

INSTITUTO POLITÉCNICO NACIONAL

CENTRO DE INVESTIGACIÓN EN CIENCIA APLICADA Y TECNOLOGÍA AVANZADA

UNIDAD QUERÉTARO

POSGRADO EN TECNOLOGÍA AVANZADA

Generación de mapas 3D globales a partir de primitivas geométricas locales

TESIS

QUE PARA OBTENER EL GRADO DE

MAESTRÍA EN TECNOLOGÍA AVANZADA

PRESENTA Ing. Denis Eduardo Hernández García

DIRECTORES DE TESIS

Dr. José Joel González Barbosa Dr. Juan Bautista Hurtado Ramos



Santiago de Querétaro Qro. Mayo del 2011

SIP-14-bis



INSTITUTO POLITÉCNICO NACIONAL SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de		Querétaro	siendo las	10:15	horas del día	30	del mes de
mayo	del	2011 se reuniero	n los miembros	de la Co	misión Revisora	de Te	esis, designada
por el Colegio de Profesores de Estudios de Posgrado e Investigación de CICATA-Querétaro							TA-Querétaro
para exami	nar la	tesis titulada:					

" Generación de mapas 3D globales a partir de primitivas geométricas locales "

Presentada por el alumno:

Hernández	García	D	enis I	Edua	rdo			
Apellido paterno	Apellido materno		Nor	nbre(s)			_	
		Con registro: B	0	9	1	6	4	2

aspirante de:

Maestría en Tecnología Avanzada

Después de intercambiar opiniones los miembros de la Comisión manifestaron **APROBAR LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Directores de tesis 1a, Losa ta Hurtado Ramos Dr. José Joel González Barbosa Dr. Julan E Dr. Francisco Javier Ornelas Rodríguez Dra. María Teresa García Ramírez Dr. Eduardo Castillo Castañeda PRESIDENTE DEL COLEGIO DE PROPE MEXICO INSTITUTO POLITECNICO NACIONAL CENTRO DE INVESTIGACIÓN EN CIENCIA APLICADA TECNOLOGIA AVANZADA UNIDAD QUEPÉTARO Dr. Jorge Adalberto Huerta Ruelas



INSTITUTO POLITÉCNICO NACIONAL secretaría de investigación y posgrado

CARTA CESIÓN DE DERECHOS

En la Ciudad de Querétaro el día 07 del mes junio del año 2011, el que suscribe Denis Eduardo Hernández García alumno del Programa de Maestría en Tecnología con número de registro B091642 adscrito al Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del Dr. José Joel González Barbosa y Dr. Juan Bautista Hurtado Ramos, y cede los derechos del trabajo intitulado "Generación de mapas 3D globales a partir de primitivas geométricas locales", al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección Cerro <u>Blanco</u> No. 141. Colinas del Cimatario, Santiago de Querétaro, Querétaro, C.P. 76090. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Dennis Hernandez 6 Denis Eduardo Hernândez García

Nombre y firma

Agradecimientos

Esta sección es dedicada a todas aquellas personas que me han ayudado, sin ellas, hubiese sido del todo imposible el éxito de la elaboración de este proyecto, en la que tanta ilusión he puesto.

Gracias...

- De forma muy especial al Dr. José Joel González Barbosa, por ser un verdadero maestro y amigo a lo largo de la tesis, por su valiosa dirección y apoyo en la realización de este trabajo.
- A mis maestros de Análisis de Imágenes del CICATA, por compartir su sabiduría y amistad en mi formación de maestro. Dr. Francisco Javier Ornelas Rodríguez y Dr. Juan Hurtado Ramos.
- Con cariño al Dr. Joaquín Salas Rodríguez, por su incondicional apoyo y facilitación de ingresar al programa de maestría.
- A mis amigos del CICATA, por su apoyo, alegrias y confianza. Muy en especial a la Lic. Edith Muñoz Olin.
- A mis padres por guiarme por el camino de la vida, el cual me llevo hacia el CICATA, unidad Querétaro.
- Con todo mi amor a Yasmina Zamora de Hernández mi esposa, compañera y amiga, por su inestimable apoyo y comprensión para sobrellevar el abandono y estrés al que ha estado sometida durante todas las horas que he dedicado a este trabajo.

 A Dios todopoderoso, que nos regala la sabiduría y el sol cada día de nuestras vidas, para compartir la dicha con todos los seres vivos.

De igual manera agradezco a las instituciones que me han dado el apoyo académico y económico para realizar este proyecto.

Gracias...

- Al Instituto Politécnico Nacional, que por medio del Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada me acogieron en sus instalaciones.
- Al Consejo Nacional de Ciencia y Tecnología de México (CONACYT) por el apoyo financiero otorgado para la realización de este proyecto de investigación.
- A la Universidad Nacional de Ingeniería de Nicaragua (UNI) por apoyarme en mi formación academica y acogerme en su gran familia de académicos.

Resumen

En los últimos años el interés en modelos de ciudades 3D se ha incrementado. Existen dos enfoques predominantes en la reconstrucción 3D: el primero es usando un sistema estéreo y el segundo es la extracción de características de la nube de puntos proporcionadas por un sensor láser. La tecnología LIDAR (acrónimo del inglés Light Detection and Ranging) está emergiendo como la oferta más eficiente en modelos digitales de ambientes, ya que los datos que proporciona son una reconstrucción 3D. Sin embargo, los datos aunque reconstruyen el ambiente, carecen de modelos matemáticos que describan el comportamiento del ambiente De esta necesidad surge el objeto de este trabajo de investigación. El estudio reconstruido. consiste en modelar por medio de planos los datos 3D entregados por el LIDAR. El plano principal corresponde al suelo y los otros planos presentes en la nube corresponden a objetos estructurados, que pueden ser las fachadas de las edificaciones, aceras, muros entre otros. Los planos encontrados en la modelización del ambiente corresponden a un conjunto de puntos. El conjunto de puntos dentro del plano puede estar definido por varios clusters. La segmentación de los cluster se lleva a cabo usando MeanShift, y nos permite subdividir el plano en regiones del plano limitados por segmentos de rectas. Para probar los algoritmos propuestos, se desarrolló un prototipo de sistema de información mediante JOGL¹, en el cual se realizaron distintas pruebas de ejecución. En el trabajo se muestra la reconstrucción para dos adquisiciones particulares de una esquina de la plaza Zénea, ubicada en el centro histórico de Querétaro, México y el estacionamiento del CICATA², unidad Querétaro.

¹Java Open Graphics Library

²Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada

Abstract

The interest in 3D city models has increased in recent years. Two approaches are predominant in 3D reconstruction: the first is using a stereo system and the second is the feature extraction of the cloud of point provided by the LIDAR. LIDAR technology is emerging as more efficient supply of digital models for environments, since the data is already providing a 3D reconstruction. In this work, the environment has been characterized by the mathematical model. The focus of our research is to model structured environment by planes through the 3D data provided by the LIDAR. The main plane corresponds to the ground and other planes present in the cloud correspond to structured objects, which may be the facades of buildings, sidewalks, walls and more. With different planes and segments of points, we proceed to calculate the clusters belonging to each segment, defined by a plane. The plane equation computing in the modeling of the environment correspond to a set of points. The set of points in the plane can be defined by several clusters. The cluster segmentation is performed using MeanShift, and allows us to subdivide the plane into regions of the plane bounded by line segments. To test the proposed algorithms, we developed a prototype information system using JOGL3, in which various tests were run. The work shows the reconstruction for two acquisitions particular, corner of the plaza Zenea in the historical center of Querétaro and parking CICATA, Querétaro unit.

Índice general

Agra	adecimientos	I
Rest	ımen	[]
Abst	ract	v
Índi	ce general	v
Índi	ce de figuras	[]
Índi	ce de cuadros	X
Intro	oducción	1
1.1.	Antecedentes en el CICATA, Unidad Querétaro	2
1.2.	Justificación	3
1.3.	Objetivos	4
1.4.	Estructura del documento	5
Esta	do del Arte	6
2.1.	Reconstrucción 3D	7
2.2.	Reconstrucción 3D mediante tecnología LIDAR	9
2.3.	Segmentación de puntos LIDAR	1
	2.3.1. Transformada de Hough	4
	2.3.2. Técnica <i>MeanShift</i>	5
	Agra Resu Abst Índia Ín	Agradecimientos III Resumen III Abstract III Índice general VIII Índice de figuras VIII Índice de figuras VIII Índice de cuadros VIIII Introducción III 1.1. Antecedentes en el CICATA, Unidad Querétaro IIIIIII 1.2. Justificación IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII

3.	Met	odología Experimental	16		
	3.1.	Preprocesamiento de Nube de puntos	17		
	3.2.	Generación del modelo para Mapas Locales	20		
		3.2.1. Segmentación del Plano Principal	21		
		3.2.2. Segmentación de otros planos presentes			
		(Transformada de Hough)	28		
3.3. Refinamiento del Modelo		Refinamiento del Modelo	35		
		3.3.1. Segmentación de objetos	35		
	3.4.	Optimización del Modelo	37		
		3.4.1. Primitivas Geométricas de cada segmento	37		
	3.5.	Reconstrucción de Mapas Globales	38		
		3.5.1. Transformación entre nube de puntos	38		
	3.6.	Sistema de Información	40		
4.	Rest	iltados Experimentales	44		
	4.1.	Generación del modelo para Mapas Locales	45		
		4.1.1. Segmentación del plano principal	46		
		4.1.2. Segmentación de planos secundarios	50		
	4.2.	Refinamiento del Modelo	52		
	4.3.	Optimización del Modelo	55		
	4.4.	Reconstrucción de Mapas Globales	57		
Co	onclus	siones y Trabajo Futuro	59		
Bi	bliogr	rafia	61		
Ap	oéndic	ces	68		
A.	Cara	acterísticas del LIDAR HDL-64E	69		
B.	Adq	uisición de datos	72		
C.	Inte	C. Interfaz de usuario sistema de información			

D. Artículos realizados

Índice de figuras

3	.1.	Metodología para construcción de un mapa local.	17
3	.2.	Puntos del LIDAR en función de $r, \theta, \phi, \ldots, \ldots, \ldots, \ldots, \ldots$	18
3	.3.	Transformación de coordenadas paramétricas a coordenadas cartesianas	18
3	.4.	(a) Sistemas de coordenadas para elevación y rotación del HDL-64E.(b) Co-	
		rrección en la elevación (ϕ) para cada rayo láser. (c) Corrección al ángulo de	
		rotación (θ) para cada láser	19
3	.5.	Sistema de Reconstrucción, h es la altura existente entre el suelo y el LIDAR,	
		en este sistema el plano principal tiene una normal \vec{n} , el campo de vista en el	
		eje vertical del HDL-64E es de 26.8 grados, las líneas punteadas corresponden	
		a los rayos láser emisores del LIDAR, que van de 164 rayos láser, d es la	
		distancia radial que constituye el área no reconstruida por los rayos láser	22
3	.6.	Diferencia entre sistemas de coordenadas, Π_0 coordenadas en el mundo, Π_i	
		coordenadas en sistema LIDAR. Para cada θ el láser <i>i</i> con distancia radial ρ_i	
		forma un triangulo con la altura y el plano principal.	23
3	.7.	Definición de puntos <i>inliers</i> y <i>outliers</i> , d_i es la distancia entre un punto y el	
		plano analizado Π_i , α es el umbral permisible para la clasificación, \vec{n} es la	
		normal del plano.	25
3	.8.	Representación de las iteraciones para encontrar el plano principal óptimo, α	
		es el umbral para definir los <i>inliers</i> , Π es el plano encontrado, h es la altura	
		entre el plano principal y el sistema de referencia del LIDAR	27
3	.9.	Recta en el espacio paramétrico formada por dos puntos alineados	31
3	.10.	Representación gráfica de los puntos de la nube en el espacio paramétrico (ρ, θ) .	32

3.11.	Espacio discretizado para la transformada de Hough	33
3.12.	Sistema de acumulación para el espacio discretizado.	33
3.13.	Dos adquisiciones en diferentes posiciones modelizadas por planos	39
3.14.	Diagrama de paquetes y clases para el sistema de información	42
4.1.	Equipo móvil utilizado para reconstrucción dentro de la zona urbana de Que- rétaro.	45
4.2.	(a) Fotografía de satélite plaza Zenea, (b) fotografía de satélite estacionamien-	
	to del CICATA, (c) adquisición en plaza Zenea, (b) adquisición en estaciona-	
	miento CICATA.	46
4.3.	Nube asociada al plano principal: (a) plaza Zenea con $\alpha = 40$, (b) plaza Zenea	
	$con \alpha = 30, (c)$ estacionamiento $con \alpha = 40. \ldots \ldots \ldots \ldots \ldots$	48
4.4.	Nube <i>outliers</i> resultantes al quitar el plano principal de cada nube	49
4.5.	Planos encontrados en la nube de puntos <i>outliers</i>	51
4.6.	Representación gráfica de los clusters en cada plano, cada color representa el	
	cluster encontrado.	52
4.7.	Histogramas para dos nubes de cluster	54
4.8.	Representación gráfica de los clusters: (a) y (b) son las captura en el estacio-	
	namiento, (a) es la captura en la plaza Zenea, cada color representa un cluster.	55
4.9.	Mapas locales para una nube de puntos capturada	56
4.10.	Mapas globales e imagen del ambiente.	58
A.1.	Muestra de una nube de puntos generada por LIDAR	70
A.2.	Diseño del LIDAR HDL-64E.	70
A.3.	Ejemplo de la secuencia de captura de datos	71
C.1.	Pantalla inicial del sistema de información.	74
C.2.	(a) Mensaje de confirmación una vez que los datos fueron leído y transforma-	
	dos a coordenadas cartesianas, (b) pantalla que indica que todo esta listo para	
	realizar las siguientes operaciones.	75
C.3.	Interfaz principal para realizar la visualizaciones de las nubes y operaciones	76
C.4.	Área de visualización para las operaciones realizadas	77

Índice de cuadros

3.1.	Algoritmo de cálculo del plano principal.	24
3.2.	Algoritmo de Transformada de Hough	30
3.3.	Algoritmo de segmentación por Mean-Shift	36
B .1.	Estructura de un bloque de 100 bytes del LIDAR	72
B.2.	Resolución angular del HDL-64E.	73

Capítulo 1

Introducción

La experiencia del mundo no consiste en el número de cosas que se han visto, sino en el número de cosas sobre las que se ha reflexionado con fruto. Gottfried Wilhelm Leibniz

A lo largo de este capítulo se muestra los trabajos previos realizados en CICATA, Unidad Querétaro, en relación al tema de investigación, así como los objetivos para la investigacion y su justificación. El final del siglo pasado y los albores del presente, han sido testigos de un avance exponencial en las capacidades de cálculo y gestión de datos de las computadoras, este avance ha inducido un rápido desarrollo de todas las disciplinas vinculadas a la computación, y entre ellas, la computación gráfica ha sido una de las más beneficiadas.

El análisis de imágenes mediante una computadora, ha alcanzado avances significativos para hablar hasta de fotorrealismo, entendiendo a este como la técnica mediante la cual se pasa de un modelo matemático a la imagen visible en la pantalla, y viceversa. La representación de sistemas físicos mediante modelos virtuales basados en simulaciones matemáticas es cada vez más compleja y realista, esto gracias a las reconstrucciones que se realizan mediante grandes conjuntos de datos.

1.1. Antecedentes en el CICATA, Unidad Querétaro

La reconstrucción de objetos y ambientes dentro del área de análisis de imágenes en el Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, desde sus inicios ha estado relacionada con imágenes provenientes de cámaras digitales. En [1] se hace una reconstrucción de mapas con imágenes panorámicas indexadas por sus histogramas. Estas son las primeras interacciones con reconstrucciones de ambientes, el objetivo era la localización de un robot dentro de los mapas almacenados en una base de datos. El ambiente construido es controlado por la base de datos, la cual fue almacenada previamente en la fase de aprendizaje del robot. La reconstrucción del mapa se realiza con información en 2D.

Desde el año 2008 se ha utilizado el láser para la reconstrucción. Los rayos láser poseen la ventaja de generar datos 3D, con lo cual se tiene una mejor reconstrucción. La complejidad con los datos de los rayos láser está en modelar las diferentes nubes de puntos. Uno de los dispositivos láser disponibles en CICATA es el HDL-64E¹, el cual es un tipo de LIDAR que obtiene datos 3D presentes en ambientes urbanos o rurales, en los anexos A y B se muestran las características del sensor láser y su funcionamiento al adquirir datos,

¹High Definition Lidar- 64 Emisor

respectivamente. El uso del LIDAR surge por la necesidad de reconstrucción de ambientes urbanos, lo que trae consigo otras necesidades, como la de mejorar la precisión en sus datos. Para la obtención de datos más precisos, en [2] se trabajó con una metodología de calibración del dispositivo, lo cual ayuda a la reducción de incertidumbre de reconstrucción con respecto al sensor, la incertidumbre por el sensor es de $30 \ cm$ de fábrica, con parámetros proporcionados por la guía técnica del sensor se reduce a $5 \ cm$ y mediante la metodología planteada es de $1,56 \ cm$ con varianza de $2,22 \ cm$. La reducción de la incertidumbre en el sensor colabora al modelado de las nubes de puntos. Los resultados obtenidos son la mejora en la precisión de los parámetros devueltos por el sensor, al mismo tiempo que mejora la reconstrucción obtenida del ambiente.

En [3] se realiza una reconstrucción de mapas, se encuentra el plano del suelo y otros planos presentes en la nube de puntos. La técnica utilizada es conocida como RAN-SAC(Random Sample Consensus). El proceso consiste en encontrar el plano principal que corresponde al suelo. Los puntos son proyectados a este plano principal para generar una imagen bidimensional. La imagen generada nos permite encontrar la transformación de rotación sobre el eje z y la traslación en (x,y) entre dos adquisiciones diferentes. Una vez encontrada la transformación entre las adquisiciones se procede a fusionar los diferentes mapas. Esta técnica es correcta siempre y cuando el LIDAR se desplace en un plano.

Los modelos obtenidos hasta el momento poseen la limitante de no trabajar con los datos del sensor en sus tres dimensiones. Se necesita encontrar modelos matemáticos que describan las fachadas de las edificaciones tomando en cuenta la información 3D proporcionada por el LIDAR.

1.2. Justificación

En el grupo de investigación de Análisis de Imágenes en CICATA, Unidad Querétaro, se están desarrollando diferentes sistemas de reconstrucción. En el área de teledetección con fuentes de tipo láser se encuentra el uso del LIDAR HDL-64E. El dispositivo genera nubes

de puntos 3D carentes de modelos matemáticos. Aunque la nube de puntos por si misma es una reconstrucción 3D del entorno, se hace necesario obtener un modelo que describa las fachadas de las edificaciones y otras estructuras con formas topológicas, presentes en la nube. Este estudio tiene como propósito dar solución a esta necesidad. Con un modelo se pretende obtener una mejor reconstrucción y ajuste en la nube de puntos con un mapa local y precisión en la unión de dos o más adquisiciones del LIDAR. El modelo también ayudará a obtener una mejor vista de visualización inmersiva, precisión en la medición con escala computacional, menor uso de recursos computacionales y localización de objetos estructurados.

Una vez realizada la reconstrucción del modelo del entorno, tendrá un amplio impacto en la sociedad, tomando en cuenta que la digitalización de la información y el acceso a la misma hoy en día es mucho más evidente. Las aplicaciones se presentan en muchas áreas como el turismo, video juegos, historia, Internet y muchas más. La reconstrucción 3D de ciudades ayuda a establecer una base para la reorganización de las estructuras de una ciudad y es un requerimiento importante para la toma de decisiones.

La mayor ventaja del modelo 3D es su movilidad y su efecto para convencer a los usuarios de los futuros procesos en la toma de decisiones. La reconstrucción de los objetos estructurados presentes en la nube de puntos es una fase en la reconstrucción global de un ambiente donde existen objetos no estructurados con formas no topológicas como arbustos, vehículos, entre otros.

1.3. Objetivos

Objetivo General

Desarrollar algoritmos computacionales que permitan detectar primitivas geométricas locales con las cuales se construyan mapas globales 3D.

Objetivos Específicos

- 1. Modelizar por medio de planos los datos entregados por el LIDAR.
- 2. Construcción de mapas 3D locales a partir de la asociación de planos .
- 3. Construcción de mapas globales.
- 4. Desarrollar una aplicación para capturar y visualizar la información 3D.

1.4. Estructura del documento

Este documento, reflejo de la investigación realizada durante dos años se divide en cuatro capítulos.

El capítulo 1 describe la introducción, la cual está compuesta por los trabajos que anteceden a este estudio en el centro de investigación CICATA, unidad Querétaro, la justificación, los objetivos y la estructura de la tesis.

El capítulo 2 del documento son las bases teóricas que sustentan los métodos a utilizar, el capítulo 3 describe la metodología aplicada para la obtención de resultados en la reconstrucción de mapas locales y globales. El capitulo 4 son los resultados obtenidos por la aplicación de métodos sobre la nube de puntos.

El documento termina con las conclusiones, en donde se menciona la información relevante obtenida en el transcurso del estudio así como las recomendaciones para el trabajo futuro en la reconstrucción de ciudades mediante el HDL-64E.

Capítulo 2

Estado del Arte

Lo que sabemos es una gota de agua; lo que ignoramos es el océano. Isaac Newton

En este capítulo se muestra el estado de la ciencia acerca de la reconstrucción 3D, así como la reconstrucción mediante la tecnología LIDAR. Se citan las diferentes técnicas de segmentación existentes, resaltando las técnicas a utilizar a lo largo del documento. La reconstrucción de un modelo geométrico tridimensional permite la recuperación de información geométrica 3D de los objetos. En este sentido la reconstrucción implica determinar la relación geométrica y topológica de las partes de un objeto. Los primeros estudios sobre la reconstrucción geométrica tridimensional se pueden encontrar resumidos en los trabajos presentados en [4–7]. El libro de Sugihara [4] presenta una recopilación de la historia de la interpretación automática de dibujos técnicos. Najendra y Gujar [5] publicaron un estudio de varios métodos de reconstrucción de objetos tridimensionales a partir de sus vistas 2D. Wang y Grisnstein [6] describen una taxonomía de la reconstrucción de objetos 3D a partir de dibujos lineales de proyección bidimensional, estos mismos presentaron un algoritmo para extraer automáticamente los bloques primitivos de un poliedro. Yan [7] completó la sistematización de uno de los métodos más desarrollados y efectivos, el cual trata acerca de la reconstrucción de poliedros a partir de vistas múltiples, estos métodos asumen que el objeto 3D puede ser construido a partir de cierto conjunto de primitivas combinadas siguiendo una cierta jerarquía, la reconstrucción que realiza Yan es mediante proyecciones ortográficas 2D.

2.1. Reconstrucción 3D

Las personas están cada vez más conscientes de las aplicaciones de modelos de ciudades 3D, estas son usadas en áreas como la navegación de vehículos autónomos, el turismo, los servicios de navegación y comercialización, la arquitectura y la planeación urbana. El interés en la generación de modelos de ciudades 3D ha aumentado considerablemente en los últimos años, históricamente la reconstrucción 3D se ha realizado mediante fotogrametría, en [8] se menciona que las actuales estaciones de trabajo fotogramétricas digitales no son las más eficiente solución para crear modelos de ciudades 3D. Esto a que realizan mediciones desde una vista vertical aérea de objetos estructurados principalmente por segmentos lineales. El artículo menciona que la principal carga de trabajo en la generación de modelos 3D sigue siendo la construcción del modelo para grandes cantidades de datos.

En [9] se menciona que el desarrollo de la tecnología en términos de hardware y software en

los recientes años ha permitido la visualización de grandes cantidades de datos, minimizando esta limitante en la reconstrucción 3D. Brenner en [10] menciona que una estructura geométrica de una escena puede ser reconstruida usando un escaneo láser o por el uso de imágenes. En los últimos años, dos enfoques han tenido la investigación predominante en fotogrametría digital, el primero es usando un sistema estéreo y el segundo es la extracción de características de la nube de puntos proporcionadas por el LIDAR. El primer enfoque, es una técnica ya tradicional que se basa en la búsqueda de puntos comunes en dos o más imágenes 2D, que fueron adquiridas desde una perspectiva diferente. En [11] se menciona que los sistemas estéreos de reconstrucción trabajan bien cuando la superficie de la escena es plana y entre las diferentes capturas de imágenes existe una relación de puntos comunes en una secuencia de imágenes. En el segundo enfoque, los datos del LIDAR ya contienen información inherente en las tres dimensiones, pero requieren procesamiento para extraer superficies. Esta técnica trabaja bien cuando las nubes de datos poseen una alta densidad de puntos en el muestreo. Debido a los avances de hardware y software en procesamiento de datos, la visualización de datos provenientes de tecnología LIDAR han permitido que emerja como una de las áreas predominantes de investigación y se han encontrado aplicaciones en una amplia variedad de campos.

El uso de un sistema estéreo para la reconstrucción de ciudades es muy utilizado, se han presentado varios métodos mediante este sistema de reconstrucción, por ejemplo en [12] se presenta la reconstrucción de un modelo digital de superficie (DSM ¹ por sus siglas en inglés) usando el acoplamiento convencional de imágenes, mediante aproximaciones morfológicas para extraer regiones de interés y así realizar la reconstrucción. Otro método es presentado en [13], en el cual se utiliza imágenes aéreas para la extracción de segmentos lineales en 3D, para luego hacer uso de planos hipotéticos ortogonales al suelo para la reconstrucción. En [14] se usan imágenes aéreas utilizando agregación ² jerárquica, es un método basado en representar un modelo de reconstrucción, por medio de las esquinas de segmentos lineales extraídos de las imágenes, en los cuales se hace un acoplamiento entre dichas esquinas por agregación jerárquica. En los sistemas estéreos existen múltiples vistas

¹iniciales de Digital Surface Model o Modelo Digital de Superficies en español

²utilizando esta palabra como unión o adición de una parte de un todo

de los ambientes, la reconstrucción presentada en [15] es basada en un modelo poliédrico, en donde las vistas tienen un papel fundamental en la reconstrucción. El método utiliza seis imágenes para la reconstrucción, primero extrae segmentos de línea 2D en cada imagen las cuales puedan acoplarse, se computa una parte de un plano mediante la homografía entre la imagen y la región vecina a la línea.

Aunque la extracción de objetos mediante imágenes y rayos láser ha sido estudiado ampliamente, con nuevas aplicaciones en el área de sistemas de navegación, en [10] se muestran algunas de las dificultades de las reconstrucciones, entre ellas: la automatización, la eficiente extracción de características o modelo del ambiente, la incertidumbre de reconstrucción y la consistencia topológica.

2.2. Reconstrucción 3D mediante tecnología LIDAR

La tecnología LIDAR es una tecnología óptica de detección alejada que mide características de la luz dispersa para encontrar la distancia y/o la otra información de un blanco distante. El método para determinar la distancia a un objeto o a una superficie es utilizar pulsos de láser, la distancia a un objeto es determinada midiendo el retraso entre la transmisión de un pulso y la detección de la señal reflejada. La precisión de los datos dependen de la frecuencia del pulso y el diámetro del rayo láser, los datos proporcionados por está tecnología proveen una ideal visualización 3D, así es mencionado en [9], particularmente en un ambiente inmersivo, debido a su naturaleza 3D y la capacidad de integrarse con otros sensores de alta resolución.

La tecnología LIDAR se está moviendo rápidamente hacia la oferta más eficiente de modelos digitales de terreno (DTM ³ por sus siglas en inglés) para aplicaciones como la de generación de imágenes ortográficas digitales y cartografía para el entorno. Algunos de los usos de la tecnología LIDAR son la teledetección láser en la caracterización y gestión del medio fluvial, así lo muestra [16]. En este trabajo se describen otros usos del LIDAR en la digitalización urbana, como la estimación del perfil longitudinal de un río, análisis

³Digital Terrain Model

de la microtopografía fluvial, entre otros. Las áreas donde se utiliza está tecnología es la arqueología, geografía, geomorfología, sismología, detección alejada y física atmosférica. En la geología y sismología una combinación del LIDAR AIRBORNE y GPS se usan como una herramienta importante para detectar averías y medir levantamientos. Las dos tecnologías producen modelos muy exactos en el cálculo de mapas de elevación del terreno, el LIDAR como monitor aerotransportado tiene la capacidad de detectar cantidades sutiles de crecimiento del terreno o declinación. Los mapas topográficos también se pueden generar de un LIDAR, mediante la medición de distancias con láser para aplicaciones de mapas a gran escala, de igual manera se puede utilizar como fuente de datos para procesos de contorno y generación de curvas de nivel para ortofotos digitales. En [3] se mencionan una amplia lista de aplicaciones de está tecnología, entre las cuales figuran planificación y modelado urbano, turismo virtual, simulación, restauración, arquitectura, ingeniería, construcción entre otras.

Los sistemas LIDAR se categorizan en dos clases según [17], los generados por los escáner de láser aéreos (ALS⁴ por su siglas en inglés) y los escáner de láser terrestres (TLS⁵ por sus siglas en inglés). La clase de LIDAR más estudiada ha sido el mapeo de modelos urbanos de nube de puntos con LIDAR ALS, se han diseñado equipos exclusivamente para este tipo de móviles. En [18] se mencionan algunos de ellos, sin embargo, con este tipo de reconstrucción no se tienen detalles de las fachadas de las construcciones verticales, y presentan muchas oclusiones en sus capturas. En [19], se muestran las fuentes por las cuales los datos de estos LIDAR tienen incertidumbre, por ejemplo debido a submuestreos donde existan agujeros y a la poca densidad en la nube de puntos a causa de que el ángulo de incidencia es pequeño, la adquisición de los datos mediante sistemas LIDAR difiere de acuerdo a la distribución de los láseres. Por su forma de barrido el LIDAR puede ser de tipo lineal, de zig zag, elíptico y de fibra óptica. Los TLS se consideran como herramientas muy eficientes para la adquisición de grandes cantidades de datos debido a su velocidad, dándoles un considerable potencial para la recopilación de datos en el modelado 3D. La nube de puntos 3D final depende de muchos factores como el color, la rugosidad de las diferentes super-

⁴Airborne Laser Scanner

⁵Terrestrial Laser Scanner

ficies, la resolución del instrumento TLS y aparecen con un espesor finito de superficies planas.

Dentro de las aplicaciones de la tecnología LIDAR se encuentra la reconstrucción 3D de ciudades mediante el uso de primitivas geométricas. En [20] se especifican las características del LIDAR Velodyne HDL-64E, el cual tiene un campo de vista horizontal (FOV⁶ (por sus siglas en inglés) de 360 grados, en la parte vertical los 64 láseres están ordenados y cubren un rango desde -24.8° a $+2^{\circ}$ con aproximadamente $0.4^{\circ}de$ incremento. También se menciona que la utilidad de los datos del sistema está supeditada a una serie de factores como la tasa de reconstrucción, la densidad de reconstrucción, la cobertura azimut, la elevación, el alcance, ruido inherentes en los sensores, error de calibración y sensibilidad de las condiciones ambientales. El método de Brenner presentado en [10], utiliza DSM, partiendo de la descomposición de primitivas en 2D, luego realiza una estimación de parámetros 3D los cuales ocupa para construir primitivas 3D.

La alta calidad de los datos del LIDAR es reflejada en varios aspectos según [21]:

- Alta precisión. Un sistema típico de LIDAR proporciona datos con 15 cm de precisión vertical y menos de 50 cm de precisión horizontal.
- Alta consistencia de la exactitud.
- Alta consistencia en cobertura, es decir, los puntos se distribuyen uniformemente en el área.

2.3. Segmentación de puntos LIDAR

El proceso de segmentación se define generalmente como una agrupación de elementos en una región, los cuales comparten propiedades similares. Esta característica presente en objetos hechos por el hombre tiene su importancia en los procesos de extracción de superficies planas, los cuales son considerados como un primer paso en la creación del modelo 3D.

⁶Field Of View

En [22] se menciona que la segmentación es el paso más importante en el proceso de extracción de características, por tanto, este es el objetivo más importante del tratamiento de nubes de puntos no estructuradas con diferentes densidades. Gracias a la segmentación, el volumen de datos y la complejidad en términos jerárquicos disminuyen considerablemente, por tanto se optimizan los resultados experimentales respecto al tiempo de ejecución. Las características incluyen generalmente la posición de cada punto, estimar la región ó superficie descrita por una agrupación de puntos y los residuos de las mejores superficies de ajuste. Sin embargo, estas características podrían ser afectadas por los puntos de ruido y, en consecuencia, afectan directamente los resultados de segmentación.

En [22] se hace una clasificación de las técnicas comunes de segmentación y algunos de sus usos, estos métodos son:

- Región creciente (*region-growing*). Estos métodos asumen que hay una parte del conjunto de datos donde todos los puntos dentro de una distancia específica pertenecen a la misma superficie, se basa en criterios geométricos para la agrupación de regiones homogéneas que están presentes en los datos, puede ser visto como una combinación de dos pasos:
 - Identificación y
 - Crecimiento de la superficie identificada.

Algunos ejemplos de algoritmos que utilizan región creciente están mostrados en [23–25].

Agrupación de características (*Clustering Features*). Este método basado en la agrupación de características ofrece una manera general y flexible para identificar patrones de homogeneidad en los datos, está basado igual que el anterior en criterios geométricos para realizar la agrupación, sin limitarse a un patrón específico, puede ser visto como una combinación de dos procesos:

- Identificación de patrones en los datos basados en los atributos.
- Agrupamiento de datos en grupos (clusters).

Algunos ejemplos de aplicación de este modelo están en [26–30].

 Ajuste del modelo (*Fitting*). Este modelo está basado en el ajuste de las formas primitivas geométricas, que se pueden representar matemáticamente como superficies. Los puntos están conformados por las representaciones matemáticas que se podrían agrupar como un segmento, algunos ejemplos de aplicación para esta técnica están descritos en [31–33].

Con la técnica ajuste del modelo hay dos métodos ampliamente conocidos, estos son el Consenso de la muestra escogida al azar (RANSAC ⁷ por sus siglas en inglés) introducido por Fischler y Bolles en 1981 y la transformada de Hough introducido por Paul V.C. Hough en 1962, si bien estos dos algoritmos son muy utilizados para el procesamiento de nubes de puntos de forma automática, con el objetivo principal de la construcción de modelos de edificios 3D, una importante comparación de eficacia y efectividad en varios aspectos ha sido explicada en [34] usando ALS, en la cual la transformada de Hough muestra su fortaleza con respecto a tiempo de procesamiento y la sensibilidad a la nube característica.

La reconstrucción 3D generada por medio de la segmentación de primitivas geométricas definidas, presente en datos de nube de puntos, es una forma de modelizar una ciudad. En [35] se explica un modelo de obtención de primitivas geométricas en la nube de puntos, en el cual lo primero que se debe realizar para la reconstrucción del área en estudio, debe ser la clasificación de estas figuras geométricas. Las figuras encontradas pueden ser en formas regulares, irregulares y superficies para construir un mapa local, en el cual se tiene una vista 3D de la nube en términos de las figuras geométricas. En [36] se presenta una segmentación en 2D para la nube de puntos del LIDAR, en la cual se utiliza el algoritmo de RANSAC para hacer la extracción del plano del suelo, el cual es el plano predominante dentro de la nube de puntos, en otras palabras el los puntos que pertenecen a este plano son los que poseen mayor

⁷Random Sample Consensus

densidad en una captura de datos, una vez realizada la segmentación para el plano del suelo se procede con los demás objetos presentes en la nube, en [37] se muestra una reconstrucción 3D para la fachada de una edificación utilizando LIDAR terrestre.

2.3.1. Transformada de Hough

Paul V.C. Hough en 1962 introdujo un algoritmo conocido como la Transformada de Hough [38]. Es un algoritmo muy utilizado hoy en día en el área de imágenes, como reconocimiento de patrones para detectar primitivas geométricas, la versión más simple consiste en encontrar líneas. El algoritmo consiste en encontrar puntos alineados que puedan existir en una imagen, cada punto se evalúa de tal manera que sea posible averiguar si es parte de una línea, se le aplica una operación dentro de cierto rango, con lo que se averiguan las posibles líneas de las que puede ser parte el punto. Este proceso continúa para todos los puntos dentro de la imagen, al final se determina que líneas fueron las que más puntos posibles tuvieron y esas son las líneas en la imagen.

La transformada de Hough emplea una representación paramétrica de formas geométricas, por tanto, para aplicarla se hace necesario realizar una transformación entre el plano imagen y el espacio de parámetros utilizados por la transformada. El trabajar las rectas en su forma paramétrica elimina el problema de análisis de la recta cuando tiene una posición cercada a la vertical, lo cual posee una pendiente que tiende al infinito. Una vez realizada la transformación se discretiza el espacio de parámetros en una serie de celdas denominadas celdas de acumulación, luego se evalúa la ecuación de la recta en su forma normal. Cada punto de la imagen se debe de evaluar de tal manera, que si se cumple la ecuación de la recta se incrementa en uno el número de votos en la celda, un número de votos elevado indica que el punto pertenece a la recta.

La ventaja de este algoritmo es que evita singularidades, como por ejemplo rectas de pendiente infinita, como es demostrado en[39] en donde se realiza una comparación de varios algoritmos de segmentación, la transformada de Hough sobresale por parámetros decisivos como velocidad y precisión. Algunas de las aplicaciones presentadas para este algoritmo

están descritas en [39, 40], Hough además de localizar líneas rectas se utiliza para construir contornos de polígonos o curvas tales como círculos o elipses.

El algoritmo de la transformada de Hough ha sido extendido al mundo 3D por [41–43], los cuales lo utilizan para la extracción de fachadas planas distribuidas irregularmente en la nube de puntos. El principio de Hough es la representación ajustada de un conjunto de puntos, definidos inicialmente en el espacio Euclidiano y trasladados en otro espacio. Esta transformada permite detectar los puntos que componen una primitiva geométrica específica.

2.3.2. Técnica MeanShift

MeanShift es una técnica de segmentación de tipo no paramétrica. La técnica consiste en un procedimiento de localización de los máximos de una función de densidad descrita por datos discretos. Introducida como método para analizar cluster en una nube de puntos por [44], [45]. Con esta técnica el análisis de un espacio de características con densidades complejas multimodales, no requiere conocimiento apriori del número de clusters presentes en la nube de puntos. La técnica es aplicada en visión por computadora para filtrado y segmentación. En segmentación el uso de la técnica está ligado a la aplicación recursiva del método, para encontrar el punto estacionario más cercano de la función de densidad, lo cual ayuda a encontrar las modas de la misma. Se analizan solamente las modas más significativas dentro de un cluster de puntos, una vez localizada esta moda, se analizarán todos los puntos que se encuentren dentro de una vecindad cercana a cada moda significativa, generando así zonas con propiedades similares o clusters de nubes de puntos que comparten la característica de moda estadística.

La razón de utilizar esta técnica se debe a que la nube de puntos se puede considerar como la función de densidad de probabilidad del parámetro representado. En [46] se muestra una generalización de la técnica, así como el comportamiento de la técnica en el análisis de clusters con aplicación en la transformada de Hough. *Meanshift* en el espacio de parámetros trabaja con mejores resultados, de esto que una vez aplicado el algoritmo de Hough, se puede utilizar como técnica de optimización de las regiones homogéneas.

Capítulo 3

Metodología Experimental

Si consigo ver más lejos es porque he conseguido auparme a hombros de gigantes. Isaac Newton

En este capítulo se muestra la metodología a implementar para la reconstrucción de mapas locales y globales, el objeto del capítulo es explicar las técnicas para modelar las diferentes nubes de puntos capturadas por el LIDAR. El capítulo está definido en tres secciones, las dos primeras secciones se resumen en la figura 3.1, en su primera fase el reto es la reconstrucción de un mapa local, el cual está compuesto por las primitivas geométricas 3D presentes en una nube de puntos correspondiente a un giro del LIDAR. Esta reconstrucción es iniciada con la transformación del espacio de los puntos, seguida por la obtención del modelo presente en la nube, con el modelo definido se prosigue a utilizar técnicas de refinamiento del modelo para concluir con la optimización del mismo, de esta manera al juntar toda la información se construye un Mapa Local.



Figura 3.1: Metodología para construcción de un mapa local.

3.1. Preprocesamiento de Nube de puntos

El tipo de LIDAR con el que se realizará la reconstrucción es el Velodyne HDL-64E¹, las características técnicas de este dispositivo están explicadas en el anexo A. La salida del dispositivo son puntos 3D en coordenadas paramétricas, estos puntos como se muestra en la figura 3.2 están en función del número de rayos láser, campo de vista horizontal (θ), campo de vista vertical (ϕ), distancia radial entre objeto, el láser (r) y la intensidad del láser. Para el millón

¹High Definition LIDAR - 64 Emisor

de puntos promedio que el dispositivo genera por segundo, se necesita preparar cada punto para su manipulación, a esta etapa del proceso de reconstrucción de un mapa local se le conoce como preprocesamiento, mostrada en la figura 3.1. El primer proceso a realizar en la etapa de preprocesamiento es la adquisición de datos 3D, estos datos están en coordenadas paramétricas. El segundo proceso es la transformación de los puntos de coordenadas paramétricas a coordenadas cartesianas, la figura 3.3 resume la transformación de sistemas de coordenadas.



Figura 3.2: Puntos del LIDAR en función de r, θ, ϕ .



Figura 3.3: Transformación de coordenadas paramétricas a coordenadas cartesianas.

La transformación de coordenadas es retomada de la técnica mostrada en [2], la cual parte tomando en cuenta los diferentes parámetros de calibración para cada láser, estos factores son: una corrección en la distancia por el tiempo de vuelo, una corrección en cada láser para el campo de vista horizontal (θ) y vertical (ϕ), corrección en el parámetro r, el cual de ahora en adelante llamaremos parámetro d, que representa la distancia retornada para un láser que se ha reflejado en una superficie determinada, una compensación en el desplazamiento vertical para cada láser (v_{OSC}) y una compensación en el desplazamiento horizontal (h_{OSC}).







Figura 3.4: (a) Sistemas de coordenadas para elevación y rotación del HDL-64E.(b) Corrección en la elevación (ϕ) para cada rayo láser. (c) Corrección al ángulo de rotación (θ) para cada láser.

La figura 3.4 retomada de la referencia [2] muestra en 3.4a los sistemas de coordenadas para el ángulo de elevación u rotación del sensor. En la figura 3.4b se muestra la elevación, donde d es la distancia radial proporcionada por el escaneo del sensor, d_s es la distancia radial con orientación (θ_i , ϕ) tomando en cuenta la corrección vertical, d_x es la componente de d en el plano XZ, la cual está descrita por la ecuación 3.1. La figura 3.4c muestra la corrección del ángulo de rotación donde d_{xy} es la proyección del vector d en el plano XY.

$$d_x = [d_s * \cos(\phi_i)] - v_{OSC} \sin(\phi_i) \tag{3.1}$$

Las coordenadas de un punto del LIDAR en coordenadas cartesianas P(x, y, z), para una posición determinada por el ángulo de vista horizontal y tomando en cuenta los factores de compensación en el desplazamiento horizontal y vertical, están dadas por las ecuaciones 3.3,3.2 y 3.4.

$$x = d_{xy}\cos(\theta) + [h_{OSC} * sen(\theta)]$$
(3.2)

$$y = d_{xy}sen(\theta) + [h_{OSC} * cos(\theta)]$$
(3.3)

$$z = d_s sen(\phi) + [v_{OSC} * cos(\phi)]$$
(3.4)

Una vez que los puntos de la nube se encuentran en coordenadas cartesianas, están listos para su manipulación en el cálculo de las primitivas geométricas presentes en la nube. El análisis de la incertidumbre en la reconstrucción con respecto al ambiente reconstruido según se menciona en [2] tiene una media de 1,56 cm con una varianza de 2,20 cm.

3.2. Generación del modelo para Mapas Locales

La siguiente etapa en la metodología de construcción de mapas locales es la generación de un modelo. El propósito principal de esta etapa, es encontrar un modelo matemático que describa el comportamiento de las diferentes estructuras que se encuentran presentes en las nubes de puntos, estas estructuras pueden ser fachadas de edificios, muros de concreto, el suelo, entre

otros. La etapa de generación del modelo consta de dos procesos, la segmentación del plano principal y la segmentación de otros planos presentes en la nube.

Los puntos en la nube por si mismos poseen una forma topológica, las cuales permitirán visualizar las primitivas geométricas inmersas en las nubes, la principal primitiva geométrica tridimensional a trabajar es el plano. El primer plano a encontrar debe de ser el que describe el suelo, al cual lo conoceremos como plano principal, debido a que este tiene asociado la mayor densidad de puntos 3D de la nube.

3.2.1. Segmentación del Plano Principal

En primera instancia, se debe de tener en cuenta que existen dos sistemas de referencias para las operaciones. El primero es el sistema de referencia del mundo y el otro, es el sistema de referencias del LIDAR.

La figura 3.5 presenta los dos sistemas de referencia, los cuales no coinciden debido a la diferencia que existe entre la posición del vehículo automotor (sistema de referencia del mundo) y la inclinación del sistema de LIDAR ensamblado en el vehículo (sistema de referencia del LIDAR). La diferencia entre ambos sistemas de referencia se muestra en la figura 3.6, en donde Π_0 corresponde al plano principal en el sistema de referencia del mundo y Π_i es el plano principal en el sistema de referencia del LIDAR. El plano principal recibe este nombre por ser el plano con mayor cantidad de puntos en un ambiente abierto, esto debido a la inclinación con que los rayos láser reconstruyen la cual es de -24,8 grados por debajo de la horizontal del sistema de referencias del LIDAR y 2 grados por encima de la horizontal, esto es representado gráficamente en la figura 3.5, en la cual se muestran los 26,8 grados de campo de vista vertical, en la figura A.2 del apéndice A se muestra los grupos emisores y el funcionamiento de reconstrucción horizontal y vertical del HDL-64E.

La segmentación del plano principal se centra en encontrar los puntos que describen el plano ortogonal al eje vertical del sistema de referencias del LIDAR, así como los parámetros que describen a este plano. En las adquisiciones realizadas el plano principal es coincidente con el plano que describe el suelo del ambiente reconstruido. La adquisición de datos 3D que



Figura 3.5: Sistema de Reconstrucción, h es la altura existente entre el suelo y el LIDAR, en este sistema el plano principal tiene una normal \vec{n} , el campo de vista en el eje vertical del HDL-64E es de 26.8 grados, las líneas punteadas corresponden a los rayos láser emisores del LIDAR, que van de 1...64 rayos láser, d es la distancia radial que constituye el área no reconstruida por los rayos láser.

se realizó en la etapa anterior, almacenó los puntos tomando como sistema de referencia el LIDAR, sin embargo, el plano del suelo se encuentra en el sistema de referencias del mundo, el algoritmo presentado en el cuadro 3.1 describe la metodología utilizada para encontrar el plano principal.

El plano principal, en el sistema de referencias del mundo posee una ecuación cartesiana de la forma mostrada en la ecuación 3.5, donde [A, B, C], es un vector normal al plano y D es la distancia existente entre el suelo y el sistema de referencias del LIDAR, en la figura 3.5 se representa al coeficiente D como h, ya que coincide con la altura entre el suelo y el LIDAR, la distancia d presentada en la figura, es la distancia radial que constituye el área no vista por el sensor, esta distancia es de aproximadamente 200 cm en el sistema de reconstrucción utilizado, la cual es la distancia radial mínima proporcionada por el sensor en



Figura 3.6: Diferencia entre sistemas de coordenadas, Π_0 coordenadas en el mundo, Π_i coordenadas en sistema LIDAR. Para cada θ el láser *i* con distancia radial ρ_i forma un triangulo con la altura y el plano principal.

las adquisiciones realizadas.

$$Ax + By + Cz + D = 0 \tag{3.5}$$

El propósito es encontrar la ecuación del plano principal con respecto al sistema de referencia del LIDAR, para ello debe coincidir el plano principal con los puntos almacenados en un mismo sistema de referencia. La figura 3.6 muestra la situación descrita anteriormente así como los parámetros utilizados.

Mediante mediciones realizadas por cinta métrica se obtuvo el valor inicial de la altura entre el suelo y el LIDAR, asignándole este valor a h_0 , este es el valor inicializador ($h \leftarrow h_0$) para la búsqueda de la altura óptima mediante el método de optimización del plano principal mostrado en el cuadro 3.1. De manera similar, tomando como eje ortogonal al suelo al eje z,
Nube de puntos	
$h \leftarrow h_0$,	h_0 altura del LIDAR respecto al suelo
$\vec{n} \leftarrow < A, B, C >,$	\vec{n} normal del plano
$Inliers_k \leftarrow 0$	
Hacer:	
(1)	Calcular $dist(x_i, \Pi_i), i = 1, 2,n$
	$dist(x_i, \Pi_i) = \frac{ \vec{n}\vec{x_i}^T + h }{\vec{n}\vec{x_i}}$
(2)	Segmentar <i>inliers</i> y <i>outliers</i> ,
	Inliers _k $\leftarrow \{x_i \in \mathbb{R}^d dist(x_i, \Pi_0) < \alpha\}$
(3)	Calcular parámetros del plano mediante:
	$GX = 0$, donde $X^T = \langle A, B, C, h \rangle$
(4)	$\langle \vec{n}, h angle \leftarrow \{X G = USV^T\}$
	
Hasta:	$ Inliers_k < Inliers_{k-1} $
	$Dd(d; t(t_1, \Pi)) > t_1$
<i>Outliers</i> _k $\leftarrow \{x_i \in$	$n^{-} aist(x_i, n_0) > \alpha \}$
	Nube de puntos $h \leftarrow h_0,$ $\vec{n} \leftarrow < A, B, C >,$ <i>Inliers_k</i> $\leftarrow 0$ Hacer: (1) (2) (3) (4) Hasta: <i>Outliers</i> $\leftarrow [m, C]$

Cuadro 3.1: Algoritmo de cálculo del plano principal.

se tendría como vector normal inicial al plano principal, la normal unitaria para el plano XY, con $\vec{n} = \langle A, B, C \rangle$.

De lo anterior, los coeficientes inicializadores para el plano principal son $[\vec{n}, h_0]$, donde $\vec{n} = \langle A, B, C \rangle$, para optimizar el plano principal debemos de buscar una ecuación que modele la distribución de cierta cantidad de puntos, para ello, lo siguiente será definir la distancia entre cada punto de la nube con respecto al plano inicializador, la cual es calculada por la ecuación 3.6.

$$dist(x,\Pi) = \left| \frac{A x_i + B y_i + C z_i + D}{\sqrt{A^2 + B^2 + C^2}} \right|$$
(3.6)

Para asociar los puntos al plano, realizamos una comparación entre la distancia encontrada y un umbral con rango permisible, todos los puntos pertenecientes a la nube con distancia menor a este umbral se les conocerá como puntos *inliers* y los puntos con distancia mayor al umbral como *outliers*, esta relación se muestra en la figura 3.7. Los puntos *inliers* son considerados como parte del plano principal y cada punto asociado al plano debe de cumplir con la ecuación del mismo, que como se dijo anteriormente su forma cartesiana está dada por la ecuación 3.5.

La primera iteración se realiza con el plano inicializador, luego con los *inliers* encontrados se recálcula la ecuación del plano, modelando los puntos *inliers* a un plano, el modelo se obtiene al resolver un sistema de ecuaciones, mostrado en la ecuación 3.7, donde (x_i, y_i, z_i) son las coordenadas de los puntos *inliers*.



Figura 3.7: Definición de puntos *inliers* y *outliers*, d_i es la distancia entre un punto y el plano analizado Π_i , α es el umbral permisible para la clasificación, \vec{n} es la normal del plano.

$$f(x, y, z) = \begin{cases} Ax_1 + By_1 + Cz_1 + D = 0\\ Ax_2 + By_2 + Cz_2 + D = 0\\ \vdots\\ Ax_i + By_i + Cz_i + D = 0 \end{cases}$$
(3.7)

Escribiendo el sistema de ecuaciones en forma matricial tendríamos la ecuación 3.8, las matrices en su forma ampliada se muestran en la ecuación 3.9, donde las incógnitas son los parámetros del vector columna $X = \langle A, B, C, D \rangle$, este es el paso (3) dentro del ciclo del algoritmo presentado en el cuadro 3.1.

$$GX = 0 \tag{3.8}$$

$$\begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ \vdots & & & \\ x_i & y_i & z_i & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$
(3.9)

$$A = USV^T \tag{3.10}$$

Las matriz G de orden $m \times n$, con m > n puede encontrarse por la técnica de transformada SVD² (Descomposición de valores singulares) como se muestra en el paso (4) del ciclo presentado en el algoritmo del cuadro 3.1, mediante la factorización mostrada en la ecuación 3.10, donde U es una matriz unitaria de $m \times m$, S es una matriz diagonal de mxn con números no negativos sobre la diagonal conocidos como valores singulares de G y V^T es una matriz unitaria de nxn que denota la conjugada traspuesta de V, las columnas de U y V^T son un conjunto de vectores ortonormales, considerados como vectores básicos. La aplicación de SVD para resolver el sistema está asociada con el vector nulo resultante del sistema, así el menor valor singular no negativo de la matriz S y la matriz de eigenvectores por la derecha GG^T dan la solución al sistema. Con lo anterior se obtienen los valores de los parámetros de la normal al plano, que son los coeficientes de las variables en la ecuación cartesiana del

²Singular Value Decomposition

plano, $\vec{n} = \langle A, B, C \rangle$ y *D*. Los coeficientes de la normal encontrada son normalizados y evaluados como el nuevo plano inicializador. Con la nube de puntos y el nuevo plano inicializador se realiza una segunda iteración, de tal manera que se pueda hacer una comparación con los *inliers* anteriores, este proceso es iterativo hasta que los nuevos *inliers* sean menores que los anteriores, en otras palabras, que los puntos *inliers* ya no sean incrementales, cada iteración recálcula (\vec{n}, D). La figura 3.8 muestra el proceso de iteración para encontrar el plano principal.



Figura 3.8: Representación de las iteraciones para encontrar el plano principal óptimo, α es el umbral para definir los *inliers*, Π es el plano encontrado, h es la altura entre el plano principal y el sistema de referencia del LIDAR.

Una vez localizados los puntos *inliers* pertenecientes al plano principal y sus parámetros que lo definen como modelo, los puntos restantes (*outliers*) de la nube de puntos, describen las diferentes fachadas de las edificaciones existentes en el ambiente, así como otros elementos no estructurados como vehículos automotores, árboles, entre otros. Las fachadas de edificaciones serán descritas en las siguientes secciones como primitivas geométricas 3D secundarias o planos secundarios.

3.2.2. Segmentación de otros planos presentes (Transformada de Hough)

Siguiendo la metodología de reconstrucción de un mapa local mostrada en la figura 3.1, el siguiente proceso a realizar una vez encontrado el plano principal, es la segmentación de otros planos presentes en la nube de puntos restante. El propósito es encontrar los planos que corresponden a las fachadas de las paredes del entorno.

Las fachadas de las edificaciones son objetos estructurados y presentan una serie de relaciones geométricas con respecto al plano principal y entre ellos mismos, la principal relación es la ortogonalidad existente entre el plano del suelo y los planos de las fachadas. Al igual que en el plano principal, en los planos secundarios el objetivo es encontrar los puntos y el modelo cartesiano del plano que describa la fachada de las paredes de los edificios. Para realizar la segmentación de estos planos utilizaremos la técnica conocida como Transformada de Hough.

El plano principal encontrado se encuentra sobre el sistema de referencia del LIDAR, para garantizar la relación ortogonal existente entre los planos buscados y el suelo, se transformaran los puntos de la nube del sistema de referencia del mundo al sistema de referencias del LIDAR. La ecuación 3.11 muestra la operación de transformación, donde $H_{N \leftarrow N'}$ es la matriz de transformación y N' es la nueva nube de puntos.

$$N' = H_{N \to N'} N \tag{3.11}$$

A medida que los puntos se alejan del origen del sistema de referencia, la densidad de la nube de puntos disminuye significativamente, por tanto el nivel de dispersión de datos igual incrementa. Mediante los resultados descritos en [3] y experimentaciones realizadas con un perímetro de radio 5,000cm se obtiene una nube de puntos con poca incertidumbre en los datos.

La transformada de Hough es basada en la detección de fronteras. El uso de esta téc-

29

nica es debido a que su objetivo es encontrar puntos alineados que puedan existir en la nube de puntos, para luego encontrar un modelo geométrico que describa su comportamiento. El algoritmo mostrado en el cuadro 3.2 describe la aplicación de esta técnica para encontrar los planos secundarios.

Los planos a encontrar deben ser ortogonales al plano principal encontrado, por lo que los puntos *outliers* se deben de proyectar al sistema de coordenadas del mundo, donde fue localizado el plano principal. El nuevo sistema de referencia estará dado por los vectores mostrados en las ecuaciones 3.12, 3.13, 3.14, donde el vector director $\vec{v_3}$ corresponde a la normal del plano del suelo, $\vec{v_2}$ es un vector paralelo al plano principal, se obtiene haciendo cero la coordenada z y con valor de 1 la coordenada de x, al despejar y de la ecuación del plano se tiene su ordenada, toda esta operación muestra como inicialización en el algoritmo del cuadro 3.2.

Una vez obtenidos los vectores directores del nuevo espacio, se normalizan para crear una base y luego se proyectan los puntos.

$$\vec{v_3} = \vec{n_s} \tag{3.12}$$

$$\vec{v_2} = [1, -A/B, 0] \tag{3.13}$$

$$\vec{v_1} = \vec{v_3} \otimes \vec{v_2} \tag{3.14}$$

La transformada de Hough trabaja en un espacio paramétrico, en función de los parámetros θ y ρ . Para transformar los puntos del sistema de coordenadas del mundo al espacio paramétrico, se aplica la notación paramétrica o normal de la recta, mostrada en la ecuación 3.15, donde ρ es la longitud de una recta normal desde el origen hasta la línea encontrada y θ es el ángulo de ρ con respecto al eje x. Se trabajó con la transformada de Hough en 2D, debido a tres razones:

 La coordenada z no aporta información significativa para encontrar la alineación de los puntos, debido a que consideramos que las fachadas son perpendiculares al plano del suelo. Cuadro 3.2: Algoritmo de Transformada de Hough.

Entrada:	Nube de pun	tos (<i>outliers</i>)
Inicialización:	$nube_proy = base = [v_1v_2 \\ v1 \leftarrow (v_3 \times acum[\rho_i, \theta_i]]$ Hacer:	$= base \times outliers$ $[v_2, v_3], v_3 \in \Pi_0, v_2 \leftarrow [1, -A/B, 0], (A, B) \in \Pi_0$ $[v_2) \leftarrow 0$
	(1)	$\forall p \mid p_i(x_i, y_i, z_i) \in nube_proy, \ i = 1, 2, \rho_{max}$
	(2)	$p = x \cos v_i + g \operatorname{serie}_i$ posición a almacenar: $k = \frac{\rho}{\Delta \rho} + \frac{\rho_{max} - \rho_{min}}{2\Delta \rho}$ incrementar acumulador $a \operatorname{carm}[k, h]$
	(3)	encontrar máximo local en $acum$, max $\leftarrow acum(k, -h) \mid max > \gamma$
	(4)	$ \begin{array}{l} \max \leftarrow \operatorname{acan}(\kappa_{max}, n_{max}) \mid \max > \gamma \\ \operatorname{con} \rho_{max} = \rho_{min} + \left[\Delta\theta \times h_{max}\right] \\ \operatorname{del} \max \operatorname{maximo} \operatorname{local} \operatorname{se} \operatorname{calcula} \operatorname{el} \operatorname{plano} \operatorname{inicializador} \\ \operatorname{con} \operatorname{forms} \operatorname{polar} \end{array} $
	(5)	en forma polal. $\vec{n} = [A, B, C], h \leftarrow D$ \vec{n} y h son los parámetros necesarios para aplicar algoritmo mostrado en el cuadro 3.1
	(6)	con \vec{n}, h y <i>nube_proy</i> , aplicar algoritmo plano principal
	(7) Hasta:	obtener \vec{n}, D , <i>outliers</i> óptimos max < γ
	0.11	

	$Outliers_k \leftarrow \{x_i \in R^d dist(x_i, \Pi_0) > \alpha \}$
Salida:	$Planos(\vec{n_i}, h_i)$, puntos <i>outliers</i> , $i = 1, 2N$

 γ es un umbral, $\theta \in [0, \Delta\theta, 2\Delta\theta, ..\pi]$ y $\rho \in [\rho_{min}, \rho_{min} + 2\Delta\rho, \rho_{max}]$ acum es una matriz de contadores, ρ es la magnitud del vector normal desde el origen a la recta, θ es el ángulo de inclinación, \vec{n} es la normal del plano



Figura 3.9: Recta en el espacio paramétrico formada por dos puntos alineados.

- La complejidad de la transformada depende del número de coordenadas y coeficientes de la función.
- La utilización de recursos computaciones se reduce significativamente.

$$\rho = x \cos(\theta_i) - y \sin(\theta_i) \tag{3.15}$$

Cualquier línea recta en 2D es representada en un punto simple (ρ_i, θ_i) en el espacio paramétrico (ρ, θ) como se muestra en la figura 3.9, en la cual se puede observar el comportamiento de dos puntos alineados (recta l) en el espacio paramétrico, así como sus distancias con respecto al origen (ρ_i, ρ_j) dadas con orientación θ_i y θ_j .

Después de la transformación al espacio paramétrico, los puntos de la nube $(x_i, y_i, 0)$ con i = 1, 2, 3...N, son transformados en curvas sinusoidales en el plano (ρ, θ) , ver figura 3.10. Los

puntos de intersección de las curvas en el espacio parámetrico, corresponden a los parámetros (ρ_k, θ_k) de las posibles rectas que se encuentran en la nube de puntos.



Figura 3.10: Representación gráfica de los puntos de la nube en el espacio paramétrico (ρ, θ) .

Para una correcta aplicación de la transformada, es necesario discretizar el espacio de parámetros en una serie de celdas denominadas celdas de acumulación. En el algoritmo del cuadro 3.2 se denota este espacio como $acum[\rho_i, \theta_i]$, la discretización se realiza sobre los intervalos (ρ_{min}, ρ_{max}) y ($\theta_{min}, \theta_{max}$), mostrados en la figura 3.11.

Una vez realizada la transformación de los puntos de la nube de coordenadas cartesianas a coordenadas paramétricas, se ubican en el espacio discretizado, la posición de un punto en este espacio está dado por (k, h), donde k es calculada mediante la ecuación 3.16 y h es un valor entre $(0, \pi)$. El sistema de acumulación depende del nivel de tolerancia $(\Delta \rho)$ para cada celda acumuladora, la figura 3.12 muestra esta operación, el cual es el proceso (2) en el cuadro 3.2. Si las diferentes magnitudes ρ_i son cercanas, se incrementa en uno el número de votos



Figura 3.11: Espacio discretizado para la transformada de Hough.

de la celda.

$$k = 1 + \left(\frac{\rho_i}{\Delta\rho} + \frac{\rho_{max} - \rho_{min}}{2\Delta\rho}\right)$$
(3.16)



Figura 3.12: Sistema de acumulación para el espacio discretizado.

Con los puntos de la nube entrante ubicados en las celdas acumuladoras, se debe de encontrar la posición (ρ, θ) correspondiente al valor máximo de las celdas, el cual conoceremos como

máximo local, con coordenadas (k_{max}, h_{max}) . El máximo local es un pico, o un número elevado de votos para una celda en particular, la cual representa una recta.

En el pico encontrado se calcula la recta que describe a esa celda, la cual corresponde a un plano descrito por las ecuaciones 3.17, 3.18, 3.19, la distancia D esta representada por la posición h de la celda con máximo local, esta posición es una distancia respecto al área de trabajo, la cual posee al LIDAR en su centro de referencia.

$$A = \rho \cos(\theta) \tag{3.17}$$

$$B = \rho \, sen(\theta) \tag{3.18}$$

$$C = 0 \tag{3.19}$$

El tamaño del área de trabajo en el espacio paramétrico discreto está dado por la ecuación 3.20, donde d_{max} es la distancia radial permisible en la reconstrucción del sensor, dentro de esta distancia los datos presentan menor dispersión y ruido. Los puntos de la nube entrante están entre una distancia mínima de 200 cm y una máxima denotada por d_{max} . El parámetro D de la ecuación del plano está en un rango de $\pm [(\sqrt{2\rho^2}) d_{max}] - \Delta\rho$, su valor es tomado del espacio acumulador de acuerdo a la posición del máximo local.

$$d = \sqrt{8 \times d_{max}} \tag{3.20}$$

El plano encontrado y ajustado por los puntos que pertenecen al máximo local en el espacio acumulador está descrito por $\vec{n} = [A, B, C]$ y $D \leftarrow h_{max}$, este plano es el plano inicializador en un proceso iterativo de optimización. El proceso (6) del cuadro 3.2 se refiere a la optimización del plano encontrado mediante el algoritmo propuesto para la optimización del plano principal, con \vec{n} , h y la nube de puntos se aplica el algoritmo del cuadro 3.1 y se tendrá de salida \vec{n} y D optimizados.

Al encontrar el plano, los puntos en el pico corresponden a los *inliers* y los demás a los *outliers*, por tanto se elimina el pico para evaluar el siguiente en el espacio discretizado. El proceso se repite hasta encontrar todos los planos que describen las fachadas.

3.3. Refinamiento del Modelo

3.3.1. Segmentación de objetos

Los planos encontrados que describen las fachadas de los edificios tienen asociados una nube de puntos, las cuales solo están consideradas por la alineación de sus puntos. Las nubes de cada plano poseen regiones con diferentes densidades de puntos, los vacíos que separan estas regiones son espacios vacíos dentro de los planos encontrados, estos espacios son discontinuaciones en las fachadas de los edificios como puertas, espacios entre paredes, entre otros.

Utilizaremos la técnica de *MeanShift* también conocida como corrimiento de medias, para segmentar las diferentes regiones de puntos en las nubes de los planos secundarios, es una técnica de agrupamiento mediante características notables en regiones y no requiere del conocimiento *a priori* del número de regiones existentes. Utilizaremos la media como característica para analizar la convergencia entre los puntos.

El algoritmo a aplicar está descrito en el cuadro 3.3. La entrada es la nube de puntos que describe un plano (Z_0), una media inicializadora (\overline{M}_1) y un ancho de banda (γ) que limita el perímetro radial alrededor de la media.

Sea $\overline{M} = [\overline{x}, \overline{y}, \overline{z}]$ las medias de cada coordenadas y un punto de la nube representado por $M_i = [x_i, y_i, z_i]$, la distancia entre cada punto y la media está dada por la ecuación 3.21. La distancia encontrada es comparada con el ancho de banda o distancia en el dominio espacial (γ), de manera que se tiene un cluster de puntos cercanos a la media.

$$dist_{(\bar{M},M_i)} = \sqrt{(\bar{x} - x_i)^2 + (\bar{y} - y_i)^2 + (\bar{z} - z_i)^2}$$
(3.21)

Para obtener una región concentrada en base a la media se debe de analizar la convergencia de dicha media. La convergencia es analizada mediante la diferencia de dos medias $\bar{M}_1 - \bar{M}_2$ y comparada con una distancia, esto da el nombre de corrimiento de medias a la técnica. Con el cluster encontrado se calcula (\bar{M}_2), en el cuadro 3.3 δ es la distancia permitida entre

Cuadro 3.3: Algoritmo de segmentación por Mean-Shift.

Entrada: Nube de puntos $Z_0 = M_i | M_i = (x_i, y_i, z_i), i = 0, 1, 2..n$ **Inicialización:** $n \leftarrow \operatorname{tamaño}(Z)$ $M_1 \leftarrow random(M_i)$ $anchobanda \leftarrow qamma$ Hacer: Calcular $dist(\overline{M}_1, M_i), i = 1, 2, ...n$ if $(dist(\overline{M}_1, M_i) < \gamma)$ $cluster = M_i, M_i \in Z$ $\overline{M}_2 \leftarrow \overline{M}_1$ $\bar{M}_1 \leftarrow media(cluster)$ **if** $[(M_1 - M_2) < \delta]$ $RegionConcentrada_p \leftarrow cluster$ $Z \leftarrow (Z - RegionConcentrada)$ $n \leftarrow tamaño(Z)$ Mientras: n > 0

```
Salida: RegionesConcentradas<sub>p</sub> \in Z_0, p = 1, 2..m
```

las medias. Cuando esta distancia está dentro del dominio del rango (δ), se agrupan los puntos del cluster en una región concentrada $RegionConcentrada \subset Z_0 | M_i = [x_i, y_i, z_i] \in RegionConcentrada, i = 1, 2, ...w. La nueva nube de puntos está dada por la diferencia de la nube actual y los puntos de la región concentrada. El proceso es iterativo finalizando cuando todos los puntos de la nube que describe cada plano es analizado y clasificado dentro de una región concentrada.$

3.4. Optimización del Modelo

3.4.1. Primitivas Geométricas de cada segmento

Siguiendo la metodología mostrada en la figura 3.1 la siguiente etapa en la reconstrucción de un plano local es la optimización del modelo. Esta etapa consiste en el proceso de delimitación de los clusters encontrados para cada plano mediante primitivas geométricas.

Cada plano encontrado tiene asociado una ecuación de la forma AX + BY + CZ + D = 0, donde la normal está dada por $\vec{n} = [A, B, C]$ y D es la distancia ortogonal entre la orientación del plano y el origen del sistema de referencia. La ecuación asociada a cada nube describe un plano con dimensiones al infinito en cualquier dirección. La delimitación del plano que describe una nube de puntos, se realiza mediante la asociación de cada cluster encontrado en la etapa anterior con la ecuación del plano.

Hasta el momento la nube de puntos fue segmentada en nubes más pequeñas, las cuales son descritas por un plano y subdividida en clusters. Utilizamos la dispersión de los datos en cada cluster para delimitar el plano en dicho cluster. Realizamos una transformación de base de manera que la dispersión de los puntos se acentúe en uno de los ejes de coordenadas. La transformación está dada por la ecuación 3.22, donde N es la nube de puntos del cluster y N' es la nube transformada, $H_{N\to N'}$ es la matriz de transformación dada por los siguientes vectores $[\vec{n}_{suelo}, \vec{n}_{plano}, \vec{n}_{suelo} \otimes \vec{n}_{plano}], \vec{n}_{plano}$ es el vector normal del plano al cual se está manipulando en ese instante.

$$N' = H_{N \to N'} N \tag{3.22}$$

La nueva nube transformada muestra la mayor dispersión en el eje ortonormal al suelo. Mediante un histograma de frecuencias, graficando la dispersión existente en el eje vertical al suelo se obtiene un mínimo y un máximo, fijados por la densidad de la nube de puntos en sus extremos. Una primera aproximación geométrica del plano está dada por un conjunto de cuatro puntos (Q), obtenidos mediante combinaciones de los valores mínimo, máximo y la media del cluster de la nube.

Para optimizar los puntos (P') que describen geométricamente cada plano, se proyectarán de manera ortonormal al plano con $\vec{n} = [A, B, C]$, mediante la ecuación 3.23, tomada de [47] donde P es un punto perteneciente al plano y \vec{n}^T es la traspuesta de la normal unitaria del plano.

$$P' = Q + [(P - Q) \vec{n}^T] \vec{n}$$
(3.23)

Al graficar los puntos (P') que acotan a los clusters de los planos, se delimita el plano que describe esa nube de puntos y al visualizar todos planos delimitados se obtiene un modelo de mapa local de la nube de puntos leída en la adquisición del HDL-64E.

3.5. Reconstrucción de Mapas Globales

Cada giro con un $\theta = 360^{\circ}$ del LIDAR corresponde a una adquisición de nube de puntos. Los mapas locales son calculados en base a una adquisición, al hacer coincidir dos o más adquisiciones mediante los modelos matemáticos de sus mapas locales, se reconstruye un mapa global.

3.5.1. Transformación entre nube de puntos

La reconstrucción de mapas globales es posible con adquisiciones consecutivas. La figura 3.13 presenta dos adquisiciones donde la referencia del LIDAR se ha trasladado y rotado respecto a los planos encontrados en el ambiente. De acuerdo a [48], el mismo plano referenciado en dos sistemas de coordenadas que han sido rotados y trasladados está definido como :

$$\begin{bmatrix} \vec{n}'_i \\ D' \end{bmatrix} = H^{-T} \begin{bmatrix} \vec{n}_i \\ D \end{bmatrix}$$
(3.24)



Figura 3.13: Dos adquisiciones en diferentes posiciones modelizadas por planos.

donde H^{-T} es la matriz traspuesta de la inversa de H, $H = \begin{bmatrix} R & T \\ \mathbf{0}^T & 1 \end{bmatrix}$, R y T corresponde a una matriz de rotación y traslación de 3×3 y 3×1 respectivamente. **0** corresponde a un vector nulo de 3×1 . $\vec{n_i}$ y $\vec{n'_i}$ corresponden a la normal unitaria de dos planos con similar orientación y distancia respecto al sistema de coordenadas del LIDAR, este plano no es fijo. D corresponde a la distancia entre un plano y el sistema de coordenadas del LIDAR en una adquisición y D' es la misma distancia del mismo plano en otra adquisición del sensor.

La ecuación 3.24 se puede reescribir de la siguiente manera:

$$\begin{bmatrix} n'_i \\ D' \end{bmatrix} = \begin{bmatrix} R & \mathbf{0}^T \\ -(R*T)^T & 1 \end{bmatrix} \begin{bmatrix} n_i \\ D \end{bmatrix}$$
(3.25)

Las ecuaciones de los planos adquiridas en diferentes posiciones puede usarse para encontrar la traslación y rotación entre ambas adquisiciones. La ecuación 3.25 puede reescribirse de la siguiente forma :

$$F = \begin{bmatrix} n'_i \\ D' \end{bmatrix} - \begin{bmatrix} R & \mathbf{0}^T \\ -(R*T)^T & 1 \end{bmatrix} \begin{bmatrix} n_i \\ D \end{bmatrix} - \lambda(1 - \det(R))$$
(3.26)

donde F es la función objetivo que se va ha minimizar conteniendo la rotación, traslación y una restricción para que la matriz de rotación sea unitaria, propiedad de la matriz de rotación presentada en [49] como $\lambda(1 - det(R))$. La matriz de Rotacion (R) la podemos descomponer alrededor de cada uno de los ejes mediante ángulos de Euler, tal que:

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_x) & -\sin(\theta_x) \\ 0 & \sin(\theta_x) & \cos(\theta_x) \end{bmatrix} \begin{bmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) \\ 0 & 1 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) \end{bmatrix} \begin{bmatrix} \cos(\theta_z) & -\sin(\theta_z) & 0 \\ \sin(\theta_z) & \cos(\theta_z) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Para encontrar la solución de traslación y rotación mostrada en la ecuación 3.26 se necesitan dos planos en dos adquisiciones consecutivas, entre mas planos se utilicen se tendrá un mejor resultado, el método de optimización utilizado para minimizar la función objetivo es *Levenberg-Marquardt's*, y se obtiene por la solución de las ecuaciones:

$$\frac{\partial F}{\partial \Gamma} = 0 \quad ; \quad$$

donde $\Gamma = \{\theta_x, \theta_y, \theta_z, T_x, T_y, T_z\}$. Tomando en cuenta que $T = \begin{bmatrix} T_x & T_y & T_z \end{bmatrix}$

Los valores de inicialización de Γ para el método de optimización, los cuales son la primera aproximación de la solución de traslación y rotación, son calculados mediante la técnica de fusión de nubes de puntos mostrada en [3].

Una vez realizado el proceso para dos adquisiciones se repite para n adquisiciones que se deseen fusionar.

3.6. Sistema de Información

Con la finalidad de llevar a cabo los algoritmos planteados se desarrolló un sistema de información. El sistema fue montado en una computadora de escritorio sobre la plataforma de programación Java y JOGL³. Haciendo uso de la metodología orientada a objetos para desarrollo de software UML⁴ (Lenguaje unificado de datos) se realizó un diagrama de clases, en el que se muestran los paquetes, las clases principales y objetos de estas clases que se

³Java Open Graphics Library

⁴Unified Modeling Language

implementaron, el diagrama de clases realizado se muestra en la figura 3.14, los paquetes mostrados son *PLANO PRINCIPAL, BD, PLANOS SECUNDARIOS, INTERFAZ, LECTURA DE NUBE, JOGL, MATCHING MAPAS, MAPAS LOCALES y MEANSHIFT.*

El paquete *lectura de nube* realiza la función de leer la nube de puntos en un formato *vlcd*, de igual manera se leen los parámetros de calibración a tráves de la clase *leer_parametros*.

Una vez leída la nube de puntos se almacena la información en variables que pertenecen a la clase *Pts3D* del paquete *BD*. El paquete *plano principal* realiza el cálculo del plano principal. La clase *Fitplane* realiza el ajuste de los coeficientes del plano, las diferentes iteraciones es con base a la clase *encuentraplano* y sus métodos. Una vez calculado el plano óptimo se almacena en las variables de *BD*, de igual manera los *outliers* necesarios para realizar el cálculo de los planos secundarios.



Figura 3.14: Diagrama de paquetes y clases para el sistema de información.

3.6. SISTEMA DE INFORMACIÓN

El paquete *planos secundarios* está compuesto por tres clases las cuales son utilizadas para encontrar cada plano existente en la nube de puntos. La clase con las operaciones principales en este paquete es *hough*, en especial el método *acumulador* que es el encargado de definir cada plano dentro de un espacio discretizado. Todos los planos encontrados tienen asociados una nube de puntos y son almacenados en una variable tipo *Collection* del paquete *BD*.

El paquete *meanshift* es el encargado de encontrar los cluster que se encuentran en las nubes de puntos de los planos. Los clusters son usados en *mapas locales* para delimitar los objetos, las primitivas geométricas son representaciones gráficas de los planos, que describen objetos estructurados en el ambiente reconstruido.

Con dos mapas locales reconstruidos, la clase *matching mapas* realiza la operación de fusión de dos o más nubes de puntos mediante las clases *matching* y *operaciones*. Los resultados de las operaciones realizadas se muestran mediante un renderizado en JOGL, definido en la clase con igual nombre. Finalmente la clase *interfaz* es la aplicación que facilita la interacción entre la computadora y el usuario.

Se han mostrados las principales clases y paquetes ocupados para la implementación de la metodología, aunque existen otras clases que ayudan que la interfaz gráfica de usuario sea amigable, en el apéndice C se muestran las interfaces gráficas resultantes del sistema de información realizado en el lenguaje de programación JAVA, el cual fue escogido por sus características de flexibilidad en multiplataforma de programación, lenguaje orientado a objetos, facilidad de uso, portabilidad (encapsulamiento ejecutable), freeware(software libre), rapidez de ejecución, entre otros.

Capítulo 4

Resultados Experimentales

Si he hecho descubrimientos invaluables, ha sido más por tener paciencia que cualquier otro talento. Isaac Newton

En este capítulo se muestran los resultados obtenidos mediante representaciones gráficas. Las figuras mostradas son capturas del sistema de información desarrollado para la para la obtención de los resultados.

4.1. GENERACIÓN DEL MODELO PARA MAPAS LOCALES

A lo largo de este capítulo se mostrarán los resultados al aplicar la metodología presentada en el capítulo anterior. Se utilizarón datos capturados dentro del área urbana de la ciudad de Querétaro. Las zonas a reconstruir en el estudio son el entorno a la plaza Zenea, en el centro de la ciudad, y el estacionamiento del CICATA. El equipo utilizado para la captura se muestra en la figura 4.1, compuesto por un vehículo automotor que contiene el sistema LIDAR montado en la parte superior del mismo. Los datos son capturados por el equipo móvil de reconstrucción, transferidos y almacenados en un equipo de cómputo portátil mediante una conexión Ethernet entre el LIDAR y la computadora.



Figura 4.1: Equipo móvil utilizado para reconstrucción dentro de la zona urbana de Querétaro.

4.1. Generación del modelo para Mapas Locales

La generación del modelo es la explicación matemática de los diferentes clusters de puntos que se encuentran en la nube. Se utilizan dos técnicas para esta etapa, las que ayudan a determinar el plano principal y los planos secundarios, las técnicas son mostradas en la sección

3.2. La etapa de generación del modelo inicia una vez que los datos han sido leídos desde la computadora, el cual contiene las capturas de las nubes de puntos. Los datos se transforman a coordenadas cartesianas mediante la técnica mostrada en la sección 3.1. Una vez los datos en coordenadas cartesianas se procede a realizar la segmentación.

4.1.1. Segmentación del plano principal



Figura 4.2: (a) Fotografía de satélite plaza Zenea, (b) fotografía de satélite estacionamiento del CICATA, (c) adquisición en plaza Zenea, (b) adquisición en estacionamiento CICATA.

El equipo móvil de reconstrucción utilizado y la visión del LIDAR, hacen que el plano principal posea la mayor cantidad de puntos de la adquisición, siempre y cuando no exista un objeto con grandes dimensiones cerca del equipo móvil de reconstrucción. El plano principal corresponde al suelo del entorno a reconstruir. Las nubes de puntos a utilizar son mostradas en la figura 4.2, donde 4.2a y 4.2b son las fotografías del ambiente en plaza Zenea y estacionamiento del CICATA, respectivamente, de igual forma 4.2c y 4.2d son adquisiciones del sensor correspondientes a estos ambientes.

La metodología utilizada para la segmentación del plano principal es mostrada en el cuadro 3.1. El plano inicializador se obtiene considerando que el sistema de coordenadas del LIDAR coincide con el sistema de coordenadas del mundo. Con el escenario planteado, la normal al plano principal está dada por $\vec{n} = [0, 0, 1]$ y la distancia inicial entre los dos sistemas de coordenadas (D) es de 190 cm, obtenida mediante mediciones con cinta métrica, este plano inicializador se describe en la ecuación 4.1.

$$0X + 0Y + Z = 190 \tag{4.1}$$

Para aplicar el algoritmo propuesto en el cuadro 3.1 se necesita un umbral que delimite los puntos válidos para el plano principal, existen varios factores que introducen incertidumbre en los puntos que pertenecen al plano del suelo, este umbral debe de considerar todos estos factores de incertidumbre los cuales son : incertidumbre debido a la adquisición del sensor de $1,56 \ cm$ con una varianza de $2,2 \ cm$ lo cual genera una incertidumbre del sensor con un 99% de probabilidad que los datos se encuentren en un rango de de $12,01 \ cm$, incertidumbre debido a la altura de las acera $16 \ cm$ según la norma oficial mexicana NOM-233-SSA1-2003, incertidumbre debido a la precisión en la medición de la altura del sistema de referencia del LIDAR y el suelo la cual es de $190 \ cm$ con una tolerancia estimada de $6 \ cm$ y $6 \ cm$ debido a una incertidumbre introducida por peso del vehículo e irregularidades en la carretera, acumulados todos ellos suman $39,76 \ cm$ los cuales se redondearon a $40,01 \ cm$.

Se realizaron cálculos con umbrales por debajo a 30 presentaron poca información del plano principal, y los umbrales mayores a 50 tomaron información de objetos estructurados

como parte del plano principal.

Al utilizar un umbral de $\alpha = 40 \ cm$, aplicar el algoritmo del cuadro 3.1 a las nubes



Figura 4.3: Nube asociada al plano principal: (a) plaza Zenea con $\alpha = 40$, (b) plaza Zenea con $\alpha = 30$, (c) estacionamiento con $\alpha = 40$.

mostrada en la figura 4.2 y luego de las iteraciones para encontrar el plano óptimo, se obtienen nubes de puntos que pertenecen al plano principal, la figura 4.3a es el plano del suelo para la plaza Zenea y 4.3b es la misma representación con un umbral de 30 cm, se puede observar en esta última que hay regiones de puntos que no fueron clasificadas como parte del plano principal. La figura 4.3c representa el plano del suelo para el estacionamiento CICATA. Una vez encontrados los planos principales los puntos restantes son considerados *outliers*, la figura 4.4 muestra las diferentes nubes de *outliers* para las nubes utilizadas.



(b) estacionamiento del CICATA

Figura 4.4: Nube outliers resultantes al quitar el plano principal de cada nube.

4.1.2. Segmentación de planos secundarios

Los planos secundarios son extraídos de la nube de puntos restante conocidos como *outliers*, a medida que los datos se alejan del sistema de referencia la dispersión de los datos es mucho mayor, a una distancia radial de 5,000 cm, ya existe mucho ruido en los puntos 3D, por tanto, para un mejor ajuste de los planos secundarios delimitaremos el perímetro hasta un radio de 5,000 cm.

Con la nueva nube de puntos acotada a una distancia de $5,000 \ cm$ se aplica la técnica de transformada de Hough presentada en la sección 3.2.2, al implementar la técnica se obtuvieron los planos y las nubes de puntos que describen objetos estructurados presentes en dicha nube.

La figura 4.5 muestra los resultados de la segmentación de las nubes de puntos que describen los planos secundarios en las nubes analizadas. Primeramente se muestran dos de los planos encontrados en una sola adquisición y posteriormente se colocan todos los planos encontrados. Las primeras nubes de puntos corresponden a la plaza Zenea, las figuras 4.5a, 4.5b muestran dos nubes de puntos que describen dos planos en esta adquisición y 4.5c son todos los planos encontrados en dicha adquisición .

Las figuras 4.5d, 4.5e y 4.5f corresponden al estacionamiento del CICATA, con el mismo orden que las anteriores figuras, las dos primeras son nubes de puntos que describen planos en el ambiente y por último se muestran el conjunto de planos encontrados por el algoritmo.

Hasta el momento hemos encontrado un modelo matemático para explicar el comportamiento de las nubes de puntos. Cada plano encontrado no está aún delimitado con respecto a la nube de puntos y en cada nube de puntos existen objetos estructurados. Se realizará una segunda segmentación para mejorar los modelos obtenidos en base a la dispersión de los datos y los coeficientes de los planos encontrados.



(e) Un segundo plano, captura estacionamiento



Figura 4.5: Planos encontrados en la nube de puntos outliers.

4.2. Refinamiento del Modelo

Cada plano describe una nube de puntos con la orientación $\vec{n} = [A, B, C]$ y una distancia al origen *D*. El refinamiento del modelo realiza una segmentación en la nube perteneciente a cada plano, para encontrar objetos estructurados en la misma. Al aplicar la técnica de corrimiento de medias a cada plano presentada en la sección 3.3, el resultado son pequeños regiones concéntricas a una media.



Figura 4.6: Representación gráfica de los clusters en cada plano, cada color representa el cluster encontrado.

4.2. REFINAMIENTO DEL MODELO

Con un ancho de banda de 300 cm y tomando como media inicial un punto aleatoriamente de la nube, al analizar la convergencia de la media mediante el algoritmo mostrado en el cuadro 3.3 se obtuvieron los resultados mostrados en la figura 4.6. La figura 4.6c y 4.6d muestran los diferentes clusters para la captura del estacionamiento, en 4.6d además de los clusters se muestra el plano del suelo encontrado, con iguales características aplicando el mismo algoritmo a la captura en la plaza Zenea, se obtuvieron los resultados mostrados en las figuras 4.6a y 4.6b.











(d) eje y



Figura 4.7: Histogramas para dos nubes de cluster.

4.3. Optimización del Modelo



(a)



Figura 4.8: Representación gráfica de los clusters: (a) y (b) son las captura en el estacionamiento, (a) es la captura en la plaza Zenea, cada color representa un cluster.

La optimización del modelo es la representación gráfica mediante una primitiva geométrica 3D de cada cluster. Se realizó una transformación a los puntos de manera que su mayor dispersión estuviera sobre un eje. La figura 4.7 muestra los histogramas para dos clusters despúes de la transformación, en los ejes x, y no se presenta dispersión alguna de los datos, la mayor dispersión se muestra en el eje z, favoreciendo esto a la delimitación de los objetos. Claramente en la figura 4.7e se encuentran cuatro clusters, en cambio en la figura 4.7f se observan dos clusters.



(a) Nube de puntos original

(b) Mapa reconstruido



(c) Nube de puntos original

(d) Mapa reconstruido

Figura 4.9: Mapas locales para una nube de puntos capturada.

Con los clusters delimitados se procedió a representarlos gráficamente. No todos los clusters son información aprovechable, hay algunos que representan puntos muy dispersos, por lo que, se tomaran válidos los clusters de puntos con un tamaño mayor que doce. Los resultados son mostrados en la figura 4.8.

Con la ejecución de la fase de optimización del modelo se completa la metodología mostrada en la figura 3.1. Para mostrar los resultados de esta sección en las figuras 4.9a y 4.9c se muestran las nubes leídas inicialmente y las figuras 4.9b y 4.9d son mapas locales, con nubes de puntos modeladas mediante planos.

4.4. Reconstrucción de Mapas Globales

La metodología de reconstrucción de mapas globales ha sido explicada en la sección 3.5, la fusión entre planos de dos adquisiciones se hace al menos con dos planos en la misma orientación con respecto al ambiente en ambas adquisiciones. Se realizó el cálculo de la transformación inicial mediante la metodología utilizada en [3] para ambas nubes de puntos, con la transformación inicial entre las nubes de puntos se optimizó la fusión mediante la metodologia explicada en la sección 3.5.

En la figura 4.10a se muestra un mapa global, donde se han fusionado cuatro adquisiciones del LIDAR. Se logra ver que la densidad de datos a medida que vamos fusionando adquisiciones es mayor, resultando una nube con mayor veracidad en la información, de igual manera la fusión de adquisiciones ayuda a superar la incertidumbre creada por las oclusiones. La figura 4.10b muestra la imagen del ambiente reconstruido, tomado con la aplicación *google earth*.



(a) Mapa global reconstruido con cuatro adquisiciones



(b) Imagen de satélite correspondiente al ambiente reconstruido

Figura 4.10: Mapas globales e imagen del ambiente.

Conclusiones y Trabajo Futuro

Conclusiones

En este trabajo se desarrollarón algoritmos computacionales para detectar primitivas geométricas locales, concluyendo que:

- La densidad de puntos depende de la dispersión de los datos, la cual es reducida al utilizar los parámetros de calibración resultantes de aplicar la metodología mostrada en [2]. El ambiente a reconstruir debe de ser un ambiente que posea presencia de objetos estructurados fijos. La reconstrucción se ve afectada por objetos dinámicos presentes en el ambiente, los cuales generan oclusiones en la adquisición del sensor.
- Debido a la forma del sistema de reconstrucción móvil, el plano principal siempre será el plano con mayor densidad de puntos, el cual se encuentra ortogonal al eje vertical del sistema de referencias del LIDAR, en las condiciones del sistema de reconstrucción este plano es asociado al suelo del ambiente.
- El uso de recursos computacionales depende del tamaño de la nube de puntos, entre más grande es el tamaño, mayor uso de recursos computacionales y mayor la dilación de tiempo de operación.
- Los planos secundarios que describen las fachadas de los objetos estructurados fijos presentes en la nube de puntos, son calculados con la nube de puntos restante al quitar el plano principal, a la cual se le conoce como nube de *outliers*.
- La segmentación de los clusters mediante la técnica de corrimiento de medias fue muy exitosa, sin embargo, el tiempo usado para su ejecución ha sido muy prolongado en nubes de gran tamaño, por ejemplo segmentos de nubes de puntos con 17,000 puntos 3D. Un factor que influye en el tiempo de ejecución es el ancho de banda utilizado, que determina la convergencia del método, entre más pequeño sea el ancho de banda mayor detalle se tendrá de la nube de puntos, para ambientes exteriores con un ancho de banda de 300 cm se obtienen muy buenos resultados.
- Los mapas locales están compuestos por conjuntos de puntos que describen los planos, los clusters existentes en los conjuntos de puntos y la representación de los planos gráficamente acotados por los diferentes clusters.
- El trabajo de investigación está direccionado en su mayoría en el control del plano principal y los otros planos presentes en la nube de puntos, con lo que se tiene una reconstrucción de un mapa local correspondiente a una adquisición. Se utilizan los planos encontrados para la fusión de dos o más adquisiciones y tener un mapa global del ambiente.

Trabajo Futuro

El algoritmo presentado para encontrar mapas locales está implementado para ambientes estructurados, como trabajo futuro se considera trabajar en ambientes dinámicos, que se generan por lo general en ambientes rurales. En ese tipo de ambiente lo importante a reconstruir son objetos no estructurados presentes en las nubes de puntos adquiridas.

Se recomienda realizar una optimización en la reconstrucción de mapas globales, para mejorar la fusión de nubes de puntos y la visualización de los mismos con respecto al uso de recursos computacionales.

Otra manera de realizar la fusión de mapas locales es mediante la utilización de GPS y giroscopios, lo cual se recomienda para reducir el costo computacional, obtener menos error

en la fusión de adquisiciones y menor tiempo de ejecución. Por otro lado, el uso de SLAM (acrónimo de Localización y mapeo simultanea) permitiría reconstruir el ambiente con mayor precisión y mejorar la posición y orientación dada por el GPS y los giroscopios.

Bibliografía

- J. J. G. Barbosa and S. Lacroix, "Rover localization in natural environments by indexing panoramic images," *Pattern Recognition, International Conference on*, no. 4, pp. 1365– 1370, 2002.
- [2] G. A. Jimenez, J. J. G. Barbosa, J. B. H. Ramos, and R. G. Barbosa, "Lidar velonyne calibration for enviroment urban," *Sensors ISSN 1424-8220*, p. En revision.
- [3] J. de Jesús Rico Jiménez, "Construcción de mapas 3d a partir de la extracción de primitivas geométricas obtenidas de datos de un lidar," Master's thesis, Instituto Politécnico Nacional, CICATA unidad Querétaro, 2009.
- [4] K. Sugihara, "Machine interpretation of line drawings," *MIT Press*, 1986.
- [5] I. V. Nagendra and U. G. Gujar, "3-d objects from 2-d orthographic views a survey," *Computers and Graphics*, vol. 12, no. 1, pp. 111–114, 1988.
- [6] W. W. D. Sc and G. G. Grinstein, "A survey of 3d solid reconstruction from 2d projection line drawings," *Computer Graphics Forum*, vol. 12, no. 2, pp. 137–158, 1993.
- [7] Q.-W. Yan, C. P. Chen, and Z. Tang, "Efficient algoritm for the reconstruction of 3d objects from orthographic projections," *Computed Aided Design*, vol. 26, no. 9, pp. 147– 161, 1991.
- [8] Y. Zhang, Z. Zhang, J. Zhang, and J. Wu, "3d bulding modelling with didital map, lidar data and video image sequences," *The Photogrammetric Record*, vol. 20, no. 111, pp. 285–302, 2005.

- [9] S. Ghosh and B. Lohani, "Development of a system for 3d visualization of lidar data." Map World Forum, Hyderabad, India, 2007.
- [10] C. Brenner, "Building reconstruction from laser scanning and images," *International Journal of Applied Earth Observation and Geoinformation*, vol. 6, no. 3-4, pp. 187–198, 2005.
- [11] H. Prudhvi Gurram, J. K. Rhodv, S. Lach, and E. Saber, "3d scene reconstruction through a fusion of passive video and lidar imagery," 36 th Applied Imagery Pattern Recognition Workshop, IEEE Computer Society, pp. 133–138, 2007.
- [12] N. Haala, *Gebäuderekonstruktion durch Kombination von Bild- und Hö hendaten*. PhD thesis, Universität Stuttgart, Institut für Photogrammetrie, 1996.
- [13] O. Henricsson and E. Baltsavias, "3d building reconstruction with aruba: A qualitative and quantitative evaluation," in AUTOMATIC EXTRACTION OF MAN-MADE OBJECTS FROM AERIAL AND SPACE IMAGES (II), pp. 65–76, Verlag, 1997.
- [14] A. Fischer, T. H. Kolbe, F. Lang, A. B. Cremers, W. Förstner, L. Plümer, and V. Steinhage, "Extracting buildings from aerial images using hierarchical aggregation in 2d and 3d," *Computer Vision and Image Understanding*, vol. 72, no. 2, pp. 195–203, 1998.
- [15] C. Baillard, C. Schmid, A. Zisserman, A. Fitzgibbon, and O. O. England, "Automatic line matching and 3d reconstruction of buildings from multiple views," in *In ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery, IAPRS Vol.32, Part* 3-2W5, pp. 69–80, 1999.
- [16] F. Magdaleno and R. M. Romero, "Aplicaciones de la teledeteccion laser (lidar) en la caracterización y gestión del medio fluvial," *Ingeniería civil*, vol. 142, pp. 29–44, 2006.
- [17] O. Kreylos, G. W. Bawden, and L. H. Kellogg, "Immersive visualization and analysis of lidar data," *Springer-Verlang Berlin Heidelberg*, vol. I, pp. 846–855, 2008.
- [18] A. T. D. Office, "A white paper on lidar mapping," Ambercore, 2008.

- [19] T. C. Yapo, C. V. Stewart, and R. J. Radke, "A probabilistic representation of lidar range data for efficient 3d object detection," *DARPA Computer Sciencie Study Group, IEEE*, 2008.
- [20] R. Halterman and M. Bruch, "Velodyne hdl-64e lidar for unmanned surface vehicle obstacle detection," SPIE: Unmanned Systems Technology, vol. XII, pp. 5–9, 2010.
- [21] Z. Wang and T. Schenk, "Building extraction and reconstruction from lidar data," *International Archives of Photogrammetry and Remote Sensing*, vol. XXXIII, no. B3, 2000.
- [22] Z. D. Y. Z. Tarek M. Awwad, Qing Zhu, "An improved segmentation approach for planar surfaces from unstructured 3d point clouds," *The Photogrametric Record*, vol. 25, no. 129, pp. 5–23, 2010.
- [23] A. D. Hofmann, H.-G. Maas, and A. Streilein, "Knowlege-based building detection based on laser scanner data and topographic map information," *International Archives of the Photogrammetry, Remote Sensing*, vol. 34, no. 3/A, p. 169–174, 2002.
- [24] C. Dold and C. Brenner, "Automatic matching of terrestrial scan data as a basis for the generation of detailed 3d city models," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 35, no. (B3), p. 1091–1096, 2004.
- [25] S. Pu and G. Vosselman, "Automatic extraction of building features from terrestrial laser scanning," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, no. (5), p. 5, 2006.
- [26] S. Filin, "Surface clustering from airborne laser scanning data," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 34, no. (3/A), pp. 119–124, 2002.
- [27] A. D. Hofmann, "Analysis of tin-structure parameter spaces in airborne laser scanner data for 3-d building model generation," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 35, no. B3, pp. 302–307, 2004.

- [28] J. L. Lerma and J. M. Biosca, "Segmentation and filtering of laser scanner data for cultural heritage," *CIPA 2005 XX International Symposium*, p. 6, 2005.
- [29] S. Filin and N. Pfeifer, "Segmentation of airborne laser scanning data using a slope adaptive neighborhood," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 60, no. 2, pp. 71–80, 2006.
- [30] J. M. Biosca and J. L. Lerma, "Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 63, no. 1, pp. 84–98, 2008.
- [31] J. Bauer, K. Karner, K. Schindler, A. Klaus, and C. Zach, "Segmentation of building models from dense 3d points-clouds," *Proceedings of 27th Workshop of the Austrian Association for Pattern Recognition*, pp. 253–258, 2003.
- [32] F. Bretar and M. Roux, "Extraction of 3d planar primitives from raw airborne laser data: a normal driven ransac approach," pp. 452–455, 2005.
- [33] H. Boulaassal, T. Landes, P. Grussenmeyer, and F. Tarsha-Kurdi, "Automatic segmenttion of building facades using terrestrial laser data," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, no. 3/W52, pp. 65–70, 2007.
- [34] F. T. Kurdi, T. Landes, and P. Grussenmeyer, "Hough transform and extended ransac algorithms for automatic detection of 3d building roof planes from lidar data," *ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007*, vol. XXXVI, no. 3, pp. 407– 412, 2007.
- [35] S. You, J. Hu, U. Neumann, and P. Fox, "Urban site modeling from lidar," *ICCSA*, Springer Verlag Berlin Heidelberg, pp. 579–588, 2003.
- [36] G. Forlani, C. Nardinocchi, M. Scaioni, and P. Zingaretti, "Complete classification of raw lidar data and 3d reconstruction of buildings," *Pattern Analysis Applications, Springer Verlag London Limited*, vol. 8, pp. 357–374, 2006.

- [37] S. Becker and N. Haala, "Refinement of building fassades by integrated processing of lidar and image data," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 36, pp. 19–21, 2007.
- [38] P. V. Hough, "Method and means for recognizing complex patters," U.S. Patent, no. 3069654, 1962.
- [39] V. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart, "A comparison of line extraction algorithms using 2d laser rangefinder for indoor mobile robotics," *IEEE/RSJ Proceedings Int. Conference on intelligent robots and systems*, 2005.
- [40] E. R. Davis, "Application of the generalised hough transform to corner detection," *IEEE proceedings*, vol. 135, no. 1, pp. 49–54, 1988.
- [41] J. Overby, L. Bodum, E. Kjems, and P. M. Ilsoe, "Automatic 3d building reconstruction from airbone laser scanning and cadastral data using hough transform," *International Archives of Photogrammetry and Re mote Sensing*, vol. XXXV, no. part B3, 2004.
- [42] G. Vosselman and S. Dijkman, "3d building model reconstruction from point clouds and ground p plans," *International Archives of Photogrammetry and Re mote Sensing*, vol. XXXIV-3/w4, pp. 37–43, 2001.
- [43] K. Khoselham, "Extending generalized hough transform to detect 3d objects in laser range data," *IAPRS*, vol. XXXVI, no. part 3/W52, pp. 206–210, 2007.
- [44] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEE Trans. Information Theory*, vol. 21, no. 8, pp. 790–799, 1995.
- [45] Y. Cheng and K. S. Fu, "Conceptual clustering in knowledge organization," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 7, pp. 592–598, 1985.
- [46] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE)*, vol. 17, no. 8, pp. 790–799, 1995.

- [47] L. M. González and E. S. Aláez, *Álgebra Lineal con métodos elementales*. ITES-Paraninfo, 2006.
- [48] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. New York, NY, USA: Cambridge University Press, 2 ed., 2003.
- [49] J. J. M. Benjamín, Mecánica Newtoniana. Barcelona Digital, SL, 2000.

APÉNDICES

Apéndice A

Características del LIDAR HDL-64E

Funcionamiento

El LIDAR HDL-64E es un sensor láser de segunda generación y su principio de operación es bastante simple, obtiene las distancias de los objetos situados a su alrededor utilizando el principio del medidor láser de distancia. Un emisor láser hace incidir un rayo el cual rebota en la superficie de un objeto y un receptor lo captura calculando la distancia de acuerdo al tiempo que tardó en regresar, si tarda más tiempo del esperado o no regresa entonces no se toma en cuenta. Con los ángulos en que se proyectó el láser y la distancia, se construye un vector, y así se obtiene un punto en el espacio R^3 , y acumulando cada uno de estos se forma la **nube de puntos** (ver figura A.1). Cuando esta colección de puntos es *renderizada* se crea una imagen tridimensional del escenario. Entre más densa sea la nube de puntos mayor resolución tendrá la imagen.

El HDL-64E tiene montado un arreglo de 64 rayos láser (ver figura A.2), divididos en un bloque superior y otro inferior de 32 cada uno, y toda la unidad gira en el eje vertical ofreciendo un campo de visión horizontal de 360 grados, al girar en un ángulo θ , y uno vertical de 26.8 grados, que va desde 2° hasta -24.8° por el acomodo de los láseres en un ángulo ϕ . Cada láser es disparado miles de veces por segundo, proveyendo más datos de puntos y una nube de puntos mucho más robusta que los diseños convencionales, con un alcance



Figura A.1: Muestra de una nube de puntos generada por LIDAR.

de hasta 120 metros y una precisión de fábrica de 5 cm. El láser es de clase 1M, seguro para la vista, con una longitud de onda de 905 nanómetros y con pulsos de 10 nanosegundos.



Figura A.2: Diseño del LIDAR HDL-64E.

Este dispositivo opera con 12 Volts y de 4 a 6 Amperes, cualquier voltaje mayor a 16 puede dañar la unidad. Cuenta con interfaces Ethernet y serial, con conectores RJ-45 y RS-232 DB9

respectivamente, y el cable de alimentación de poder. Una vez que el LIDAR alcanza cierta velocidad de rotación comienza una transmisión simple de paquetes UDP que son capturados por la computadora por medio de su interfaz Ethernet, es necesario un software que funcione como escucha de manera que cuando estos se reciban inmediatamente sean convertidos en la información requerida y se pueda continuar recibiendo los paquetes restantes. La cabeza rota horizontalmente en un ángulo θ y dispara los láseres por grupos ya sea superior o inferior. Para obtener una mayor resolución a grandes distancias se emiten y capturan tres veces el grupo de láser superior por solo uno del inferior, este proceso es mostrado en la A.3.



Figura A.3: Ejemplo de la secuencia de captura de datos.

Apéndice B

Adquisición de datos

El LIDAR HDL-64E entrega como salida paquetes UDP Ethernet a 100 Mbps. Los paquetes están divididos en 12 bloques de 100 bytes (ver cuadro B.1), seguidos de seis bytes al final de cada paquete con información de estado que contienen un contador de giro y versión del firmware o temperatura. Cada paquete puede ser tanto para el conjunto de láser superior o el inferior, y cada conjunto contiene información de 32 láseres.

Encabezado	Ángulo	Distancia	Distancia	 Distancia	Distancia
		Láser 0	Láser 1	Láser 30	Láser 31
2 bytes	2 bytes	3 bytes	3 bytes	 3 bytes	3 bytes

Cuadro B.1: Estructura de un bloque de 100 bytes del paquete UDP entregado por el LIDAR.

Encabezado. Contiene 2 bytes que indican cuando un paquete pertenece al bloque superior o inferior. El bloque superior se representa con 0xEEFF y el inferior con 0xDDFF.

Angulo. Son 2 bytes que representan un entero entre 0 y 35999, es necesario dividir entre 100 para obtener al ángulo real.

Distancia. Conjuntos de 3 bytes con dos de información de la distancia, en incrementos de .2 centímetros, y uno con información de intensidad, de 0 a 255. Y si la distancia es cero o menor de tres pies es un dato no valido.

Estado. Muestra la temperatura interna de la unidad o el número del firmware. La temperatura

se muestra con 2 bytes seguidos por la cadena "DegC" en ASCII. El firmware con el formato de cadena "Vn.n" en ASCII.

Estos datos son leídos con un código en c++ que por medio de sockets o librerías como WinPcap realiza la conexión y transferencia de los paquetes desde el LIDAR a la computadora. También se puede hacer una modificación en la velocidad de rotación, esto genera que aumente o disminuya la resolución de la imagen ya que la tasa de muestreo de datos no cambia. La velocidad rotación default es de 600 rpm (10 Hz), y puede girar desde 300 rpm (5 Hz) hasta 900 rpm (15 hz). Estas modificaciones se realizan con una conexión serial RS-232 COM [1].

En el cuadro B.2 se muestran los diferentes valores de resolución angular que puede tomar el HDL-64E según su variación de velocidad.

Bloque inferior						
RPM	RPS	Puntos por	Puntos por	Resolución		
		revolución	revolución	angular		
			por láser	(grados)		
300	5	50,000	1562.5	0.2304		
600	10	25,000	781.25	0.4608		
900	15	16,667	521	0.6912		
	Bloque superior					
RPM	RPS	Puntos por	Puntos por	Resolución	Resolución	
		revolución	revolución	angular	angular	
			por láser	(grados)	(Post-	
					Bloque	
					inferior)	
300	5	100,000	6250	0.2304	0.1152	
600	10	200,000	3125	0.4608	0.2304	
900	15	66,667	2083	0.6912	0.3456	

Cuadro B.2: Resolución angular del HDL-64E.

Apéndice C

Interfaz de usuario sistema de información



Figura C.1: Pantalla inicial del sistema de información.

La interfaz de usuario inicial del sistema informático es donde se selecciona el archivo que se va ha abrir, estos archivos deben ser en formato *vlcd*, el cual es el formato que genera el sensor una vez que los datos fueron transferidos a un dispositivo de almacenamiento primario,

la figura C.1 presenta la pantalla inicial del sistema.

Al seleccionar un archivo que se desea leer, se realiza la transformación de coordenadas y si todo es de manera correcta, el sistema envía un mensaje de confirmación al usuario, el mensaje de confirmación es mostrado en la figura C.2a.



Figura C.2: (a) Mensaje de confirmación una vez que los datos fueron leído y transformados a coordenadas cartesianas, (b) pantalla que indica que todo esta listo para realizar las siguientes operaciones.

Con la información lista para realizar operaciones, damos un clic en el comando *ver gráficas* presentado en la figura C.2b, con lo que se abrirá la pantalla mostrada en C.3, en la cual se realizan la mayoría de operaciones como: cálculo del plano principal, cálculo de la transformada de Hough, cálculo de *meanshift*, entre otros.



Al realizar cualquier operación se cuenta con un menú Ver con el que se tiene acceso

Figura C.3: Interfaz principal para realizar la visualizaciones de las nubes y operaciones.

a la visualización de las diferentes nubes de puntos, como la nube leída, la nube del plano principal, las diferentes nubes de los planos secundarios, las nubes con las primitivas geométricas entre otras, se muestra un ejemplo en la figura C.4, en la cual se logran observar los diferentes planos encontrados luego de aplicar la técnica de Hough.



Figura C.4: Área de visualización para las operaciones realizadas.

Apéndice D

Artículos realizados





Browse > Conferences > Electrical Communications and

3D city models: Mapping approach using LIDAR technology

Hernandez-Garcia, Denis-Eduardo Gonzalez-Barbosa, Jose-Joel Hurtado-Ramos, Juan-Bautista Ornelas-Rodriguez, Francisco-Javier Castillo Castaneda, Eduardo Ramirez, Alfonso Garcia, Angel Ivan Gonzalez-Barbosa, Ricardo Avina-Cervantez, Juan Gabriel CICATA, Intituto Politecnico Nacional, 141 Cerro Blanco Col. Colinas del Cimatario, Queretaro, Mexico

This paper appears in: Electrical Communications and Computers (CONIELECOMP), 2011 21st International Conference on Issue Date : Feb. 28 2011-March 2 2011 On page(s): 206 Print ISBN: 978-1-4244-9558-0 Digital Object Identifier : 10.1109/CONIELECOMP.2011.5749361 Date of Current Version : 11 abril 2011

ABSTRACT

We propose a map building system for generation of 3D city models using a vehicle equipped with LIDAR (Light Detection And Ranging) technology, which is capable of obtaining more than one million points per second. This approach is based in two methods: plane segmentation and local map construction. The First method introduces a fast method to extract the ground plane. The rest of the points are used to extract planes using a general plane extraction. For the general plane extraction process, is used a method that choose a random point and its neighbor in order to compute a plane; this plane is evaluated with respect to the inliers number, and the best plane is the one containing the biggest number of points. Finally, using the planes, we construct an environment map.

INDEX TERMS

Available to subscribers and IEEE members.

REFERENCES

Available to subscribers and IEEE members.

CITING DOCUMENTS

Available to subscribers and IEEE members.

© Copyright 2011 IEEE – All Rights Reserved





Organizing Committee:

General Chair Rubén Alejos Palomares University of the Americas ruben.alejos@udlap.mx

Technical Program Felix Eduardo Guerrero University of the Americas felixe.guerrero@udlap.mx

Alfredo Sánchez Huitrón University of the Americas alfredo.sanchez@udlap.mx

Daniel Vallejo Rodríguez University of the Americas <u>daniel.vallejo@udlap.mx</u>

Tutorial courses.

Eduardo Jiménez López University of the Americas eduardoj.jimenez@udlap.mx

Ofelia Cervantes Villagómez University of the Americas ofelia.cervantes@udlap.mx

Finances.

José Luis Vázquez González University of the Americas josel.vazquez@udlap.mx

José Luis Zechinelli University of the Americas joseluis.zechinelli@udlap.mx

Web Master Felix Guerrero Castro University of the Americas felix.guerrero@udlap.mx

Local arrangements Eduardo López Sánchez University of the Americas eduardo.lopez@udlap.mx

21st International Conference on Electronics, Communications and Computers

CONIELECOMP 2011

February 28th –March 2nd,2011 Cholula, Puebla, México

The 21st International Conference on Electronics, Communications and Computers (CONIELECOMP 2011) is organized by the *Department of Computer Sciences, Electronics and Mechatronics* at Universidad de las Américas-Puebla, México, and co-sponsored by *Center for Research in Information and Automation Technologies of the Universidad de las Américas* and *IEEE Section Puebla*. The CONIELECOMP will be held in Cholula, Puebla, México, from February 28th to March 2nd, 2011.

About the Conference

The aim of CONIELECOMP is to provide a forum for scientists and engineers throughout the world to present and exchange ideas and results from original research and applications in the listed technical areas but not restricted to them. The first CONIELECOMP took place in 1990 and it has been held annually since then. The conference will feature keynote speakers, tutorial presentations, lectures, and poster sessions.

Call for Papers

Prospective authors are invited to submit a 4-6 pages full paper of their contributions for the proceedings. Once the submitted papers are reviewed and accepted, they will be placed into the conference schedule. The preliminary program will appear in the conference website. Deadlines for all submissions are shown below. All submissions will be handled electronically. The CONIELECOMP 2011 website <u>http://www.conielecomp.org/2011</u> will provide details regarding paper submission, registration, program, events, accommodations, and tourist attractions.

Technical Areas

Including but not limited to:

Electrical Engineering and Mechatronics

- VLSI Analog and Digital Circuit Design - Computational Intelligence
- Instrumentation
- Bioelectronics and Biomedical Systems
- Computer Architecture
- Computer Networks
- Communications
- Control Systems
- Optoelectronics
- Power Electronics
- Semiconductors
- Signal Processing - Robotics
- MEMS and Nanotechnology
- Embedded Systems

Computer Systems and Information Technologies

- Security
- E-Learning
- Artificial Intelligence
- Computer Vision
- Software Engineering
- Human-Computer Interaction
- Multimedia Systems
- Data Bases and Data Mining
- Operating Systems
- Computer Graphics
- Virtual Reality
- Computability
- Distributed Systems
- Geographic Information Systems

Important Deadlines

Paper Submission: Notification of acceptance: Advanced registration deadline: Publication of Technical program



Sección Puebla

October 31, 2010 November 21, 2010 January 14, 2011 January 28, 2011



3D City Models: Mapping Approach using LIDAR technology

Denis-Eduardo Hernández-García, José-Joel Gonzalez-Barbosa, Juan-Bautista Hurtado-Ramos, Francisco-Javier

Ornelas-Rodríguez, Eduardo Castillo Castaneda, Alfonso Ramírez, Angel Ivan García,

CICATA, Intituto Politecnico Nacional, 141 Cerro Blanco Col. Colinas del Cimatario, Queretaro

jgonzalezba@ipn.mx

Ricardo Gonzalez-Barbosa, Juan Gabriel Aviña-Cervantez Campus Irapuato-Salamanca. Universidad de Guanajuato.

ABSTRACT

We propose a map building system for generation of 3D city models using a vehicle equipped with LIDAR (Light Detection And Ranging) technology, which is capable of obtaining more than one million points per second. This approach is based in two methods: plane segmentation and local map construction. The First method introduces a fast method to extract the ground plane. The rest of the points are used to extract planes using a general plane extraction. For the general plane extraction process, is used a method that choose a random point and its neighbor in order to compute a plane; this plane is evaluated with respect to the inliers number, and the best plane is the one containing the biggest number of points. Finally, using the planes, we construct an environment тар.

I. INTRODUCTION

An accurate surrounding environment model permits to a robot to complete complex tasks quickly, reliably, and successfully. Map based navigation calls three processes: map learning, localization, and path planning [1]. These three processes may rely on three different distinct sources of information available to a robot; the first source enables the robot to sense its own personal configurations and its relationship to the environment; the second is required to safely move in the mission's environment by detecting and avoiding obstacles along the desired trajectory toward its target; and the third is to recognize the associated requirements and features of the targeted mission and assure its successful execution. There are many ways to integrate these information sources in a representation that is useful for robot navigation. The drawbacks and advantages of these information sources are complementary. For the path planning purpose and obstacle avoidance a detailed analysis is not necessarily required, and high speed and rough of the environment understanding around the robot can be sufficient.

Velodyne (see Figure 1) has developed and produced a High Definition LIDAR (HDL) sensor designed to satisfy the demands for autonomous vehicle navigation, stationary and mobile surveying, mapping, industrial use and other applications. The Velodyne HDL unit provides 360-degree azimuth field of view (θ) and 26.5-degree elevation field of view (ϕ), up to 15 Hz frame refresh rate, and a rich point cloud populated at a rate of one million points per second. The HDL-64E operates on a rather simple premise: instead of a single laser firing through a rotating mirror, 64 lasers are mounted on upper and lower blocks of 32 lasers each and the entire unit spins. This design allows for 64 separate lasers to each fire thousands of times per second, providing exponentially more data points per second and a much richer point cloud than conventional designs. The HDL-64E is rated to provide usable returns up to 120 meters.

II. RELATED WORK

In this work, we present a method for computing an environment model from dense 3D point clouds. In our approach we extract the ground plane and the main façade planes. The ground plane is extract



Fig. 1. (a) Velodyne mounted in the car. The Velodyne is composed of 64 lasers. Each laser orientation is in function of ϕ and θ . The ϕ angle is fixed for the lasers, and the θ angle is done by the Velodyne rotation.(b) Example of the points cloud. The data 3D points in this Figure correspond to only one Velodyne rotation, the number of points are more that 1000 000.

fast method designed for this specific problem. For the 3D point cloud all dominant façade planes are detected with a random plane extraction method, finally a map for robot navigation is constructed.

A. Extraction of the planes

Several works proposed for the façade plane segmentation on modeling city areas. Image-based methods were proposed by Zisserman et. al. [2], where planes are detected by a sweeping method; and Teller et. al. [3] where are extracted the facade plane in microstructures. Another approach uses dense 3D point clouds resulting from image matching for environment modeling. Those point clouds should be transferred into a low-polygon representation while still preserving fine detail e.g. edges. Proposed methods vary from general methods [4] and [5], which do not consider the specific demands on building model simplification approaches, to very specific problem approaches [6]. When dealing with buildings the most prominent features are façade planes and in those planes the indents from windows, doors etc. and protrusions created e.g. by balconies and friezes.

B. Maps construction

Mapping refers to solving the problem of generating robot environment models from sensor data. The problem importance is due to the fact that many successful mobile robot systems require accurate maps for their operation. Examples include the Helpmate hospital delivery robot [7], Xavier [8], and several museum tour-guide robots [9]. In the literature is possible to find a variety of different representation variety. In the indoor systems context, three concepts prevail: geometric, topologic, and occupancy probability grids.

Topologic representations aim at representing environments by graph-like structures, where edges correspond to places, and arcs to paths between them. In [10] the topological map is constructed using a panoramic annular lens (PAL), where the main advantage is the rotational invariant, scalable description of the whole 360 degrees environment. In this work, the map is described by a graph structure, the nodes correspond to recognizable landmarks where different actions can be taken, and links are associated with regions where some environmental feature can be used by the robot to control its position.

Geometric models use primitives for environment representation. The primitives can be used for polygons environment representation, which are generally simple and have a regular structure, more powerful models are needed for environments like houses, of which the structure may be complex and not easy to describe with simple geometric primitives. In a metric maps the introduction of a metric involves a high precision of environment features and localization [11], [12].

A very popular probabilistic approach to represent the environment is the called occupancy probabilistic grids, which is an approximate method. In [13], the authors present the way how is constructed a linear map of an unknown indoor environment using a mobile robot equipped with a single sonar sensor. The proposed method consists of two main steps. In the first step, the robot measurements surroundings are mapped into a twodimensional occupancy grid map. In the second step, Radon transform is used to extract the line parameters from the grid map occupancy. These parameters are subsequently used to represent the profiles of the detected objects as a robot environment representation. Applying a wall extension process to the resulting line map completes the process within the limits of resolution of the grid map.

III. PLANES EXTRACTION

The number of points in the data acquisition increases the plane extraction time. In this section, we present two methods. In the first one, we introduce a method dedicated to extract the ground plane (see Section III-A). The number of 3D points lying in the the floor plane depends of the environment, but generally is a considerable amount. The second method uses the 3D points excluding the points belonging to the floor plane. These points are used by the general plane extraction process(see Section III-B). The Table I shows the map construction algorithm, P and Π' refers to the set of 3D points cloud and planes found respectively, π represents each plane, A_0 , B_0 , C_0 , D_0 refers the coefficients of the main plane, S_0 are the points outside the main plane. We introduce a method to extract the ground plane. The points that cannot be modeled with this method are processed with a random plane extraction method in order to obtain all the environment planes.

A. Floor plane extraction

The velodyne is mounted in the car (see Figure 1), the relation between the Velodyne frame (O_v) and the plane frame (O_f) is shown in the Figure 2.

For a given θ angle, the laser *i* (where *i* = 1, 2, ... 64) forms a triangle with *h* and the plane. We can express the relation in the Figure 2 by:

$$-h + r_i \cos(\phi_i) \cos(\phi_0) + r_i \sin(\phi_i) \sin(\phi_0) = 0$$

Where r_i is the distance between the Velodyene frame 0_v to the object in direction (θ, ϕ_i) of the laser *i*, and ϕ_i is the angle formed by r_i and the *Z*- axis of the Velodyne frame. The last equation can be defined by :

$$Ax = 0 \tag{1}$$

TABLE I

GENERAL ALGORITHM FOR LOCAL MAP BUILDING

	Input:				
	• 3D points				
	$P = \{ p_1 p_2 p_3 \cdots p_n \}$				
	Output:				
	• $\Pi' = (\pi_0, \pi_1, \pi_2, \cdots \pi_r)$				
	• Local Map (LM)				
(i)	Computing the principal plane				
	$\pi_0 \leftarrow \begin{bmatrix} A_0 & B_0 & C_0 & D_0 \end{bmatrix}^T$				
(ii)	Computing subset S_0				
	$S_0 \leftarrow \{S_0 \in P \mid d(p_i, \pi_0) > \varepsilon\}$				
	where d is the Euclidean				
	distance between π_0 and p_i				
(iii)	RANSAC planes fitting				
	using S_0				
	$\Pi' \leftarrow \{\pi_1, \pi_2 \pi_r\}$				
(iv)	Local Map building				
	using $\{\pi_0, \pi_1, \pi_2 \pi_r\}$				

TABLE II

PRINCIPAL PLANE

	Input:				
	• 3D points represented as :				
	$(\theta_i, \phi_i, r_i(\theta_i))$				
	$\theta_i \leftarrow \text{LIDAR orientation}$				
	$\phi_i \leftarrow \text{orientation of the laser } i$				
	$r_i(\theta_i) \leftarrow$ measure distance by				
	the laser <i>i</i> in orientation θ_i				
	Output:				
	• $\pi_0 \leftarrow \begin{bmatrix} A_0 & B_0 & C_0 & D_0 \end{bmatrix}^T$				
(i)	initialization				
	$j \leftarrow 0$				
do					
(ii)	For each θ_i solve the equation 1				
	$T \leftarrow \{(\phi_0^j, h_i, \theta_i, \sigma_i)\}$				
(iii)	$i \leftarrow i + 1$				
while	$(\theta_s < 360)$				
(iv)	$n'_{a} \leftarrow (\phi^{k}_{a}, h, \theta, \sigma_{a})$				
(1)	σ_{1} is the small values in T				
(\mathbf{x})	\mathcal{D}_{k} is the small values in \mathcal{I}				
(\mathbf{v})	$n_0 \leftarrow \begin{bmatrix} A_0 & D_0 & C_0 & D_0 \end{bmatrix}$				
	where				
	$A_0 = \cos(\phi_0^\kappa) \cos(\theta_k)$				
	$B_0 = \cos(\phi_0^k)\sin(\theta_k)$				
	$C_0 = \sin(\phi_0^k)$				
	$D_0 = h_k$				

where

$$A = \begin{bmatrix} -1 & r_n(\theta)\cos(\phi_n) & r_n(\theta)\sin(\phi_n) \\ -1 & r_{n+1}(\theta)\cos(\phi_{n+1}) & r_{n+1}(\theta)\sin(\phi_{n+1}) \\ \vdots & \vdots & \vdots \\ -1 & r_{n+k}(\theta)\cos(\phi_{n+k}) & r_{n+k}(\theta)\sin(\phi_{n+k}) \end{bmatrix}$$
$$\leq n+k \leq 64, \quad \text{and} \quad x = \begin{bmatrix} h \\ \cos(\phi_0) \\ \sin(\phi_0) \end{bmatrix}$$

1



Fig. 2. For a given θ angle, the laser *i* forms a triangle with the height *h* and the ground plane



Fig. 3. Plane representation computed with the nearest neighbors x_i .

 $A = U\Sigma V^T$ is the singular value decomposition of A, the x solution is given by v_k , where v_k is the eigenvector that correspond to the smallest eigenvalues σ_k different to zero.

The Figure 2 shows that the Equation 1 is correct when the normal vector of the plane belongs to the plane formed by the laser directions, for these case the σ values are small. The Table II shows the algorithm porposed, where T refers to the set of parameters after solving the equation 1, n_0 is the normal at each point, A_0 , B_0 , C_0 , D_0 are the coefficients of the normal with lowest eigenvalue. In the other case, the Equation 1 cannot be used to compute the normal and the error of this equation will increase.

B. Random plane extraction

In this work the algorithm fits the model plane in a 3D space. We consider that the points are sorted, so when a random point is select, we use its neighbors to compute the plane model. With this model, we computed the distances between the 3D points and the plane model. The inliers correspond to points whose distance is smaller than ϵ . These points are used for compute the final plane model. The random algorithm is shown in the Table III.

The mathematical model is given by:

$$Bx + Cy + Dz + E = 0 \tag{2}$$

This model can be computed using the n nearest neighbors (see Figure 3). In this case the model is represented by :

$$\begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & z_n & 1 \end{bmatrix} \begin{bmatrix} B \\ C \\ D \\ E \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$
(3)
$$Ax = 0$$

The model is computed by the solution of the Equation 3. This solution is computed by the decomposition of $A = U\Sigma V^T$, where $V = [v_1, v_2, v_3, v_4]$, $diag(\Sigma) = [\sigma_1, \sigma_2, \sigma_3, \sigma_4]$ and $\sigma_1 > \sigma_2 > \sigma_3 > \sigma_4$. The x solution is given by v_k , where v_k is the eigenvector that corresponds to the smallest eigenvalue σ_k different to zero.

IV. MAPS CONSTRUCTION

The random algorithm computed the plane equation and the 3D points that belong to each plane $((\Pi_0, S_0), (\Pi_1, S_1), (\Pi_2, S_2)...)$. The floor plane is unbounded, because we consider that the floor is not completely built by the occlusions of the walls and objects. However, the planes that describe the walls are discontinues because their contains door and corridors for example. The 3D points S_i are segmented used Mean Shift algorithm [14]. These 3D points segmented and the plane equations are used for computed the plane segment and describe the map environment.

V. EXPERIMENTAL RESULTS

In this work, we present a mapping approach for robot navigation. This approach is optimal for structured environments, where the floor points are fitting a plane and the number of façade is high. Using the extracted planes and the unclassified 3D points we construct an environment map. We use a sensor capable of obtain more than one million of 3D points per second, for this reason we present

A	LGORIT	HM : RANDOM 3D PLANE EXTRACTION
Input:		Points cloud S. Minimum number of inliers
mput.		permits for the model N . Threshold ϵ Number
		of points permit in the final process M
Output		Equations of planes (Π_{i}) and the 3D points
Output.		belong to the plane (S_1) and the SD points
		belong to the plane (by)
		C is the number of point in S
Do		0
		$S_i \leftarrow null$
		$j \leftarrow 0$
	(i)	Randomly select one point and their neighbors
		s of the data points from S
	(ii)	Compute the model plane. Substituing s to the
		Equation 3, and the model plane is computed
		using their solution
	(iii)	Determine the set of data points S_i which are
		within a distance threshold ϵ of the model (see
		Figure 3). The set S_i defines the inliers of S .
		R_i is the number of point in S_i
If		$R_i > N,$
	(iv)	$j \leftarrow j + 1,$
	(v)	$S_j \leftarrow S_i$
	(V1)	Final plane model. Substituing S_j to the
		Equation 3. Their solution correspond
	<i>(</i> ···)	to the final model plane (Π_j)
and	(V11)	Remove the points S_i of the points clouds S
end		(C < M)
unui		$(U \leq M)$

TABLE III

a method for ground extraction and the rest of the planes are extracted by a general plane extraction method.

A. Floor Plane extraction

The original information is show in Figure 1 (b). The Figure 4 shows the (θ, ϕ, h) results versus the smallest eigenvalues (σ_k) computed with the Equation 1. The figure 4 (a) shows that the θ values can be approximated to a parabolic function around the local minimum. The θ value is selected for the minimum σ_k using a parabolic fitting. The ϕ and h values are computed by extrapolation when $\sigma_k = 0$ using the data of the Figure 4 (b) (c) respectively. The (θ, ϕ) are used to compute the normal vector to the floor; however, the h values correspond to distance between the floor and the velodyne in direction of the floor normal. The floor normal and the distance h are used for floor segmentation, the result is shown in the figure 5.

B. Random plane extraction

The plane model computed by the random method (III-B) doesn't take into account the point



Fig. 4. Results : θ , ϕ and h respect to the eigenvalues (σ_l) given in the solution of the Equation 1



Fig. 5. Floor Segmentation: (a) 3D floor points and (b) rest of 3D points

cloud discontinuity. The result of this plane extraction method is shown in the figure 6, the points belonging to the planes are not continuous.



Fig. 6. Planes extraction by the random method describe in Section III-B

C. Map Construction

The method described in the section III-B permits us to classify the points in several model planes. Each subset of points S_i is segmented by Mean Shift algorithm; This method allows plane subdivide (see figure 7). The new segment planes have a continuous 3D points.



Fig. 7. The planes are segmented using the Mean Shift Method

VI. CONCLUSIONS AND FUTURE WORK

The approach described in this paper aims to segment automatically and extract planar surfaces from a building façade captured by terrestrial laser scanner. Firstly, the point cloud is segmented into several planes using a random plane extraction algorithm. The results obtained are satisfactory, because they are produced based on the unique assertion that the best plane is the plane containing a maximum of points. Thus, considering that no additional constraint is needed, the global accuracy is better than expected. Therefore, the proposed methodology enables reliable facade segmentation with decreased processing time using terrestrial laser scanner data. It constitutes a first and primordial step in the generation of complete 3D building models.

The second part is based on the enhancement of segmentation operation. In order to avoid the plane models with discontinue points clouds and increase the global and relative accuracy of the results, we considered a geometrical and topological constraint and the planes are subdivided by a Means Shift algorithm

In future works, a precision GPS will be used in order to join several maps. A panoramic camera will be installed with the purpose of add texture to the map building. Acknowledgements

This work was supported partially by CONA-CYT grants CB-2005-01-51004 and SIP-IPN grant 20101726.

REFERENCES

- L. T.S. and L. D.T., "Qualitative navigation for mobile robots," *Artificial Intelligence*, vol. 44, no. 3, pp. 305–360, 1990.
- [2] A. Zissermana, T. Werner, , and F. Schaffalitzky, "Towards automated reconstruction of architectural scenes from multiple images," *In Proc. 25th workshop of the Austrian Association for Pattern Recognition*, pp. 9–23, 2001.
- [3] S. Teller, XiaoguangWang, S. Totaro, F. Taillandier, and A. Hanson, "Recovering façade texture and microstructure from real-world images," *In Proceedings, ISPRS Commission III Symposium on Photogrammetric Computer Vision*, pp. 381–386, 2002.
- [4] C. Andujar, P. Brunet, and D. Ayala., "Topology-reducing surface simplification using a discrete solid representation.," *ACM Transactions on Graphics*, p. 21(2), 2002.
- [5] M. Garland and P. S. Heckbert., "Surface simplification using quadric error metrics.," *In Proceedings of SIGGRAPH '97*, pp. 209–216, 1997.
- [6] H. Mayer, "Scale-space events for the generalization of 3dbuilding data," in *in: International Archives of Photogrammetry and Remote Sensing*, pp. 639–646, 2000.
- [7] S. King and C. Weiman, "Helpmate autonomous mobile robot navigation system," *In Proc. Of the SPIE Conference on Mobile Robots*, vol. 2352, pp. pages 190–198, 1990.
- [8] R. Simmons, R. Goodwin, K. Haigh, S. Koenig, and J. O'Sullivan, "A layered architecture for office delivery robots," *In Proc. of the First International Conference on Autonomous Agents (Agents)*, 1997.
- [9] W. Burgard, A. Cremers, D. Fox, D. Hähnel, D. S. G. Lakemeyer, W. Steiner, and S. Thrun, "Experiences with an interactive museum tour-guide robot," *Artificial Intelligence*, vol. 114, no. 1-2, 2000.
- [10] Z. Vamossy, "Topological map building from pal images with navigation purpose," 6th International Symposium on Applied Machine Intelligence and Informatics, pp. 217–220, 2008.
- [11] D. Hähnel, W. Burgard, D. Fox, and S. Thrun, "A highly efficient fastslam algorithm for generating cyclic maps of largescale environments from raw laser range measurements," *in Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, no. October, 2003.
- [12] A. Davison, Y. G. Cid, and N. Kita, "Real-time 3d slam with wideangle vision," *in Proc. IFAC Symposium on Intelligent Autonomous Vehicles, Lisbon*, vol. July, 2004.
- [13] M. L. R. Jaradat, "Line map construction using a mobile robot with a sonar sensor," *International Conference on Advanced Intelligent Mechatronics*, pp. 1251–1256, 2005.
- [14] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence (IEEE)*, vol. 17, no. 8, pp. 790–799, 1995.



Office Outlook Web Access	Escriba aquí para realizar la búsqued: Esta carpeta	🕮 🔛 libreta de direcciones 🛛 🔛 Opciones 🔍 🖤 Cerrar sesión		
Correo	Responder Responder a todos Reenviar	liminar Cerrar 🗢 👻 🔤		
Calendario	Re: Correo Carmen Espino			
Contactos	Congreso Ingeniería UAQ [librocongresing@gmail.com]			
	Enviado el: jueves, 07 de abril de 2011 23:33			
Borradores [3]	Para: Jose Joel Gonzalez Barbosa			
🐻 Correo no deseado	ſ			
Deleted Items				
inbox (19)				
Sent Items				

Haga clic para ver todas las carpetas. 😼

🎒 Administrar carpetas...

Estimado Jose Joel Gonzalez B, como se le ha informado el estatus de su capítulo: "Detección de objetos tridimensionales" es de ACEPTADO.

Ahora bien, el trabajo de Jose Guadalupe Rico Espino no lo tenemos registrado en nuestra base de datos, por lo que consideramos que a lo mejor lo mando a otra sección del Congreso y no aquí. Por lo que lo invitamos a que lo someta nuevamente en la siguiente edicion del libro. Gracias

2011/4/7 Jose Joel Gonzalez Barbesa <<u>jgenzalezba@ipn.mx</u>> Estimado Comité

Tengo dos preguntas.

A). El trabajo que fue enviado el 7 de Febrero, tiene el título Detección de objetos tridimensionales Three-dimensional object detection Alfonso Ramírez Padraza, Denis-Eduardo Hernández-García, José Joel González Barbosa, Francisco, Javier Órnelas Rodríguez y Juan Bautista Hartado Ramos

mas sin embargo, José Guadalupe envío otro artículo después del 14 de Febrero con el título

Sistema de visión de reconstrucción 3D con lente telecéntrica 3U reconstruction vision system with a telecentric lense José Guadalupe Rico Espino, José Joel González Darbosa, Roberto Augusto Gómez Loenzo

¿LA ACEPTACIÓN DEL ARTÍCULO A CUAL DE LOS DOS ARTÍCULOS CORRESPONDE?.. (A LOS DOS?

B). Podrían enviarme el formato de la siguiente carta:

 Envie su conformidad para poder publicar bajo las normas de estilo del libro a este correo para ser publicado

Saludos

José Joel González Barbosa

Detección de objetos tridimensionales Three-dimensional object detection

Alfonso Ramírez Pedraza, Denis-Eduardo Hernández-García, José Joel González Barbosa, Francisco Javier Órnelas Rodríguez y Juan Bautista Hurtado Ramos

Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Instituto Politécnico Nacional

RESUMEN. La segmentación de objetos sobre datos 3D es un tema fundamental en el procesamiento en tres dimensiones. La calidad de la segmentación determina en gran medida el éxito de la recuperación de información de los objetos. La segmentación aplicada en este artículo, consiste en dos etapas. La primer etapa toma en consideración que el ambiente es estructurado, así el suelo es segmentado y los planos verticales usando la transformada Hough. La segunda etapa considera que los puntos 3D restantes corresponden a objetos no estructurados. Estos objetos son segmentados por el algoritmo Mean Shift. Finalmente se presentan resultados de las diferentes etapas de la segmentación. *Keywords: Nube de Puntos 3D, Lidar, Segmentación.*

1. INTRODUCCIÓN

Hoy en día existen muy pocos métodos para llevar a cabo segmentaciones de objetos dentro de la nube de puntos. Cada uno de los diferentes métodos de segmentación de objetos se pueden clasificar según su funcionamiento y las técnicas para lo cual fue desarrollado.

Una de las ventajas que nos brinda el escáner laser Lidar es, exactitud y precisión comparándolo con otros sistemas o métodos de reconstrucción 3D. Igualmente la facilidad de hacer estudios de campo, en donde solo nos preocupamos por levantar datos, la obtención de información como pudiesen ser distancias, tamaños, formas y longitudes nos la brinda el mismo sistema (Lidar), y estas pueden hacerse fuera de línea.

La importancia de llevar a cabo la segmentación sobre Nube de Puntos 3D, es el poder reconocer que objetos fueron reconstruidos. Por ejemplo, se puede reconstruir una zona grande como una ciudad y reconocer la flora presente en la reconstrucción para hacer un inventario del tipo de flora y ubicación de esta.

El problema de la segmentación de objetos en las nubes de puntos 3D es un gran reto. Los datos del mundo real son muy ruidosos. Así que llevar a la práctica estas técnicas requieren de grandes habilidades y destrezas. A diferencia de las imágenes, no podemos utilizar colores o texturas como señales, los problemas de la segmentación, es la entrada de una nube de puntos ruidosa que representa una escena, en lugar de una superfície limpia del área o de un objeto individual.

El objetivo particular de este trabajo es desarrollar una metodología para detectar y segmentar los objetos reconstruidos. En la literatura se ha abordado poco el problema de segmentación de objetos en nube de puntos 3D. El detectar, segmentar, reconocer y posiblemente eliminar todo objeto no perteneciente al ambiente tiene mucha mayor importancia de lo que actualmente se cree. Presentar visitas virtuales 3D de ciudades sin objetos que obstaculicen la vista real de un determinado panorama, es mucho más atrayente para los usuarios, que guiar una visita 3D con objetos que no corresponden al medio ambiente, o bien; recopilar información y llevar un inventario respecto a que flora se encuentra en algún determinado paisaje ya sea urbano o rural.

La metodología desarrollada para la segmentación de objetos 3D se desarrolla en dos etapas. La primera de ellas segmenta los objetos estructurados del medio ambiente reconstruido usando la transformada Hough. Estos objetos segmentados corresponden al suelo y las paredes verticales a este. La segunda etapa trabaja con los puntos 3D restantes los cuales pertenecen a objetos no estructurados. La segmentación de estos objetos es llevada a cabo por el método MeanShift. La segmentación de objetos no estructurados nos permitirá en un futuro reconocerlos y eliminarlos, ya que por ejemplo, los carros y las personas no son parte del ambiente reconstruido. Igualmente nos permitirá hacer un inventario de la flora incluyendo la ubicación real de esta en un mapa 3D, etc.

Las técnicas utilizadas son la Transformada de Hough, la cual fue pensada y patentada por Paul Hough el 18 de Diciembre de 1962 como una técnica para detección de curvas en una imagen binaria. Esta técnica consiste en aislar características de forma particular dentro de una imagen. La idea básica es encontrar curvas que puedan ser parametrizadas como líneas rectas, polinomios y círculos. La técnica MeanShift es una técnica general no paramétrica para encontrar grupos en espacios de características. Este método fue implementado por D. Comaniciu y P. Meer en el año 2002 (Comaniciu and Meer, 2002), pero originalmente fue desarrollado por

Fukunaga y Hostetler en 1975 (Fukunaga and Hostetler, 1975). La razón para utilizar una técnica general no paramétrica de estimación de la densidad se debe a que el espacio de características se puede considerar como la función de densidad del parámetro representado.

En el siguiente capítulo presentamos los trabajos relacionados a la segmentación de objetos en una nube de puntos 3D, mientras que en la sección 3 hacemos una descripción de la metodología de segmentación de objetos que hemos implementado, en la sección 4 se presentan los resultados obtenidos y finalmente una discusión de estos.

2. TRABAJOS RELACIONADOS

Dado que el levantamiento de datos a gran escala en ambientes de exterior proporcionan una gran variedad de fuentes de datos, no existen muchos trabajos que describan las segmentaciones sobre la Nube de Puntos 3D en exterior. La mayoría de los trabajos se centran en la extracción de primitivas geométricas como planos (Rabbani, Van Den Heuvel, and Vosselmann, 2006) y (Unnikrishnan and Hebert, 2003), pero no en la segmentación de objetos no estructurados.

Algunos trabajos como el presentado por (Rabbani, Van Den Heuvel, and Vosselmann, 2006) usan puntos 3D que corresponden a un solo objeto, el método consiste en descomponer dicho objeto en varios sub-objetos o manchas con características geométricas semejantes. En el trabajo de (Rabbani, Van Den Heuvel, and Vosselmann, 2006) emplean algoritmos que modelan por medio de mallas los puntos 3D y luego llevan a cabo la segmentación. La segmentación se lleva a cabo por la modelización de la malla en función de su normal y curvatura.

En el levantamiento de datos en ambientes de exterior, es difícil porque muchos datos son ruidosos y los objetos son codificados de manera incompleta. En general la segmentación de los puntos se basa en la estimación de la curvatura. La extracción de objetos en ambientes de exterior es relativamente nueva, y está impulsada por la reciente disponibilidad de explorar áreas de difícil acceso para el ser humano y sobre todo áreas que son complicadas de manipular por la gran cantidad de información que pudiesen tener. El algoritmo presentado por (Golovinskiy, Vladimir, Funkhouser, 2009) se compone de cuatro etapas principales las cuales son: localización, segmentación, caracterización y clasificación de grupos en puntos 3D. El reconocimiento se lleva a cabo dentro de la nube de puntos 3D en exteriores. El proceso es el siguiente, primero agrupan los puntos cercanos según su forma, en función de las localizaciones de los objetos; segundo, segmentan la localización de los puntos del frente y de atrás usando un algoritmo graph-cut; tercero, se construye un vector de características de cada punto del grupo; por último se etiquetan las características de los vectores utilizando un clasificador entrenado.

En robótica móvil la segmentación de objetos 3D y la clasificación es importante para las tareas de localización del robot o para la generación de trayectorias. Los trabajos en esta área se centran en el reconocimiento de diferentes clases de objetos en ambientes familiares en donde se pueden encontrar cientos de objetos cotidianos. Por ejemplo en (Wohlkinger, Vincze, 2010) usan un sistema de visión basado en categorizar los diversos objetos, para lo cual primero reconocen o identifican las diferentes restricciones del medio ambiente, y posteriormente realizan las segmentación de los diversos objetos.

Los trabajos de (Strom, Richardson, Olson, 2010) presentan un algoritmo para segmentar dentro de la nube de puntos. La segmentación de los datos se realiza por medio de un sensor, el cual permite realizar tareas en un alto nivel, captar obstáculos y clasificar el terreno. El método propuesto permite la combinación de información como el color en un campo amplio de visión con la nube de puntos 3D. La segmentación aplicada es robusta y usa como principal característica las normales y el color. Los resultados de segmentación son más robustos que el solo usar una cámara o escáner laser, el método es probado en escenas de interiores como en exteriores.

(Brenneke, Wulf, Wagner, 2004) presenta un enfoque llamado SLAM (Localización Simultanea y ping-mapa) la cual se basa en el escaneo de los desniveles o inclinaciones del suelo, el método es una combinación de la percepción 3D y cartografía 2D. Lo métodos presentados permiten establecer una ruta de navegación usando algoritmos de planificación, con costos computacionales bajos. El método se divide en cuatro etapas: la primera es la adquisición de datos, en la segunda la segmentación de los obstáculos, en la tercera escanean los desniveles o inclinaciones que pudiesen existir y finalmente los resultados obtenidos se usan para el SLAM. El método brinda los mejores resultados en áreas de exteriores.

Trabajos como el de (Moosmann, Pink and Stiller, 2009) presenta un algoritmo capaz de hacer frente a enormes cantidades de datos 3D, para lo cual se utiliza un enfoque gráfico basado en el suelo y los objetos explorados por el Lidar, el criterio que utiliza es basado en las medidas de convexidad local. Los resultados son mejores en entornos urbanos, incluyendo superficies curvadas de la carretera.

3. METODOLOGÍA

La metodología que utilizamos para la segmentación de objetos está definida en dos etapas. La primera segmenta los objetos estructurados, tales como suelo y paredes perpendiculares a este. El método de Hough permite agrupar los clúster que definen un modelo geométrico como planos. La segunda etapa utiliza los puntos restantes para segmentar los objetos no estructurados con el método de MeanShift. Este método nos permite calcular la media de los diferentes clústeres y en base a ese dato poder agruparlos según el objeto al cual pertenecen.

3.1. Segmentación de objetos estructurados

Los objetos estructurados corresponden al suelo y a las paredes de las construcciones. La referencia del Lidar no siempre coincide con la referencia del suelo, por esta razón el suelo es segmentado como primer parte del procedimiento. El plano del suelo y los puntos que no corresponden a él son calculados de acuerdo al algoritmo 1.

Algoritmo 1. Segmentación del suelo.	Algoritmo 2. MeanShift
Entrada: Nube de puntos X donde $x_i \in \Re^d; i = 1 \dots n$ Inicialización $h \leftarrow h_0$ h_0 es la altura del Lidar $n \leftarrow -[0 \ 0 \ 1]$ n_0 normal del plano $m_k \leftarrow 0$	Entrada: Datos X donde $x_i \in \Re^d$; $i = 1 \dots n$ ancho de banda h for all $x_i \in X$ do $x_t \leftarrow x_i$ while $m_h(x_i) \neq \text{umbral do}$ $Calcular m_h(t);x_t \leftarrow x_i + m_h(t).$
while $ m_k - m_{k-1} < \alpha$ $d_i = \frac{ nx_i + h }{nn^T}$ $Y \leftarrow \{x_j d_j < \beta\}; j = 1 \dots m$ $m_k \leftarrow m$ Se recalcula (n, h) a partir de Y	End While $GrupoCent_i = \{x \in D \setminus dist (x, x_t)esm$ If $GrupoCent_i$ no existe them $Grup o_i = nuevo GrupoId;$ end if end for
$\frac{\text{End}}{Z \leftarrow \{x_j \mid d_j > \beta\}}$	Salida: Vector Grupo ∈ ℜ ^d
Salida: Parámetros del plano del suelo (n, h) y puntos que no pertenecen al suelo Z	

Una vez segmentado el suelo, todos los puntos 3D son rotados al eje de referencia del suelo. De esta forma, podemos conocer que los planos a encontrar son perpendiculares al plano del suelo. A partir del conjunto de puntos Z se calculan los planos perpendiculares al suelo con el método de Hough.

La transformada de Hough es una técnica muy efectiva cuando se trata de detectar puntos alineados que pueden existir en un conjunto de datos, Hough trabaja en un espacio paramétrico como una función $f(\rho, \theta)$, donde ρ, θ son los parámetros de la normal que pasa por del origen, esta técnica se aplica con propósito de realizar una segmentación de las primitivas geométricas presentes dentro de la nube de puntos restantes luego de extraer el plano principal, para encontrar los diferentes planos que están integrados en la nube, caracterizados por la densidad y dirección de los puntos, en el espacio paramétrico una tripleta ordenada (x, y, z) representa una superficie sinusoidal, la implementación de esta técnica para este artículo se resumen en el algoritmo 3.

Algoritmo 3. Transformada de Hough

Entrada: Nube de puntos Z, donde $z_i \in \Re^d$, i = 1..n

Hacer

Discretizar el espacio paramétrico acum(ρ , θ)

 $acum[\rho_i, \theta_i] \leftarrow 0$

Para cada punto $z_i(x_i, y_i, z_i), i = 1, 2, ... \rho_{max}$

a. Hacer $\rho = x \cos \theta_i + y \sin \theta_i$

- b. Encontrar posición con respecto a ρ , $k = \frac{\rho}{\Delta \rho} + \frac{\rho_{\text{max}} \rho_{\text{min}}}{2\Delta \rho}$
- c. Incremente matriz de contadores acum[k, h]

Encontrar Máximo local max $\leftarrow acum(k_{max}, h_{max})$ tal que max > γ

Con $\rho_{\max} = \rho_{\min} + \Delta k_{\max}$ y $\theta_{\max} = 0 + \Delta \theta h_{\max}$ del máximo local se describe el plano inicializador en forma polar

 $n \leftarrow [A, B, C] h \leftarrow D$ Que son valores inicializadores para aplicar Algoritmo 1

Aplicar algoritmo 1

Obtener parámetros del plano [n, h] y puntos que no pertenecen al plano encontrado Mientras max > γ

Salida Planos encontrados

 γ es un umbral, $\theta \in [0, \Delta\theta, 2\Delta\theta, ...\pi]$ y $\rho \in [\rho_{\min}, \rho_{\min} + \Delta\rho, \rho_{\min} + 2\Delta\rho, \rho_{\max}]$, *acum* es una matriz de contadores inicializada en 0, ρ es la magnitud del vector normal desde el origen a la recta, θ es el ángulo de inclinación, distStep son los saltos entre intervalos en el acumulador, n es la normal del plano encontrado con coeficientes A, B, C, D.

Una vez localizados los objetos estructurados presentes en la nube de puntos, los puntos no asociados a estos (outliers) corresponden a objetos no estructurados, los cuales son encontrados mediante la técnica explicada en la siguiente sección.

3.2. Segmentación de objetos no estructurados

La segmentación por el algoritmo de corrimiento de media (Comaniciu and Meer, 2002), MeanShift (MS), se puede definir como la descomposición del objeto a analizar en regiones homogéneas, las cuales comparten características tales como intensidad y distribución espacial.

El uso de esta técnica se basa en aplicar iterativamente el método para encontrar el máximo local o punto estacionario de la función de densidad más cercano a un punto Ξ del conjunto de datos Ξ . La mayor ventaja que presenta este algoritmo es que no es necesario el conocimiento a priori del número de grupos y además permite la forma arbitraria de estos.

El algoritmo de agrupamiento basado en MeanShift consiste en la búsqueda de modas, básicamente son dos puntos principales:

- ✓ Para los puntos del conjunto de datos se deja correr el procedimiento de búsqueda de puntos estacionarios de *Fk*. h(x) o modas.
- ✓ Deshacerse de los puntos intermedios reteniendo sólo el máximo local.

En otras palabras, el algoritmo MeanShift agrupa un conjunto de datos de dimensión d asociando a cada punto con la moda o el pico de la función de densidad de probabilidad del conjunto de datos.

Para cada punto, el algoritmo calcula la moda correspondiente y la asocia a él. En primer lugar, se define un kernel radial centrado en el punto y con ancho de banda ^h, se calcula la media de los puntos que caen dentro del área definida bajo el kernel. Después, el algoritmo desplaza la ventana o el kernel hacia la media y se va repitiendo el proceso hasta que converge, es decir, hasta que el vector calculado recursivamente se mantiene constante o hasta que su variación es inferior a un umbral determinado. En cada iteración, la ventana se desplaza hacia zonas de mayor densidad de puntos hasta que se alcance el pico, donde los datos están distribuidos de la misma manera en la ventana. El algoritmo 2 muestra el pseudocódigo del agrupamiento MeanShift. Al finalizar el algoritmo, todos los puntos asociados a una moda de valor similar son asignados a un mismo grupo.

4. RESULTADOS EXPERIMENTALES

En la sección 4.1 se muestran los resultados experimentales obtenidos por la segmentación de planos, incluyendo el suelo. El conjunto de puntos que pertenecen a los planos son quitados de la nube de puntos, por lo que la nube de puntos restantes contiene objetos no estructurados. Los objetos son segmentados por el método MeanShift, los resultados se muestran en la sección 4.2

4.1. Segmentación de objetos estructurados

La figura 1 muestra una nube de puntos típica adquirida por el sistema Lidar observada desde diferentes ángulos. La figura 2 muestra la segmentación del suelo usando el Algoritmo 1. Usando la orientación del plano del suelo, todos los puntos son rotados con respecto al suelo. Con esta rotación aseguramos que las paredes son perpendiculares al plano x - y, así al aplicar el método de Hough podemos despreciar la coordenada z para encontrar los planos, ya que la normal de los planos coincidirá con el plano x - y. De esta forma se reduce el espacio de búsqueda de la dirección de la normal a solo encontrar orientación con respecto al eje x. Aplicando el método de Hough segmentamos diferentes planos, los cuales son mostrados en la Figura 3.



Figura 1. Nube de puntos original. Vista desde diferente perspectiva y acercamiento



Figura 2. Segmentación del suelo y el resto de puntos



Figura 3. Planos de la nube de puntos

Figura 4. Nube de puntos objetos

De la nube de puntos original se extrae el suelo y las paredes, por lo que la nube de puntos restante contiene objetos no estructurados como puede verse en la figura 4. Los planos que se observan en esta figura son debido a que el número de puntos 3D que definen el plano es menor al umbral definido como parámetro V.

4.2. Segmentación de objetos no estructurados

La Figura 5 muestra los objetos segmentamos por el método de MeanShift, los puntos que pertenecen a un objeto son pintados del mismo color, los clúster de colores separados corresponden a diferentes objetos. La figura 7 muestra el mismo resultado con un zoom. En la figura se han usado colores repetidos debido a las limitantes gráficas. Si el número de puntos que corresponden a un objeto es superior a 150 puntos podemos considerar que el clúster representa un solo objeto. Este umbral elimina objetos más pequeños, así como ruido, el resultado es mostrado en la figura 6 y 8.



Figura 5. Nube de objetos segmentada por color



Figura 6. Objetos segmentada con MeanShift



Figura 7. Nube de objetos segmentada por color



Figura 8. Objetos segmentada con MeanShift

5. CONCLUSIONES Y TRABAJOS FUTUROS

Los resultados los podemos dividir en dos partes: La extracción de planos del medio ambiente y la segmentación de objetos 3D. En este trabajo, el Lidar adquirió datos en una sola posición, para trabajos futuros el Lidar se desplazará, así que los diferentes planos deberán ser unidos con el objetivo de crear un mapa tridimensional del entorno adquirido. Sin embargo, en las adquisiciones existen datos que no corresponden al entorno, tal es el caso de autos, personas, etc. Una vez segmentados los objetos estos deberán ser identificados, por lo que los puntos que corresponden a autos y personas deberán identificarse y ser eliminados del mapa construido.

7. REFERENCIAS

Brenneke Christian, Wulf Oliver, Wagner Bernardo, Using 3D Laser Range Data for SLAM in Outdoor Environments Institute for Systems Engineering, University of Hannover, Germany, 2004.

Comaniciu D. and Meer P. Mean shift: a robust approach toward feature space analysis, Pattern Analysis and Machine Intelligence IEEE Transactions on, vol. 24, no. 5, 2002.

Fukunaga, Keinosuke; Larry D. Hostetler, The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition IEEE Transactions on Information Theory (IEEE) 21 (1): 32 a 40. doi:10.1109/TIT.1975.1055330. January 1975.

Golovinskiy Aleksey, Kim G. Vladimir, Funkhouser Thomas, Shape-based Recognition of 3D Point Clouds in Urban Environments, Princeton University, 2009.

Moosmann Frank, Pink Oliver and Stiller Christoph, Segmentation of 3D Lidar Data in non-at Urban Environments using a Local Convexity Criterion Institut fur Mess- und Regelungstechnik Universitat Karlsruhe (TH), 76128 Karlsruhe, Germany, 2009.

Rabbani T., Van Den Heuvel F., and Vosselmann G, Segmentation of point clouds using smoothness constraint. In IEVM06, 2006.

Strom Johannes, Richardson Andrew, Olson Edwin, Graph-based Segmentation for Colored 3D Laser Point Clouds, Department of Computer Science and Engineering, University of Michigan, 2010.

Unnikrishnan R. and Hebert M., Robust extraction of multiple structures from non-uniformly sampled data. In IROS, volume 2, pages 132229, October 2003.

Wohlkinger Walter, Vincze Markus 3D Object Classification for Mobile Robots in Home-Environments Using Web-Data Automation and Control Institute Vienna, University of Technology, Austria, 2010.