 1997) of point clouds and has been extensively reported on single, isolated objects. A similar approach does not suffice for point cloud representing large domains involving multiple objects, where the reconstructed model of the unified data set is topologically inconsistent and does not help in extracting desired geometries/features from unified point cloud. Specifically, the presence of multiple objects in the scanned scene is a deterrent to accurate information extraction. Hence, it is crucial to identify the correct points representing specific geometries, features or objects for improving the reliability of the extracted features, besides handling large size of the point cloud.

It is also important for the post processors to handle the large, unified point cloud of large, occupied building interiors. Data decimation is an important part of any feature extraction process, however in this case, it is essential to preserve specific features during the simplification process. Thus, a suitable set of post-processors must be designed to handle the unified point clouds of occupied building interiors and convert them into functional information models representing individual surfaces or feature segments. It is beneficial to use direct point based tools that can simplify and correlate spatial points to its respective feature for extracting objects of interest.

4.2 Building Features

It is essential to identify the type of features to formulate their corresponding extraction strategies by generating specific post-processing tools. Due to the vast possibilities of the features present in a typical interior scanning scene, it is not possible to devise a universal approach to data post-processing (decimation, segmentation or feature extraction) and hence adaptive approaches are to be devised for specific applications that conform to the desired characteristics of the feature extraction. Identification of these features mainly depend on the semantic level at which these features are to be extracted as shown in Figure 4.1.
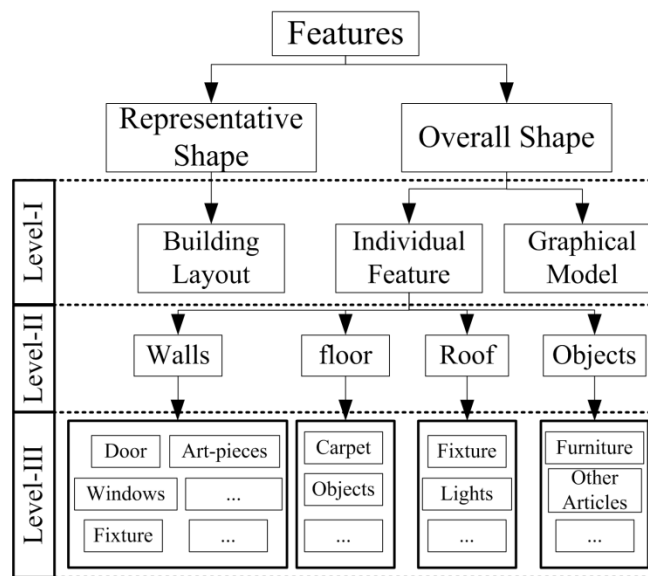
Figure 4.1:   Various levels of extractable features from point cloud of building interiors.

The first semantic level defines the features based on the kinds of shapes that need to be extracted from the scanned data (representative or overall shape). This can be further categorized into its constituents in the next level and that can be further subdivided into its own individual features, surfaces, edges and so on, as desired. Thus, different features can be extracted from the united point cloud based on the specific requirements of the given application. This chapter presents a representative shape extraction in terms of *data decimation and layout extraction* in an occupied building interior environment, which works directly on point cloud data. This two stage simplification and layout extraction algorithm extracts building layouts in a user defined direction by identifying and retaining the layout points and decimating others. It achieves point decimation by about 88-95% and the proposed

algorithm has diverse application. Subsequent chapters compile the developed algorithms for segmenting and discontinuous boundary detections that transform the captured point cloud into meaningful cluster and region discontinuity.

4.3   Related Work

Most of the existing simplification techniques have been extensively demonstrated on intermediate surface models or spatial data points of isolated objects. Pierre et al. (Pierre, Giuliana, Craig *et al.*, 2007) have recently compiled a detailed survey on the re-meshing of surfaces for graphical applications. However, extendibility of such simplification methods (Amenta, Choi, Dey *et al.*, 2000; Dey, Giesen, Leekha *et al.*, 2001; Hoppe, DeRose, Duchamp *et al.*, 1992; Schroeder, Zarge and Lorensen, 1992) is quite limited. Generating decimated polygonal models using an adaptive triangulation (Akkouche and Galin, 2001; Araújo and Jorge, 2005) of implicit surfaces is computationally expensive ($O(n^3)$).

Direct point cloud simplification(Chang and Chang, 2002; Lee, Woo and Suk, 2001) is an effective alternative for direct simplification of point clouds that attempts to reduce the number of data points without generating any intermediate surface. Pauly *et al*. (Pauly, Gross and Kobbalt, 2002; Pauly, Kobbalt and Gross, 2004) modified the mesh-based simplification theory to achieve point-based simplification for point-based graphics and multi-resolution surface modeling using the moving least square (MLS) method and iterative simplification process. This method re-computes the MLS surface after each point removal and is computationally expensive. However, an idea of removing trivial points without removing points corresponding to desired feature (boundary) by Song and Feng (Song and Feng, 2008) can be exploited to reduce data points without losing feature (layout). The applicability of direct point cloud simplification method is advantageous due to the availability of visualization tools such as pointshop3D (Zwicker, Pauly, Knoll *et al.*, 2003).

Layout extraction on the other hand was initially inspired by contour extraction in rapid prototyping that have been used extensively(Lee and Woo, 2000; Park, Chang and Park, 2007; Shin, Park and Park, 2004) to reduce the total reverse engineering time (Chen, Ng and Wang, 1999) of isolated object. Automated layout extraction from 3D buildings is a relatively new research area and relied on 2D aerial images to compute the building footprints as layouts. Suveg and Vosselman (Suveg and Vosselman, 2002) presented a simple

automatic 3D building reconstruction methodology based on a knowledge-based system and combining the basic primitives (flat, gable, and hip roof building) extracted from aerial images. An attempt to combine images with sparse point clouds was presented (Hongjian and Shiqiang, 2006) using the Laplacian sharpening approach, where a threshold segmentation approach is used to extract 2D edges from images and the building height is determined from sparse points. Another improved approach of building feature extraction from terrestrial laser data (Pu and Vosselman, 2006) uses the segmentation approach with different feature constraints based on the data segments, which may not be suitable for domains with no predefined constraints. Su *et al*. (Su, Hashash and Liu, 2006) presented a basic framework using 3D scanning in geotechnical monitoring of an urban excavation through planning, execution, and data processing phases of collecting accurate construction information with an unprecedented level of detail on the as-built site conditions. These approaches have only been demonstrated on external shapes to extract building footprints or exterior layouts.

Very few algorithms have been demonstrated on layout extraction from interior point clouds. A feasibility study (Shih and Hu, 2007) that emphasizes the usability of 3D range data in interior design applications is presented, where the captured data is utilized in visual inspection, design modeling, production drawing, construction monitoring and follow-up interior maintenance and management. Tseng *et al*. (Tseng, Tanaka and Leeladharan, 2002) attempted to extract internal profiles using a laser-based approach. This approach requires a dedicated controller to guide the laser for estimating the profile shape of the scanned structure. The information extraction from laser scanned data has been recently addressed (Kim, Lee, Kang *et al.*, 2008), where authors have attempted to extract geometric information on a highway using terrestrial scanned data. The study has been demonstrated on pavement design, and highway geometry extraction using local geometric parameters such as normal and data density. Most of these methodologies and frameworks are demonstrated on the terrestrial scanned data and their usability for building interiors is limited. The difficulties inherent in the feature reconstruction of interiors are different from the problems that come from the extraction of external shapes of buildings, thus they have to be treated differently. This is also the reason why model reconstruction and visualization of generic indoor scenarios is still a difficult task (Furukawa, Curless, Seitz *et al.*, 2009).

Floor plan extraction from point cloud normally relies on the identification of planar surfaces from scan data sets (Budroni and Böhm, 2009; Hähnel, Burgard and Thrun, 2003;

Johnston and Zakhor, 2008), Hough transforms (Tarsha-Kurdi, Landes and Grussenmeyer, 2007) or probability model (Thrun, Martin, Liu *et al.*, 2004). Most of these methods extract section layouts or floor plans as a by-product of the simplified model reconstruction process. These methods are intended to solve a more complex problem than floor plan extraction and tend to be computationally demanding, due to the large size of the cumulative data set representing multiple objects. This brings out the scope of the proposed work that intends to devise post-processors that are directly applicable on the point data sets and simplify them while preserving and extracting the desired feature.

4.4   Data Decimation and Layout Extraction Algorithm

The proposed algorithm decimates the data sets while retaining the data points representing individual features and the basic layout. An important characteristic of this approach is that it exploits the regions representing layouts using a measure of planarity and trivial (non-layout) point decimation.

The algorithm works in two distinct stages. In the first stage, the data set is sliced along the simplification direction, which can be computed as a normal of the plane fitted to the floor point cloud or a user defined value. In the second stage, the sliced data points are decimated by retaining the feature points and decimating others and thereby improving visibility of the desired features (individual features and section layouts). The data reduction is achieved till the interior points are minimized to the desired level and the layout points are clearly distinguishable. The following paragraphs define the related terminology and elaborate the proposed algorithm. An overview of the algorithm is shown in Figure 4.2.

The algorithm attempts to access the shape of the scanned object by extracting 2D section layouts directly from its 3D point cloud data. The challenge here is to identify the points representing the layouts. Thus, the points corresponding to other features are identified and adaptively simplified before the section contours are generated. The novelty of the proposed algorithm lies in the fact that it works directly on points, does not require any intermediate surface model and it handles point data involving multiple objects. The overall process requires the user to define certain key terms to achieve the desired results. These key terms include a vector direction ($\hat{n}$), number of simplifying planes (N_p), no. of neighbouring points (K_r), average angular deviation threshold ($\Psi_{thresold.}$) of neighbouring distance vectors

from a user defined vector ($\hat{n}$) and the close vicinity region thickness (t_v). Other terms which are computed indirectly, based on these user defined values, are the overall data size defined by point cloud data height (*H*), reduction ratio (*r*), slice thickness (*t*), and the point's importance in terms of the average angular deviation value ($\Psi_{avg.}$). In the following paragraphs, the overall process is described.
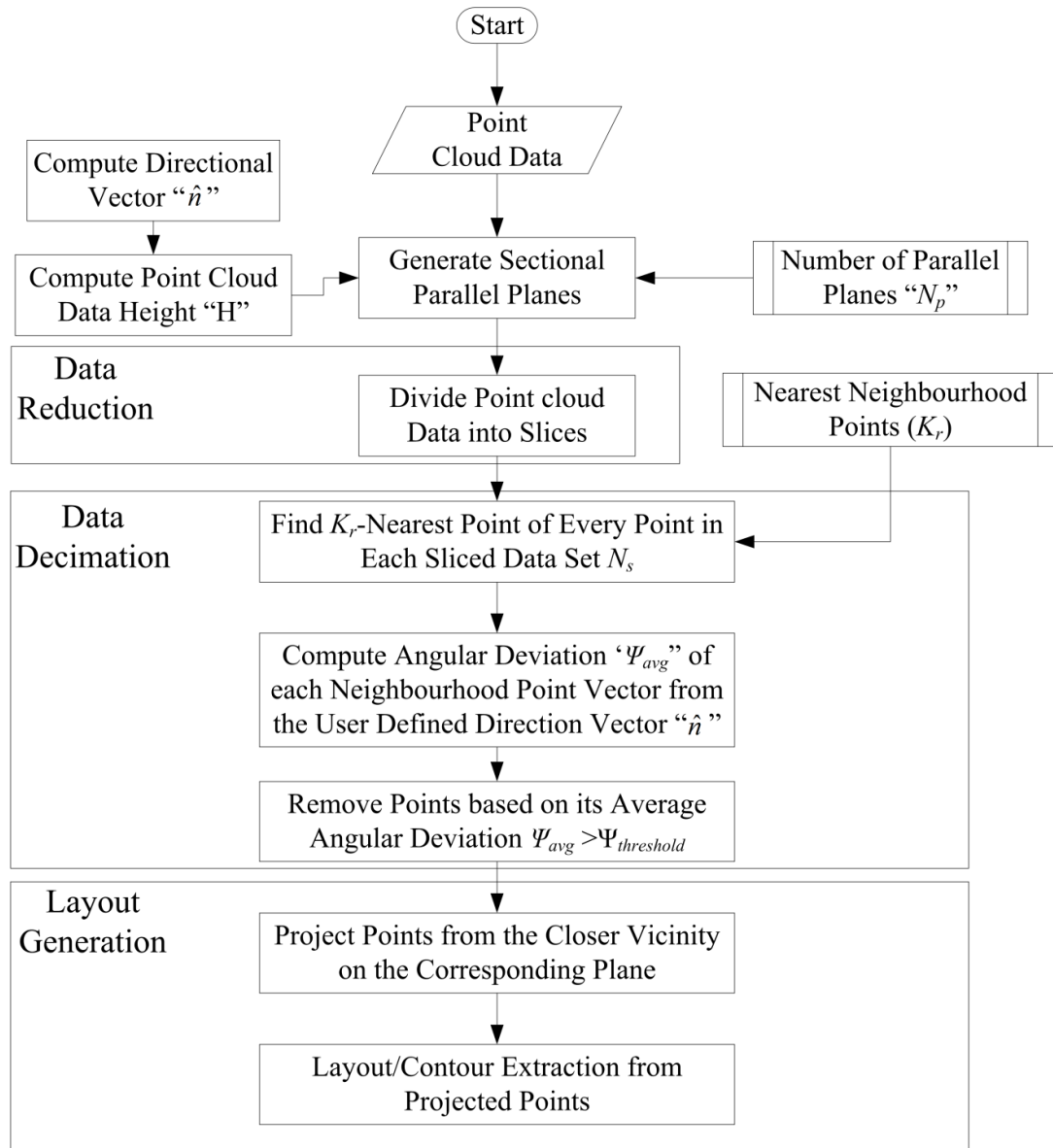
Figure 4.2: The overall process of layout extraction algorithm.

4.4.1 Slicing a point cloud data set

The first stage subdivides the point cloud into slices along the user defined direction using sectional planes, where the layouts are to be extracted. This subdivision diminishes the global influence on the extracted layouts and reduces the required computational effort. A vector direction ($\hat{n}$) is defined by the user either by using two spatial data points or by fitting a planar surface to the set of floor data points. The scanned data size along this direction is measured as the overall point cloud data height (H) through a bounding box.

A series of user defined parallel planes (N_p) with common normal ($\hat{n}$) are generated to subdivide the overall point cloud data height (H) into slices with 't' as the slice thickness as shown in Figure 4.3. The slice thickness (t) is defined as the ratio of data height (H) and the number of slices (N_p-1) as defined below:

$$t = \frac{H}{(N_p - 1)} \tag{4.1}$$

The number of planes (N_p) used to generate these slices depends upon the complexity of the scanned data and the required level of simplification. A large value of 'N_p' can extract more layouts and thus, can represent complex geometries, but will result in less simplification. On the other hand, greater simplification can be achieved using fewer planes at the cost of losing feature details. A single slice of the pre-defined slice thickness can also be used at the desired location along the direction normal ($\hat{n}$) for single layout extraction. In this case, a user determines the location along direction normal to identify an appropriate location for layout extraction. The point cloud data set in the vicinity of the simplifying plane represents its local shape and can be easily handled to determine the local shape of the scanned geometry.

The number of points in each slice is a representative of the local surface complexities and provides a means of selecting the data slice for extracting section layouts. The points in each slice along the data height (H) of a typical room scanned data are shown in Figure 4.4, which clearly shows that the slice including the ceiling data points have large number of points due to the unobstructed viewpoint of the scanner during the data capturing process. Further, the slice includes furniture such as table, chairs and other objects laying on the table

also thus exhibits large data set. These objects are also responsible for surface occlusion of the wall and reduce the captured data sets in its immediate neighboring data slices (occluded regions- I and II). It is evident that a relatively less occluded region is most suitable for extraction of section layouts. However, such unobstructed regions may not always be readily available and the data has to be adaptively decimated to extract the desired layout. During the decimation process, each point is retained or removed based on its contribution towards the actual layout.
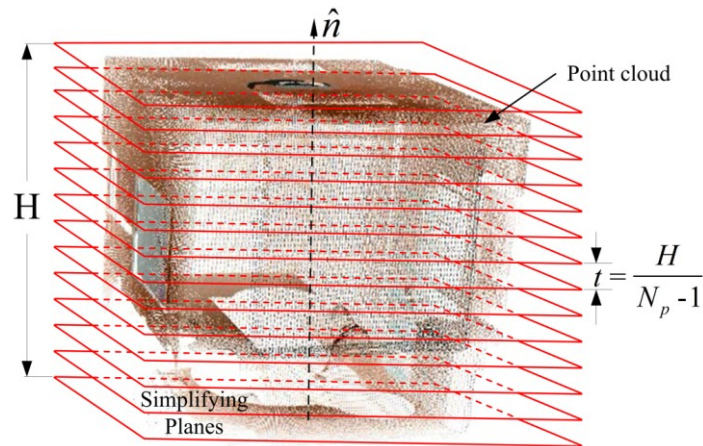
Figure 4.3: Data segmentation of point cloud by parallel planes.
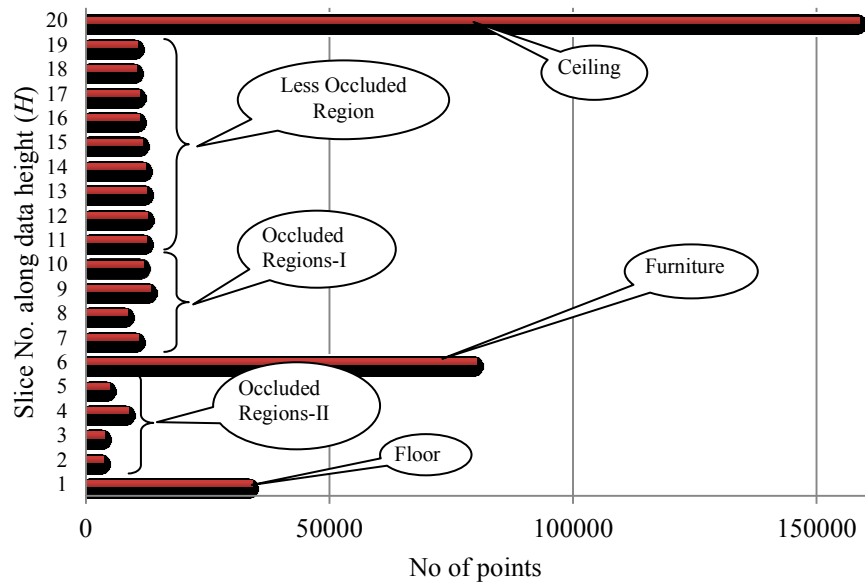
Figure 4.4: The height histogram showing the number of points in each slice.

4.4.2   Evaluating point's importance

The sliced point cloud includes points, representing different objects in the scanned scene. The proposed methodology identifies the layout points from each sliced data.

The layout points are lying along the vertical directional vector ($\hat{n}$). A point's importance is computed by measuring its average angular deviation value this directional vector ($\hat{n}$), as shown in Figure 4.5
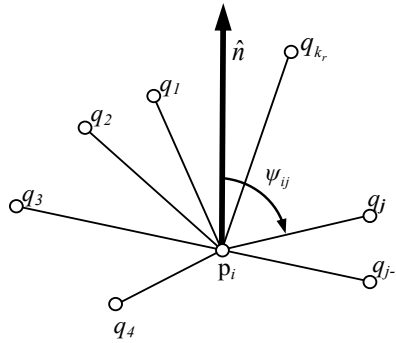
Figure 4.5:   Angular deviation calculations of neighboring vectors.

A neighborhood searching algorithm (Sankaranarayanan, Samet and Varshey, 2007) is used to identify the K_r-nearest spatial points (q_j) of every data point (p_i) in the sliced data set. A distance vector ($d_{ij}=q_j-p_i$) for each of these neighboring points is defined for computing their corresponding angular deviation value (Ψ_{ij}). An average angular deviation value (Ψ_{avg}) of each point is computed as cumulative mean of the deviation values of its neighboring data points and is a good measure of point's importance to be a layout point. The points with angular deviations more than a threshold value ($\Psi_{threshold}$) represent the points lying in horizontal planes and not on the walls, and thus does not represent layouts. Such points can be removed during decimation process. This approach will also remove some points from vertical wall data especially in regions where wall meets the floor or other horizontal regions. However, a large number of data points are retained which are sufficient for extracting section layouts. Algorithm 4.1 describes the steps used for computing the point's importance in terms of its angular deviations and achieving the desired data decimation. A controlled set of algorithmic parameters also helps in identifying the interior objects during this data decimation process as shown in the results (Section 4.5).

Algorithm 4.1:　　**Angular Deviation and Point Reduction algorithm**

Parameters:　　　N, $\hat{n}$, N_p

Find $\mathbf{p_e^T}$ and $\mathbf{p_e^B}$ (Extreme data points in N along $\hat{n}$)

Compute Data Size along $\hat{n}$

$$\mathbf{H} = (\mathbf{p_e^T} - \mathbf{p_e^B}).\hat{n}$$

Generate (N_p - 1) equal thickness data slices along $\hat{n}$
For each slice with N_s data points
Calculate K_r points for every $p_i \in \{N_s\}$

initialize Ψ_{ij} , Ψ^i_{avg}

　　while $i \leq N_s$ **do**

　　while $j \leq K_r$ **do**

　　Compute $d_{ij} = q_j - p_i$

$$\Psi_{ij} = \cos^{-1} \frac{(q_j - p_i).\hat{n}}{|q_j - p_i|}$$

$$\Psi_{ij} = \Psi_{ij} \qquad if \quad 0 \leq \Psi_{ij} < \pi/2$$

$$\Psi_{ij} = \pi - \Psi_{ij} \qquad if \ \pi/2 \leq \Psi_{ij} \leq \pi$$

$$\Psi^i_{avg} = \frac{\sum \Psi_{ij}}{K_r}$$

　　end *while*
　　end *while*

Remove　 p_i for which $\Psi^i_{avg} > \Psi_{threshold}$

return Updated N_s,

4.4.3　Projection based data decimation

The decimation process in the previous step separates the points from the sliced data set, which do not lie along the directional vector ($\hat{n}$). To limit the global impact on the extract layouts, only the immediate neighboring points around a simplifying plane are considered. A locality distance is defined around each plane from which the points are used for local shape extraction. This locality region is defined by a planar vicinity thickness (t), spanning equally ($\pm t/2$) on either side of a slicing plane as shown in Figure 4.6.
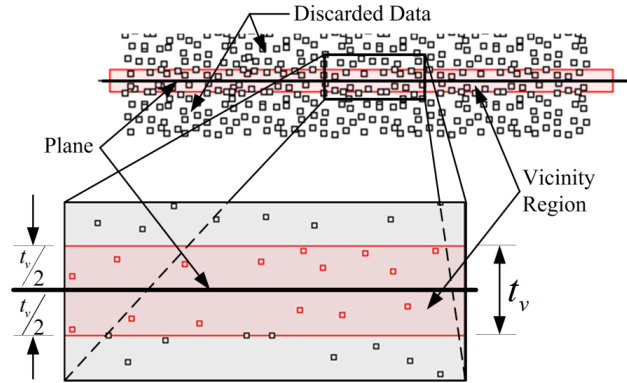
Figure 4.6: Vicinity region definition around a section plane.

The selection of the vicinity region thickness (t_v) is a compromise between local shape retention and the level of simplification. A very low value of vicinity region thickness fails to extract enough points for effective local layout extraction due to under-sampling, whereas a large value superimposes a global effect on the extracted layouts and diminishes the basic objective of the data simplification process. The maximum value of a vicinity region thickness (t_v) is denoted as t_v^{max} and it is defined, as follows:

$$t_v^{max} = \frac{H}{\mu N_p}$$

(4.2)

where N_p is the number of planes defined by the user, H is the overall point cloud data height and μ a factor that controls the density of the data sets, finally selected for layout extraction. All the points in this vicinity region are projected on its corresponding plane along its normal plane using a projection operator as shown in Figure 4.7.
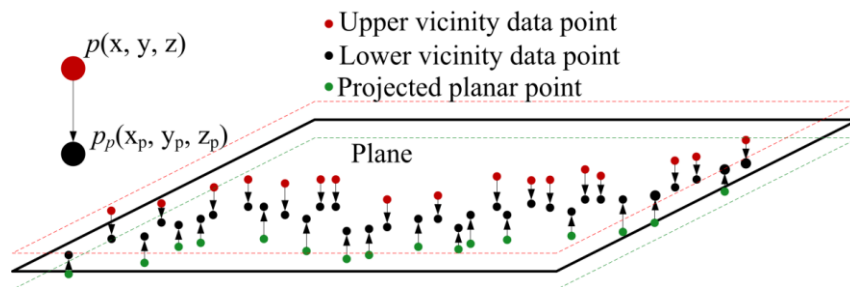
Figure 4.7: Plane with projected points from its predefined close vicinity.

The projection operator transforms the spatial data points from the vicinity region to coplanar data points in their corresponding simplification plane. The projected point $(p_p(x_p,y_p,z_p))$ is computed on the plane $(ax+by+cz+d=0)$ for each spatial point $(p(x,y,z))$ in the vicinity region thickness as below:-

$$x_p = x - a\frac{ax+by+cz+d}{a^2+b^2+c^2} \tag{4.3}$$

$$y_p = y - b\frac{ax+by+cz+d}{a^2+b^2+c^2} \tag{4.4}$$

$$z_p = z - c\frac{ax+by+cz+d}{a^2+b^2+c^2} \tag{4.5}$$

Thus, the data set used to generate the final layout actually corresponds to a fraction of the total point cloud data set, which has been adaptively decimated in two stages.

4.4.4   Section layout extraction

The projected coplanar points are used to generate the section layout in the desired plane. The layout is extracted and initial linear degree curve is generated through progressively searching and joining its neighborhood data points and maintaining an appropriate connectivity rule. A standard polyline tool in Rhino3D® is customized to generate this section layout through multiple coplanar points. This approach connects each point to its closest neighborhood points and maintains a connectivity index of 2 for each point. For example, in Figure 4.8, point '*A*' is already connected to '*C*' on one side. Point '*A*' finds its nearest points. In this case '*B*' is nearer to '*A*' than '*D*', thus, the progressive connectivity approach connects '*A*' to 'B' and the maximum connectivity index avoids its connectivity to any other point such as '*D*'. Further, point '*B*' finds the next connectivity point and in the process some points like '*D*' may be omitted. It has been found in this investigation that most of these omitted points are redundant and are not the true representation of the local layout contour shapes, thus, it is justified to omit them while extracting the approximate local section contours. Further, extracted contours are compensated for probable intersection of sharp bends.
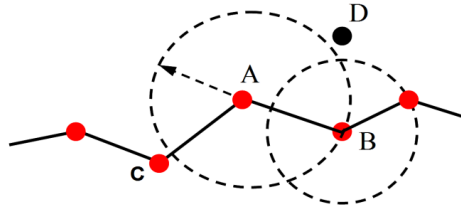
Figure 4.8: Linear contour generation from projected points

The initial contour expansion in this way may encounter some local edge intersections between neighboring segments as shown in Figure 4.9. However, their tendency is generally local and global intersections are not encountered due to the high data density and a restricted threshold value of the permissible distance (d_{ij}) between neighboring data points. The data density increases further during the projection stage and it does not allow the contour to cross its geometric confidence map (Pauly, Mitra and Guibas, 2004) and connect to a distant data point, thereby avoiding the global intersection situations. Further, few local intersections may occur in the extracted contour and small deviations. Although these small deviations can be handled effectively with synthetic curve regeneration, yet such intersections are discouraged at this stage. The possible local intersections are concurrently identified by constantly checking the intersections of reconstructed segments with its preceding s-segments (s=10). In case and intersection occurs, such as in Figure 4.9(a), the intersecting contour segments are identified and the contour is reset to the last point (D) of the first intersecting segment. Now, it omits the previous nearest point –E from the data set and identifies new neighbor-F and contour is advanced. Thus, new progressive points are identified and the intersections are avoided.
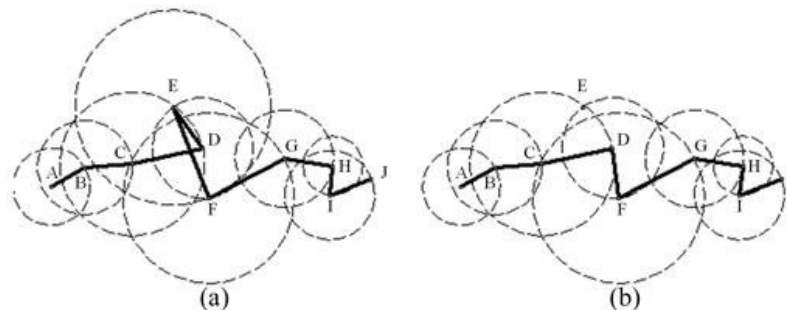
Figure 4.9: Avoiding self intersections during linear contour generations.

4.4.5 Synthetic simplification

The extracted layouts in the previous stage may be very sharp in certain cases. Although these sharp layout are often local and do not correspond to large kinks, these can be smoothened with best fitted synthetic curves (cubic B-splines) using less number of points, which achieves further data reduction (Figure 4.10). Here, the data can be decimated using a reduction ratio (r) that defines the ratio of points used to generate initial and regenerated synthetic layouts.
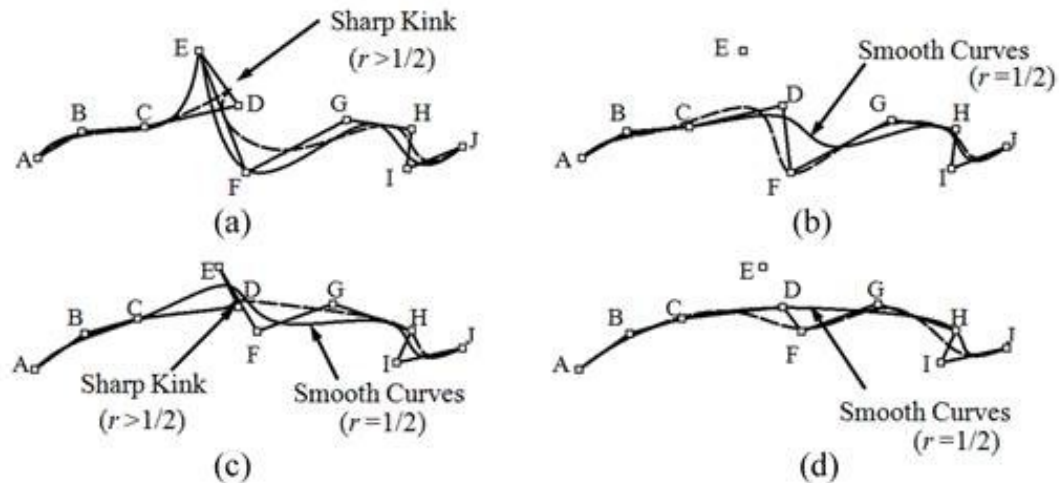
Figure 4.10:   Synthetic curve generation and removal of tiny imperfections.

Here the intersecting curve can be smoothened with desired reduction ratio. Initial intersecting linear curves and sharp synthetic kinks (Figure 4.10 (a)) can be removed using intersection checks and synthetic cubic b-spline curve approximation using higher data reductions (Figure 4.10 (b)). A similar approach also works for curves with no intersection but sharp initial kinks (Figure 4.10 (c)) that can be smoothened with higher point reductions and synthetic curve smoothening (Figure 4.10 (d)).

This type of smoothing and corresponding simplification may not always be required for applications involving building interiors. However, it can really facilitate generating complex contours and geometric models of organic shapes as demonstrated on facial scanned data by the authors (Sareen, Knopf and Canas, 2009) Essentially, this stage reduces the number of

points by removing the points which are (a) not laying in the close vicinity region thickness of any plane and (b) points which are omitted during contour generation

4.5   Results and Discussions

The proposed *layout extraction algorithm* is illustrated with a raw scanned data set of a room with multiple objects, captured with a FARO® scanner (Model: FS0880). The algorithms were developed using web-based PHP language with SQL server data-base. The results of the processed points are visualized with a Rhino3D software package and are compiled in the following sections.

4.5.1   Room scan data

A room scanned data set (Figure 4.11) with 1550035 data points is used to illustrate this proposed layout extraction pipeline with different algorithmic parameters. The point cloud data set is sliced using four sets of parallel planes N_p (20, 30, 40 and 50) and the corresponding data set is decimated using proposed angular deviation analysis. The average angular deviation value (Ψ_{avg}^i) used for retaining contour points from each slice, depends upon the relative topology of the neighboring points and the type of features around the simplification plane.

| (a) | (b) | (c) |
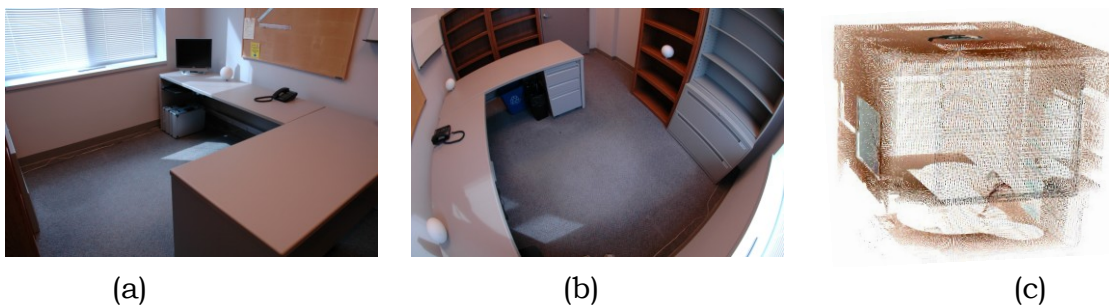
Figure 4.11:   (a) Room view from front (b) Room's view from back (c) Scanned point cloud data set of the room.

4.5.2   Data simplification

Room data decimation is first achieved by segmenting the data set into individual slices, reducing the data points using the proposed angular deviations values and then finally reducing the resulting data set by defining the *vicinity region thickness*. A small value of the vicinity region is defined using $\mu=10$, which reduces the global effect on the extracted contours. In the process, the total data decimation achieved is defined by the overall reduction ratio (*r*). The effectiveness of the proposed methodology is measured in terms of the layout point identification and the accuracy of the shape of these extracted contours. The point cloud data, slices and different stages of data decimation of a particular layer is shown in Figure 4.12.

The algorithm can be implemented on any slice and it is advisable to select the data slice from the less occluded regions (Figure 4.4) for an accurate layout extraction. However, there will be cases of occupied building interiors in practice, where less clutter data slices are not available. An effective algorithm must be able to compensate such cluttered data set as well. So, a relatively complex data slice with maximum number of points is selected to prove the effectiveness of the algorithm (Figure 4.12).

It can be seen from Figure 4.12 (c-d) that the data points in the slice correspond to multiple objects (table, reference sphere, phone and computer screen). Points with average angular deviation higher than a threshold value are removed during the data decimation process. Based on this allowable deviation threshold ($\Psi_{threshold}$), the decimated points in the data slice can be used for effective scene visualization (Figure 4.12(c-e)) or extracting points representing section layouts (Figure 4.12(f)). A relatively higher threshold value ($\Psi_{threshold} = 80°\text{-}85°$) mainly removes the planar data set and the data representing the interior objects and furniture pieces are retained, which formulates a quick means of identifying interior objects. A further reduction of this angular deviation threshold value removes points from the horizontal regions and only retains the data points representing vertical regions. These highly decimated points (Figure 4.12(f)) represent the walls and can be used to extract the section layouts as the points corresponding to the interior objects (furniture and other individual objects) are removed during the simplification process.

(a) Data slicing

(b) Sliced data

(c) Processed data
(K_r=10, $\Psi_{threshold.}$=85°)

(d) Processed data
(K_r=10, $\Psi_{threshold.}$=80°)

(e) Processed data
(K_r=10, $\Psi_{threshold}$=40°)

(f) Processed data
(K_r=10, $\Psi_{threshold.}$=10°)
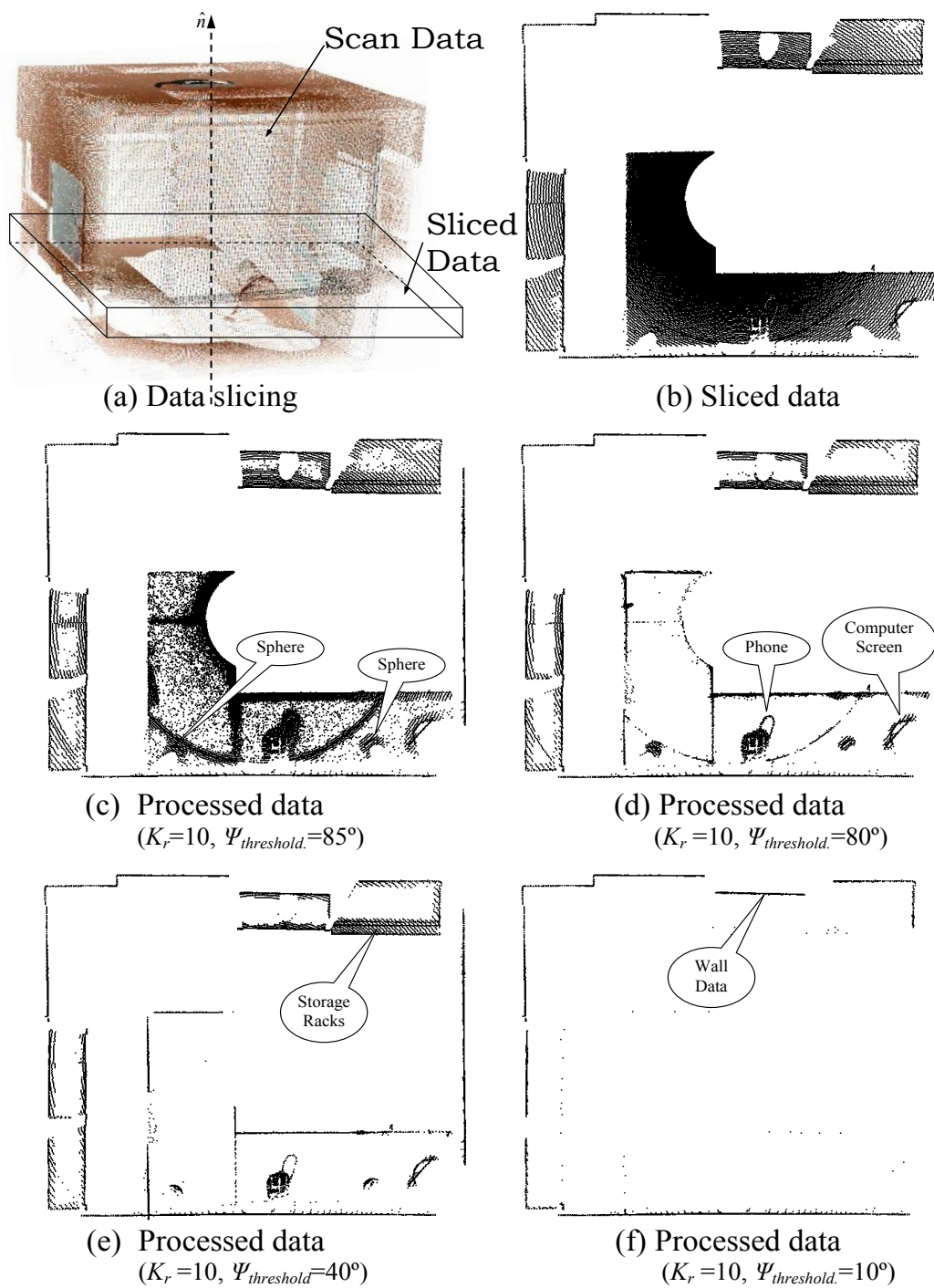
Figure 4.12:   (a-b) Segmented data slice and (c-f) data set of various data reduction stages.

4.5.3   Layout extraction

The layouts are represented by the boundary data points, which are retained from a very thin slice around the parallel plane. Thus all other points lying outside the thin layer (thickness = t_v^{max}) around each simplifying plane are removed to decimate the data set further. The decimated data set is then projected on the plane and then these projected, coplanar points are connected together to generate single degree curve as explained in the Section 4.4.4. e.g. the dataset from Figure 4.12 (d) is projected on its corresponding plane and then the coplanar data set is used to generate the section layout as shown in Figure 4.13.

(a)

(b)   (c)
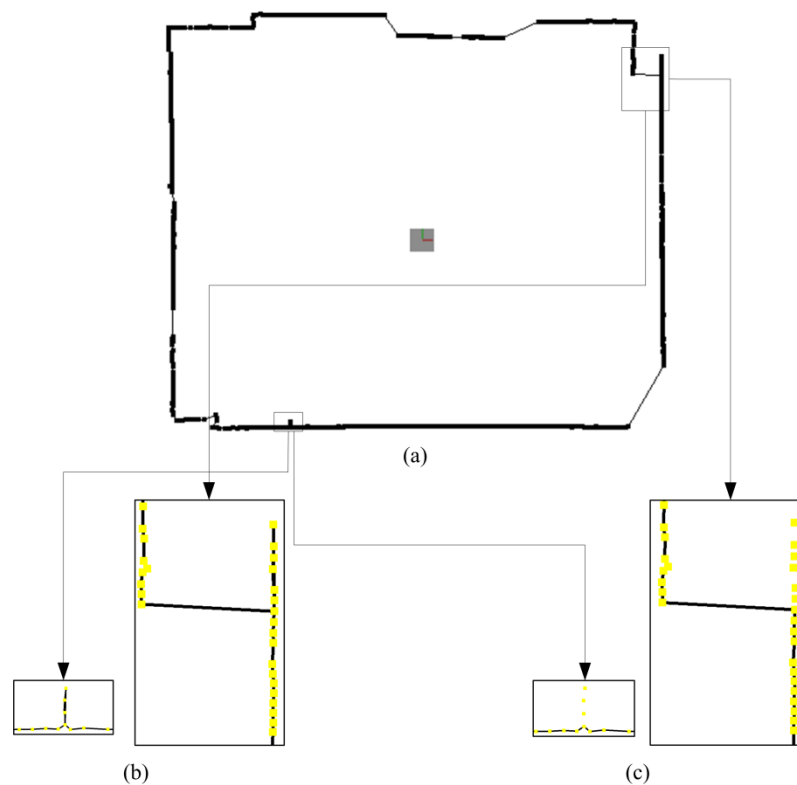
Figure 4.13:   (a) Point clouds representing section layouts (b) Layout generated using single degree
polyline with Rhino®3D, and (c) Layout extracted with connectivity index of 2 using improved
polyline methodology.

The extracted contour represents the domain layout. Figure 4.13(c) represents the layout extracted using the proposed methodology and handles the multiple diversion really well, which otherwise would generate multiple local contours at these diversions as shown in

Figure 4.13(b). Thus, based on the location of the simplifying plane, the multiple layouts can be extracted as shown in Figure 4.14.

(a) 50 Layouts ($r = 0.12$)   (b) 40 Layouts ($r = 0.09$)

(c) 30 Layouts ($r = 0.07$)   (d) 20 Layouts ($r = 0.05$)
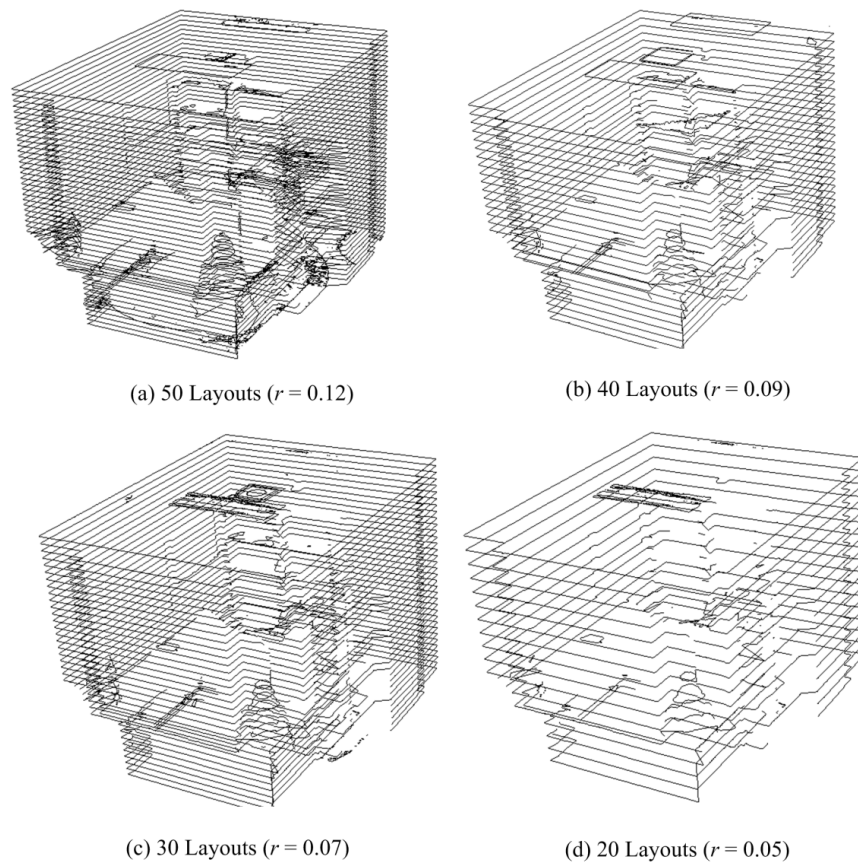
Figure 4.14: Extracted layouts from scanned data in user-defined planes and their corresponding reduction ratio.

4.5.4   Discussions

The results of the proposed algorithm suggest that the proposed methodology is effective for both data simplification and layout extraction from complex data set of multiple objects. The algorithm efficiently identifies the point's importance based on its location and contribution towards the layout. Thus, the complex point cloud can easily be decimated to point cloud representing the layouts and important features in the point cloud.

The decimation based on the angular deviation of points from its neighboring point data set retains points representing important features in the point clouds. Point data decimation does not remove data points based on a constantly changing parameter unlike most other

simplification algorithms. Instead, it simplifies data set based on a pre-computed angular deviation value and does not require the recursive computation of the point's importance with each removed data point and thus, is computationally very effective. The decimated data set gives a good estimation of the objects in the scanned environments and thus provides a foundation for identifying and regenerating these objects.

A highly decimated data set is achieved using a very low value of angular deviation value, which is used as a threshold value for point cloud reduction. It is evident from Figure 4.14 that better layouts are extracted from regions with less cluttered data. Multiple, local and premature layouts are often generated from decimated data set from layers exhibiting point clouds from multiple objects. A less cluttered region on the other hand, often retains points with less occlusion and thus minimum discontinuities and highly accurate section layouts are extracted from such data set. Sometimes, the layouts extracted from highly cluttered data set with discontinuous data sets are often not closed (Figure 4.14) due to restrictive connectivity lengths, which are necessary for avoiding irregular layout extractions. However, the partial layout extracted in these sections is also a good representation of the actual scanned data set around that plane. Thus, it is necessary to use suitable sliced data for extracting the required layout. Figure 4.15 shows two different sliced data sets from the room scan, which are used to extract the building layouts. The deviation values that appear in the occluded regions (region A, B and C) in this Figure 4.15 (a) figure are due to the absence of the scanned data in these regions. Whereas another sliced data Figure 4.15(b), involving relatively less occlusions, generate accurate results with maximum deviation of 3mm only. Thus, the regions with good data representation are quite accurate.
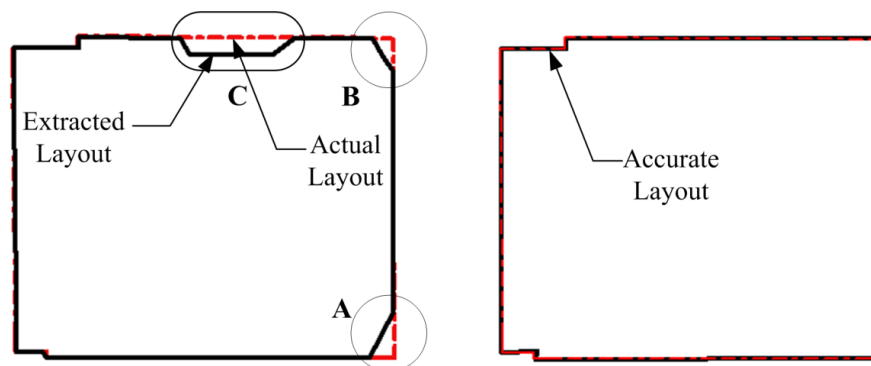
Figure 4.15: (a) Extracted layout involving multiple objects with cluttered data and occluded regions and (b) Extracted layout from less cluttered data set.

Further, it is observed that using a small number of nearest neighborhood points, a point's importance can be computed accurately and thus is an effective approach of data simplification and subsequent layout extraction. This is due to the fact that large numbers of nearest neighborhood points superimpose a global effect on the angular deviation values, hence its average values becomes comparable over a large domain. This low variation in average angular deviation value makes it difficult to distinguish feature points effectively.

It is to be pointed out that the primary objective of this research is to develop an effective simplification strategy to minimize the data confusion by avoiding the inclusion of non-layout points for extracting the layout. The simplification strategy works directly on the points and does not require recursive computation for assessing the point's importance. The simplification strategy can be extended to an interior scenario involving multiple room, pathways or industrial spaces. However, extracting their corresponding layouts is a complex task and requires advanced graph theory to identify accurate bifurcation for multi-direction layout propagation.

4.6   Concluding Remarks

The process of converting the point clouds into usable format is a complex, but essential step in digitization of the building interiors. The complexity increases manifolds when the scanned data represent multiple objects and occluded geometries.

In this chapter, we have presented a *data simplification and layout extraction algorithm* from a point cloud data set with multiple objects in the scanned scene. The process identifies and simplifies the data set by removing non-feature points. The process exploits the fact that simple geometries such as planes can be represented with fewer data points and thus can be reduced. Reduction of data points from these planar regions retains other critical non-planar features, which can be efficiently reconstructed, as required. The approach attempts to simplify the point clouds with the intention of retaining the layout points and regenerate the layout from this reduced data set.

The simplification approach reduces the points by removing planar points along a given direction. The approach effectively retains the feature points using a computationally effective data decimation process due to its non-recursive computational approach. The data simplification process is directly applicable on the points and does not require any

intermediate geometric model. A highly decimated data set with this approach is then used to generate the layout contours by projection-based contour extraction. The projected data set is joined to its nearest neighborhood points by a connectivity rule to extract the layout contour in the user defined planes. It has been shown that accurate layouts can be extracted using the proposed methodology. These extracted contours need only a fraction of the actual scanned data points (about 5-12%). A small vicinity region and smaller neighborhood points (K_r) improves the layout extraction performance by reducing the global influence on local shapes. The simplification approach is extendable to large data sets of occupied building interiors involving multiple rooms and interior structures.

The data decimation process facilitates a point based simplification and an effective layout extraction algorithm from the point cloud. However, some of the features such as planar regions along the simplification directions (such as table tops) are lost. Moreover, the data decimation process reduces the points from all features based on their angular deviation from the simplification direction. In such cases, the data density at these regions may become so sparse that extracting corresponding features becomes difficult. Thus, a suitable approach of feature extraction is to segment the point cloud into individual object data sets. The next chapter compiles few segmentation strategies developed for the point cloud data set of occupied building interiors.

CHAPTER 5   COLOR-BASED SEGMENTATION OF POINT CLOUDS

5.1   Introduction

Range scanning of large man-made structures, exterior building facades and ill-defined interior spaces generates an immense 3D point cloud of spatial coordinate data of discrete surface points and, often, the corresponding RGB color information. The extraction of accurate geometric and spatial information from these millions of scattered data points is a complex task due to varying geometric complexity, presence of multiple objects in the scanned scene and extensive computational demands.

Segmenting the large point clouds into meaningful clusters reflecting surfaces with common characteristics is often the first step [Ning *et al*., 2009] for applications such as reverse engineering, computer graphics, computer vision, as-design-as-built checking and virtual reality (VR) modeling of pre-existing interiors. The segmented data clusters greatly improve the data usability of such cohesive point clouds. Once properly segmented, it is possible to use individual data clusters as the application demands. Each application uses the data sets in its own unique way according to its own explicit set of requirements. Thus, segmentation strategies of one application may not be applicable to another. For example faster computational speed is essential for computer vision applications whereas segmentation accuracy is of topmost importance for individual object reconstruction.

Accurate segmentation of point clouds of building interiors (Budroni and Böhm, 2009; Dorninger and Nothegger, 2007; Rabbani, Heuvel and Vosselman, 2006; Wolfart, Sequeira, Ng *et al.*, 1999) and architectural shapes (Alharthy and Berthel, 2004; Budroni and Böhm, 2009; Rabbani, Heuvel and Vosselman, 2006) introduce unique challenges that are often not encountered when partitioning range data taken from a single isolated object and building exteriors. It is crucial to handle immensely diverse geometric disparities (planar walls as well as intricate objects) and data discontinuities due to the presence of multiple, overlapping objects in the scanned scene, besides compensating for traditional scanning errors.

Traditional geometry based approaches tend to over-segment simple surfaces and under-segment complex shapes that have fine features or details (Zhan, Liang and Xiao, 2009). Moreover, generating geometric derivatives (surface normal and curvature) from the point

clouds for region differentiation is a computationally expensive and error-prone task due to the vast size and geometric complexity of the point cloud. Inconsistent results are especially observed in areas where surfaces from multiple objects with similar geometric properties are closely spaced, partially occluded, or overlapped.

For buildings and urban scanned data, most current segmentation methods (Alharthy and Berthel, 2004; Dorninger and Nothegger, 2007; Rottensteiner, Trinder, Clode *et al.*, 2005) will apply local planar fitting techniques and a priori knowledge of building structure to extract flat roofs, walls, floors and ground regions from either the points clouds (Alharthy and Berthel, 2004) or triangulate the captured data to generate cumulative surfaces from the cloud (Rottensteiner, Trinder, Clode *et al.*, 2005). Unfortunately, only primitive partial shape information is obtained from such methods and the results cannot be readily extended to the whole scanned scene. To further automate the process, computationally efficient scan-line based segmentation algorithms (Sithole and Vosselman, 2004; Vosselman, Gorte, Sithole *et al.*, 2004) have been used with some success to identify discontinuities and associate similar data points from neighboring regions in the 3D point cloud. These methods work well for extracting planar regions from well-structured point clouds and, in few cases, have been used to extract information about architectural shapes (Ning, Zhang, Wang *et al.*, 2009), industrial pipes (Vosselman, Gorte, Sithole *et al.*, 2004) and trees (Barnea, Filin and Alchanatis, 2007). However, the requirement of highly structured point clouds limits its use for clustering range data captured from unstructured point clouds of building interiors.

Relatively few segmentation methods have been applied successfully to range scans of building interiors. Wolfart *et. al* (Wolfart, Sequeira, Ng *et al.*, 1999) proposed a hybrid strategy that performed triangulation during data re-sampling, local surface fitting, and discontinuous edge detection to solve the problem of segmenting building interiors. Although the technique produced satisfactory results for large planar regions such as walls and cabinets, the algorithm could not accurately cluster points that lie on small objects with freeform shapes due to inadequate seed points. To address this problem, Rabbani (Rabbani, Heuvel and Vosselman, 2006) used normal residuals to identify reliable seed points in the cloud and a normal deviation-based smoothness parameter for region growing. The method was able to successfully extract unique clusters from a scanned industrial environment. The computed geometric parameters were, however, not highly reliable and prone to significant errors in overlapping regions which lead to data over-segmentation. In contrast, Budroni and

Böhm (Budroni and Böhm, 2009) proposed a plane-sweep algorithm for segmenting 3D point clouds into planar clusters. The relative advantage of this approach is that it works directly on sampled points instead of tessellated surface-based approaches (Rabbani, Heuvel and Vosselman, 2006; Wolfart, Sequeira, Ng *et al.*, 1999). Although effective, the technique omits all non-planar data points and, therefore, cannot be used to segment building interiors that contain desks, chairs and personal items.

Many surface reconstruction methods demonstrated on range data of large civil structures and building interiors often exploit known shape or size *semantics* and generate cumulative building models. Wang and Luebke (Wang and Luebke, 2003) generated triangulated models of the scanned data by using normal and confidence-level semantics along with weighted color and texture coordinates. A cumulative triangulated model can enhance the overall visualization of the scanned scene, but it does not permit surface segmentation and reconstruction of individual objects in the scene. Bahmutov *et al.* (Bahmutov, Popescu and Mudure, 2006) demonstrated a rapid building reconstruction method for large structures by using shape semantics such as rectangular cuboids for pathways walls. The super-positioning of registered color information on this sparse depth map generates photorealistic models with over-simplified geometries. These over-simplifications often lose critical geometric details. It is, therefore, difficult to have a single algorithm that simultaneously handles simple planar shapes and complex, organically shaped objects. Furthermore, most geometry based segmentation methods require intermediate surface generation (Wolfart, Sequeira, Ng *et al.*, 1999) or geometric parameter computation (Rabbani, Heuvel and Vosselman, 2006) which are not reliable if the scanned scene contains multiple overlapping objects.

In this context, there are two major research challenges in segmenting the scanned data of building interiors: (1) developing an algorithm that can handle a broad range of geometries from planar walls to complex free-form objects, and (2) selecting a reliable measure of similarity between neighboring points assigned to the same cluster or underlying surface region.

To address these challenges, an inherent similarity parameter of the point cloud (color) is exploited to form a reliable differentiating parameter for clustering the unified point clouds. A key feature of the technique is the application of a robust HSV (Hue, Saturation, Value) color perception model rather than the RGB (Red, Green, Blue) sensor values. This approach

enables "color" to be defined by a single parameter called hue, and eliminates shading and intensity effects associated with the "dominance of the hue" (saturation) or "color brightness" (value) due to variations in room illumination.

First color is used alone to generate a rapid segmentation algorithm(Sareen, Knopf and Canas, 2010) for quick visualization of the scanned scene and subsequently combined with geometric parameters to formulate a robust shape-hierarchy based hierarchical segmentation algorithm(Sareen, Knopf and Canas, 2011) as compiled in the following sections.

5.2   Colored Point Cloud Acquisition

The colored data set is generated by combining the spatial data points captured by the 3D range senor with their corresponding colored pixels from colored images mapped together in the optical plane as explained below.

5.2.1   Colored range scanning

The colored point cloud data set is generated by capturing a spatial domain using a stationary laser scanner and bracket mounted digital camera as shown in Figure 5.1
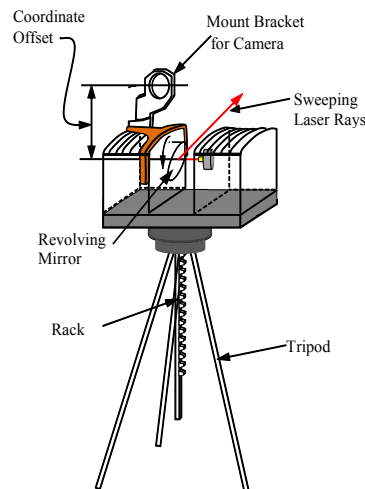
Figure 5.1:   A schematic diagram of a FARO® scanner with camera mounted bracket.

The spatial geometry of a given domain is captured in terms of 3D points (x y z) is first captured by laser scanner (FARO® LS880) where the laser rays are reflected towards the

scanning range by the revolving mirror. These surface points (x y z) are captured in a spatial coordinate system with the laser origin as its perspective centre. In this scanning position, the bracket-mounted camera is moved to the extreme left (Figure 5.2(a)) so that it does not block the scanning path of the laser. Then, the tripod is lowered and bracket is moved to the centre to synchronize the coordinate systems (Figure 5.2(b)) of the previously captured point cloud data set and colored images that are to be taken by the digital camera (Nikon D70) with a fish-eye lens (Nikkor 10.5mm f/2.8G ED). The camera captures the color information in the form of multiple, colored images of the scanned scene as shown in Figure 5.3.
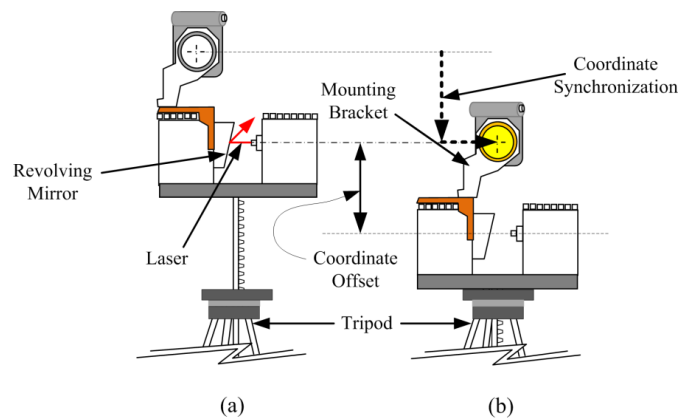
Figure 5.2: (a) Scanning position and (b) Colored imaging position of the FARO® scanner.
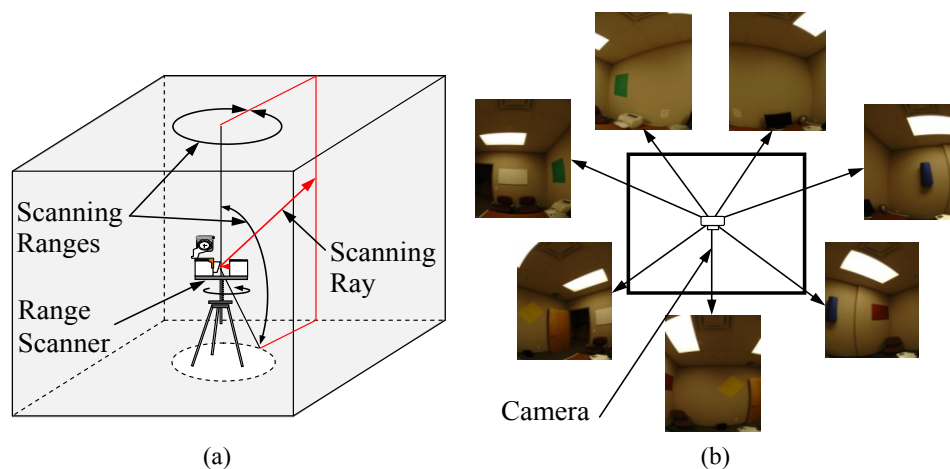
Figure 5.3: A typical shape capturing process using (a) 3D point clouds and (b) Color information in terms of multiple colored images.

5.2.2   Colored image mapping

A panoramic image of the whole scanned scene is generated by combining these individual images in an optical plane such that the focal length (f_l) of this image capturing system is defined as the shortest distance between its optical (O) and the perspective centre (O_p) as shown in Figure 5.4. Subsequently, a transformation model computes the 2D coordinates of each spatial data point such as point-p in the same image plane using methodology explained by Xu, Ye and Fan (Xu, Ye and Fan, 2002).   In this manner, each captured spatial data point p (x y z) corresponds to a unique point (x_a y_a) in this optical plane, which coincides with its matching pixel position ($u,$ v) in the panoramic colored image. Consequently, the color attributes (r g b) of each synchronized pixel is assigned to its corresponding spatial data point (x y z) and the colored point cloud data set (x y z r g b) is generated. In order to capture accurate color information with minimum intensity distortions, the images are captured without enhanced artificial illuminations using a tripod mounted camera. The captured images are processed to generate a panoramic view of the scanned domain. This extracted panoramic image can be smoothened to remove image distortions before mapping the color coordinates to its its corresponding spatial points.

Figure 5.5 shows (a) the original range scanned data with no color information, (b) the panoramic view of the color images of the scanned domain in the optical plane, and (c) the resultant colored range scan data.
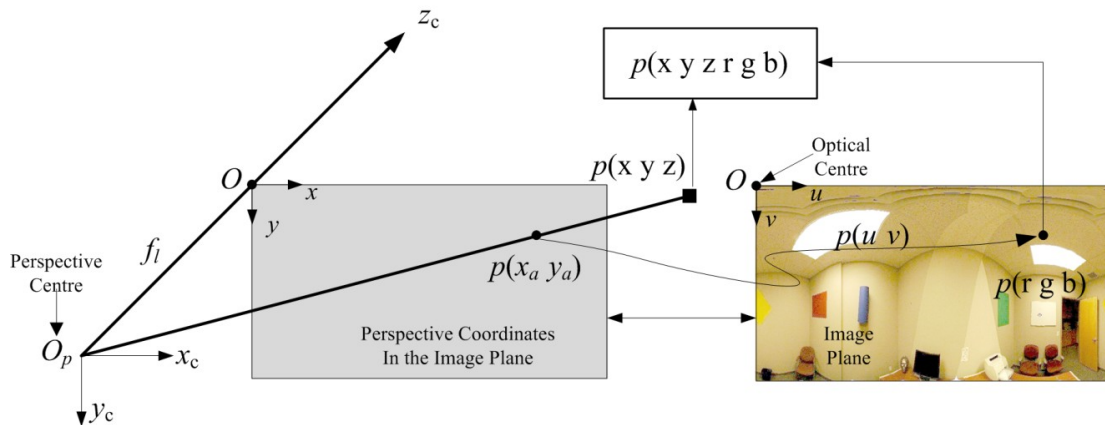
Figure 5.4:   The principle of generating colored point cloud from point set using FARO® scanner and panoramic color image from digital camera.
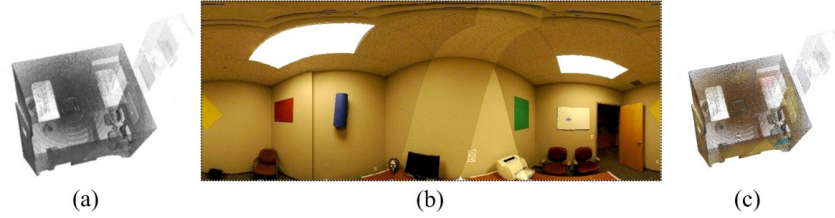
Figure 5.5: (a) Spatial data set without color information (b) A panoramic colored view of the whole room, and (c) Resultant color range scanned data of the room.

It is to be noted that the color information is transferred from the image to their corresponding spatial data points. Thus, even though the colored images may have highly detailed information, each pixel color may not be retained in the final data set depending upon the spatial data density of the captured data. Thus, generating an initial, dense point cloud data set is an absolute necessity.

5.2.3   Color model conversion

The red, green and blue (RGB) color model (Figure 5.6 (a)) is extensively used in display devices but it does not support a robust color segmentation strategy due to (a) high correlation among its three color sub-spaces and (b) its dependency on intensity parameters (Zhang and Wang, 2000). A RGB based segmentation strategy (Zhan, Liang and Xiao, 2009) demonstrated on point clouds generates over-segmented data sets and subsequent region growth and merging strategies were required to refine the segmentation results. Further, this highly correlated RGB colored data set does not provide an intuitive way of selecting initial seed point. To avoid these limitations of this color model, it is converted into a more efficient hue, saturation and value (HSV) color model (Figure 5.6(b)).
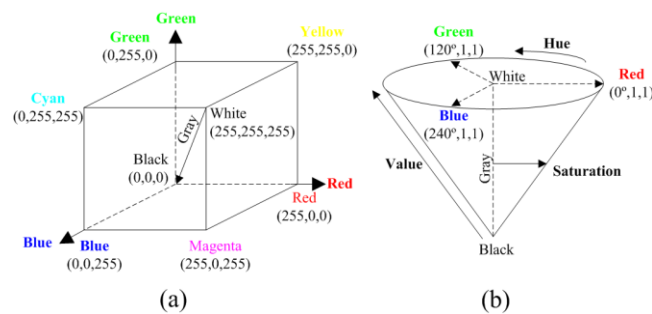
Figure 5.6: (a) RGB color model cube (b) HSV color model cone.

The basic color in the HSV model is characterized by its Hue (H) parameter (Figure 5.6(b)), whereas, the Saturation (S) and Value (V) represent variations of the same Hue. Thus, the points can be clustered based in their Hue similarity, which reduces the segmentation complexity due to fewer discriminating parameters. This HSV based segmentation strategy handles the usual limitations of RGB color model effectively and facilitates a single parametric segmentation approach. The following methodology is used to convert the color model from the available RGB to HSV model.

The conversion process maintains the correspondence between the spatial scanned data points and its color information. Thus, each data point (x y z r g b) is converted to its corresponding data point (x y z h s v). The processed point cloud is used to achieve the segmentation.

Algorithm 5.1:   RGB to HSV Color Model Conversion

Input : *r g b* Value (0-255)

      Normalize RGB values from 0-1
          norm_*r* = (R/255); norm_*g* = (G/255); norm_*v* = (B/255);

Compute: Maximum and Minimum Normalized Value
      Norm_Min = *min* (norm_*r*, norm_*g*, norm_*b*)
      Norm_Max = *max*(norm_*r*, norm_*g*, norm_*b*)
      del_Max = Norm_Max - Norm_Min

              Value (V) = Norm_Max
If (Norm_Max = 0)
      Hue (*h*) = 0º; **Saturation** (*s*) = 0;
else
      S = del_Max / Norm_Max
      ***for* Hue** (*h*)
              Case 1: Norm_Max = norm_R
                  $h = 60 \bullet \{(norm\_g - norm\_b)/del\_Max\}$
              Case 2: Norm_Max = norm_G
                  $h = 60 \bullet \{(norm\_b - norm\_r)/del\_Max\}$
              Case 3: Norm_Max = norm_B
                  $h = 60 \bullet \{(norm\_r - norm\_g)/del\_Max\}$
      end *for*
    if *h* < 0
        $h = 360° + h$
    end *if*
end *if*

5.3 Color-Based Clustering

The traditional geometry based segmentation algorithms use derived geometric derivatives (surface normal and curvature) as differentiating parameters for segmentation. It is computationally very expensive to compute these parameters for a large, unified data set of building interiors. Many real time applications such as virtual reality modeling, as-design as-built evaluations, computer vision, and web-based monitoring require a rapid segmentation strategy that can accelerate the clustering process so that meaningful data segments are quickly identified for better understanding of the scanned scene.

The clustering process can be speeded up by avoiding the computation of these additional geometric attributes such as normals and saving their corresponding computational cost ($O(n^2)$ using triangulation and $O(a \cdot n)$ through surface fitting, where a-defines the number of neighboring points used for plane approximation). Instead of computing these approximated surface properties, an effective approach is to formulate a segmentation strategy that uses primary attributes of the point clouds such as color and spatial proximity to differentiate object clusters and subsequently, segment them. The computational cost required by this methodology ($O(n)$ for structured and $O(n\log n)$ for unstructured data sets) is drastically reduced. Hence, a pure color-based segmentation algorithm is investigated as a starting point, as discussed in the following section.

A pure color-based segmentation algorithm formulates clusters from neighboring points with similar hues. First of all, rough clusters are quickly segmented based on the hue, which are subsequently refined to extract accurate data clusters as shown in Figure 5.7.

5.3.1 Rough segmentation

This initial stage extracts data clusters for a specific hue range with limited deviation span for user defined seeds. The points are grouped together into a cluster if their Hue deviation (ΔH) is within a permissible limit. A point data saturation parameter (S_r), controls the data extraction process based on an allowable hue deviation value (Algorithm 5.2). The saturation parameter (S_r) is defined as a ratio of the increase in number of extracted points to the cumulative extracted points in the previous stage. It has been seen that the number of extracted points increases by increasing the permissible angular variation value, however it stabilizes (Figure 5.8) at a specific hue deviation value. This maximum permissible angle is

denoted by *Max_Ang* and is defined as the maximum angular span (ΔH) around a specified hue (H) value, which constitutes a specific color in the given scanned scene. This approach identifies different segments representing specific colors and thus identifies a roughly segmented data representing different objects. A saturation criterion keeps a check on the segmentation process and avoids the under-segmentation. It is to be noted that *Max_Ang* can be specified externally by the user to any lower value than this computed value.
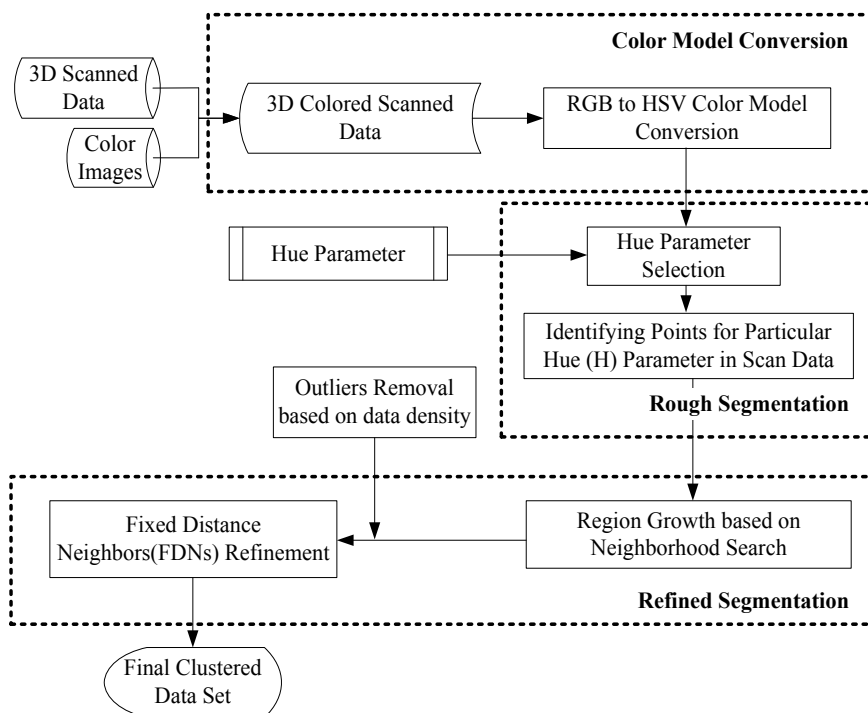
Figure 5.7: An overview of a pure color-based clustering methodology.
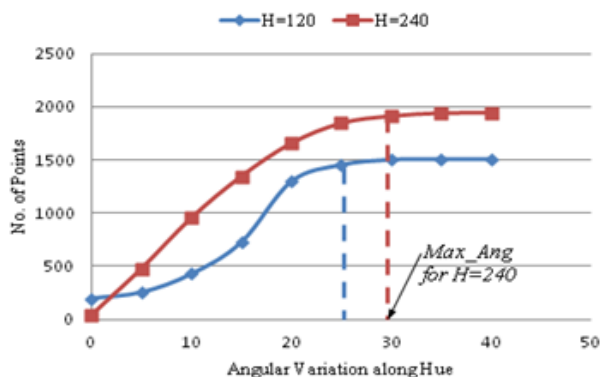
Figure 5.8: Point cloud stabilization with increased variation in hue value.

The point cloud is clustered using the Algorithm 5.2 where the points for a given hue are segmented until the corresponding cluster is saturated. The extracted cluster often includes outliers and thus, such clusters need to be refined to extract accurate data clusters with steps shown in Algorithm 5.3.

Algorithm 5.2: Color Based Clustering

Input : *Point cloud* (PC)*data with color info (i j x y z h s v)*
   Main Hue seed values in data set $(h_1, h_2, h_3..... h_n)$
   Permissible Hue Array $(\Delta h_1, \Delta h_2, \Delta h_3, …… \Delta h_m)$

Define: *Cluster for hue h =* $\mathbf{C_h}$*; Point with Hue h =* p_h*;*

Initalize: n_h =size($\mathbf{C_h}$)=0; \$Prevn_h = 0; S_r=0.03; \$Prev Δh=0; N_h=0;
 for each Hue Seed *h*, **Do**
     $p_h \xrightarrow{\text{add}} \mathbf{C_h}$; n_h =size($\mathbf{C_h}$)
 for k = 1 to m
    for each point $p_q \in$ PC
      if $h - \dfrac{\Delta h_m}{2} \leq h_q \leq h + \dfrac{\Delta h_m}{2}$;
         $p_q \xrightarrow{\text{add}} \mathbf{C_h};$ $n_h = n_h + 1$;
       end if
    end for
    Checking the Data Saturation Condition
      if $\dfrac{n_h - \$\text{Prev}n_h}{\sum N_h} \geq S_r$
         \$*Prev* $n_h = n_h$; $N_h = N_h + n_h$; \$*Prev* Δh= Δh_m; $n_H = 0$;
      else Break;
      end if
 end for
 end for
 return *Updated Cluster* $\mathbf{C_H}$

5.3.2   Cluster refinement

The rough clusters for a particular Hue represent two type of data sets (a) Outliers or spurious data and (b) Rough object clusters. In this refinement stage, spurious data and outliers are removed and object's clusters are refined. This refinement is achieved using a distance based outlier detection and removal process. The saturated regions are extracted using region growth approach through neighborhood investigation (Algorithm 5.3). A local density based FDN$_{threshold}$ defines the minimum number of points required within a pre-computed fixed average distance (d_f) based on average data density. The data cluster is extracted by searching

neighboring points within a fixed distance ($\beta \cdot d_f$) using a user defined β. The process ensures that the expansion process have sufficient marching seeds to extract large clusters. $FDN_{threshold}$ is also used to compare the data density and detecting outliers. The points with the least neighboring point count are defined as outliers and are removed and the remaining data points represent data clusters.

Algorithm 5.3:   Cluster Refinement

Input : Rough Data Cluster [$\mathbf{C}_h$] for Hue h, $FDN_{threshold}$;
   n_h=size($\mathbf{C}_h$); K_p-*neighboring finding function*
   FDN_p = No. of neighboring points around a point (p) within a fixed distance(d_f);
   [$\mathbf{FC}_h$] = final cluster; Distance Factor =β; Average Distance= d_f

Outlier Removal

for i=1 to n_h
   if $FDN_i \leq FDN_{threshold}$
   Remove p_i; Update [$\mathbf{C}_h$];
end *for*

Refined Cluster Extraction

for i=1 to size($\mathbf{C}_h$)
   $p_i \xrightarrow{\text{add}} [\mathbf{FC}_h]; p_i \xrightarrow{\text{remove}} [\mathbf{C}_h]$
Call Function **MngCluster** (p_i, $\mathbf{C}_h$)
end *for*
   Function **MngCluster** (p_i, $\mathbf{C}_h$)
      $\{K_{pi}\} \longleftarrow K_p(p_i)$
      for j=1 to size$\{K_{pi}\}$
         Compute $d_{ij}=(p_j-p_i)$
         if $d_{ij} \geq \beta \cdot d_f$
         $p_j \xrightarrow{\text{add}} [\mathbf{FC}_h]; p_j \xrightarrow{\text{add}} TempArray;$
         $p_j \xrightarrow{\text{remove}} [\mathbf{C}_h]$
        else Continue
      end *for*
         while *TempArray* is not empty **Do**
            T=TempArray(1)
            Delete TempArray(1) ; Call **MngCluster** (T, $\mathbf{C}_h$)
         end *while*
         Return;

5.3.3   Color based segmentation results and discussions

The proposed color based approach is implemented on a colorized data set of (a) a curvature continuous data of a round ball with opposite quadrants characterized by the same color and (b) an office interior environment involving multiple objects and a few colored sheets on the

walls. Figure 5.9 shows the results of extracted clusters from the ball data (β=1.5, h=120º and 240º) and Figure 5.10 and Figure 5.11 shows the rough and refined clusters of the room scanned data (h= 0º, 120º, and 240º, β =2, d_f =1.5cm, FDN$_{threshold}$=5, S$_r$=3%), which are extracted from the office scan data using color based rough clustering and subsequent refinement by spatial proximity investigation using FDNs, respectively. The wall data is not shown in Figure 5.11 to enhance the visual clarity.

The proposed methodology extracts unique data clusters based on a HSV color model. Initial rough clusters, extracted with a large permissible variation in hue component (20-25º), and followed by a concise cluster refinement using restricted FDN$_{threshold}$, parameters is an effective cluster extraction approach. A multiple factor (β) of 1.5-2 extracts large homogeneous clusters. Further, unique data clusters can be extracted even from curvature continuous regions (quadrants, Figure 5.9) and from geometrically consistent data set (color sheets from walls, Figure 5.10). The observed inconsistencies can be refined to some extent by removing the outliers (Figure 5.11) in the refinement stage, however, some spurious data is also identified in regions with inconsistent color information and lack of color contrasts. The extracted data set represents objects which may be further segmented based on geometrical properties if required for reverse engineering. However, here the main objective was to extract data clusters for quick visualization and scene understanding of the scan domains and the proposed approach can be effectively used for such applications.

This pure color-based clustering algorithm segments the point clouds into individual clusters representing individual shaped objects. A robust HSV color model helps in devising a hue-based differentiating criterion to formulate non-recursive clustering methodology.

Although the approach generates clusters representing different hues in the scanned scene, there are few limitations of this approach. The approach works better where the domain have distinct contrast colors and not grey, black and white regions. Further, areas from different objects with similar hue values are often clubbed together into a single cluster due to the absence of unique differentiating parameters in such regions e.g. floor and person data set. Further, no distinction between simple (planar) and complex (interior objects) geometries have been made during the segmentation process. To avoid this total color dependence and extract planar as well as complex data clusters from this unified data set, a more robust hierarchical approach is proposed as follows.

(a) Original ball surface

(b) Ball point cloud data (*x y z*)

(c) Segmented data set (*h*=120°, Δ*h* =6°)

(d) Segmented data Set (*h*=240°, Δ*h* =6°)
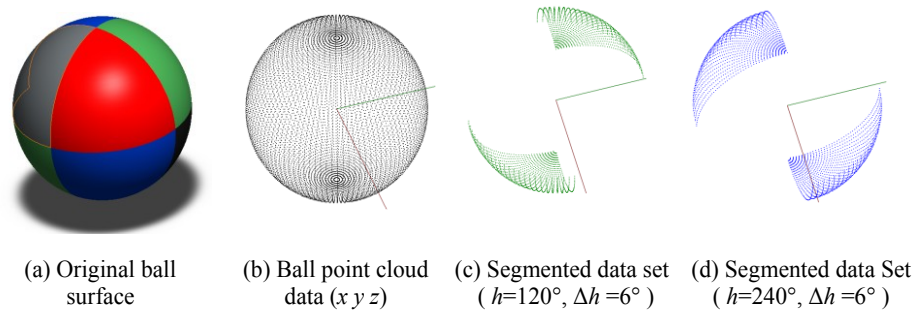
Figure 5.9: Point cloud clustering results of a colored ball (a) Colored ball surface (b) Point cloud data of the ball and its segmented results using the proposed algorithm based on (c) Green and (d) Blue hue parameters.
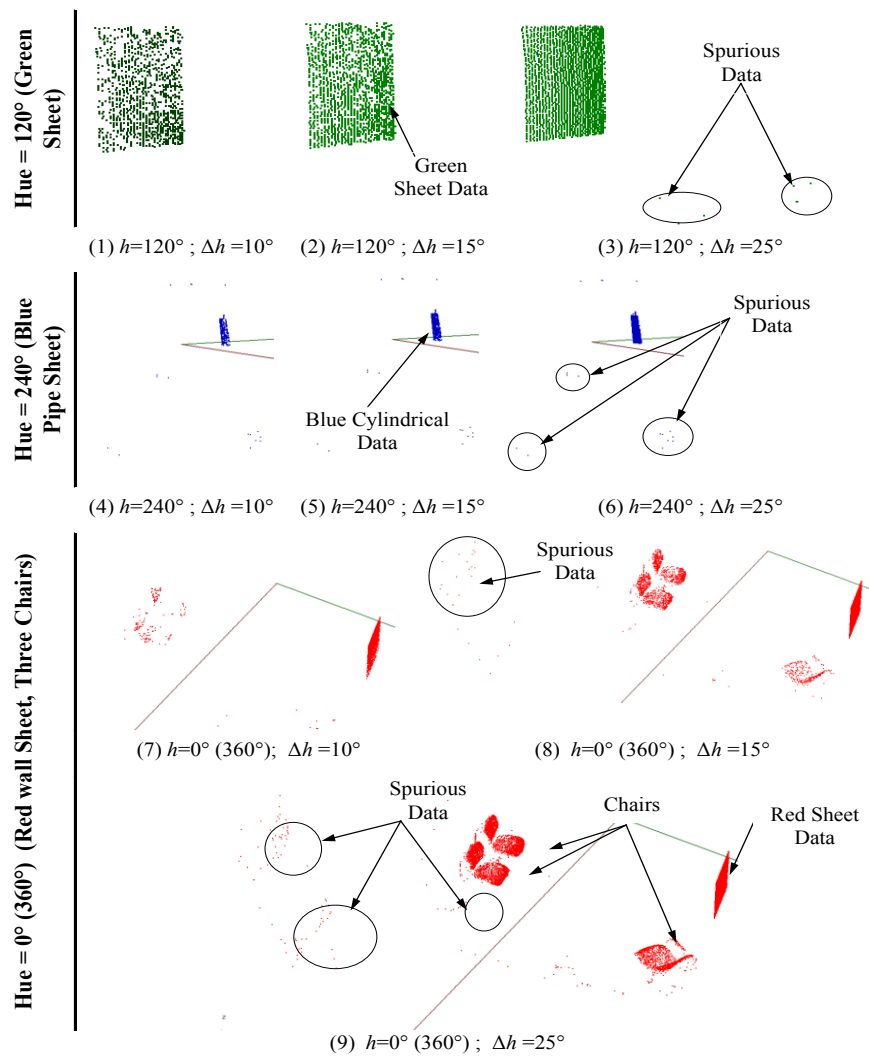
Figure 5.10: Point cloud clustering results of a room scan using the proposed algorithm for (1–3) Green (4-6) Blue and (7-9) Red hue with corresponding clustering parameters.
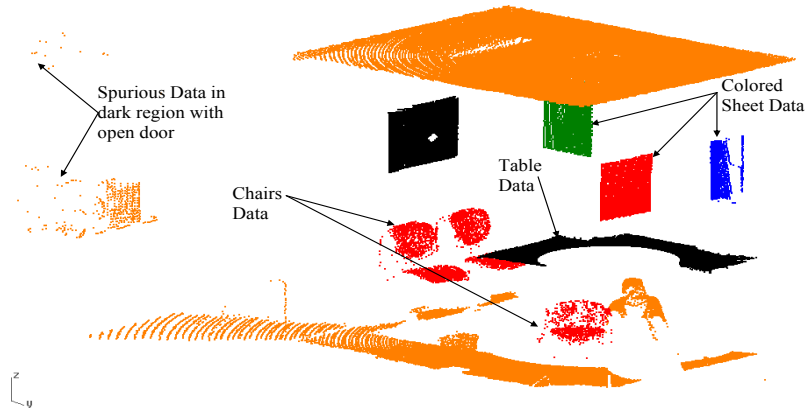
Figure 5.11:   Refined interior clusters, extracted using the proposed region growth and FDN investigation ($\beta=2$; $S_r=3\%$, $d_f=1.5$cm.)

5.4   Hierarchical Segmentation Algorithm

The point cloud in building interiors can be divided into two main categories based on their geometric shape complexities: (1) large regions with planar geometries such as walls, floors, and ceilings, and (2) small regions representing complex, freeform interior objects. It is difficult to develop a single comprehensive clustering algorithm that will satisfactorily partition such real-world scenes having multiple objects with vast variety of surface complexity. However, dividing the segmentation task into stages can simplify the problem to some extent. Further, combining color and geometry characteristics can be used to formulate a robust segmentation strategy to handle spatial uncertainties.

In the proposed hierarchical segmentation approach, a shape based hierarchy is proposed to identify planar and complex data regions in two successive stages. It simultaneously exploits the combined color and geometry based segmentation strategies to formulate a robust algorithm. In the first hierarchical step, large regions of the acquired data cloud representing flat regions are identified using constraints that combine color and local measure of planar alignment. The remaining data points are then assigned to individual clusters based on local surface normal and hue deviation information. A suitable seed selection strategy and cluster expansion method (Sections 5.4.1and 5.4.2) is used to extract accurate data clusters in two hierarchical stages (Sections 5.4.3 and 5.4.4).

An effective seed selection and cluster expansion process will enable the 3D point cloud to be segmented into a simplified, but accurate, representation of the various contiguous surfaces that comprise the scanned scene. Random seed selection followed by location optimization (Hoover, Jean-Baptiste, Jiang *et al.*, 1996; Wani and Arabnia, 2003) is often used to improve the overall segmentation accuracy but this is a very computationally expensive process. Selecting too many seed points that lie along transition zones (cluster boundaries) and unreliable cluster expansions often leads to over-segmentation. Thus, it is essential to devise effective strategies for accurate seeds selection and cluster expansion process.

5.4.1   Seed selection

The main challenge in selecting appropriate seeds in colorized point clouds is to ensure that the identified seed point has the same color and geometric properties as its immediate surroundings and it does not lie along a cluster boundary. The seeds can be selected manually for large clusters or can be selected automatically for interior objects.

In this regard, the hue and geometric properties of each seed candidate are compared with its K_r-nearest neighbors, and the average deviation from this group is then used to determine its suitability as a seed point. This is accomplished by first computing the average hue deviation (Δh) for the selected point. If the computed value is within an acceptable limit, the neighborhood around the identified point is then checked for its geometric disparity using the *average angular residual value* (a_{rp}). This parameter represents an average deviation between the *normal vectors* of the seed point (n_p) and its K_r-nearest data points (n_j) as follows: -

$$a_{rp} = \frac{\sum_{j=1}^{K_r} f(n_p, n_j)}{K_r} = \frac{\sum_{j=1}^{K_r} \lambda_{n_p,n_j}}{K_r} \quad \text{where } \lambda_{n_p,n_j} = \left| \cos^{-1}(n_p \bullet n_j) \right| \quad (5.1)$$

Points with large angular residuals ($a_{rp} > 30°$) often lie on the region boundaries or surface edges of building interiors, and are not good candidates for seed points. Thus, if a_{rp} is large then another point from same hue set must be picked and checked for its suitability as the seed point.

In other words, a data point from the colorized 3D cloud is selected as a seed point only if exhibits hue ($\Delta h < 10°$) and geometric ($a_{rp} < 10°$) coherence with its immediate neighbors as illustrated in Figure 5.12. The hue parameter of point p_1 in Figure 5.12(a) is consistent with its surroundings; it lies in a region with significant surface normal deviations and is, therefore, not a suitable candidate for a seed point. However, another point (p_2) from the same hue region (Figure 5.12(b)) satisfies both hue and geometric deviation constraints and can be successfully used as a seed point. To prevent multiple seed selection from the same group, the initiated cluster is completed before selecting the next seed.

A further advantage of a color-based seed selection procedure is that it can initiate additional seed points from regions representing different objects having similar surface geometry such as a door or art work on the wall. A pure geometry based algorithm will combine such segments into a single identified cluster. In contrast, the proposed similarity criterion will not allow the inclusion of points with different hue in the geometrically similar regions, which will be available for future seed selection.
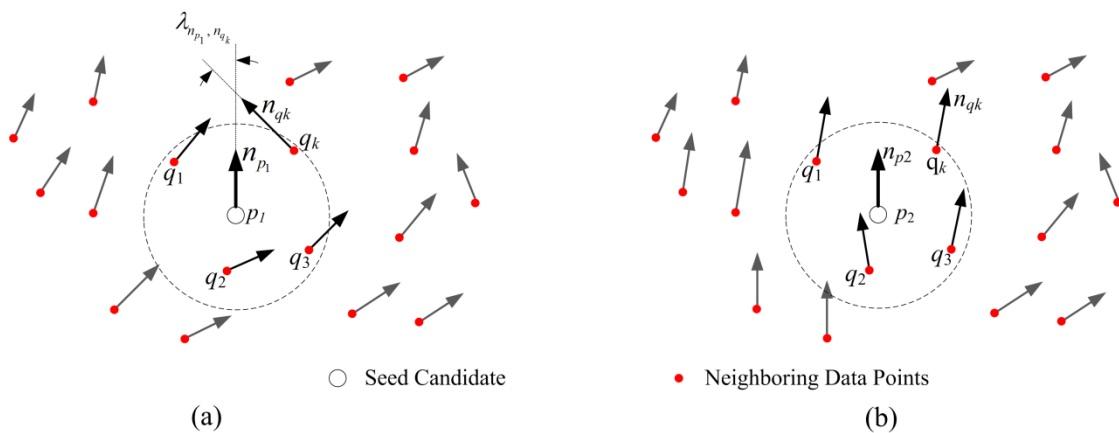
Figure 5.12:   Selection criterion of initial seed points (a) Bad seed data point (b) Good seed data point.

5.4.2   Cluster expansion

The algorithm forms clusters by analyzing the data points within a *fixed neighborhood distance* (d_f) around the selected seed point for hue and geometric coherence. Those *fixed distance neighbors* (FDNs), that satisfy the similarity constraints are included in the cluster and are used as anchor points to investigate subsequent FDNs around them from the

remaining data points. This process continues until expanding cluster is not able to find any candidate point to investigate within its FDNs. The clusters are extracted sequentially and the process continues until most of the data points are segmented into clusters.

The *fixed neighborhood distance* (d_f) is defined adaptively for each data cluster based on the local data density, which varies due to the relative position of the scanner from the scanned surfaces. This *fixed neighborhood distance* (d_f) is computed as the multiple of *adaptive average density distance* (AADD) and a user defined, non-dimensional *Distance Factor* (DistF), where AADD is defined as the arithmetic mean of distances between the seed point and its *N*-nearest neighbors ($d_1, d_2, d_3, ...d_N$) in the scanned data set as shown in Figure 5.13. A suitable *distance factor* (DistF = 1-2) ensures that the cluster expansion process continuously finds sufficient neighboring points within its fixed distance to define new anchor points and prevents pre-mature cluster termination and thus, works in sparse and dense data regions, alike. On the other hand, a restricted *distance factor* (DistF) also prevents the cluster expansion over a discontinuous boundary and prevents under-segmentation. For example, two probable seed candidates-p_1 and p_2 in two dissimilar regions (Figure 5.13) compute a different value of AADD and their corresponding *fixed distance* (d_f). Each cluster uses its own fixed distance value computed at its respective seed point. As a result, the clusters will be extracted from both these regions irrespective of their spatial density.
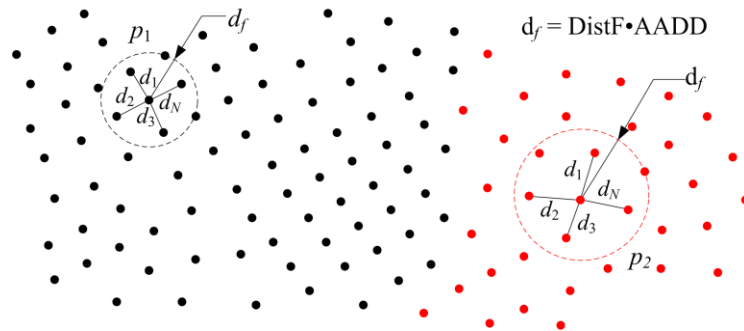
Figure 5.13:   Computation of fixed neighborhood distance (d_f) from local data density defined in terms of AADD.

This color-assisted, geometry-driven methodology is used in two hierarchical stages of this algorithm to extract planar (Section 5.4.3) and complex free-form (Section 5.4.4) clusters from a colored scanned data set of building interiors as shown in Figure 5.14 and described in the following sections.

5.4.3 First stage of hierarchical clustering (planar shapes)

The first stage of the hierarchical clustering algorithm exploits the fact that there are often large planar geometries (walls, doors, windows, floors and ceilings) in building interiors, therefore identifying these planar regions greatly reduces the volume of data to be analyzed in the second stage where clusters associated with multiple-interior objects are extracted. Large planar regions are initially identified from the cloud using constraints that combine hue and a measure of local planarity called a *planar alignment factor* (*PAF*).

The *PAF* is a dimensionless parameter that reflects whether a sampled point (p_i) in a predefined region of the cloud lies on or near a plane fitted through all points in the selected region. This measure enables the clustering algorithm to quickly identify captured coordinate points that have been originated from flat interior surfaces such as walls, ceilings or floors. The *PAF* parameter is computed by first fitting a plane through the coordinate points that reside within a predefined neighborhood of a selected seed point p_s. A region with a large number of neighboring points (K_r) is used to fit the plane and determine the corresponding surface normal ($\hat{n}$). Each neighboring data point p_i (where $i = 1, 2, ..., r$) is then checked to see if it lies on the plane defined by $\hat{n}$. This is achieved by looking at the immediate neighborhood of p_i (Figure 5.15) and determining the cumulative mean ($\theta^i_{avg} = \Sigma\theta_{ik}/k$) of the angular deviations of coordinate vectors from the ith point to its kth neighbor and the pre-computed surface normal ($\hat{n}$).

The *planar alignment factor* (*PAF_i*) for point p_i is defined as the ratio of the threshold angular value ($\theta^{threshold}_{avg} = 80°$) and the complementary deviation angle ($\pi/2 - \theta^i_{avg}$). Mathematically, this is given by

$$PAF_i = \frac{\theta^{threshold}_{avg}}{(\pi/2 - \theta^i_{avg})} \tag{5.2}$$

Points with $PAF_i \geq 1$ are geometrically consistent with the locally fitted plane defined by $\hat{n}$ and, therefore, can be clustered together during planar region expansion process. For example, a cluster initiated from seed point p_s in Figure 5.15 includes points p_2 and p_3 but omits points p_4 and p_5 based on their $PAF_i < 1$.

Start

Point Cloud Data Set

Colored Images

Panoramic Color Image Extraction and Range Synchronization

Colored Point Cloud Data Set

(x y z r g b)

Color Model Conversion (RGB to HSV)

(x y z h s v)

First Hierarchical Level
(Extraction of large planar clusters with simple geometries)

Geometric Constraints (*Dir. Normal and Planar Alignment Factor*)

(x y z *PAF* h s v)

$\theta_{avg} = \dfrac{\sum \theta_{ij}}{K_p}$

$PAF = \dfrac{\theta_{avg}^{threshold}}{(\pi/2 - \theta_{avg})}$

Spatial Position+PAF+Color

Seed

Neighbouring Point Search

Region Growth

Region Growth Similarity Criterion

Geometric Similarity (PAF Value)

Combined Similarity (PAF valu, Hue dev.)

Remaining Points

Large Planar Segments

Walls, Floors, Ceiling, Doors, Windows and other Planar Regions

Geometric Indecisiveness

Uncertain Interior Data

(x y z h s v)

Compute Approximated Normal (n_X, n_Y, n_Z)

Second Hierarchical Level
(Extraction of small, multiple interior clusters)

Spatial Position+ Normal+Color

[x y z n_x n_y n_z h s v]

Region Seed Selection

Region Growth

Region Growth Similarity Criterion

Geometric Similarity (Normal dev.)

Combined Similarity (Normal, Hue dev.)

Remaining Points

Multiple Interior Clusters

Segmented Data Representing Different Clusters/Objects

Figure 5.14:   An overview of the overall hierarchical data clustering methodology.
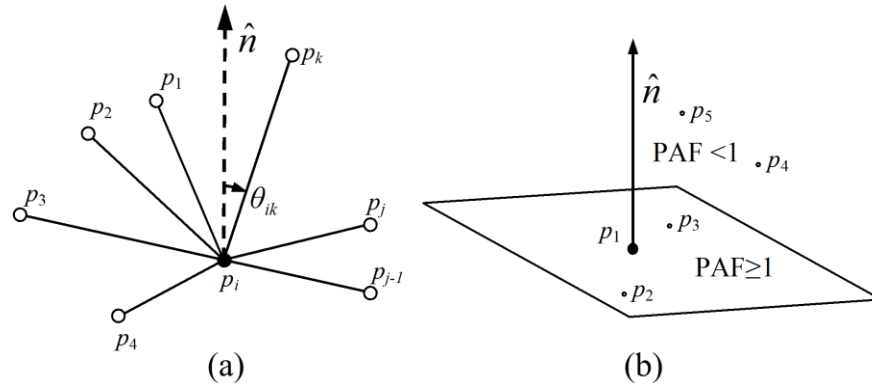
Figure 5.15:   (a) Angular deviation calculations of neighboring vectors (b) Data clustering based on angular deviation threshold.

The PAF_i parameter on its own will not be able to distinguish between multiple planar objects that may overlap (e.g. posters on walls) or lie in close spatial proximity (e.g. planar light fixtures).   This under-segmentation problem can be corrected, in part, by using a combined hue and geometry-based similarity criteria.   In other words, geometrically consistent points ($PAF_i \geq 1$) on flat surfaces but with a significantly different hue will not be included in the cluster as it grows.   These neglected points will remain available in the cloud for subsequent seed selection and/or assignment to another expanding cluster region.   The region-growing algorithm include a point p_i in the expanding cluster only if (a) $PAF_i \geq 1$ and (b) its hue variation is within a threshold value (h^{thres}) as described in first hierarchical clustering algorithm (Algorithm 5.4). In this manner, unique clusters for doors, windows or display posters on the walls, contrasting colored rugs and light panels on the ceiling can also be efficiently extracted.

Algorithm 5.4:   First hierarchical clustering algorithm (Point data clustering based on PAF and hue constraint)

Parameters: Point cloud $\{P\}$, Seed Points $\{P_s\}$, Remaining Points$\{RP\}$
Neighboring point finding function $K_p(p), K_r, K_p, \text{h}^{thres}, \theta_{avg}^{thres}$

Initiate: Initial Cluster with its seed point.
for each seed point $p_s \in \{P_s\}$ ***Do***

$\quad p_s \xrightarrow{\ add\ } \{C_s\}$ $\qquad\qquad\qquad$ % (Cluster initiation)

$\quad$ Compute Normal Vector $\hat{n}$ from K_r -nearest points using plane fitting

$\quad$ Remaining Points $RP = \{P\} - p_s$

Call Function **MngCluster1**($p_s, \hat{n}, RP$)

$\quad$ *if* size $\{C_s\} < 50$;
$\quad\quad \{C_s\} \xrightarrow{\ add\ } \{RP\}$; $s{+}{+}$
$\quad$ *end* if
end for

Function **MngCluster1**($p_s, \hat{n}, RP$)

$\quad \{K_p\} \xleftarrow{\ find\ } \Psi(p_s)$ $\qquad\qquad$ % (Find K_p-neighboring finding function $\Psi(p_s)$ based on AADD)

$\quad\quad$ *for* q=1 to size$\{K_p\}$

$\quad\quad\quad$ call Function **PAFCompute** (p_s, RP, $\hat{n}$)

$\quad\quad\quad\quad$ *if* $PAF_q \geq 1$ and $\{(h_q - h_s) \leq \text{h}^{thres}\}$);

$\quad\quad\quad\quad\quad p_q \xrightarrow{\ add\ } \{C_s\}$ $\qquad$ % (Add p_q to cluster C_s)

$\quad\quad\quad\quad\quad p_q \xleftarrow{\ remove\ } \{RP\}$ $\qquad$ % (Remove p_q from remaining points RP)

$\quad\quad\quad\quad\quad p_q \xrightarrow{\ add\ } TempArray$ $\quad$ % (add p_q to a Temporary array for expanding cluster C_s)

$\quad\quad$ *else* Continue
$\quad\quad$ *end* for
$\quad\quad$ *while* TempArray is not empty ***Do***
$\quad\quad\quad$ *for* m=1 to size(TempArray)
$\quad\quad\quad\quad$ T=TempArray(1)
$\quad\quad\quad\quad$ Delete TempArray(1)
$\quad\quad\quad\quad$ Call **MngCluster1**(T, RP)
$\quad\quad\quad$ *end* for
$\quad\quad$ *end* while
$\quad$ **Return**;

$\quad$ Function **PAFCompute** (p_i, RP, $\hat{n}$)

$\quad\quad$ Initiate $\theta^i = 0$; $\{K_p\} \xleftarrow{\ find\ } K_p(p_s)$

$\quad\quad$ *for* j=1 to size$\{K_p\}$
$\quad\quad$ Compute $p_j - p_i$

$$\theta_{ij} = \cos^{-1} \frac{(p_j - p_i) \cdot \hat{n}}{\left| p_j - p_i \right|} \qquad \begin{cases} \theta_{ij} = \theta_{ij} & if \quad 0 \leq \theta_{ij} < \pi/2 \\ \theta_{ij} = \pi - \theta_{ij} & if \ \pi/2 \leq \theta_{ij} \leq \pi \end{cases}$$

$$Sum\,\theta^i = \theta^i + \theta_{ij} \,;\, \theta_{avg}^i = Sum\,\theta^i \Big/ size(K_p)$$

$\quad\quad$ *end* for

$\quad\quad$ Planar Alignment Factor $(PAF_i) = \dfrac{\theta_{avg}^{threshold}}{\pi/2 - \theta_{avg.}^i}$

$\quad$ **Return**;

5.4.4 Second stage of hierarchical clustering (complex surfaces and objects)

The remaining data point set in the 3D cloud represent complex interior objects and, therefore, must be assigned to new clusters using robust geometric and hue similarity criteria. This is the role of the second stage in the hierarchical clustering process, where normal vectors and hue deviations are used to measure the cluster coherence.

5.4.4.1 Computation of geometric parameters

To minimize the effect of noise in the data set, an approximated normal vector $\hat{n}_i$ is computed for each remaining data point p_i by fitting a plane through its nearest neighbors. However, utilizing the nearest neighbors over the same fixed distance for all points often results in erroneous normal vectors due to significant variations in data density. The problem is addressed by selecting the neighboring points based on the adaptive average density. Thereby points within the adaptive fixed neighboring distance (d_f) are used to compute point normal as shown (Figure 5.16). This fixed neighboring distance (d_f) reduces in high density zone and vice-versa, which helps in generating consistent point normal for each data point. Thus, each remaining point $p_i \in$ [RP] in the cloud has a distinct spatial location (x y z), approximated unit normal (n_x n_y n_z), and color information (h s v).
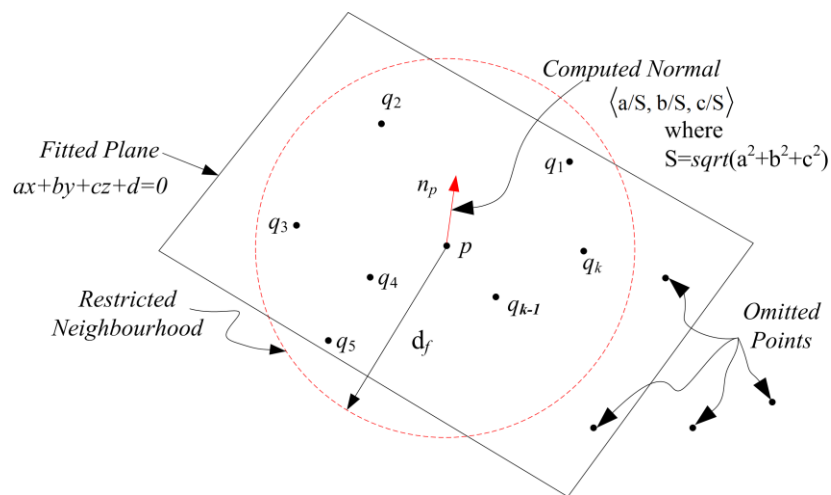
Figure 5.16: Normal vector computation from restricted neighborhood data points.

The approximated surface normal for interior point cloud is shown in Figure 5.17, which is used to extract individual clusters in the second hierarchical stage using combined hue and normal deviation criteria.

(a)   (b)

Figure 5.17:   (a) Interior point cloud data and (b) Interior points along with their approximated surface normal.

5.4.4.2 *Data clustering based on color and surface normal*

This step extracts coherent clusters using a combined hue and surface normal-based similarity constraint. A seed point (p_s) initiates a new cluster and expands it by investigating its K-nearest neighbors (p_1, p_2, …..p_K) using within the *fixed investigating distance* (d_f) around the point. The similarity constraint criteria requires the hue ($\leq \mathrm{h}^{thres}$), normal vector deviation ($\leq \lambda_{normVector}^{thres}$) and the spatially conformity ($\leq d_f$) of the data point to be within permissible limits for its inclusion in the expanding cluster. The clusters are extracted sequentially using the *adaptive average density distance* (AADD) approach described in Section 5.4.2. Second hierarchical clustering algorithm (Algorithm 5.5) summarizes the steps of this hue and geometric similarity based segmentation stage. The results are compared with the extracted clusters, when the hue or geometric similarity constraints are implemented individually and are compiled in the next section.

Algorithm 5.5: **Second hierarchical clustering algorithm (Point data clustering based on normal and hue constraint).**

Parameters: Colored Point cloud data [**P**] with point normal, Similarity threshold values h^{thres}, $\lambda_{normVector}^{thres}$

Neighbor finding function $K_p(p)$

for each seed point (p_s)

Do

Initialize: Cluster region

   Call Function **MngCluster2** (p_s, RP)

    if size $C_s < 50$;

     $\{C_s\} \xrightarrow{\quad add \quad} \{RP\}$; s++

    end *if*

end *for*

Function **MngCluster2**(p_s, RP)

   $\{K_r\} \xleftarrow{\quad find \quad} K_p(p_s)$ % (Find K_p-neighboring finding function $K_p(p_s)$ based on AADD

   for q=1 to size$\{K_p\}$

      Compute normal Vectors (n_q) for each data point p_q

       if $(h_q - h_s) \leq h^{thres}$ && ($\cos^{-1}(n_{p_s}.n_q) \leq \lambda_{normVector}^{thres}$);

         $p_q \xrightarrow{\quad add \quad} \{C_s\}$ % (Add p_q to cluster C_s)

         $p_q \xleftarrow{\quad remove \quad} \{RP\}$ % (Remove p_q from remaining points RP)

         $p_q \xrightarrow{\quad add \quad} TempArray$ % (add p_q to a Temporary array for expanding cluster C_s)

       else Continue

       end *if*

   end *for*

       while TempArray is not empty **Do**

         for m=1 to size(TempArray)

         T=TempArray(1)

         Delete TempArray(1)

          Call **MngCluster2**(T, RP)

         end *for*

       end *while*

Return;

5.4.5 Hierarchical clustering results and discussions

The hierarchical clustering algorithm is implemented on the colorized 3D range data of an office interior that includes multiple objects such as chairs, tables, computer monitors, a statue head and a printer. In addition, a few colored sheets with different orientations are also taped on the flat walls to evaluate the suitability of this method in identifying unique clusters from regions with similar surface geometries. The clustered results from each hierarchical stage, with different algorithmic parameters are compiled in the following sub-sections.

5.4.5.1 *Planar shape clustering results*

In the first hierarchical stage, the large planar data clusters (walls, floor and ceiling) are extracted by analyzing the planar alignment factors (*PAFs*) and the permissible hue deviation threshold (h^{thres}) from its neighboring points. The surface normal at the selected seed point is initially computed by fitting a plane through a large number of neighboring points ($K_r = 50$) using a very tight fit ($a_{rp} \leq 10º$), which ensures that the seed point is not selected at a boundary region. The cluster then grows from the selected seed point by investigating the coordinate points in its closest vicinity (DistF = 1.0 - 1.5) using two permissible angular deviation ($\theta_{avg}^{threshold}$ = 7° and 15°) and hue deviation threshold (h^{thres} = 15°) constraints. The number of extracted clusters and their corresponding point counts are computed and tabulated in Table 5.1.

Table 5.1: Number of extracted clusters and points in first hierarchical stage using angular deviation and hue constraint.

Points: 385655		Regions	*No. of Clusters(Cluster points)*		*% Scanned data*	
Outliers: 92			Without Color	With Color	Without Color	With Color
DistF=1.0; $\theta_{avg}^{threshold}$=7° h^{thres}=15°	1	Floor	1(9594)	1(9741)	2.48	2.52
	2	Ceiling	1(92719)	2(110290)	23.97	28.52
	3	Walls	4(170887)	9(172313)	44.19	44.55
	4	Boundary	1(6574)	1(5121)	1.70	1.32
	5	Interior	1(106973)	1(89282)	27.66	23.09
DistF =1.5; $\theta_{avg}^{threshold}$=15° h^{thres}=15°	1	Floor	1(11704)	1(13135)	3.03	3.40
	2	Ceiling	1(121170)	3(131839)	31.33	34.09
	3	Walls	4(178068)	10(179422)	46.04	46.39
	4	Boundary	1(4096)	1(2088)	1.06	0.54
	5	Interior Data	1(67052)	1(60263)	17.34	15.58

Table 5.2: Clustering results of first hierarchical stage.

Similarity Criterion			
DistF =1.0		**DistF =1.5**	
Pure Geometry ($\theta_{avg}^{threshold}=7°$)	Combined Geometry and Hue ($h^{thres}=15°$; $\theta_{avg}^{threshold}=7°$)	Pure Geometry ($\theta_{avg}^{threshold}=15°$)	Combined Geometry and Hue ($h^{thres}=15°$; $\theta_{avg}^{threshold}=15°$)

Floor

Ceiling

Walls

Boundary

Remaining Data

% Data Distribution

Floor Data · Ceiling Data · Walls data · Boundary Data · Remaining Data

Table 5.2 presents a comparison between the clustered results obtained using a pure geometry-based approach and the proposed hue and geometry based approach. An initial set of seed points extracts the large planar clusters from the floor, walls and ceiling. All interior points corresponding to planar regions such as tables are not extracted as no initial seed initiates these data clusters. The use of color helps in identifying multiple clusters from the planar wall regions and ceiling data (Table 5.2, 2nd column). Further, increasing investigation distance (DistF =1.5) helps the expanding cluster to overcome small spatial discontinuities and extract large clusters by including doors into wall clusters and light fixtures and air-ducts into ceiling (Table 5.2, 3rd column). These are individually segmented using color and geometric based segmentation (Table 5.2, 4rd column). The hierarchical approach extracts large and multiple planar clusters from its initial point clouds and a small amount (16-28%) is carried forward to second stage for further classification.

5.4.5.2 *Complex surface clustering results*

The second hierarchical stage works on the remaining points in the 3D data cloud and attempts to extract accurate clusters using normal vector deviation and hue variation criterion. Local normal and color coherence forms the basis of similarity analysis in the second stage along with spatial coherence and hue deviations.

For illustrative purposes, the second stage of the clustering process is implemented on the remaining data points of the test set (DistF = 1.5, $\theta_{avg}^{threshold}$ = 15° and h^{thres} = 15°). It uses (a) a pure-hue constraint, (b) a normal deviation constraint, and (c) a combined normal deviation and hue constraint with an *adaptive average density distance* (AADD) measure using two values of DistF (1.0 and 1.5). The extracted clusters from this second stage are shown in Table 5.3. It is to be noted that the segmented colored data sets only represent the extracted clusters and have no correspondence to the original hue of the object data. The first column shows the results obtained using the hue variation analysis only, where three values of hue thresholds (h^{thres} = 10°, 15° and 20°) are used. The second column represents the data clusters based on the normal vector deviations alone with three normal vector deviation thresholds ($\lambda_{normVector}^{thres}$ =10, 15, 20°) and finally, the third column compiles the results obtained by implementing the proposed hue and normal deviation methodology ($\lambda_{normVector}^{thres}$ = 10°, 15°, 20°; h^{thres} = 15°), simultaneously. To facilitate an effective comparison criterion for these clustering results, the same initial set of seed points are used in all three cases.

Table 5.3: Clustering results of second hierarchical stage.

| Similarity Criterion | | |
Pure Hue ($\lambda^{thres}_{normVector}$=N/A)	Pure Normal Vector Deviation (h^{thres} = N/A)	Combined Color and Normal Vector Deviation (h^{thres}=15°)
DistF=1.0		
h^{thres}=10°	$\lambda^{thres}_{normVector}$=10°	$\lambda^{thres}_{normVector}$=10°
h^{thres}=15°	$\lambda^{thres}_{normVector}$=15°	$\lambda^{thres}_{normVector}$=15°
h^{thres}=20°	$\lambda^{thres}_{normVector}$=20°	$\lambda^{thres}_{normVector}$=20°
DistF=1.5		
h^{thres}=10°	$\lambda^{thres}_{normVector}$=10°	$\lambda^{thres}_{normVector}$=10°
h^{thres}=15°	$\lambda^{thres}_{normVector}$=15°	$\lambda^{thres}_{normVector}$=15°
h^{thres}=20°	$\lambda^{thres}_{normVector}$=20°	$\lambda^{thres}_{normVector}$=20°

5.4.5.3 Hierarchical clustering discussions

It has been demonstrated that the proposed hierarchical approach is effective in extracting accurate data clusters from both planar and geometrically complex regions of a scanned data set of building interiors. The division of the segmentation exercise into two stages is advantageous because a large number of the original scanned points (72 to 84%) correspond to planar regions and segmenting these points in the first stage significantly reduces the size of the remaining data points for the second segmentation stage. Further, the results show that significant improvements in data clustering can be achieved using a combined geometric and color similarity constraint, when compared with clusters obtained using a pure color or geometry-based approaches.

The pure geometry based segmentation in the first hierarchical stage groups the points based on their local *planar alignment factor* (PAF) using a permissible angular deviation value ($\theta_{avg}^{threshold}$). It is evident from Table 5.1 and Table 5.2 that the pure geometry-based approach results in an over-segmented data with numerous clusters. A strict similarity constraint in terms of low permissible angular threshold ($\theta_{avg}^{threshold} < 7°$) generates isolated data clusters and a low neighboring distance (DistF ≤ 1.0) fails to cross small occluded boundaries and thus, leads to under-segmentation of floors, doors and intrusions on walls, light fixtures and air vents in the ceiling. The clustering inconsistencies can be improved with a higher angular threshold value ($\theta_{avg}^{threshold} = 10°$-$15°$) and distance factor (DistF = 1.5). Unfortunately, increasing these parameters tend to combine dissimilar regions and fails to differentiate unique hue objects having similar geometries.

The combined hue and geometry-based similarity criterion successfully extracts multiple planar clusters (Table 5.2) from geometrically similar regions representing colored sheets on the wall, light and vent sections in the ceiling and doors in the walls. The additional differentiating parameter (hue) allows a larger investigation distance (d_f), that helps in checking more neighboring points for parametric coherence. This prevents over segmentation by combining multiple regions, separated by occlusions such as a protruding wall or door boundary and generates saturated and large contiguous planar data regions. Although, few of these extracted sheet clusters (e.g. S-1, Table 5.2) have missing sharp boundaries, however, it is due to the reflective properties and hue uncertainty in these regions, which can be improved by accurate capturing of hue parameters during range scanning.

A pure hue-based clustering approach fails in the regions lying very close to the scanner (computer table), where the captured data does not have any color information. The absence of color generates multiple, over-segmented data clusters representing computer monitor, man, printer and dustbin etc. This over-segmentation issue can be improved to some extent using large distance factor (DistF) or hue deviation threshold (h^{thres} = 10°), however, the segmentation pattern remains practically similar. Further a relatively higher hue threshold (h^{thres} =15°-20°) leads to under-segmentation by combining multiple regions. As can be seen from Table 5.3, the color-based clustering generates best results with higher DistF (= 1.5) and intermediate hue deviation thresholds (h^{thres}=10°-15°). However, the pure hue-based approach is not effective in segmenting complex shaped objects and regions with hue uncertainty (S=0, V=1; S=0, V=0). Thus, the hue parameter alone cannot robustly extract accurate clusters from a multi-object scenario.

Normal deviation criterion on the other hand (Table 5.3, column 2) extracts refined but un-saturated clusters using a lower value of DistF and permissible normal vector deviation threshold ($\theta_{avg}^{threshold}$), especially from the regions with lower data density such as a computer screen and a statue placed on the table. Increasing the permissible thresholds improves clustering accuracy in the region but tends to unite multiple clusters at elevated normal vector deviations ($\theta_{avg}^{threshold}$=20°). On the other hand, increasing the DistF value has a lesser impact on the cluster extraction efficiency from the sparse data set and cluster unification still persists at large angular deviation values, as before.

The proposed approach generates better segmentation results (Table 5.3, column 3) by combining hue as well as normal vector deviation constraints. These dual similarity constraints can effectively compensate the geometric uncertainty in regions with multi-objects and overlapping regions. Accurate data clusters are extracted even from the sparse data regions (computer screen and statue head) around the discontinuous boundaries using a restricted investigating distance (DistF=1.0) with limited threshold values ($\lambda_{normVector}^{thres}$=15°, h^{thres} =10°). Increasing the investigating distance (DistF=1.5) effectively expands the cluster to its connected segments (e.g. arm rest of the chair) and as such, extracts cumulative data sets for objects, which will be quite effective for pattern recognition and object identification applications. It has been observed that a local density-based (DistF=1.0-1.5) investigating

distance with low similarity threshold values ($\lambda_{normVector}^{thres}$=10º-15º and h^{thres}=10°) is most appropriate algorithmic parameters for extracting accurate data clusters without encountering any major segmentation inconsistencies. A higher permissible angular deviation value ($\lambda_{normVector}^{thres} \geq$15º) with a greater investigation distance (DistF≥1.5) tends to combine multiple regions with poor hues (grays, blacks and white). This is critical at discontinuous boundaries where the geometric properties do not change drastically. Hence, it is important to identify such discontinuous boundaries, which is addressed in the next chapter.

It can be stated that the proposed hue and geometry based segmentation methodology is very effective in extracting accurate data clusters from point clouds representing multiple objects of building interiors. The extracted data clusters are closely associated with the object's shape in the scanned scene and therefore can be used for pattern recognition or object identification applications. The segmentation thresholds can be controlled to achieve desired segmentation level for extracting objects or individual surfaces. However, the approach shows some inconsistencies at discontinuous boundaries (computer screen and chair edges, Table 5.3) and regions with hue uncertainty (over-exposed colored sheet regions, Table 5.2). Thus, it will be beneficial to identify discontinuous boundaries in the point cloud to detect occluded geometries.

5.5   Concluding Remarks

The initial, pure color based clustering algorithm helps in segmenting the data clusters which enhances the scene clarity and helps in quickly identifying the multiple clusters in the scanned scene. These clusters are a good representation of the scanned scene and help the user to quickly identify the objects present in the scanned scene. However, it works best for a scanned scene with contrast colors. It also requires a compensating parameter to ensure that multiple regions from different objects are not merged together.

This geometric complexity based hierarchical clustering approach improves the color based algorithm and it can segment the cohesive data sets of occupied building interiors. A combined hue and geometry based similarity criterion was used to segment data sets representing multi-objects and overlapping regions. This approach instigated an effective seed selection strategy and permitted the extraction of accurate date clusters, even from

geometrically consistent data regions. The shape based hierarchy was used to extract large planar clusters and complex freeform shape clusters in two sequential steps. The first hierarchical stage assigned about 72-84% of scanned data to planar clusters using planar alignment factors and hue-based constraints and the remaining data points were further segmented in the second stage using normal and hue-based approach. The adaptive average density distance approach extracted accurate data clusters from dense as well as sparse data regions. The availability of this additional hue based similarity criteria helps in investigating a greater number of neighboring points with uncertain geometric parameters and extracts saturated freeform clusters from geometric similar regions and overlapping regions. A suitable set of parameters (DistF=1-1.5; $\lambda_{normVector}^{thres}$=10°-15° and h^{thres}=10°) with adaptive average density distance provides a reliable mean to control the level of segmentation. The segmented clusters can be used in various subsequent post-processing applications such as pattern recognition, object modeling and data simplification. Small clustering inconsistencies are observed in the regions involving uncertain hues and discontinuous boundaries.

CHAPTER 6   DISCONTINUOUS BOUNDARY DETECTION

6.1   Introduction

The occluded data set is an inherent characteristic of the point cloud obtained by scanning complex and occupied building interiors with multiple objects and diverse geometries. The overlapping geometries in the scanned scene generate large void regions and numerous discontinuous sections at the occluded boundaries. Moreover, the lack of known shape semantics is deterrent in differentiating overlapping data regions with similar geometries.

Most existing boundary detection algorithms (Alharthy and Berthel, 2004; Barnea, Filin and Alchanatis, 2007; Rottensteiner, Trinder, Clode *et al.*, 2005) are demonstrated on exterior point clouds obtained from airborne LIDAR scanning and very few methods (Budroni and Böhm, 2009; Ning, Zhang, Wang *et al.*, 2009; Vosselman and Dijkman, 2001) have addressed issues of point clouds representing multiple objects of building interiors. In case of exterior point clouds obtained from airborne scanner, the discontinuous boundaries are often extracted through depth analysis of the captured data, which cannot be extended to point cloud data set obtained by a stationary scanner used typically in scanning of interior buildings, where the scan depth vary continuously all along the domain.

Discontinuous boundaries are often identified as the border of the void region that are not captured during scanning process due to object occlusions, surface characteristics or feature complexity. Most of these methods (Borodin, Novotni and Klein, 2002; Ju, 2004; Liepa, 2003; Weyrich, Pauly, Heinzle *et al.*, 2004) require the construction of intermediate surface models to identify discontinuous boundaries with the main objective of filling holes. The identified void regions are often filled by interpolating the surrounding data set.

Mesh based models (Barequet and Kumar, 1997; Borodin, Novotni and Klein, 2002; Ju, 2004; Turk and Levoy, 1994; Weyrich, Pauly, Heinzle *et al.*, 2004) are normally used as intermediate surface models, where focus is to identify and repair the model to generate water tight geometric models. The limiting factor of extending this approach lies in the fact that the captured point cloud data set of building interiors represents multiple objects and generating models from this cohesive data is significant only if they represent specific objects or distinct features. Filling holes and generating models does not help in this case. The data

requires to be converted into building information model (BIM) and thus, an effective approach is to identify the occluded boundaries or boundary data directly from the point cloud.

The traditional surface boundary detection algorithms (Sampath and Shan, 2007) (Boltcheva, Bechmann and Thery, 2007; Wei, 2008; Xianfeng, Xioguang, Fan *et al.*, 2008) mainly exploit geometric characteristics of the scanned data and use their disparity to identify region boundaries from a contiguous data set. These approaches use the fact that the geometric properties such as surface normal or curvature changes drastically at a region boundary. The primary objective of these algorithms is to sub-divide the data set into individual segment to facilitate easy surface fitting and model regeneration. Extending these geometry-based approaches to identify discontinuous boundaries does not suffice as the computed surface properties at a discontinuous boundary are often consistent. The discontinuous boundaries in the data set are identified as the continuous edges of the void region, identified with in the captured data set. These discontinuous boundaries are composed of edges of the triangles, which are not shared by any other triangle (Figure 6.1(a)).

Computation of such intermediate, approximated surface models from the point cloud data set is necessary to identify these discontinuous boundaries; however, it demands significant computational cost for unified point cloud of building interiors. Moreover, surface modeling from cohesive data set representing multiple objects is neither beneficial nor required. Other approaches of identifying surface boundaries use point normal, which changes drastically at surface boundaries, but the change is not that rapid at discontinuous boundary (Figure 6.1(b)) of the void due to absence of data points on the other side. Such voids are often encountered when a scanner captured a domain involving multiple overlapping objects (Figure 6.2).
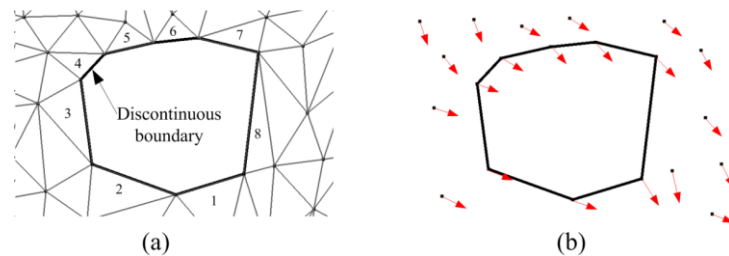
Figure 6.1: (a) Discontinuous boundary detection from triangulated data set (b) Approximated surface normal at point cloud data set.
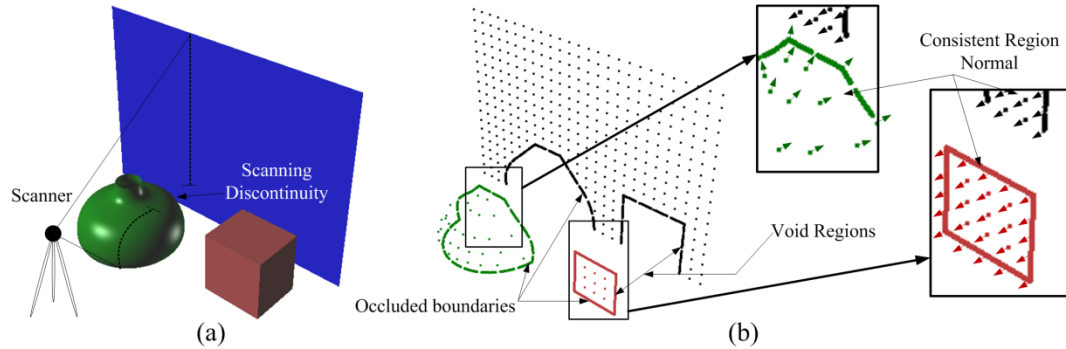
Figure 6.2: (a) Range scanning of multiple objects (b) Scanning characteristics (occluded boundaries), void regions and consistent geometric parameters in discontinuous boundary vicinity.

Thus, a direct point based algorithm is an effective approach in identifying these discontinuous boundaries without generating any intermediate surface model. This chapter proposes a side-ratio constraint methodology to identify the spatial data points at the occluded boundaries directly from the point clouds without generating any intermediate surface models. The novelty of this proposed algorithm lies in the fact that the boundary data points are identified directly from the point cloud data set unlike most other algorithms, which needed an intermediate surface model for subsequent surface completion and other post-processing tasks. It analyses the progressive scan lines and identifies the boundary data points based on their spatial position coherence.

6.2 Discontinuous Boundary Detection

Normally, the point cloud obtained by scanning multiple objects simultaneously in a building interior environment generates point clouds with discontinuous sections and voids regions. These occluded boundaries and voids correspond to the spatial discontinuities in the scanned surfaces. Identification of these spatial discontinuities is essential to accurately control the growth of the expanding cluster for successful segmentation of cohesive data set of occupied building interiors, especially in spatially apart but geometrically similar data patch. Our approach attempts to identify these boundary data points by exploiting this positional discontinuity of the point cloud of scanning lines progressively. The concept of side-ratio constraint is inspired by a work done by Xiang Feng *et al.* (Xianfeng, Xioguang, Fan *et al.*, 2008), where the dimensional constraint on the sides of the boundary triangles is used to extract complex, concave boundaries of a building exterior from aerial point clouds.

However, the algorithm proposed in this chapter, works directly on point cloud of building interior and does not require any intermediate surface model for its execution.

6.2.1 Scan-line side ratio constraint algorithm

The core of this proposed algorithm is to make full use of the inherent properties of the point cloud and exploit the positional and topological relationship between points along the scan lines. The underlying principle of abrupt variations of range depth at the discontinuous boundary is used as a measure of discontinuity. This abrupt depth variation is accessed by comparing the distance between two nearest neighbors on either side. It identifies those points as discontinuous boundary points that show large neighboring distance disparity. The basic steps of this process includes: (1) capturing the point cloud data with their mapped positional index in terms of their row/ column index for each data point (2) computing the relative dimensional side ratios between immediate neighborhood data points (3) accessing the disparity index of each data point and identifying boundary points and finally (4) reining discontinuous boundary data points by removing the spurious and outliers in the extracted data points. The basic steps are shown in a process flowchart in Figure 6.3.

The shape acquisition process uses a stationary range scanner to capture the point cloud from visible surfaces using the sweeping laser as shown in Figure 6.4(a). The scanner captures a fixed number of points from the horizontal and vertical ranges respectively, which, if unwrapped on its optical planar surface will form a 2D grid as shown in Figure 6.4(b). The total scanning range is captured using mxn-points (m-rows and n-columns). The density variation in the mapped grid is due to the varying distances of corresponding surface regions from the scanner. The density is maximum at a point nearest to the scanner (centre of the wall) and reduces away from this point on either side. Although the data density and its corresponding side ratio vary at every point, however, the variation is smooth and thus does not indicate discontinuity. A discontinuity is observed only at the point exhibiting a large spike in the side ratio. An abrupt change in the range depth or spatial data density signifies the boundary point at the discontinuous boundary. Identifying a sudden change from point cloud data set representing large buildings is computationally very expensive ($O(n^3)$), however the known grid index is quite effective in quickly identifying the probable neighbourhood and can help reducing the computationally cost ($O(n)$ for unsorted and $O(\log n)$ for sorted grid indexes).
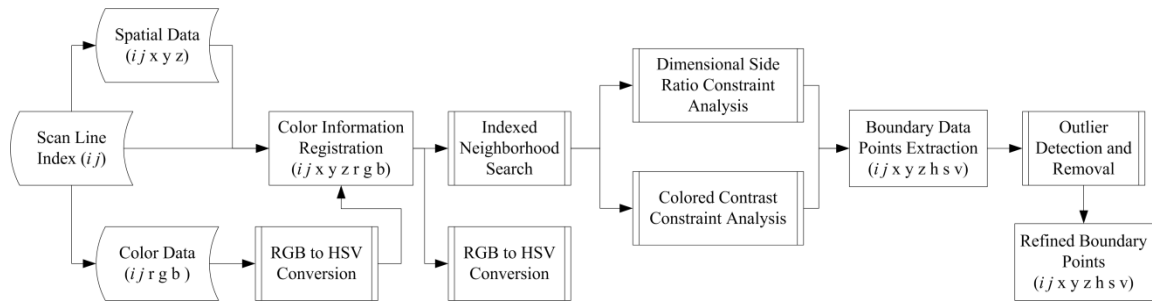
Figure 6.3:   Flow Chart of the proposed boundary data point extraction algorithm.
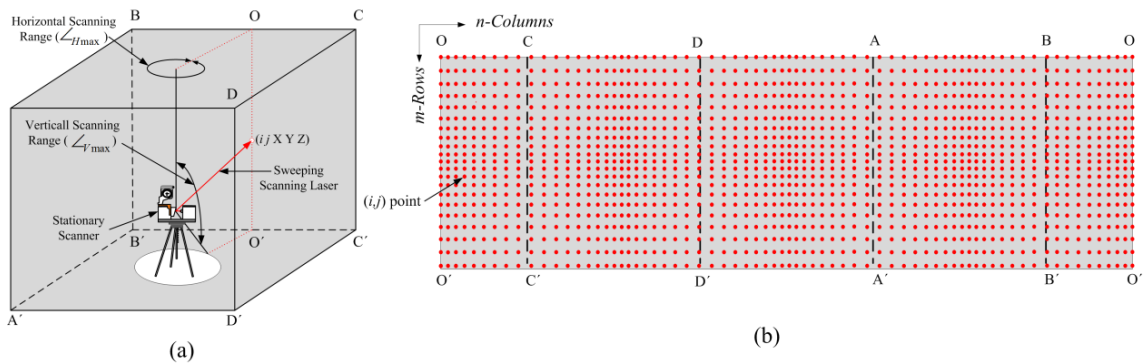
Figure 6.4:   (a) Sweeping laser scanning of building interiors (b) Unwrapped domain in the optical plane.

Each captured data point (x y z) in 3D space corresponds to a particular positional index (i, j) in the mapped grid (Figure 6.5(a)), which comes in handy for quick identification of the neighboring data points along the sweeping lines. It is to be noted that the Figure 6.5(a) shows grid indices only and their corresponding spatial points (x y z) are used to compute the Euclidean distance (Figure 6.5(b)). A similar analysis can be performed using depth analysis of data points using spherical coordinates; however the overall depth and density will vary considerably within the domain and needs to be compensated to formulate a reliable strategy. The proposed approach works on a narrow band of data sets and thus it inherently compensates for local density variations.

A given data point can identify its neighboring spatial coordinates (x y z) from their corresponding grid indices (i, j) in its adjoining rows and columns using a window as shown in Figure 6.5 (for demonstrations purposes, the gradual variation in the scanned data density is omitted in this figure), where one point window is used to elaborate the process of computing spatial distances of the corresponding neighbors on either side candidate point.

Essentially, a large window size is used around a candidate point to identify it as a boundary or non-boundary point. The spatial distances between the corresponding neighbors on either side of the candidate point are computed and an occluded discontinuity is identified by the large ratio of these spatial distances. The window size is reduced and the same process is repeated. A candidate is declared as occluded boundary data points if the result converges. A large window size facilitates the convergence criterion for a boundary data point, compensates the data noise and avoids false occluded boundary point identification. e.g. point-A in Figure 6.5(b) is a boundary data point because of high side ratio as the spatial distance of its neighboring points on either side of this point varies considerably (distance between $(i, j)^{th}$ and $(i\text{-}1, j\text{-}1)^{th}$ points is exceptionally more than the distance between $(i, j)^{th}$ and $(i+1, j+1)^{th}$ points). Thus, point-A signifies the occluded boundary point.
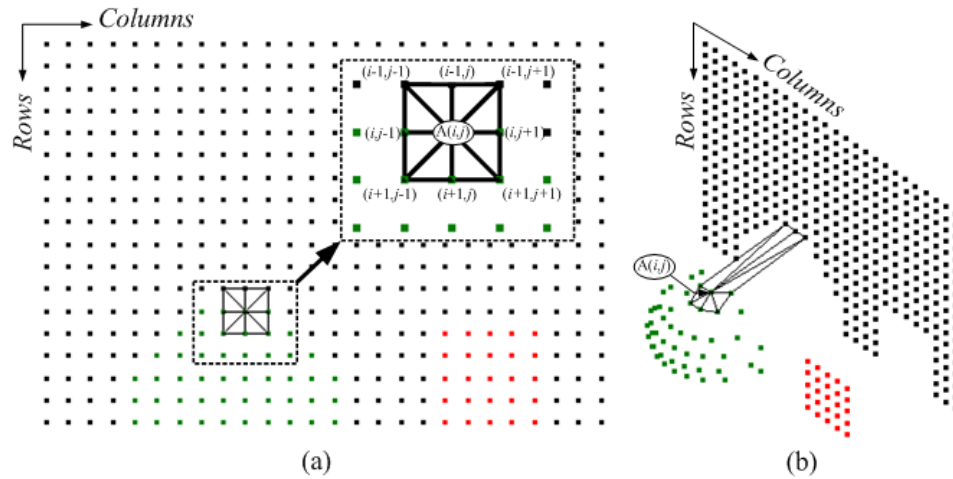
Figure 6.5: (a) Front view and (b) Isometric view of the captured data from the scan scene of Figure 6.2.

A side ratio distance is defined as the ratio of the distances that define the spatial separation of the proceeding and succeeding data points on the grid. The side-ratios are computed for each data points based on the investigation window size. Here w-is the window size that defines how many points on each side of the point cloud will be used on the grid to investigate the point to be a boundary or a non-boundary point. The side-ratio of a data point in an investigation window is defined along the row as follows:-

$$SR_r^{i,j} = \frac{\max(d_{(i,j),(i+w,j)}, d_{(i,j),(i-w,j)})}{\min(d_{(i,j),(i+w,j)}, d_{(i,j),(i-w,j)})} \qquad (6.1)$$

where $d_{(i,j),(i-w,j)}$ defines the Euclidean distance between spatial coordinates corresponding to $(i,j)^{th}$ and $(i-w,j)^{th}$ grid indices and w varies from its initial window size to 1. Similarly, the side-ratio along the column and two lateral sides are computed as follows:

$$SR_c^{i,j} = \frac{\max(d_{(i,j),(i,j+w)}, d_{(i,j),(i,j-w)})}{\min(d_{(i,j),(i,j+w)}, d_{(i,j),(i,j-w)})} \tag{6.2}$$

$$SR_{l1}^{i,j} = \frac{\max(d_{(i,j),(i+w,j-w)}, d_{(i,j),(i-w,j+w)})}{\min(d_{(i,j),(i+w,j-w)}, d_{(i,j),(i-w,j+w)})} \tag{6.3}$$

$$SR_{l2}^{i,j} = \frac{\max(d_{(i,j),(i+w,j+w)}, d_{(i,j),(i-w,j-w)})}{\min(d_{(i,j),(i+w,j+w)}, d_{(i,j),(i-w,j-w)})} \tag{6.4}$$

The ratios are close to one when computed at a regular point on a local surface. As the distance from the scanner increases, the distance between consecutive points also increases, however, the variation is gradual and for any given point, the distance between two points on a continuous surface are comparable. Such gradual variation generates consistent side ratios. However, at discontinuous boundaries, the distances between points on the grid vary abruptly and have side ratios considerably higher than the threshold value. Three threshold values (SR_r, SR_c and SR_l) are used to identify these rows, columns and lateral discontinuities. A spatial point corresponding to (i, j)- grid indices is a boundary data point if at least one of the ratios defined by eqs 6.1-6.4 is more than its corresponding threshold value.

$$r_r^{i,j} \geq SR_r \quad \text{or} \quad r_c^{i,j} \geq SR_c \tag{6.5}$$
$$\text{or} \quad \left(r_{l1}^{i,j} \; or \; r_{l2}^{i,j} \right) \geq SR_l \tag{6.6}$$

It is to be noted that the above investigation is started with large window (w=5) and the size reduced after each calculations. It continues unless the result converges and accurate boundary point is identified. The data density (distance between nearest captured data points) is dependent on its scanner parameters such as average distance from the scanner ($d_{avg.}$), number of scanned points m and n to cover horizontal and vertical scanning ranges, respectively. However, the captured side-ratio values are not dependent on these density values and are quite consistent in continuous data sets. The constant threshold values (SR_r, SR_c and SR_l) identifies discontinuities in the rows, columns or lateral directions, respectively.

The proposed algorithm accesses each data point in the sorted point cloud as a candidate for the boundary data point by computing its side ratios from the nearest, complementary data points. A boundary-point function computes these ratios with the maximum size of the window and reduces the size of the window until the result converges and boundary point is identified as shown in Algorithm 6.1. All identified boundary data points are shifted from the original point clouds to the boundary data sets.

Algorithm 6.1: Occluded boundary data point detection (Side-ratio constraint algorithm)

Parameters: Point cloud {P}, side-ratio constraint thresholds { SR_r, SR_c *and* SR_l }, Scanning parameters (m, n)
　　　　　　Investigation window size-w
Initiate: Initial Boundary Point Data Cluster {BP}
Sort: The index point cloud data $\{i\ j\ x\ y\ z\ h\ s\ v\}$

for $i = 1: m$　***Do***
　　　for $j = 1: n$　***Do***
　　　　　Call Function **Boundary-Point** $(P, w, i, j, SR_r, SR_c\ and\ SR_l)$
　　　　　if $p_{i,j}$ is a boundary point ?

$$p_{i,j} \xleftarrow{remove} \{P\}\ ;\ p_{i,j} \xrightarrow{add} \{BP\}$$

　　　　　end if
　　　end for
end for

Function **Boundary-Point** $(P, w, i, j, SR_r, SR_c\ and\ SR_l)$
for $q= w:1$ ***Do***
Compute Ratios
　　$\{SR_r^{i,j}, SR_c^{i,j}, SR_{l1}^{i,j}, SR_{l2}^{i,j}\} \xleftarrow{find}\ \xi(qp_{i,j})$ % (Side-ratio function $\xi(qp_{i,j})$ computes four side-ratios
　　　　if $r_r^{i,j} \geq SR_r$　*or* $r_c^{i,j} \geq SR_c$　*or* $\left(r_{l1}^{i,j}\ or\ r_{l2}^{i,j}\right) \geq SR_l$
　　　　else $q=q+1$
　　　　break
end for
if q=1
　　　Point $p_{i,j}$ is a boundary point
else
　　　Point $p_{i,j}$ is a not a boundary point
end if

Return

6.2.2　Outlier detection and boundary refinement

The proposed side-ratio constraint algorithm identifies boundary data point based on their local disparity. As the algorithm is implemented directly on the raw scanned point cloud data sets that includes numerous scanning errors due to signal noise, multiple regions and surface characteristics. Thus, there are instances where incorrect boundary data points are identified from such erroneous point cloud data sets. Identifying and removing such flaws is essential to

improve the reliability of the proposed algorithm. These incorrectly identified boundary data points often represent sparse density and thus can be identified as density based outliers. The overall process of this density based outlier detection algorithm is presented in Algorithm 6.2.

Algorithm 6.2 : **Refining discontinuous boundary data points (Density-based outlier detection algorithm)**

Parameters: Point cloud [P], Boundary point data cluster [BP], Scanning parameters (m, n), Density threshold $n_{threshold}$

Initiate: Initial Boundary Point Data Cluster $[BP_{refined}]=[BP]$ and average point distance $d_{avg.} = 0$;

for *every* $p_{(i,j)} \in [[P]-[BP]]$ ***Do***
    Compute
        $d_1=d_{(i,j),(i-1,j-1)}$; $d_2=d_{(i,j),(i,j-1)}$; $d_3=d_{(i,j),(i+1,j-1)}$; $d_4=d_{(i,j),(i+1,j)}$;
        $d_5=d_{(i,j),(i+1,j+1)}$; $d_6=d_{(i,j),(i,j+1)}$; $d_7=d_{(i,j),(i-1,j+1)}$; $d_8=d_{(i,j),(i-1,j)}$;

    Initiate $n_{nb}=0$;
Call Function **Average Distance** $(d_{avg.},d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, n_{nb})$
end *for*

for *each* $p_{(i,j)} \in [BP]$ ***Do***
    $[N_k] \xleftarrow{\;find\;} K_r(p_{(i,j)})$ % where $K_r(p_{(i,j)})$ is a K-nearest neighborhood points around p
    $[NN]=[N_k]$
    for $a=1:size$ $[N_k]$
        if $d_{p_{(i,j)},p_a} \geq \gamma.d_{avg.}$ % $d_{p_{(i,j)},p_r}$ is the spatial distance between p and its ath neighbors
                                  % γ is a multiple factor
            $[NN]=[NN]- p_a$
      end *if*
     end *for*
    if $size$(**NN**) $\leq n_{threshold}$
        $[BP_{refined}] = [BP_{refined}] - p_{(i,j)}$
        $[P] = [P] + p_{(i,j)}$
        Update $\{BP\}$
    end *if*
end *for*

Function **Average Distance** $(d_{avg.},d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, n_{nb})$
    for $q=1:8$
        if $d_q \neq 0$
            $n_{nb} = n_{nb}+1$;
        else
            $n_{nb} = n_{nb}$;
        end *if*
    end *for*

$$d_{avg.} = \frac{d_{avg.} + \sum_{q=1}^{8} d_q}{n_{nb}}$$

Return $d_{avg.}$

Each identified boundary point from the side-ratio constraint algorithm is analysed based on the data density in its local vicinity. The sparse data points are outliers and are removed. The local data density is computed first in terms of number of neighboring points within a pre-defined investigation distance (d_f), which is computed as a multiple-γ of an average distance ($d_{avg.}$) between data points in the continuous region. The points having lower neighboring point count in that region are outliers and are removed from the identified boundary point data set [BP] and thus, a refined point cloud data set [$BP_{refined}$] representing discontinuous boundary is extracted.

6.3   Result and Discussions

The proposed algorithm is implemented on the point cloud data set of building interiors. An office space (Figure 6.6) is first scanned using FERO® scanner using specific *m*x*n* data points. Essentially, there is a one-to-one correspondence between each (*i*, *j*) pixel (Figure 6.6(a)) and its corresponding spatial coordinate (Figure 6.6(b)). The side ratios are computed based on the Euclidean distances of its local neighboring points. These ratios are computed for each spatial data point, where the neighborhood information is identified from grid indices.

(a)   (b)

Figure 6.6:   Scanned point cloud data representing an office room (402x963).

6.3.1   Side-ratio constraint and boundary data points extraction

The captured point cloud data of an office room (Figure 6.6) is used to elaborate the process and results. Converged side-ratios are computed for each data point from its neighboring points. These side-ratios along rows, columns and lateral direction are shown in Figure 6.7, where the isometric views (Figure 6.7(a)) represent the vertical height as the side ratio at its

corresponding grid index (i, j). Figure 6.7(b) shows the top view with color grid indices reprinting occluding boundary indices. Once these indices are identified, their corresponding spatial data points (x y z) are separated from the point cloud and are identified as occluded boundary data points.

(a)   (b)
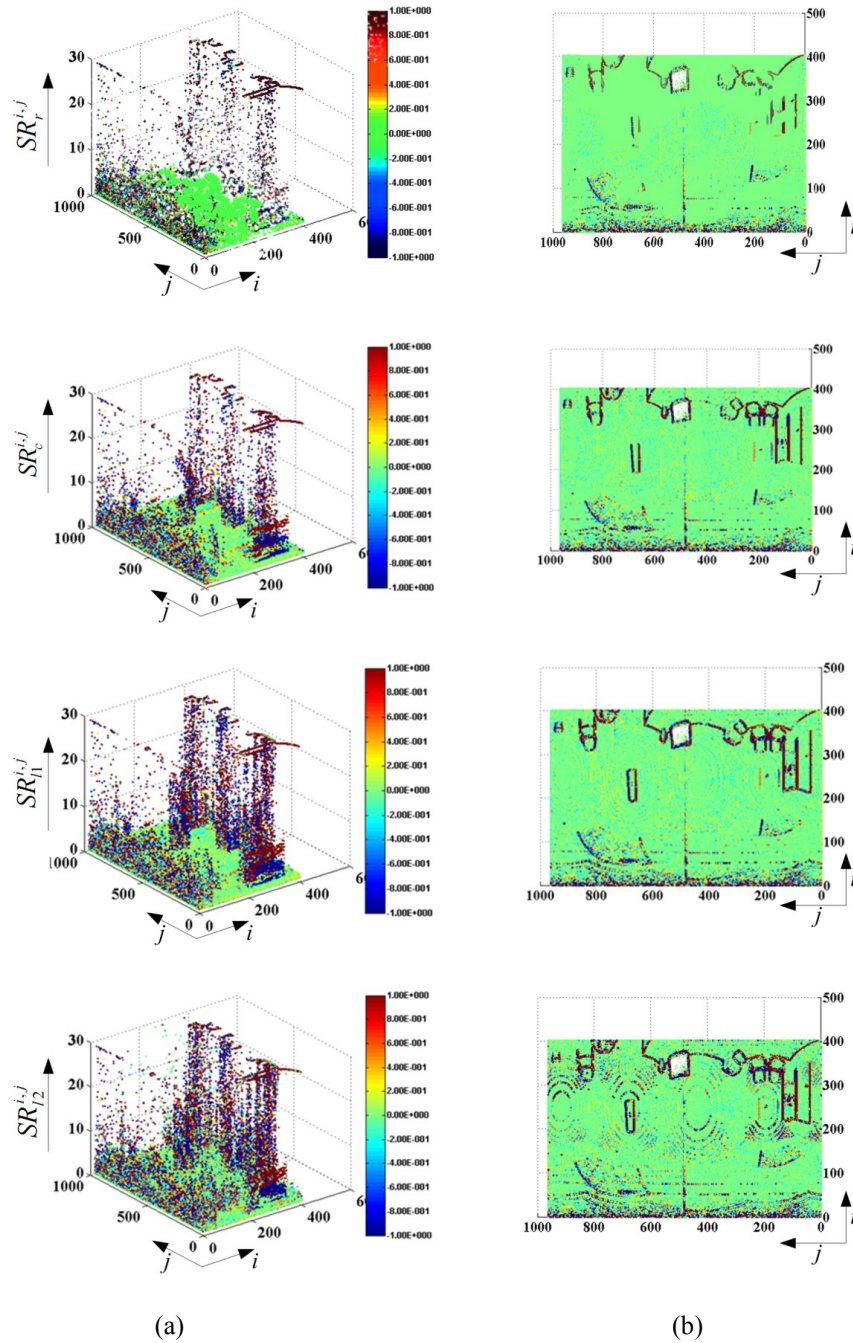
Figure 6.7: Side-Ratios ($SR_r^{i,j}$, $SR_c^{i,j}$, $SR_{l1}^{i,j}$ and $SR_{l2}^{i,j}$) of the point grid (mxn) of the scanned data of an office room.

The vertical direction shows the value of the actual side-ratios. The colors show the normalized local deviations from the unwrapped grid. Some of the point with noisy neighborhood points generates very large reduction ratios, which increases the span of the vertical axis and makes it difficult to differentiate the lower values of reduction ratios in the plot. Thus, a restricted value of this maximum reduction ratio is used for plotting the variations. It can be seen from these graphs that the although the higher side ratios ($SR_r^{i,j}$, $SR_c^{i,j}$, $SR_{l1}^{i,j}$ and $SR_{l2}^{i,j}$) give good indication of the location index (i j)) of the surface discontinuities. In this case, outliers may be avoided with suitable selection of the threshold values and subsequent outlier compensation.

A suitable threshold value of these side-ratios provide the location grid indices (i j) of the spatial points on the discontinuous boundary. Figure 6.8 compiles the identified boundary data points with different threshold values implemented on the lateral side-ratios. It can be seen from this figure that the side ratios with a suitable value of threshold can extract accurate boundary data points. However some data points from the continuous regions are also extracted as boundary data points. The data points identified as circular rings are due to the varying data density, which is observed as the surface region moves away from the scanner and data density reduces. Further, the points representing the ceiling vents are extracted only in Figure 6.8(a) where a sharp variation identifies points on surface discontinuity, which otherwise is a contiguous surface with occasional abrupt changes. This situation leads to the micro-discontinuity and thus are identified with small side-ratio threshold (low ζ). Such inconsistencies are considerably reduced and the accuracy of the identified boundaries is significantly improved with the increase in the threshold value (Figure 6.8 (b, c and d)).

Original Point Cloud Data

(a) Identified boundary data points ($\zeta=5$)

(b) Identified boundary data points ($\zeta=7.5$)

(c) Identified boundary data points ($\zeta=10$)

(d) Identified boundary data points ($\zeta=15$)
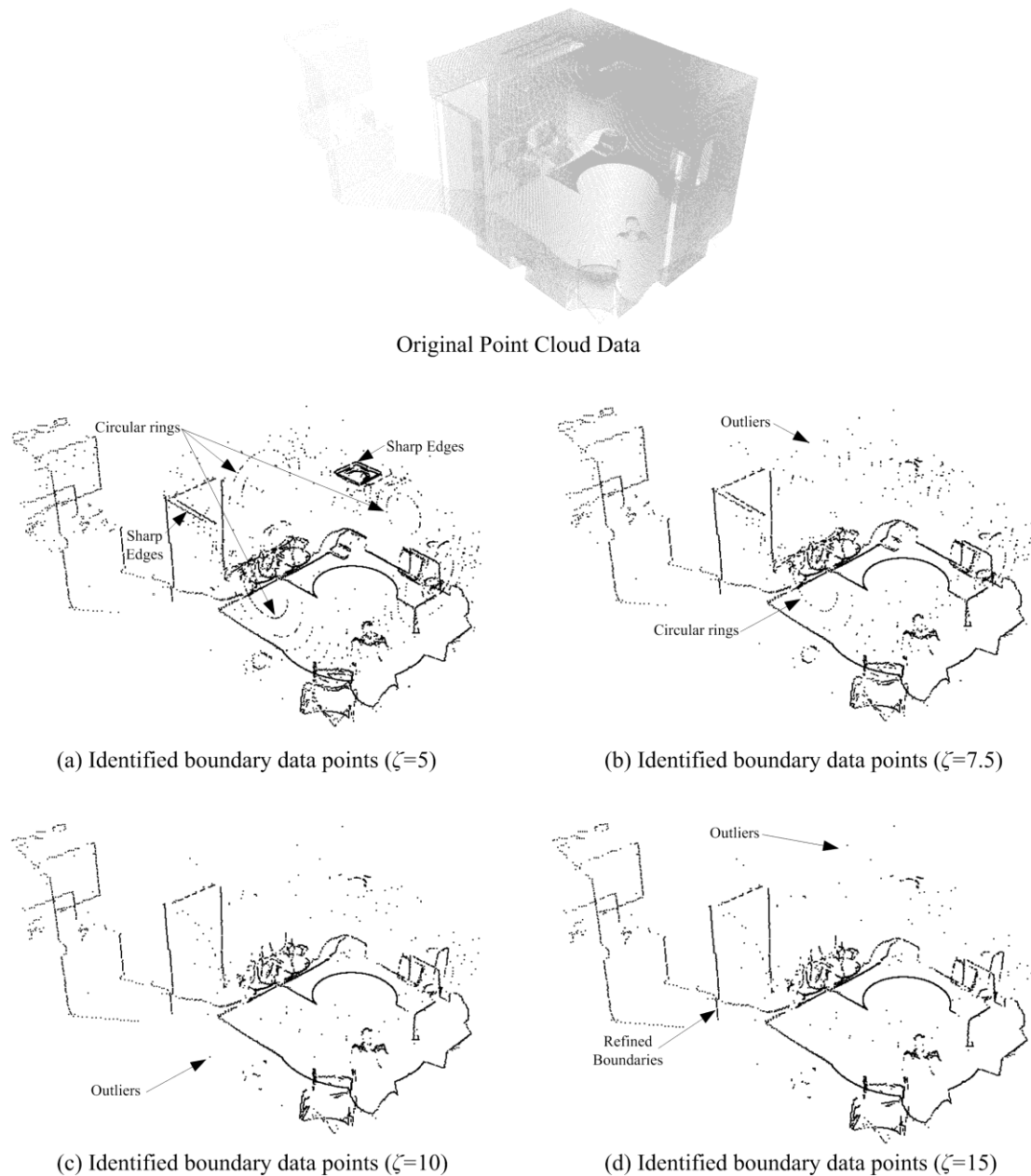
Figure 6.8:   Original point cloud data and Identified boundary data points with different lateral side-ratio constraints with $\zeta = 5$, 7.5, 10 and 15.

6.3.2   Density based boundary refinement

Boundary data points are accurately identified using side-ratio constraint algorithm with higher threshold. It takes care of the spurious data points from regions with higher data density variations and abrupt changes in the surface properties. However, some random

points are identified as boundary points due to data noise and other surface characteristics. The extracted data points are further improved by removing outliers to extract accurate discontinuous boundaries. Figure 6.9 shows the results of the refined data points identified at the discontinuous boundary after the density based outlier detection approach is implemented.

(a) Refined boundary data points ($\zeta=5$, $\gamma=2$)

(b) Refined boundary data points ($\zeta=5$, $\gamma=2$)

(c) Refined boundary data points ($\zeta=10$, $\gamma=3$)

(d) Refined boundary data points ($\zeta=15$, $\gamma=3$)
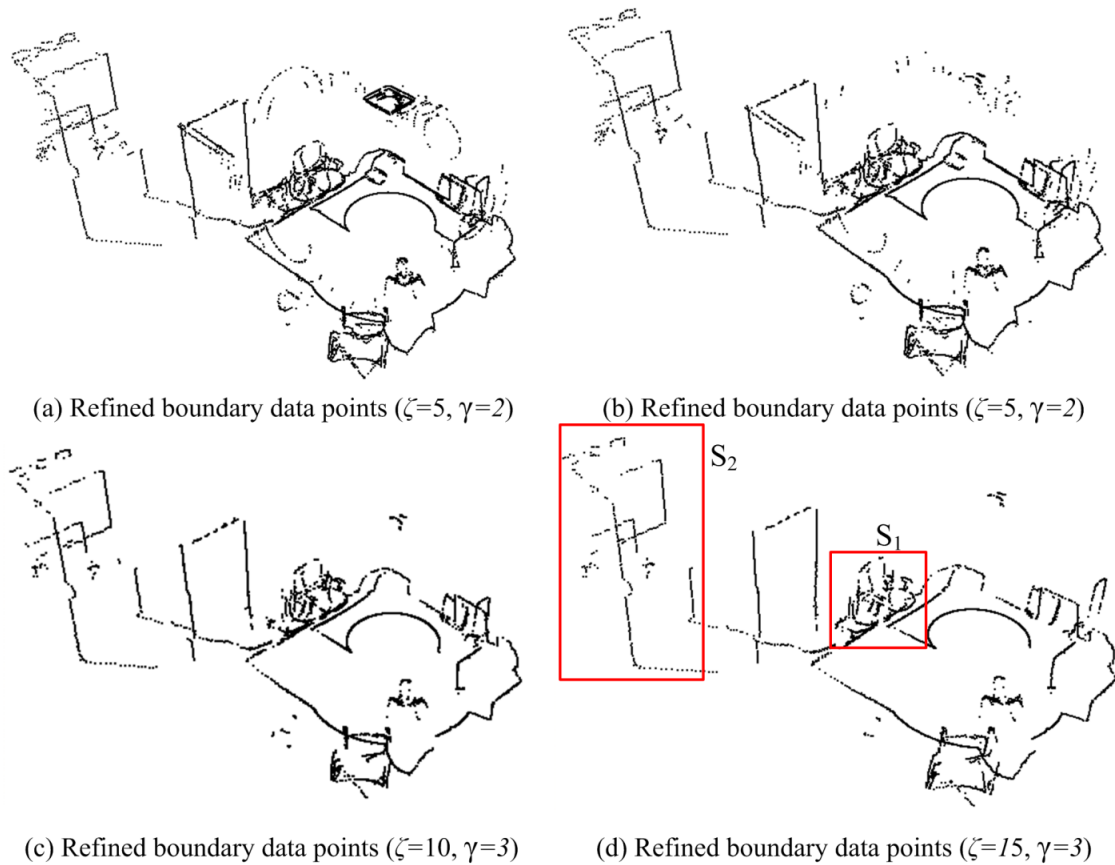
Figure 6.9:   Refined boundary data points obtained using density-based outlier detection ($n_{threshold}=4$).

The density-based refinement removes the spurious data points and generates accurate boundary data points (Figure 6.9). The local effect of varying data density in the point cloud is minimized as the points corresponding to circular rings are reduced (Figure 6.9(a-c)). It can also be seen from Figure 6.9(c) and (d) that the points corresponding to the two distinct edges of the computer monitor are accurately extracted. The discontinuous boundary data points at the far-end of the vertical scanning range (n_{max}) are also extracted as the circular pattern, which is partially captured at the table and the floor. Additionally, the multiple points from

the boundary data sets are also reduced considerably. It can be concluded from the analysis that an effective approach of extracting accurate boundary data points is to avoid very low value of side-ratios (ζ=10-15), where the spurious data points are removed by density based outliers.

There are few portions in the refined data sets such as section S_1 and S_2 in Figure 6.9(d), where the refining process seems to work ineffectively. However, these regions can be seen from an alternative view-point (Figure 6.10) that in these data set regions, discontinuous boundaries are accurately identified. It is very clear from the figure that this data set shows discontinuous boundaries of different parts of the chair (chair arm, the car seat and the back) and different parts of the occluded regions i.e. wall and floor (Figure 6.10 (a)). Further multiple boundary data points in section S_2 represent different objects such as cabinetry, wall, side wall and the back wall (Figure 6.10(b)). Hence, it can be concluded that the proposed initial boundary point extraction and subsequent refinement process is effective, even in this dense area representing multiple objects.

(a) Section S1 from a different view point   (b) Section S_2 from different view point
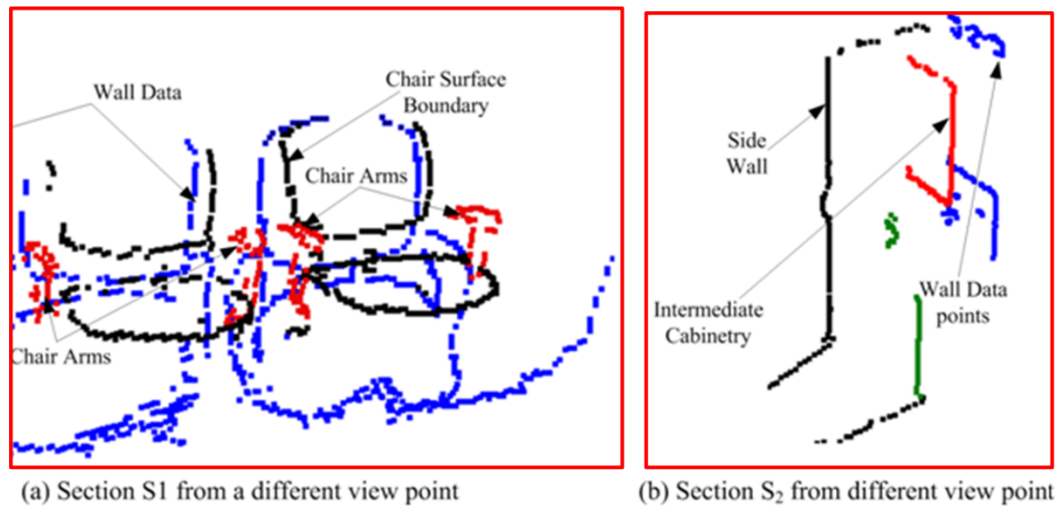
Figure 6.10:   Detailed views of section S_1 and S_2 from different view point.

6.4   Concluding Remarks

Occluded boundary point identification in point clouds representing multiple objects is an essential pre-processing step. This is especially critical for accurate representation of voids and discontinuous boundaries in the point cloud. It is essential to generate an effective

framework for application involving data registration, segmentation and geometric modeling. The proposed side-ratio methodology extracts the boundary data points directly from point clouds, which categorize each data point as an occluded boundary or non-boundary data point based on side ratios from its neighbors. The neighborhood relationship is effectively measured from the sorted point cloud data and exploiting the grid index of the scanned data sets. It is observed that the inconsistent data densities due to the varying scanner distance can be compensated by using higher thresholds of the side ratios (ζ=10-15). Further, the occluded boundary data cluster is refined by removing the spurious data points that are generated due to signal noise, varying data densities and non-homogeneous surface properties. The refinement process relies on an average density based outlier detection approach, which removes the sparse data points from the boundary data cluster. The refined boundary points represent an accurate occluded boundary, which can be effectively used in various post-processing applications involving region segmentation, pattern recognition and data simplification.

CHAPTER 7   INFORMATION MODEL EXTRACTION

7.1   Introduction

Building information models of large civil structures from their corresponding point cloud have become an important tool for designers, modelers and engineers to identify, extract and communicate geometric, topological and spatial information with others. These BIM models help to investigate the impact of any change in modeling, renovations and architectural modifications on a scanned domain. These models provide engineers with effective means to understand a complex scanned scene with informative point clouds representing something more than mere spatial positions. It helps in evaluating the scanned structures in terms of its as-built as-design evaluation, layout extraction, historical site documentation, freeway redesign, tunnel modeling and creating geographical information system (GIS).

A set of information models developed in this research transforms the raw point cloud data into a usable format (geometric features, meaningful clusters, and other shape attributes) with additional information. The BIM specifications are critically dependent on the application-specific domain and the usability of the information. However, its applicability cannot be generalized to multiple applications and an explicit information model is required for a given application. Besides, reliable information models can only be generated if the scanned data set is captured accurately in the first place. The research presented here focuses on these two critical steps of developing effective information models from point clouds: (1) capturing accurate spatial geometry and (2) developing adaptive information models.

Accurate spatial form can be captured using effective scan planning, which generates a huge point cloud with minimum scanning inconsistencies and representing everything in the scan scene. A descriptive scanning methodology is devised to improve the accuracy of the captured data so that reliable information models can be developed from the data set. Further, the information model developed in this research extracts section layouts and data clusters which are representative of the objects lying within the scanned scene. The overall research (methodologies and developed algorithms) carried out in this thesis and their probable applications are shown in Figure 7.1 and described in the following sections.
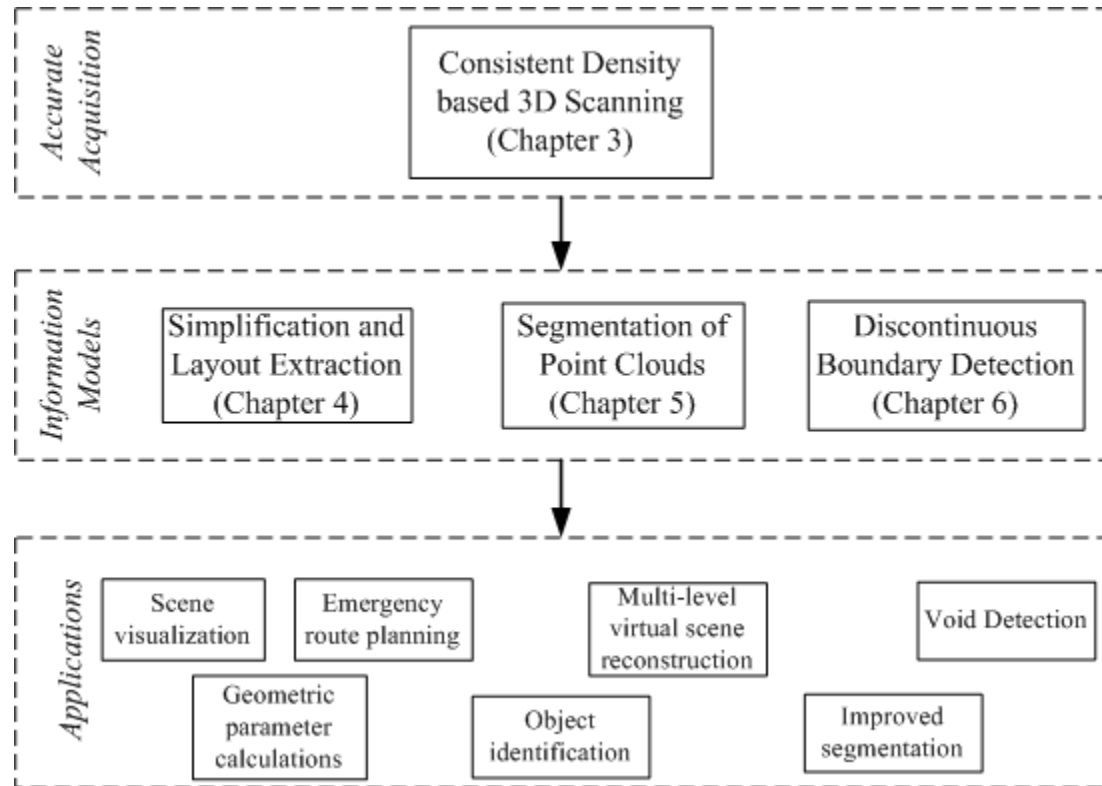
Figure 7.1: The methodologies and algorithms developed and their probable applications.

7.2 Simplification and Layout Extraction Applications

The first information model extracted from accurately captured point cloud data is the section layout. During the process, the point cloud data set is also simplified using a contour based simplification methodology. The simplification strategy removes points adaptively based on the local geometric complexity i.e. more number of points are removed from the planar region and less number of points are removed from geometrically complex regions. Thus, certain elements, features and objects are visible in the simplified scanned scene. The layouts are extracted using a slice of point cloud along a given direction and thus, based on its location, the extracted layout can be used to compute the geometric parameters such as area, volume etc. Another location of layout slice can produce the layouts representing the openings in the walls corresponding to the doors or windows. This information model can be used in the following applications.

7.2.1   Scene visualization

One of the most critical characteristic of 3D scanning of building interiors with multiple objects is that captured point cloud often corresponds to multiple objects with data density having no direct correlation with the geometric complexities of the objects. Here, simple (geometries) as well as complex (freeform) surfaces are captured coherently. Thus, the point cloud is overly populated in planar regions and it does not help to distinguish the features in the scanned scene. Thus, a simplification strategy that can directly focus on important features and extract them directly from the point cloud, is quite beneficial. It not only extract the desired feature (layout in this case), but the simplified data set helps in visualizing the scanned scene accurately. For example, Figure 7.2 shows the scanned scene, a data slice and the simplified data slice.

The original point cloud data (Figure 7.2(a)) is very dense and its visualization does not yield any information about the objects present in the scanned scene. The regions are over-populated in some regions and under-populated in others. The scanner's settings can be adjusted to capture large number of points to avoid under-populated regions. However, this also increases the number of points at the simplified region and the visualization becomes even worse. It is really difficult to identify the points representing the interior objects. The point cloud simplification strategy proposed in this research helps in reducing the internal points as shown in Figure 7.2. Here the sliced point cloud is decimated to reduce the point cloud in such a way that the point cloud corresponding to internal objects is retained. The resultant slice point cloud (Figure 7.2 (c) and (d)) shows the objects are clearly identifiable and the scene visualization is greatly improved. This facilitates a reliable means to devise effective strategies for CAD modeling and virtual reality application through reverse engineering or pattern recognition. It is to be noted that the approach is purely based on the point cloud where the point cloud is simplified and visualization is improved without generating any intermediate models or derived geometric components.

Figure 7.2 (d) retains points corresponding to the reference spheres, shelves, monitor, telephone and the boundary data point of the tables. This decimated data set can not only be used to identify the number of objects lying in the scanned scene, but can also be used to formulate effective post-processing strategies to model them. The proposed simplification

strategy also extracts layouts from the point cloud data set, which can be used for numerous applications as described below.

(a) Original Point Cloud of an Office

(b) Data Slice

(c) Simplified Data (K_r=10, $\Psi_{threshold.}$=85º)

(d) Simplified data (K_r=10, $\Psi_{threshold.}$=80º)
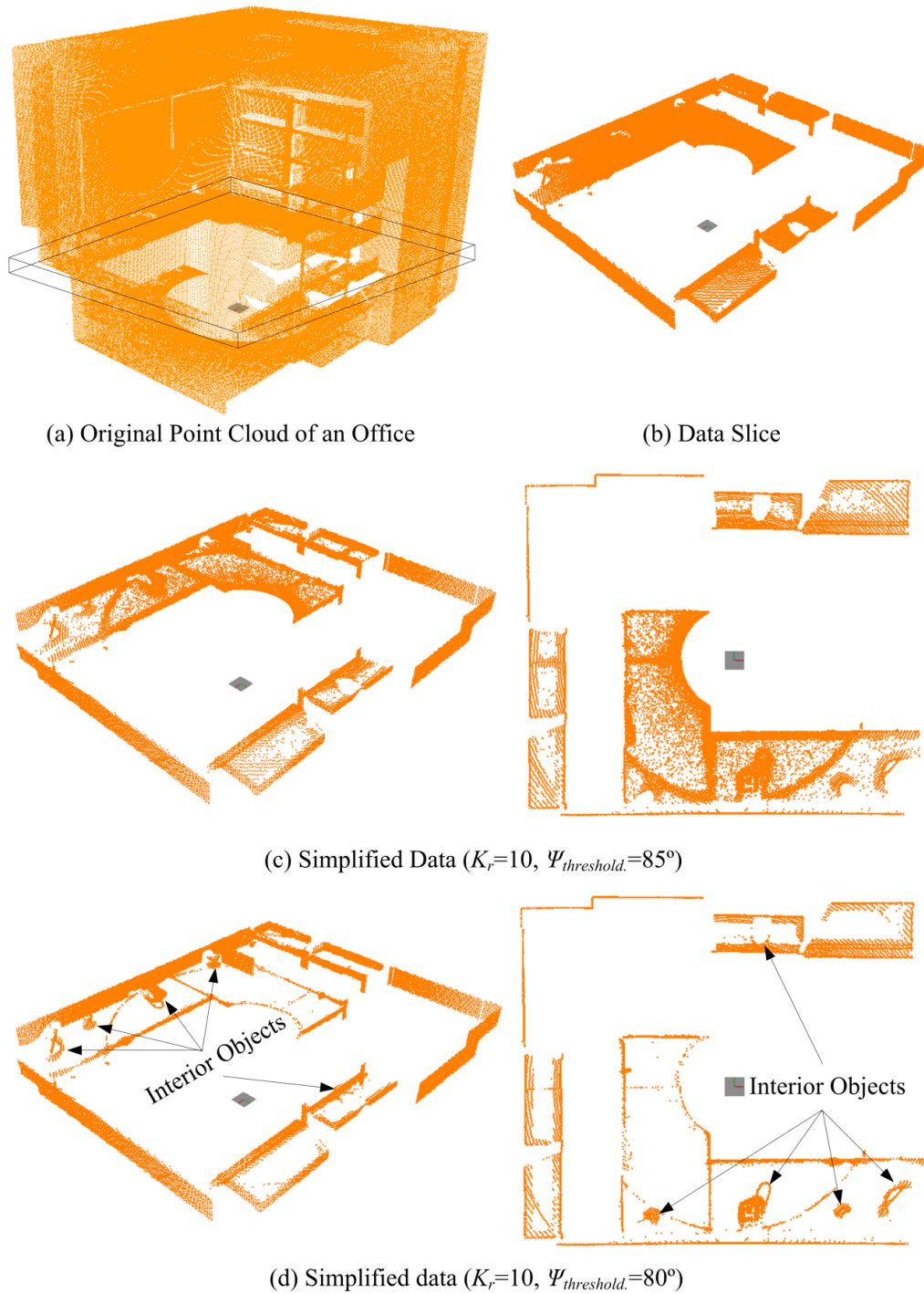
Figure 7.2: Data simplification and better scanned scene visualization.

7.2.2 Geometric parameter estimation

The layouts extracted from the simplified point cloud data set can be used to estimate the geometric parameters including dimensions, areas and volumes. The extracted shapes provide a direct means of measuring the dimensional details of the length, width or any other dimensional parameter of the scanned objects. Subsequently, other geometric properties can be derived from these dimensional parameters. The layout extracted from the sliced point cloud is shown in Figure 7.3 that gives an accurate estimation of the geometric parameters as compiled in Table 7.1. In fact, the area computed from the scanned point cloud is an accurate estimate of room because it does not compute it directly by multiplying length and width and takes into account the small variations, protrusions or recesses in the walls.

(a) Data Slices

(b) Extreme data slices used to compute the geometric parameters

(c) Extracted Layout

(d) Area Estimation with region growth
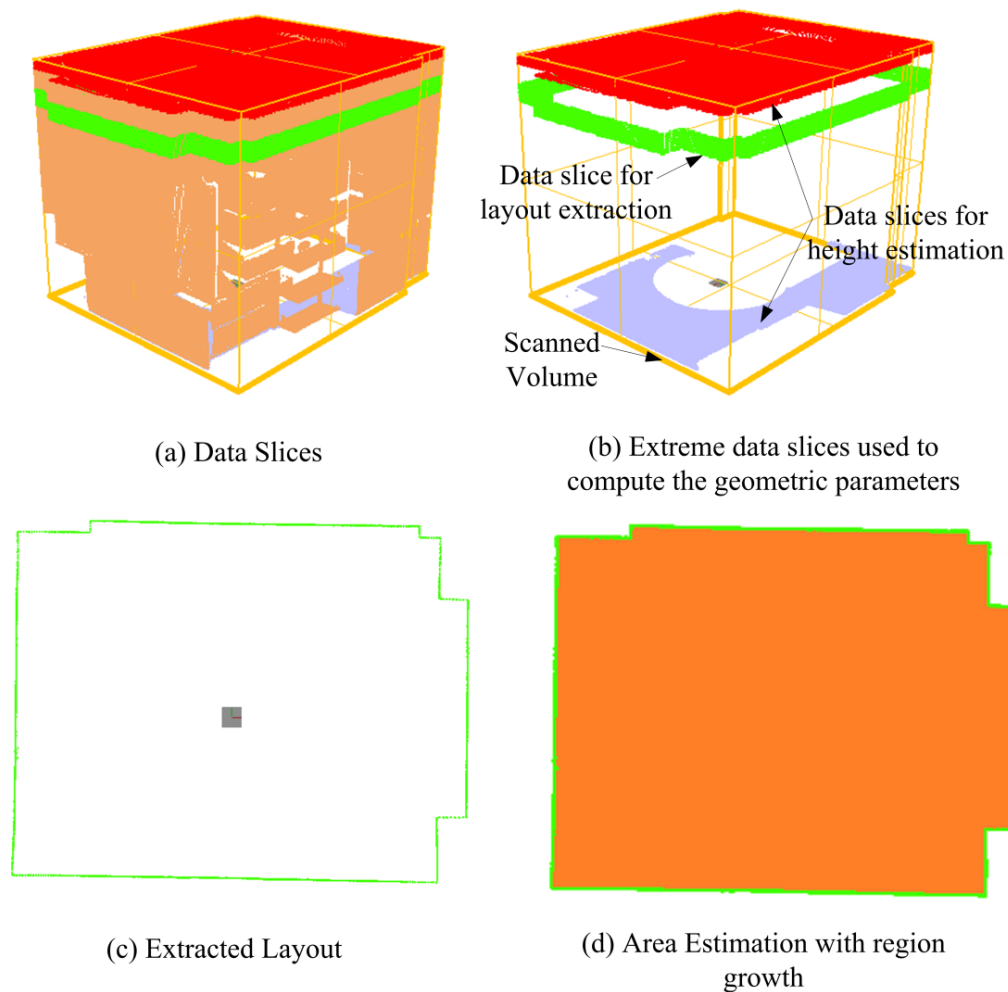
Figure 7.3: The sliced data set, computation of geometric parameters and extracted parameters.

Table 7.1: The estimated and manually measured physical parameters and percentage variations

Parameters	Estimated Value		% variation
	From Scanned Data	Manually Measured	
Max. Length (m)	4.662	4.664	0.043
Max. Width (m)	3.655	3.655	0
Max. Height (m)	3.379	3.381	0.050
Area (m^2)	16.51	17.04	3.110
Volume (m^3)	56.029	57.636	2.788

The low variation in the dimensional parameters (length, width and height) is an indicative of capturing accuracy of the scanner. The variation in areas and volume is more as the one computed through manually measurements does not take into account the variation along the span and uses measured length, width and height to estimate the area and volume values. However, the values estimated from the scanned data are more realistic. In case of area computation, the protruded regions due to pillars and wall sections reduce the internal area of the room, which is the exact representation of the corresponding parameter. Moreover, the slice position used to extract the layout for area computation can be altered to better suit the visibility of the dimensional parameters. The volume on the other hand, reduces at the inward protrusions of the wall sections and outwards protrusions of floor section and hence it is actually close to the real value.

7.2.3 Emergency route planning

The extracted contours can also be used to compute the emergency path/route planning. All path planning applications need the section layouts with sufficient information regarding the opening in the layouts for an object/ robot to move around.

As the proposed algorithm extracts the section layouts from data slice, the location of the slice can be adjusted to extract the section layout with desired sectional properties. One slice can generate the section layout with closed boundaries (Figure 7.3(c)) and thereby can be used to compute geometric properties (area). Similarly, another slice from the point cloud can be used to generate the section with recesses (Figure 7.4(d)). The recesses so computed in the extracted layout can be subsequently exploited geometrically and the path or route planning algorithms can be developed for mobile or evacuation applications. Figure 7.4 shows the slice selected from the point cloud of room scan and their corresponding section layout with

door opening. Although this figure shows a section layout of a single room, the methodology is extendable to point cloud captured from multiple locations and registered together, where multiple layouts extracted from the different scan positions can be registered together to combine the overall floor layout. This layout is then used to develop reliable route/path plan.

(a) Data Slices

(b) Data slice with door opening

(c) Simplified point cloud

(d) Extracted layout
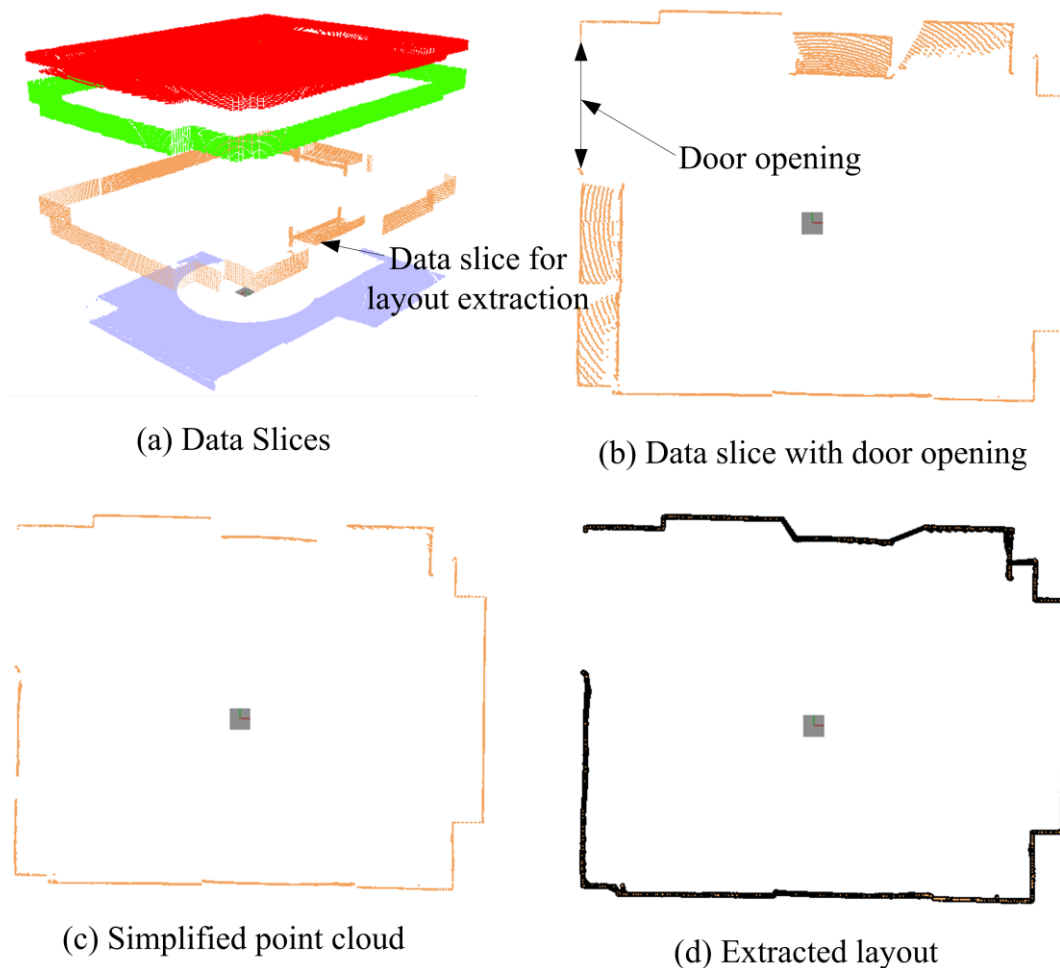
Figure 7.4:   Layout extraction with openings for path/route planning.

7.3   Segmentation and Occluded Boundary Detection Applications

The information model obtained from the color-based, hierarchical segmentation and occluded boundary identification help in segregating point clouds into small, meaningful clusters that can be used in numerous applications as discussed below.

7.3.1 Generating multi-level virtual scenes

The point cloud captured in the scanning process corresponds to a significant amount of data and rich source of information. However, this information is to be extracted from the point cloud using suitable post processing strategies. Their suitability largely depends on the application for which the information is to be used.

The algorithmic parameters of the proposed hierarchical segmentation algorithm can be adjusted to extract varying level of data clusters (Figure 7.5) details and subsequently create multi-level virtual models of the scanned scenes.

The first segmentation level extracts large clusters which are representative of the simplest geometric form of the scanned scene consisting of planar regions and internal data sets. The planar clusters extracted in this case represent the walls, floor and ceiling using geometric differentiation parameter (level I). The data set representing occupied interiors is not processed and it acts like a single cluster. The level suffices if the requirement demands very simple representation of the point cloud.

The combined color and geometric segmentation approach further segments the large planar clusters into smaller data clusters. This level segregates the points representing multiple objects in planar regions (level II) in the first hierarchical stage. Thus, this superior information model provides a means of identifying these multiple objects, which can be used to create virtual models of desired details.

The interior point cloud is difficult to subdivide into accurate individual clusters using pure geometric differentiation criterion (level I for 2nd hierarchical segmentation stage). Even with optimum algorithmic parameters, the extracted clusters are often incomplete, unsaturated and primitive in shape. The algorithmic parameters of the combined hue and geometric effectively extract accurate clusters from the point cloud. The value of these algorithmic parameters controls the level of information models and the details of the point clusters. A suitable set of parameters for the 2nd hierarchal segmentation stage (hue deviation threshold (h^{thres}), geometric threshold (θ_{avg}^{i}, $\theta_{avg}^{threshold}$), neighboring investigation distance (DistF and d_f)) are used for extracting accurate segments from the data set. Although not completely saturated, it gives a fairly accurate representation of the shape and size of object extracted in scanned scene (level II).
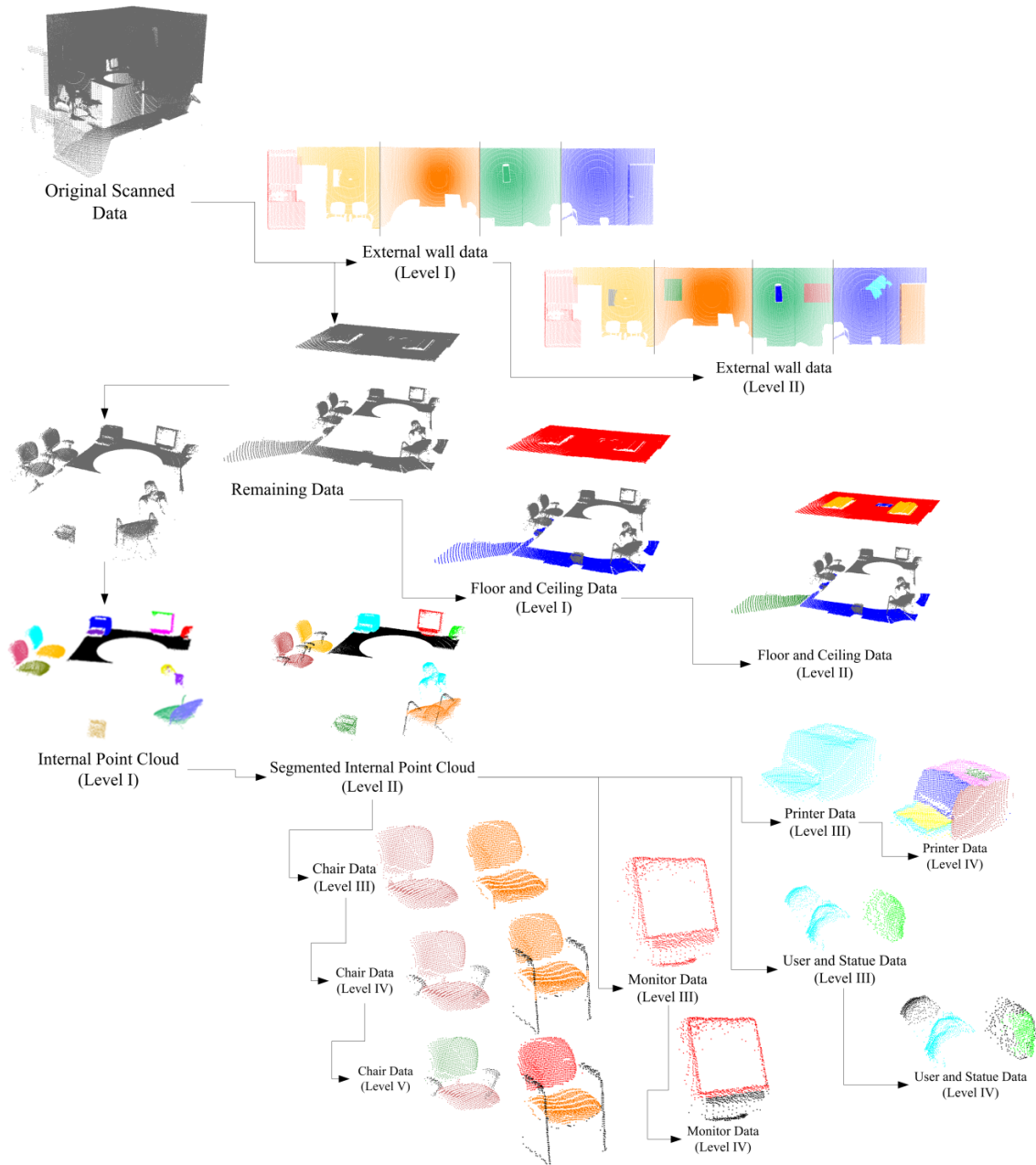
Figure 7.5: A hierarchy-based multi-level segmentation models using geometric and color based segmentation criteria and algorithmic parameters.

A relatively relaxed geometric threshold along with small hue deviation constraints combines similar hue clusters and generate unsaturated data clusters (level III), which are representative of the individual objects in the scanned scene. Selection of multiple seed

extracts additional data clusters from the data objects such as side rests for chairs. The extracted data clusters are effective for individual object identification in the scanned scene and formulate an accurate visual model.

Further, the individual object clusters can be subdivided into its constituent surfaces (level IV and V) so that if required, surfaces can be fitted to this segregated point cloud for actual geometric modeling of individual surfaces to complete the virtual model. However, the geometric accuracy of the extracted model greatly depends on the actual captured point cloud density in this region and the completeness of the data set of a given model. The point cloud representing the hidden or overlapping surfaces is often not available and hence, a complete surface model of all interior objects cannot be generated. However, based on the level of information, the extracted models can be regenerated or matched to its individual surfaces and objects.

This multi-level information models transforms the cumulative point cloud into desired format that can be directly used to generate their corresponding virtual reality model.

7.3.2 Object Identification/inspection/regeneration.

The first level information model (first stage) can help in generating primitive model of the scanned shape of the scanned scene through piecewise integration of planar regions. This can be used to compute section layout and other primitive shape information in the first stage. The level-II information model of the first stage segregates the planar regions into its constituent objects and can help in identifying planar objects such as table, ceiling light, doors and window.

The information model generated by segmenting the point cloud in the second hierarchical stage represents points from level-I (unsaturated disjoint regions) to level-II (representative objects) to level III(saturated objects) and to other levels representing individual surface geometries. As the output of this step is pure point cloud with its color and spatial information alone and no intermediate geometric model approximation, the resultant models are bound to provide and accurate representation of the objects.

The point cloud data set can be used to identify objects either through pattern recognition, objects matching or object reconstruction. The first two approaches leads to

good results especially in single scan scenario as the point cloud representing each object is often not complete and have missing surfaces.

The segmented data can also be used for investigating as-built as design investigation. It is easy to match the segmented data to its corresponding features in the scanned scene instead of using the whole data set. This approach is relatively less demanding on the computational power due to its data handling strategy, where only an individual feature, individual object is considered at a time.

Finally, the extracted data clusters can be further subdivided or approximated for shape for geometric model regeneration. The segmentation facilitates small data clusters representing special feature or object and thus can be used to generate one object at a time to complete the overall geometric model of the scanned scene.

7.4   Concluding Remarks

The information models developed in this research can be used to simplify the post-processing of the point clouds of large building interiors and transform the un-informative, cumulative point clouds into a usable format.

Direct point-based, data slice-driven simplification strategy helps in generating a visualization model for the scanned scene providing a quick means of understanding the domain complexity. The same methodology is extended for extracting sections layouts, where the proposed algorithm allows the generation of application specific layouts. Its algorithmic parameters can be changed to compute both closed as well as open layouts for applications involving the computation of geometric parameters and planning the emergency routes or mobile paths, which can be used in large number of application scenarios.

The color-based, hierarchical segmentation and occluded boundary detection algorithms provide a reliable means of segregating the cumulative point cloud into multiple, manageable and informative clusters. The extracted clusters are very accurate and the algorithmic parameters can be controlled to preserve varying level of surface details of the objects, which can be used to generate a simple representative model, exact visual model or complex geometric model, as desired.

CHAPTER 8   CONCLUSIONS AND RECOMMENDATIONS

8.1   Review of Algorithms

In this thesis, a scan planning methodology enabling accurate 3D acquisition and several post-processing algorithms generating information models for occupied building interiors were presented. The main focus was to develop a framework for efficient shape acquisition, information definition, feature identification and the development of information models. The proposed methodologies and algorithms are illustrated on data set of building interiors. However, the core algorithms are versatile for use in variety of similar applications such as tunnel scanning, sub-ways, narrow pathways and over-bridge etc. The FARO range scanner and a digital camera are used to capture the point cloud and the color parameters, respectively. The developed algorithms are purely point based and do not require an intermediate surface model during implementation.

The research investigated the limitations of scanning large and occupied building interiors and identified scanning and physical parameters of the domain that contributes to scan data inconsistencies. These inconsistencies and the domain's physical parameters are mapped and the criticality of density variation is identified in terms of domain's length/width ratio. A density based scan planning methodology is proposed that ensures coherent scan density in the captured data. This consistent scanned data could be used for post-processing applications without encountering density based issues such as incorrect geometric parameter computation, voids regions identification and erroneous data segmentation.

The proposed slice-based simplification was capable of generating representative virtual model, where all the objects are clearly identifiable in the simplified data set. The same algorithm is extended to extract the section layouts from the point cloud directly. The algorithmic parameters and the data slices were used to extract the section layout as desired. Desired information can be extracted based on the location of the data slice used to extract section layout. In fact, the same methodology can extract section layout for computation of geometric parameters (length, width, area) and sections for path planning, by changing the location of the simplifying plane and data slices.

The color-based, geometry-assisted segmentation and occluded boundary detection algorithms transform the cumulative point cloud into effective information models. A robust HSV color model was used to formulate reliable Hue-based segmentation criterion by minimizing the differentiating dependency on intensity to some extent. A rapid, hue-based segmentation algorithm does not require the additional, higher order geometric parameters (normal and curvature) and can quickly segment the point cloud. However, the presence of color in the scanned domain is an absolute necessity in this case. Therefore, a more robust, color-based, geometry-assisted hierarchical segmentation algorithm is proposed that can work even in the absence of a color/hue. It uses a shape based hierarchical approach that ensures that the segmentation parameters are suitable for both simple (planar) and complex freeform (interior objects) regions.

8.2   Novel Features of the Proposed Work

The novel features of the proposed 3D scanning methodology and information extraction algorithms are described below.

8.2.1   Density-based scanning methodology

The proposed approach addresses an important issue of digitization process to avoid inconsistent data density captured in the scanned scene. The tendency of capturing inconsistencies increases with large linear domains and stationary scanners. In this case, the proposed methodology provides a feasible solution. The proposed scanning methodology identifies the factors responsible for inconsistent densities and defines a data density. It then generates a practical scanning methodology that ensures that the variation in the captured data density is within a reasonable limit, which will guarantee that the computed parameters from the captured data are reliable for effective post-processing.

A high density data set is always beneficial. However, it is often very time consuming to capture such high density point cloud using higher scanning resolution. A significant amount of time can be saved by capturing the domain shape using multi-scan methodology, where multiple, relatively less time consuming scans can be combined to generate high density point cloud. In this thesis, a systematic scan planning methodology is presented for large linear domains.

The approach uses domain's prominent geometric parameters and computes number of scans required to capture the whole domain while ensuring that the data density is within a desired level. The method defines the restricted density value based on the feature size, shape capturing capability or the limited density variation range desired in the final data set. The method provides a systematic means of planning the point cloud acquisition.

An analytical methodology formulates the mathematical relationships for computing the consistent scanning range for each stationary scanner for a given density condition. The effectiveness of the proposed scanning methodology is demonstrated on a long, narrow corridor by generating a consistent density data set. A restricted data density distance (d_{alw}) and low incident angle (θ_l) are necessary to capture and register multiple scans in narrow spaces with small scanner distance (D). The occluded regions can be effectively captured by placing the scanner in front of these regions and thereby reducing incident angle and corresponding surface occlusions. The scanning flaws corresponding to the surface characteristics can be partially improved by multiple scanning of a given domain. The scanning methodology not only works on building interiors, but it also works on any large linear domain such as an exterior building facets and long corridors.

8.2.2   Simplification and layout extraction algorithm

The novel contribution of this layout extraction and simplification algorithms lies in the fact that it works directly on point clouds and can compensate multiple objects and occluded geometries in the scanned scene.

The simplification algorithm identifies and simplifies the data set by removing non-feature points. The process exploits the fact that simple geometries such as planes can be represented with fewer data points and thus can be reduced. Reduction of data points from these planar regions retains other critical non-planar features, which can be efficiently reconstructed as desired. The approach attempts to simplify the point clouds with the intention of retaining the layout points and regenerating the layout from this reduced data set. The approach computes point's importance using a computationally effective, non-recursive approach. The data simplification process is directly applicable on the points and does not require any intermediate geometric model.

A decimated data set generated with this approach is then used to generate the layout contours by projection-based contour extraction. The approach requires a fraction of the total point cloud to compute the section layout. Further, positioning the simplification plane and data slices can generate close or open section layouts for dimensional parameters and route planning applications, respectively.

8.2.3 Segmentation algorithms

The typical characteristics of point cloud representing occupied building interiors are: (1) vast geometric diversity ranging from planar regions to complex freeform objects and (2) geometric uncertainty due to region overlapping and presence of multiple objects. The proposed algorithm successfully tackled these issues using a novel color-based, geometry-assisted, hierarchical approach that extracts both simple as well as complex freeform object point cloud using robust similarity criteria.

A pure-hue based segmentation algorithm was initially presented for achieving rapid clustering of the point cloud. This approach uses spatial conformity and hue coherence to extract data clusters and does not require the computation of any intermediate parameter to reliably segment the data set. Although the results were promising, its dependency on pure color limits it use in colored environments only and therefore, the approach is extended by combing color and geometry to formulate robust differentiating parameters.

A combined hue and geometry based similarity criterion ensures accurate segmentation in geometrically uncertain regions (corresponding to multiple objects or overlapping regions). This combined approach instigated an effective seed selection strategy and permitted the extraction of accurate data clusters, even from geometrically consistent data regions. The shape based hierarchy was used to extract large planar clusters and complex freeform shape clusters in two sequential steps. The first hierarchical stage assigned about 72-84% of scanned data to planar clusters using planar alignment factors and hue-based constraints and the remaining data points were further segmented in the second stage using normal and hue-based approach. The adaptive average density distance approach extracted accurate data clusters from dense as well as sparse data regions. Further, the additional hue based similarity helped with investigating a greater number of neighboring points with uncertain geometric parameters and extracts saturated freeform clusters from geometric

similar and overlapping regions. A suitable set of algorithmic parameters with adaptive average density distance can control the segmentation level from objects to individual surfaces. The segmented clusters can be used in various subsequent post-processing applications such as pattern recognition, object modeling and data simplification. Small clustering inconsistencies are observed in the regions involving uncertain hues and discontinuous boundaries.

8.2.4   Discontinuous boundary detection algorithm

Occluded boundary point identification in point clouds representing multiple objects is an essential pre-processing step. The proposed side-ratio methodology extracts the discontinuous boundary data points directly from point clouds by exploiting their location index. The neighborhood relationship is effectively measured from the sorted point cloud data and exploiting the grid index of the scanned data sets. The occluded boundary data cluster is refined by removing the spurious data points that are generated due to signal noise, varying data densities and non-homogeneous surface properties. The refinement process relies on an average density based outlier detection approach, which removes the sparse data points from the boundary data cluster. The refined boundary points represent an accurate occluded boundary, which can be effectively used in various post-processing applications involving region segmentation, pattern recognition and data simplification.

8.3   Limitations and Current Work

The scan planning methodology presented in this thesis is devised for narrow and long linear domains, which experience large density variations. Specific practical scenarios and the dimensional parameters of the domains are enlisted that can benefit from this proposed methodology. The planning parameters are computed and the results are demonstrated. This methodology is applicable as long as the domain can be approximated using multiple linear sections along the path with each section having acceptable linearity. However, the scope of this methodology is limited to slender domains such as tunnels, sub-ways, building pathways and slightly curved corridors. Further, this method is affected by the user defined scanner positions and capturing accuracy of common references, which is currently controlled though

scan planning guidelines that may not work in all interior applications with diverse domain shapes.

The simplification and layout extraction algorithm has been demonstrated on single scan data sets of building interiors. The application diversity of this algorithm is also demonstrated on freeform data set such as facial scans. A systematic plan is suggested to extend the proposed layout extraction algorithm to registered point clouds obtained from multiple scan positions. However, the present version lacks the robustness in handling connectivity issues at regions where data bifurcates in multiple directions. For example a protrusion on the wall, multiple intersecting regions and common walls from adjacent scans. The present approach suggests the extraction of a layout from each point cloud individually before generating another one from different point cloud that are to be merged together, which require accurate scanner positioning for merging data sets and extracting layouts effectively.

Two hue-based segmentation algorithms have been designed and implemented for this specific application domain of building interiors. The pure-hue based algorithm is fast and effective on the data set where the scanned domain exhibits contrast colors to form a successful segmentation strategy. This algorithm helps in quickly visualizing the objects present in the scanned scene. However, the approach does not work in domains lacking distinctive color attributes. The hierarchical segmentation algorithm formulates a shape adaptive segmentation criterion to extract planar as well as complex interior objects. It uses a shape based hierarchy to define segmentation parameters and compensates for color and geometric uncertainty by formulating a robust geometry based, color assisted differentiating parameters. The approach identifies accurate data clusters from both sparse as well as dense point cloud data set. The current approach requires user defined seed points in the first hierarchical stage for extracting large planar clusters, which can be automated in the future. Further, the planar alignment factor used to extract the large planar clusters will not work for non-planar walls and curved ceilings, limiting its applicability to traditional buildings. The segmentation parameters in the first hierarchical stage can be improved to cover complex domains with severe shape deviations from the planar domains. Further, a convergence criterion for the second stage seed selection process should be defined so that only finite data clusters are extracted from the point clouds.

Lastly, the proposed occluded boundary detection algorithm works on the data set with the known grid indices. In the absence of these indices in the point cloud, the approach does

not work directly on the points and the computation of additional geometric properties is required to identify and segment the occluded boundaries.

8.4   Future Work

The research work presented in this thesis demonstrates an effective system paradigm of capturing and post-processing the scanned data of occupied building interiors. The objective of developing density-based scanning methodology and other information extraction algorithms is to transform the spurious and uninformative point clouds into usable information models. This research can be extended to make these information extraction algorithms more intuitive, robust and computationally efficient so that interactive virtual models can be generated from point cloud data.

The inputs and the outputs of the current algorithms are often the text files representing spatial points, algorithmic parameters and processed results. The resultant point clouds are subsequently viewed in visualization tool such as Meshlab and Rhino3D®. This approach leaves little leverage for *on-the-fly* changes to the algorithmic parameters and makes them non-interactive. Such interactive modifications of the algorithmic parameters are especially useful to control multiple and interdependent parameters and obtain optimum solutions. Ideally, a GUI interface with interactive sliders for the selecting algorithmic parameters and visualization framework for viewing the processed results in real time will help in improving the usability of these algorithms. Such GUIs can be developed using standard application programming interfaces (APIs) such OpenGL. This will not only help in selecting the optimum algorithmic parameters but also give an effective user-interface to the overall project.

The interactive visualization also demands an effective implementation of the algorithms in terms of computational efficiency. The implementation of algorithms was currently done in Microsoft® Visual Studio® using C++ and Matlab. During the implementation of the proposed algorithms, more emphasis could have been given on making them computationally efficient. e.g. higher order geometric parameters (surface normal at a point) could be computed through more efficient eigen decomposition of the k-neighborhood point surface patch than the standard local plane approximation. The speed and interactivity of these developed algorithms can be further enhanced using efficient open project platform such as

such as Point Cloud Library (PCL) (Rusu and Cousins, 2011), where the standard set of algorithms can be used to pre-process (data registration, point cloud filtering for noise and scanning imperfections), basic feature extraction (neighborhood identifications and geometric features computations) and direct visualization (point cloud rendering). The implementation speed of these algorithms can be further increased through parallel computing using multithreading and multi-core data processing.

The algorithms can further be improved by reducing the total number of user defined variables either by grouping them or converting some of them into derived variables. The current scan planning methodology focused on critical shape attributes of slender domains that leads to high variation in data density. Although, it is difficult to formulate a comprehensive scan planning methodology covering all possible domain shapes, it would be interesting to categorize major domains of building interior environments and devising corresponding scanning guidelines.

Certain algorithmic parameters like planar alignment factor lacks intuitiveness when implemented in non-traditional scenario. Although it works fine with the wall lying perpendicular to the floor, but if the walls are at an inclined angle, it is difficult to achieve the same level of accuracy and hence, this aspect can be investigated in future research to develop a more intuitive definition for planar data extractions irrespective of its directions. The hue based seed selection strategy in the second hierarchical stage works effectively in point clouds with distinctive colors and additional geometric properties to compare and check the suitability the selected seed. However, in the absence of hue, the seed selection process occasionally diverges and a recursive process is required to identify good clusters. Thus, the seed selection mechanism can be improved for its robustness, in the absence of contrast colors in the point cloud. It would also be useful to understand the effect of varying algorithmic parameters such as K_r and N_p on data simplification and subsequent layout extraction.

The HSV color models have been successfully used to formulate an effective segmentation strategy when used in conjunction with geometry-based differentiating parameters. The inconsistent results in over-exposed areas can further be investigated using illumination invariant color models (Chong, Gortler and Zickler, 2008; Song, Ge, Qi *et al.*, 2010) and texture based segmentation (Zheng, Sang, Liu *et al.*, 2010). Alternatively, advanced geometric parameters such as curvatures can be used in addition to neighborhood

proximity and point normal to formulate an even better segmentation strategy for point clouds, especially in the regions with higher geometric uncertainty.

8.5   Final Remarks

An effective scan planning methodology and novel information extraction models are proposed in this research. The proposed technique provides accurate acquisition of the captured geometry using density-based scanning methodology, which forms the basis of reliable information model extraction during post processing of the captured data. The information models so created, can be used to create virtual models of the scanned scenes with interactive visualization, geometric modeling and individual object

It is imperative to develop the algorithms presented in this thesis further to make them more robust and generalized for wide variety of applications domains. It will be useful to either integrate the develop algorithms to the standard software packages or develop a stand-alone visualization tools to improve the interactivity of the information models extracted in this research.

In conclusion, this research work helped in understanding several aspects of 3D scanning and information extraction challenges of occupied building interiors. The proposed work has improved the information modeling pipeline by devising an effective scan planning methodology and novel algorithms for data simplification, layout extraction, data segmentation and occluded boundary detection.

APPENDIX- A

The equations used in Section 4, are computed by the following mathematical formulations. The angular scanning resolution is defined as the minimum angle $\square$, between two consecutive rays, that are emitted circumferentially from the scanner, such that

$\alpha = \dfrac{2\pi}{N}$, where N = total number of circumferential points

So $d_1 = D\tan\alpha = x_1$

The value of d_n as shown in Figure 3.3 can be computed as compiled below:

$d_n = D[\tan(n\alpha) - \tan((n-1)\alpha)]$

$$d_n = \dfrac{D\sin(\alpha)}{\cos(n\alpha)\cos((n-1)\alpha)} \tag{A.1}$$

The allowable consistent scanning range (L_n) is recursively computed by comparing this density distance (d_n) with the three cases of allowable data density distance (d_{alw}) as compiled below.

Case I: d_{alw} = Constant == $\xi \cdot$ feature size

The permissible density distance is defined as a user defined value. Maximum value of L_n is computed for given values of D, N and f, with this restricted density condition as follows:

$$d_{alw} = \dfrac{D\sin(\alpha)}{\cos(n\alpha)\cos((n-1)\alpha)}$$

$$d_{alw}D = \dfrac{D}{\cos(n\alpha)}\dfrac{D}{\cos(n\alpha)}\sin(\alpha)$$

$$L_n^2 = \dfrac{d_{alw}D}{\sin\alpha} = \dfrac{d_{alw}D}{\alpha} \qquad \text{\{for small value of } \alpha, \sin\alpha = \alpha\}$$

Thus $L_n^2 = \dfrac{d_{alw}D}{\alpha} = d_{alw}D\dfrac{N}{2\pi}$ $\qquad\qquad$ (A.2)

Case II: $d_{alw} = f\,D$

For this condition of density condition ($d_{alw} = f\,D$), consistent scanning range (L_n) is computed for given values of D, N and f, by reducing Eqn. (A.1) to:

$$d_{alw} = \frac{D\sin(\alpha)}{\cos(n\alpha)\cos((n-1)\alpha)} = f D$$

$$\Rightarrow \frac{D\sin(\alpha)\,D}{\cos(n\alpha)\cos((n-1)\alpha)} = f D^2$$

$$\Rightarrow \frac{D}{\cos(n\alpha)}\frac{D}{\cos(n\alpha)} = \frac{f D^2}{\sin(\alpha)} \qquad \{\cos(n\alpha) \approx \cos((n-1)\alpha)\}$$

$$\Rightarrow L_n L_n = \frac{f D^2}{\sin\alpha} \Rightarrow L_n^2 = \frac{f D^2}{\alpha} \qquad \{\text{for small }\alpha,\ \sin\alpha = \alpha\}$$

Thus $L_n^2 = f D^2 \dfrac{N}{2\pi}$ $\qquad\qquad\qquad\qquad\qquad\qquad$ (A.3)

Case III: $d_{alw} = md_1 = mx_1$

If maximum value of L_n is computed for given values of D, N and m, with a restricted condition of $d_{alw} = md_1 = mx_1$, then the Eqn. (A.1) reduces to:

$$d_{alw} = \frac{D\sin(\alpha)}{\cos(n\alpha)\cos((n-1)\alpha)} = md_1 = mD\tan(\alpha)$$

$$\Rightarrow \frac{D}{\cos(n\alpha)\cos(n\alpha)} = m\frac{D}{\cos(\alpha)}$$

$$L_n = \frac{D}{\cos n\alpha} = mD\frac{\cos n\alpha}{D}\frac{D}{\cos\alpha}$$

$$L_n^2 = \frac{mD^2}{\cos(\alpha)} \qquad\qquad\qquad\qquad\qquad\qquad (A.4)$$

The permissible linear scanning range (X_n), and total number of scans (n_{scans}) can be computed as follows

$$X_n = \sqrt{L_n^2 - D^2} \qquad\qquad\qquad\qquad\qquad\qquad (A.5)$$

$$n_{scans} = \frac{(L - O_l)}{(2X_n - O_l)} \qquad\qquad\qquad\qquad\qquad\qquad (A.6)$$

REFERENCES

1. Adamson, A. & Alexa, M., 2003, "Ray tracing point set surfaces", In: *Proceeding of Shape Modeling International, IEEE Computer Society*, Washington, DC, USA, 271-279.

2. Ailing, T. Chunhui, W. Zhuangde, J. Hongjun, W. & Bingcai, L., 2008, "Study on key algorithm for scanning white-light interferometry", In: *Proceeding of SPIE Proceedings of Ninth International Symposium on Laser Metrology*, 71552N1-N8.

3. Akkouche, S. & Galin, E., 2001, "Adaptive implicit surface polygonisation using marching triangles", *Computer Graphics Forum*, 20(2), 67-80.

4. Alexa, M. Behr, J. Cohen-Or, D. Fleishman, S. Levin, D. & Silva, C. T., 2003, "Computing and rendering point set surfaces", *IEEE Transactions on Visualization and Computer Graphics*, 9(1), 3-15.

5. Alexa, M. Behr, J. Cohen-Or, D. Fleishman, S. & Silva, C. T., 2001, "Point set surfaces", In: *Proceeding of IEEE Visualization*, San Diego, California, USA, 21-28.

6. Alharthy, A. & Berthel, J., 2004, "Detailed building reconstruction from airborne laser data using a moving surface method", In: *Proceeding of International Archives of the Photogrammetry and Remote Sensing*, Istanbul, Turkey, 213-218.

7. Alharthy, A. & Bethel, J., 2002, "Heuristic filtering and 3D feature extraction from LiDAR data", *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34(Part 3/A), 29–34.

8. Alonso, J. I. S. J. Rubio, J. M. Martin, J. J. F. & Fernández, J. G., 2011, "Comparing time-of-flight and phase-shift-The survey of the royal pantheon in the basilica of san isidoro (león)", In: *Proceeding of International Society for Photogrammetry and Remote Sensing (ISPRS) Archives*, 9 pages.

9. Amenta, N. Choi, S. Dey, T. K. & Leekha, N., 2000, "A simple algorithm for homeomorphic surface reconstruction", In: *Proceeding of 16th Annual ACM Symposium on Computational Geometry*, Hong Kong, 213-222.

10. Amenta, N. & Kil, Y. J., 2004, "Defining point-set surfaces", *ACM Transactions on Graphics*, 23(3), 264-70.

11. Araújo, B. R. d. & Jorge, J. A. P., 2005, "Adaptive polygonisation of implicit surfaces", *Computer and Graphics*, 29(5), 686-696.

12. Bahmutov, G. Popescu, V. & Mudure, M., 2006, "Efficient large scale acquisition of building interiors", *Computer Graphic Forum*, 25(3), 655-662.

13. Barequet, G. & Kumar, S., 1997, "Repairing CAD models", In: *Proceeding of IEEE Visualization (editors: Yagel R, Hagen H)*, Phoenix, AZ, USA, 363-70.

14. Barnea, S. Filin, S. & Alchanatis, V., 2007, "A supervised approach for object extraction from terrestrial laser point clouds demonstrated on trees", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, XXXVI(3/W49A), 135-140.

15. Behan, A., 2000, "On the matching accuracy of rasterised scanning laser altimeter data", *International Archives of Photogrammetry and Remote Sensing*, XXXIII(Part 2B), 75-80.

16. Bernardini, F. & Rushmeier, H., 2002, "The 3D model acquisition pipeline", *Computer Graphics Forum*, 21(2), 149-172.

17. Blais, F. Picard, M. & Godin, G., 2004, "Accurate 3D acquisition of freely moving objects", In: *Proceeding of International Symposium on 3D Data Processing, Visualization, and Transmission, 3DPVT 2004*, Thessaloniki, Greece. Los Alamitos, CA, 422-429.

18. Boehler, W. & Marbs, A., 2004, "3D scanning and photogrammetry for heritage recording: A comparison", In: *Proceeding of 12th International Conference on Geoinformatics-Geospatial Information Research: Bridging teh Pacific and Atlantic* 291-298.

19. Boltcheva, D. Bechmann, D. & Thery, S., 2007, "Discrete Delaunay: boundary extraction from voxel objects", In: *Proceeding of Sixth International Conference on 3-D Digital Imaging and Modeling, 2007 (3DIM '07)*, Montreal, Quebec, Canada, 209-216.

20. Bornaz, L. Lingua, A. & Rinaudo, F., 2003, "Multiple scan registration in LiDAR close-range applications", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, XXXIV(Part 5éW12), 72-77.

21. Borodin, P. Novotni, M. & Klein, R., 2002, "Progressive gap closing for mesh repairing", In: *Proceeding of Advances in Modeling Animation and Rendering 2002 (editors: Vince J, Earnshaw R.)*, Springer Verlag; Bradford, UK, 201-13.

22. Boulaassal, H. Landes, T. Grussenmeyer, P. & Tarsha-Kurdi, F., 2007, "Automatic segmentation of building facades using terrestrial laser data", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, 3(3/W52), 65-70.

23. Brenner, C., 2005, "Building reconstruction from images and laser scanning", *International Journal of Applied Earth Observation and Geoinformation*, 6(3-4), 187-198

24. Breunig, M. Kriegel, H. Ng, R. & Sander, J., 2000, "LOF: Identifying density-based local outliers", In: *Proceeding of ACM SIGMOD Conference*, Dallas, Texas, USA, 427-38.

25. Bucksch, A. Lindenbergh, R. & Ree, J. V., 2007, "Error budget of terrestrial laser scanning: Influence of the intensity remission on the scan quality", In: *Proceeding of GeoSiberia*, Novosibirsk, Russia, 13 pages. (DOI=10.1.1.149.3692)

26. Budroni, A. & Böhm, J., 2009, "Toward automatic reconstruction of interiors from laser data", In: *Proceeding of 3D-ARCH*, Trento, Italy, (DOI: 10.1.1.153.5975).

27. Callieri, M. Cignoni, P. Dellepiane, M. & Scopign, R., 2009, "Pushing time-of-flight scanners to the limit", In: *Proceeding of The 10th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST (2009)*, St. Julians, Malta, 85-92.

28. Carr, J. Fright, W. & Beatson, R., 1997, "Surface interpolation with radial basis functions for medical imaging", *IEEE Transactions on Medical Imaging*, 16(1), 96-107.

29. Castellani, U. Livatino, S. & Fisher, R. B., 2002, "Improving environment modelling by edge occlusion surface completion", In: *Proceeding of First International Symposium on 3D Data Processing Visualization and Transmission (3DPVT'02)*, Padova, Italy, 672-675.

30. Chang, D. Y. & Chang, Y. M., 2002, "A freeform surface modeling system based on laser scan data for reverse engineering", *Advanced Manufacturing Technology*, 20(1), 9-19.

31. Chen, G. Kua, J. Shum, S. Naikal, N. Carlberg, M. & Zakhor, A., 2010, "Indoor localization algorithms for a human-operated backpack system", In: *Proceeding of 3D Data Processing, Visualization, and Transmission 2010*, Paris, France, 1-8.

32. Chen, Y. H. Ng, C. T. & Wang, Y. Z., 1999, "Generation of an STL file from 3D measurement data with user-controlled data reduction", *International Journal of Advance Manufacturing Technology*, 15(2), 127-131.

33. Cheng, H. D. Jiang, X. H. Sun, Y. & Wang, J., 2001, "Color image segmentation: advances and prospects", *Pattern Recognition*, 34(1), 2259-2281.

34. Chong, H. Y. Gortler, S. J. & Zickler, T., 2008, "A perception-based color space for illumination-invariant image processing", *ACM Transactions on Graphics (SIGGRAPH '08)*, 27(3), Article No. 61.(DOI: 10.1145/1399504.1360660)

35. Curless, B., 2000, "From range scans to 3D models", *ACM SIGGRAPH Computer Graphics*, 33(4), 38-41.

36. Curless, B. & Levoy, M., 1996, "A volumetric method for building complex models from range images", In: *Proceeding of 23rd Annual Conference on Computer Graphics and Interactive Techniques/SIGGRAPH*, New Orleans, LA, USA, 303-12.

37. Davis, J. Marschner, S. R. M.Garr & Levoy, M., 2002, "Filling holes in complex surfaces using volumetric diffusion", In: *Proceeding of 1st International Symposium on 3D Data Processing, Visualization and Transmission*, Padua, Italy, 428-38.

38. Debevec, P. E. Taylor, C. J. & Malik, J., "Modeling and rendering architecture from photographs: A hybrid geometry and Image-based approach", Technical Report No. UCB/CSD-96-893, University of California, Berkeley, January 1996.

39. Dell'Acqua, F. & Fisher, R., 2002, "Reconstruction of planar surfaces behind occlusions in range images", *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(4), 569–575.

40. Dey, T. K., "Delaunay mesh generation of three dimensional domains", Technical report 5393,

41. Dey, T. K. Giesen, J. Leekha, N. & Wenger, R., 2001, "Detecting boundaries for surface reconstruction using Cocones", *International Journal of Computer Graphics & CAD/CAM*, 16(1), 141-59.

42. Dorninger, P. & Nothegger, C., 2007, "3D segmentation of unstructured point clouds for building modeling", *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVI(3/W49A), 191-196

43. Duan, Y. Cruz, C. & Nicolle, C., 2010, "Architectural reconstruction of 3D building objects through semantic knowledge management", In: *Proceeding of 11th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, London, United Kingdom, 261-266.

44. Duda, R. Hart, P. & Stork, D., 2001, *Pattern Classification* (*Second Edition*), John Wiley & Sons.

45. Duranleau, F. & Poulin, P., 2006, "Patch-based synthesis of geometry textures with point set surfaces", In: *Proceeding of Vision, Modeling and Visualization*, Aachen, Germany, 1-8.

46. Edelsbrunner, H. & Mücke, E. P., 1994, "Three-dimensional alpha shapes", *ACM Transaction on Graphics*, 13(1), 43-72.

47. El-Hakim, S. Gonzo, L. Voltolini, F. Girardi, S. Rizzi, A. Remondino, F. & Whiting, E., 2007, "Detailed 3D modelling of castles", *International Journal of Architectural Computing*, 5(1), 199-220.

48. Elberink, S. O. & Vosselman, G., 2009, "Building reconstruction by target based graph matching on incomplete laser data: analysis and limitations", *Sensors*, 9(8), 6101–6118.

49. Fabio, R., 2003, "From point cloud to surface: the modeling and visualization problem", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, 36(5/W10), 11 pages.

50. Filin, S., 2002, "Surface clustering from airborne laser scanning data", *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXIV(3A/B), 119-124.

51. Fleischer, M. Windecker, R. & Tiziani, H. J., 2001, "Theoretical limits of scanning white-light interferometry signal evaluation algorithms", *Applied optics*, 40(17), 2815-2820.

52. Fleishman, S. Cohen-Or, D. Alexa, M. & Silva, C. T., 2003, "Progressive point set surfaces", *ACM Transactions on Graphics*, 22(4), 997-1011.

53. Forlani, G. Nardinocchi, C. Scaioni, M. & Zingaretti, P., 2006, "Complete classification of raw LiDAR data and 3D reconstruction of buildings", *Pattern Analysis & Applications*, 8(4), 357–374.

54. Furukawa, Y. Curless, B. Seitz, S. M. & Szeliski, R., 2009, "Reconstructing building interiors from images", In: *Proceeding of 12th International Conference on Computer Vision (ICCV 2009)*, Kyoto, Japan, 80–87.

55. Gonzalez-Aguilera, D. & Gomez-Lahoz, J., 2009, "Forensic terrestrial photogrammetry from a single image", *Journal of Forensic Science*, 54(1), 1376-1387.

56. Grazzini, J. Prasad, L. & Dillard, S., 2010, "Simultaneous hierarchical segmentation and vectorization of satellite images through combined non-uniform data sampling and anisotropic triangulation", In: *Proceeding of SPIE Symposium on Image and Signal Processing for Remote Sensing XVI (SPIE 7830)*, Toulouse, France, 78300F. (DOI: 10.1117/12.865047)

57. Gruen, A. Remondino, F. & Zhang, L., 2004, "Photogrammetric reconstruction of the Great Buddha of Bamiyan", *The Photogrammetric Record*, 19(1), 177-199.

58. Guarnieri, A. Vettore, A. & Remondino, F., 2004, "Photogrammetry and ground-based laser scannring: Assessing of metric accuracy of the 3D model of pozzoveggiani church", In: *Proceeding of Laser Scanning and Photogrammetry (Positioning and Measurement Technologies and Practices II)*, Athens, Greece, 15 pages.

59. Guennebaud, G. & Gross, M., 2007, "Algebraic point set surfaces", In: *Proceeding of ACM SIGGRAPH*, San Diego, California, USA, article 23.

60. Hähnel, D. Burgard, W. & Thrun, S., 2003, "Learning compact 3D models of indoor and outdoor environments with a mobile robot", *Robotics and Autonomous Systems*, 44(1), 15–27.

61. Hawkins, D., 1980, *Identification of outliers*, Chapman and Hall.

62. Hongjian, Y. & Shiqiang, Z., 2006, "3D building reconstruction from aerial CCD image and sparse laser sample data", *Optics and Lasers in Engineering*, 44(6), 555-566.

63. Hoover, A. Jean-Baptiste, G. Jiang, X. X. Flynn, P. J. Bunke, H. Goldgof, D. B. Bowyer, K. K. Eggert, D. W. Fitzgibbon, A. W. & Fisher, R. B., 1996, "An experimental comparison of range image segmentation algorithms", *Pattern Analysis and Machine Intelligence*, 18(7), 673-689.

64. Hoppe, H. DeRose, T. Duchamp, T. McDonald, J. & Stuetzle, W., 1992, "Surface reconstruction from unorganized points", In: *Proceeding of 19th Annual Conference on*

Computer Graphics and Interactive Techniques/ ACM SIGGRAPH, New York, USA, 71-78.

65. Hu, S. Zha, H. & Zhang, A., 2006, "Registration of multiple laser scans based on 3D contour features", In: *Proceeding of Tenth Conference on Information Visualization*, London, England, 725-730.

66. Huang, J. & Menq, C.-H., 2001, "Automatic data segmentation for geometric feature extraction from 3-D coordinate points", *IEEE Transactions on Robotics and Automation*, 17(3), 268-279.

67. Huber, D. Akinci, B. Tang, P. Oliver, A. A. Okorn, B. E. & Xiong, X., 2010, "Using laser scanners for modeling and analysis in architecture, engineering, and construction", In: *Proceeding of 44th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, USA, 7p.

68. Jain, A. Murty, M. & Flynn, P., 1999, "Data clustering: a review", *ACM Computing Surveys*, 31(3), 264-323.

69. Jain, A. K., 2010, "Data clustering: 50 years beyond K-means", *Pattern Recognition Letters*, 31(8), 651-666.

70. Johnston, M. & Zakhor, A., 2008, "Estimating building Floor-plans from exterior using laser scanners", In: *Proceeding of SPIE Electronic Imaging conference, 3D Image Capture and Applications (SPIE-6805)*, San Jose, California, 11p.

71. Ju, T., 2004, "Robust repair of polygonal models", In: *Proceeding of ACM SIGGRAPH 2004 (editor: Marks J)*, Los Angeles, California, USA, 888-95.

72. Kalaiah, A. & Vashney, A., 2003, "Modeling and rendering of points with local geometry", *IEEE Transactions on Visualization and Computer Graphics*, 9(1), 30-42.

73. Kim, J.-S. Lee, J.-C. Kang, I.-J. Cha, S.-Y. Choi, H. & Lee, T.-G., 2008, "Extraction of geometric information on highway using terrestrial laser scanning technology", *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVII(B5), 539-544.

74. Kim, K. & Shan, J., 2011, "Building roof modeling from airborne laser scanning data based on level set approach", *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(4), 484-497.

75. Knorr, E. Ng, R. T. & Tucakov, N., 2000, "Distance-based outliers: Algorithms and applications", *International Journal of very large data bases*, 8(3-4), 237-53.

76. Křemen, T. Koska, B. & Pospíši, J., 2006, "Verification of laser scanning systems quality", In: *Proceeding of XXIII FIG Congress on Shape the Change*, Munich, Germany, 16 pages.

77. Kua, J. Corso, N. & Zakhor, A., 2012, "Automatic loop closure detection using multiple cameras for 3D indoor localization", In: *Proceeding of SPIE 8296, 82960V*, (DOI: 10.1117/12.916639).

78. Lee, K. H. & Woo, H., 2000, "Direct integration of reverse engineering and rapid prototyping", *Computers and Industrial Engineering*, 38(1), 21-38.

79. Lee, K. H. Woo, H. & Suk, T., 2001, "Point data reduction using 3D grids", *International Journal of Advanced Manufacturing Technology*, 18(3), 201-210.

80. Lerma, J. L. Navarro, S. Cabrelles, M. Seguí, A. E. Haddad, N. & Akasheh, T., 2011, "Integration of laser scanning and imagery for photorealisitic 3D architectural

documentation", In: Laser Scanning, Theory and Applications, (ed. C.-C. Wang), InTech, (ISBN: 978-953-307-205-0)

81. Li, W. X. & Mitchell, L. D., 1995, "Laser scanning system testing–Errors and improvements", *Measurement*, 16(2), 91-101.

82. Li, Y. & Gu, P., 2004, "Free-form surface inspection techniques state of the art review", *Computer-Aided Design*, 36(13), 1395-1417.

83. Lichti, D. Gordon, S. & Tipdecho, T., 2005, "Error models and propagation in directly geo referenced terrestrial laser scanner networks", *Journal of Survey Engineering*, 131(4), 8 pages.10.1061/(ASCE)0733-9453(2005)131:4(135)

84. Lichti, D. D., 2007, "Error modelling, calibration and analysis of an AM-CW terrestrial laser scanner system", *ISPRS Journal of Photogrammetry and Remote Sensing, Volume 61, Issue 5, p.18 (2007)*, 61(7), 307-324.

85. Liepa, P., 2003, "Filling holes in meshes", In: *Proceeding of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, Aachen, Germany, 200-05.

86. Linsen, L., "Point cloud representation", Technical Report, Faculty of Informatics,

87. Liu, T. Carlberg, M. Chen, G. Chen, J. Kua, J. & Zakhor, A., 2010, "Indoor localization and visualization using a human-operated backpack system", In: *Proceeding of International Conference on Indoor Positioning and Indoor Navigation (IPIN, 2010)*, Hoenggerberg, Switzerland, 1-10.

88. Lu, S. Shi, R. & Zhu, G., 2008, "Combinative representative of TLS point cloud and 3D virtual reality for cultural heritage preservation planning", *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(B2), 1077-1080.(DOI: 10.1.1.184.1965)

89. Mayer, R., 1999, *Canadian: Invention and innovation from Canada's National Research Council (NRC)*, Raincoast Pub-Remainder (Nov 1 1999) (ISBN 1551922665).

90. Nguyen, V. Gächter, S. & Martinelli, A., 2007, "A comparison of line extraction algorithms using 2D range data for indoor mobile robotics", *Autonomous Robots*, 23(2), 97–111.

91. Ning, X. Zhang, X. Wang, Y. & Jaeger, M., 2009, "Segmentation of architecture shape information from 3D point cloud", In: *Proceeding of 8th International Conference on Virtual Reality Continuum and its Applications in Industry (VRCAI '09)* Yokohama, Japan, 127-132.

92. Nooruddin, F. S. & Turk, G., 2003, "Simplification and repair of polygonal modes using volumetric techniques", *IEEE Transactions on Visualization and Computer Graphics*, 9(2), 191-205.

93. Ohno, K. Kawahara, T. & Tadokoro, S., 2009, "Development of 3D laser scanner for measuring uniform and dense 3D shapes of static objects in dynamic environment", In: *Proceeding of IEEE International Conference on Robotics and Biomimetics*, Bangkok, Thailand, 2161-2167.

94. Oliver, A. A. & Huber, D., "Reconstruction of wall surfaces under occlusion and clutter in 3D indoor environment", Tech. Report CMU-RI-TR-10-12, Robotics Institute, Carnegie Mellon University (April, 2010) 33 pages.

95. Oliver, A. A. & Huber, D., 2011, "3D reconstruction of interior wall surfaces under occlusion and clutter", In: *Proceeding of 3D Imaging, Modeling, Processing, Visualization*

and Transmission (3DIMPVT), Hangzhou, China, 275 - 281. (DOI: 10.1109/3DIMPVT.2011.42)

96. Park, H. T. Chang, M. H. & Park, S. C., 2007, "A slicing algorithm of point cloud for rapid prototyping", In: *Proceeding of 2007 Summer Computer Simulation Conference*, San Diego, California, USA, Article No. 27.

97. Pauly, M. Gross, M. & Kobbalt, L. P., 2002, "Efficient simplification of point sampled surfaces", In: *Proceeding of 13th IEEE Visualisation Conference*, Boston, MA, USA, 163-170.

98. Pauly, M. Keiser, R. Kobbelt, L. P. & Gross, M., 2003, "Shape modeling with point-sampled geometry", *ACM Transaction on Graphics* 22(3), 641-50.

99. Pauly, M. Kobbalt, L. K. & Gross, M., "Multi-resolution modeling of point-sampled geometry", Tech. Report-37, ETH Zurich, Switzerland 2004.

100. Pauly, M. Mitra, N. J. & Guibas, L., 2004, "Uncertainty and variability in point cloud surface data", In: *Proceeding of Eurographics Symposium on Point-Based Graphics 2004*, ETH Zurich, Switzerland, 77-84.

101. Pavlidis, T. & Horowitz, S., 1974, "Segmentation of plane curves", *IEEE Transactions on Computers*, 23(8), 860-870.

102. Pfeifer, N. Dorninger, P. Haring, A. & Fan, H., 2007, "Investigating terrestrial laser scanning intensity data: quality and functional relations.", In: *Proceeding of 8th International Conference on Optical 3-D Measurement Techniques*, Zurich, Switzerland, 328-337.

103. Pierre, A. Giuliana, U. Craig, G. & Marco, A., 2007, *Recent advances in remeshing of surfaces*, Springer Publishing, Berlin.

104. Polleyfeys, M. Koch, R. Vergauwen, M. & Gool, L. V., 1999, "Hand-held acquisition of 3D models with a video camera", In: *Proceeding of 2nd International Conference on 3D Digital Imaging and Modeling*, Ottawa, Canada, 14-23.

105. Pu, S. & Vosselman, G., 2006, "Automatic extraction of building features from terrestrial laser scanning", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, 36(5), 25-27.

106. Pu, S. & Vosselman, G., 2006, "Automatic extraction of building features from terrestrial laser scanning", *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXV(6), 5p.

107. Pu, S. & Vosselman, G., 2007, "Extracting windows from terrestrial laser scanning", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, XXXVI-3(3/W52), 320-325.

108. Pu, S. & Vosselman, G., 2009, "Knowledge based reconstruction of building models from terrestrial laser scanning data", *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(6), 575-584.

109. Rabbani, T. Heuvel, F. A. v. d. & Vosselman, G., 2006, "Segmentation of point clouds using smoothness constraint", *International Archives of the Photogrammetry, Remote Sensing Information Sciences, Symposium on Image Engineering and Vision Metrology*, XXXVI(PArt 5), Sept. 25-27, 2006.

110. Richter, E., 2009, "Denoising point cloud data of small-structured free form-surfaces captured by a phase-based laser scanner", *ISPRS Laser Scanning*, 38(1), 37-42.

111. Rocchini, C. Cignoni, P. Montani, C. & Scopigno, R., 1999, "Multiple textures stitching and blending on 3D objects", In: *Proceeding of 10th Eurographics Workshop on Rendering*, Granada, Spain, 127-138.

112. Rottensteiner, F., 2003, "Automatic generation of high-quality building models from LiDAR data", *IEEE Computer Graphics and Applications*, 23(6), 42-50.

113. Rottensteiner, F. & Briese, C., 2002, "A new method for building extraction in urban areas from high-resolution LiDAR data", *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* XXXIV(3/A), 295-301.

114. Rottensteiner, F. Trinder, J. Clode, S. & Kubik, K. K. T., 2005, "Automated delineation of roof planes from LIDAR data", In: *Proceeding of ISPRS Workshop Laser scanning 2005*, Enschede, the Netherlands, 221-226.

115. Rousseeuw, P. & Leroy, A., 1987, *Robust regression and outlier detection*, John Wiley and Sons Inc.

116. Rusu, R. B. & Cousins, S., 2011, "3D is here: point cloud library (PCL)", In: *Proceeding of IEEE International Conference on Robotics and Automation (ICRA)*, Shangai, China, 4 pages.

117. Rusu, R. B. Marton, Z. C. Blodow, N. Dolha, M. & Beetz, M., 2008, "Towards 3D point cloud based object maps for household environments", *Robotics and Autonomous Systems*, 56(11), 927-941.

118. Sampath, A. & Shan, J., 2007, "Building boundary tracing and regularization from airborne LiDAR point clouds", *Photogrammetric Engineering and Remote Sensing*, 73(7), 805–812.

119. Sampath, A. & Shan, J., 2010, "Segmentation and reconstruction of polyhedral building roofs from aerial LiDAR point clouds", *IEEE Transactions on Geoscience and Remote Sensing*, 48(3), 1554-1567.

120. Sankaranarayanan, J. Samet, H. & Varshey, A., 2007, "A fast all nearest neighbour algorithm for applications involving large point-clouds", *Computers & Graphics*, 31(1), 157-174.

121. Sareen, K. K. Knopf, G. K. & Canas, R., 2009, "Contour based 3D point data simplification for facial reconstruction", *International Journal of Shape Modeling*, 15(2), 19-43.

122. Sareen, K. K. Knopf, G. K. & Canas, R., 2010, "Rapid clustering of colorized 3D point cloud data for reconstructing building interiors", In: *Proceeding of IEEE International Symposium on Opto-mechatronic Technologies* Toronto, Canada, 1-6. DOI: 10.1109/ISOT.2010.5687331

123. Sareen, K. K. Knopf, G. K. & Canas, R., 2011, "Hierarchical data clustering approach for segmenting colored three-dimensional point clouds of building interiors", *Optical Engineering*, 50(077003.(DOI: 10.1117/1.3599868)

124. Schnabel, R. Wahl, R. & Klein, R., 2007, "Efficient RANSAC for point-cloud shape detection", *Computer Graphics Forum*, 26(2), 214-226.

125. Schroeder, W. J. Zarge, J. A. & Lorensen, W. E., 1992, "Decimation of triangle meshes", *ACM SIGGRAPH Computer Graphics*, 26(2), 65-70.

126. Schuon, S. Theobalt, C. Davis, J. & Thrun, S., 2008, "High-quality scanning using time-of-flight depth super-resolution", In: *Proceeding of IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08, Anchorage, AK, USA, 1-7.

127. Shih, N.-J. & Hu, J.-Z., 2007, "The application of 3D range images in the interior design practice", In: *Proceeding of 11th International Conference Information Visualization (IV'07), IEEE Computer Society*, Washington, DC, 825-830.

128. Shin, H. Y. Park, S. Y. & Park, E. J., 2004, "Direct slicing of a point set model for rapid prototyping", *Computer Aided Design and Applications*, 1(1-4), 109-115.

129. Sithole, G. & Vosselman, G., 2004, "Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds", *ISPRS Journal of Photogrammetry and Remote Sensing*, 59(1–2), 85–101.

130. Song, D.-L. Ge, L.-H. Qi, W.-W. & Chen, M., 2010, "Illumination invariant color model selection based on genetic algorithm in robot soccer", In: *Proceeding of 2nd International Conference on Information Science and Engineering (ICISE), 2010*, Hangzhou, China, 1245-1248. (DOI: 10.1109/ICISE.2010.5689376)

131. Song, H. & Feng, H., 2008, "A global clustering approach to point cloud simplification with a specified data reduction ratio", *Computer Aided Design*, 40(3), 281-292.

132. Sotoodeh, S., 2006, "Outlier detection in laser scanner point clouds", *Journal of International Society for Photogrammetry and Remote Sensing*, 36(5), 297-302.

133. Sotoodeh, S., 2007, "Hierarchical clustered outlier detection in laser scanner point clouds", *Journal of International Society for Photogrammetry and Remote Sensing*, 36(3), 383-88.

134. Stamos, I. Gene, Y. Wolberg, G. & Zokai, S., 2006, "3D modeling using planar segments and mesh elements", In: *Proceeding of Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT)*, Chapel Hill, USA, 599–606.

135. Su, Y. Y. Hashash, Y. M. A. & Liu, L. Y., 2006, "Integration of construction as-built data via laser scanning with geotechnical monitoring of urban excavation", *Journal of Construction Engineering and Management*, 32(12), 1234-1241.

136. Sun, L. Tian, H. Wu, D. & Du, Z., 2010, "Three-dimensional geometric modeling of the spine based on reverse engineering technology", In: *Proceeding of 3rd International Conference on Biomedical Engineering and Informatics (BMEI 2010)*, Yantai, China, 1292-1295.

137. Suveg, I. & Vosselman, G., 2002, "Automatic 3D building reconstruction", In: *Proceeding of SPIE Conference on Three-Dimensional Image Capture and Applications (Vol. 4661)*, 59-69. (DOI: 10.1117/12.460181)

138. Tarsha-Kurdi, F. Landes, T. & Grussenmeyer, P., 2007, "Hough-transform and extended RANSAC algorithms for automatic detection of 3d building roof planes from LiDAR data", In: *Proceeding of ISPRS Workshop on Laser Scanning*, Espoo, Finland, 407-412.

139. Thrun, S. Martin, C. Liu, Y. Hahnel, D. Emery-Montemerlo, R. Chakrabarti, D. & Burgard, W., 2004, "A real-time expectation maximization algorithm for acquiring multi-planar maps of indoor environments with mobile robots", *IEEE Transactions on Robotics and Automation*, 20(3), 433-443.

140. Tseng, A. A. Tanaka, M. & Leeladharan, B., 2002, "Laser-based internal profile measurement system", *Automation in Construction*, 11(6), 667–679.

141. Turk, G. & Levoy, M., 1994, "Zippered polygon meshes from range Images", In: *Proceeding of 21st Annual Conference on Computer Graphics and interactive Techniques SIGGRAPH 1994*, Orlando, Florida, USA, 311-18.

142. Várady, V. Martin, R. R. & Cox, J., 1997, "Reverse engineering of geometric models-an introduction", *Computer Aided Design*, 29(4), 255-68.

143. Verma, V. Kumar, R. & Hsu, S., 2006, "3D Building detection and modeling from aerial LiDAR data", In: *Proceeding of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR'06. IEEE Computer Society*, Washington, DC, USA, 2213–2220.

144. Vosselman, G. & Dijkman, S., 2001, "3D building model reconstruction from point clouds and ground plans", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, XXXIV(3/W4), 37-43.

145. Vosselman, G. Gorte, B. G. H. Sithole, G. Rabbani, T. & Barequet, G., 2004, "Recognising structure in laser scanner point clouds", *International Archives of the Photogrammetry, Remote Sensing Information Sciences*, XXXVI(8/W2), 33-38.

146. Wang, J. & Oliveira, M. M., 2007, "Filling holes on locally smooth surface reconstruction from point clouds", *Image and Vision Computing*, 25(1), 103-113.

147. Wang, R. & Luebke, D., 2003, "Efficient reconstruction of indoor scenes with color", In: *Proceeding of Fourth International Conference on 3-D Digital Imaging and Modeling (3DIM)*, Banff, Canada 402-409.

148. Wani, M. A. & Arabnia, H. R., 2003, "Parallel edge region-based segmentation algorithm targeted at reconfigurable multi ring network", *Journal of Supercomputing*, 25(1), 43–62.

149. Wei, S., 2008, "Building boundary extraction based on LiDAR point clouds data", *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(Part B3b), 157-162.

150. Weyrich, T. Pauly, M. Heinzle, S. Keiser, R. Scandella, S. & Gross, M., 2004, "Post-processing of scanned 3d surface data", In: *Proceeding of Symposium on Point-Based Graphics*, ETH Zurich, Switzerland, 85-94.

151. Wolfart, E. Sequeira, V. Ng, K. Butterfield, S. Gonçalves, J. G. M. & Hogg, D., 1999, "Hybrid approach to the construction of triangulated 3D models of building interiors", *Computer Vision Systems, Lecture Notes in Computer Science* 1542(1999), 489-508.(DOI: DOI: 10.1007/3-540-49256-9_29)

152. Wu, X. J. Wang, M. Y. & Han, B., 2008, "An automatic hole-filling algorithm for polygon meshes", *Computer-Aided Design and Applications*, 5(6), 889-899.

153. Xianfeng, H. Xioguang, C. Fan, Z. & Jianya, G., 2008, "Side ratio constraint based precise boundary tracing algorithm for discrete point clouds", *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVII(Part B3b), 349-353.

154. Xiong, X. & Huber, D., 2010, "Using context to create semantic 3D models of indoor environments", In: *Proceeding of British Machine Vision Conference (BMVC) 2010*, Cardigan Bay, UK, 11p.

155. Xu, Z. Q. Ye, S. H. & Fan, G. Z., 2002, "Color 3D reverse engineering", *Journal of Materials Processing Technology*, 129(1-3), 495-499.

156. Yang, U. Kim, B. Toh, K.-A. & Sohn, K., 2010, "Illumination-invariant color space and its application to skin-color detection", *Optical engineering*, 49(10), 107004.

157. Zhan, Q. Liang, Y. & Xiao, Y., 2009, "Color-based segmentation of point clouds", *ISPRS Laser Scanning Workshop*, 248-252.

158. Zhang, C. & Wang, P., 2000, "A new method of color Image Segmentation Based on Intensity and Hue Clustering", In: *Proceeding of 15th International Conference on Pattern Recognition, 2000*, Boston, MA, USA, 613-616.

159. Zhang, X. Yang, J. Liu, Z. & Wang, X., 2008, "Segmenting foreground from similarly colored background", *Optical Engineering*, 47(7), 077002.

160. Zheng, J. Y., 1994, "Acquiring 3D models from sequences of contours", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2), 163–178.

161. Zheng, Q. Sang, N. Liu, L. & Gao, C., 2010, "Textured image segmentation based on modulation models", *Optical Engineering*, 49(9), 097009.

162. Zhou, Q. Y. & Neumann, U., 2008, "Fast and extensible building modeling from airborne LiDAR data", In: *Proceeding of 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM GIS 2008)*, Irvine, CA, USA, 1-8. (DOI: 10.1145/1463434.1463444)

163. Zhuang, H. & Roth, Z. S., 1995 "Modeling gimbal axis misalignments and mirror center offset in a single-beam laser tracking measurement system", *The International Journal of Robotics Research*, 14(3), 211-222.

164. Zwicker, M. Pauly, M. Knoll, O. & Gross, M., 2003, "Pointshop3D: An interactive system for point-based surface editing", *ACM Transactions on Graphics*, 21(3), 322-329.

CURRICULUM VITAE

Name:	**Kuldeep Kumar Sareen**

Post-secondary Education and Degrees:

B.E.(*Mechanical Engineering*)
Punjab Technical University
Jalandhar, Punjab, India
(1999)

M.Tech. (*Mechanical Design*)
Indian Institute of Engineering and Technology, Bombay
Mumbai, Maharashtra, India
(2001)

Ph. D.
Western University
London, Ontario, Canada
(2012)

Honors and Awards:

Early Faculty Induction Program Fellowship, IIT, Bombay
(1999-2001)

Distinction in M. Tech. (3rd rank in batch)
(1999-2001)

College Topper in Bachelor of Engg., Punjab Technical University, India
(1995-1999)

Related Work Experience

Quality Engineer (*Body-in-white, Car Exterior*)
Chrysler Canada (Brampton Assembly Plant)
Brampton, Ontario, Canada
(2012 – till now)

3D Scanning and Process Researcher and Teaching Assistant
Western University, Canada
London, Ontario, Canada
(2007– 2012)

Guest Worker, National Research Council of Canada
Institute for Research in Construction (IRC-NRC), Canada
London, Ontario, Canada
(2008–2012)

Assistant Professor and Design Consultant
Mechanical Engg. Deptt, Guru Nanak Dev Engineering College
Ludhiana, Punjab, India
(2001 – 2006)

Publications:

Theses

Sareen, K K (2012) "*Consistent Density Scanning and Information Extraction from Point Clouds of Building Interiors*". Doctor of Philosophy (Ph.D.), Department of Mechanical and Material Engineering, Faculty of Engineering. Western University, London, Ontario, Canada.

Sareen, K K (2001) *Design of Reversible Tractor*. Master of Technology (M.Tech.), Department of Mechanical Engineering, Indian Institute of Technology, Bombay, Mumbai, Maharashtra, India.

Refereed Journal Papers

Sareen, K K, Knopf, G K, Canas R (2011) "*Hierarchical data clustering approach for segmenting colored 3D point clouds of building interior*s" Optical Engineering Journal, Vol. 50 No. 7 pp. 0077003 (July 2011)

Sareen, K K, Knopf, G K, Canas R (2011) "*Consistent point clouds of narrow spaces using multi-scan domain mapping*" International Journal of Civil and Infrastructure Engineering, published online-8 Nov 2011.(DOI: 10.1111/j.1467-8667.2011.00742.x)

Sareen, K K, Knopf, G K, Canas R (2009) "*Contour based 3D point data simplification for facial reconstruction*" International Journal of Shape Modeling, Vol. 15, No. 1-2, pp 19-43.

Grewal C, Gill S and **Sareen K** (2008) "*A multi-criteria logistics outsourcing decision making using Analytic Hierarchy Process (AHP)*" International Journal of Services Technology and Management, Vol. 9, No. 1, pp 01-13.

Grewal C and **Sareen K (2006)** "*Effectiveness of ISO 9000 standards in Indian educational institutions*", International Journal of Service Technology and Management, Vol. 7, No. 3, pp 403-415.

Grewal C and **Sareen K**, (2006), "*Development of Lean Improvement Model: A Case Study*", Industrial Engineering Journal, Vol. 35, No 5, pp 24-27.

Grewal C and **Sareen K** (2005), "*Application of Deming philosophy to improve technical education*", The Indian Journal of Technical Education. Vol. 28 No 1, Jan-Mar, pp 56-59.

Grewal D S, **Sareen K K and** Grewal C (2003), "*QFD based approach to improve the quality of technical education*", The Indian Journal of Technical Education Vol. 26 No 2, pp 64- 68.

Refereed Conference Papers

Sareen, K K, Knopf, G K, Canas R (2010), "*Rapid Clustering of Colorized 3D Point Cloud Data for Reconstructing Building Interiors*," Accepted in IEEE International Symposium on Optomechatronic Technologies (Toronto, 25-27th Oct, 2010), pp. 1-6; DOI: 10.1109/ISOT.2010.5687331

Sareen, K K, Knopf, G K, Canas R (2010), "*Point cloud simplification and civil building layout extraction from large data-set involving multiple objects*," in Proceedings of the 21st IASTED International Conference on Modelling and Simulation (MS2010), July 15-17,

2010, Banff, Alberta, Canada, 2010. pp 79-86.

Sareen, K K, Knopf, G K, Canas R (2009), "*Contour-based 3D Point Cloud Simplification for Modeling Freeform Surfaces*," in Proceedings of the IEEE Symposium on human factors and Ergonomics: Modeling, International Conference on Science and Technology for Humanity (TIC-STH2010), Sept 26-27, Toronto, Canada, 2009, pp. 381-386.

Sareen, **K K,** Knopf, G K, Canas R (2009), "*Surface reconstruction from sliced point cloud data for designing facial prosthesis*," in Proceedings of the IEEE Symposium on biomedical applications, International Conference on Science and Technology for Humanity (TIC-STH2010), Sept 26-27, Toronto, Canada, 2009, pp. 6-11.

Singh, B, **Sareen**, **K K** and Singh, **C** (2005) "*Parameters for ergonomically designed sewing machine work station using worker's perceptions*", International conference on Human Factors and Ergonomics, IIT Guhawati, Dec, 10-12, 2005, pp 205-211.

Singh B, **Sareen K K** and Singh C (2004), "*Ergonomic consideration in Sewing Machine workstation design*", National conference on Precision Manufacturing, SLIET Longowal, 11-12 Dec 2004, pp 192-195.

Sareen K K and Singh C (2003), "*Product Design in Distributed Design Environment: A Modular Approach*", International Conference on Emerging Technologies, 19-21 Dec, 2003, KIIT Bhubaneswar, 38- 43.

Kumar A, **Sareen K** (2003) "*Computer Aided Design verification of Critical Components of Oil Expeller of Capacity 50 TPD*" 31st Oct-1st Nov, 2003, National Conference on Recent Development in Mechanical Engineering, TIET, Patiala, pp 9 -14.

Text-books/Book Chapters/Newsletter

- Knopf G K, **Sareen K. K**, and Canas R (2011), "Reconstructing complex scenes for virtual reality" SPIE Newsroom; DOI 10.1117/2.1201110.003920.
- **Sareen K K** and Grewal C S (2006), "*CAD/CAM: Theory and Concepts*", S. Chand Publishers, New Delhi, India, 1st Edition 2006. and 2nd edition 2007,