



UNIVERSIDAD NACIONAL DE COLOMBIA

Clasificación Automática De Anomalías Asociadas Con Ausencia De Información En Superficies Tridimensionales De Objetos De Forma Libre

Pedro Sandino Atencio Ortiz

Universidad Nacional de Colombia

Facultad de Minas, Escuela de sistemas

Medellín, Colombia

2012

Clasificación Automática De Anomalías Asociadas Con Ausencia De Información En Superficies Tridimensionales De Objetos De Forma Libre

Pedro Sandino Atencio Ortiz

Tesis presentada como requisito parcial para optar al título de:
Magister en Ingeniería de Sistemas

Director:

Ph.D. JOHN WILLIAN BRANCH BEDOYA

Co-director:

M.Sc. GERMAN SÁNCHEZ TORRES

Línea de Investigación:

Visión Artificial

Grupo de Investigación:

GIDIA

Universidad Nacional de Colombia

Facultad de Minas, Escuela de Sistemas

Medellín, Colombia

2012

Resumen

En este trabajo se propone un método computacional para clasificar anomalías relacionadas con ausencia de información sobre modelos tridimensionales de forma libre. Para ello, se hizo una exploración descriptiva de las propiedades geométricas globales y locales de las anomalías y una evaluación posterior de distintos métodos de clasificación utilizados en visión artificial y aplicaciones de reconstrucción tridimensional. El método propuesto logra un nivel de clasificación cercano al 90% y un tiempo de ejecución de alrededor de 100 milisegundos. Restringir la clasificación de acuerdo a la aplicación en específico se propone como trabajo futuro.

Palabras clave: Visión artificial, reconstrucción tridimensional, corrección de huecos.

Abstract

In this work it is proposed a computational method to classify anomalies related with information absence over free-form tridimensional models. For that, it was made a descriptive exploration of global and local geometric properties of anomalies and a posterior evaluation of different classification methods widely used in artificial vision and tridimensional reconstruction applications. The proposed method achieved a classification level near to 90% and an execution time near to 100 milliseconds. Constrain classification according to the specific application is suggested as future work.

Keywords: Artificial vision, tridimensional reconstruction, hole filling.

Contenido

Lista De Figuras.....	VII
Lista De Tablas.....	X
CAPÍTULO 1	11
INTRODUCCIÓN, OBJETIVOS Y METODOLOGÍA.....	11
1.1. Motivación.....	11
1.2. Definición del problema de investigación	11
1.3. Preguntas de investigación.....	12
1.4. Hipótesis de investigación.....	12
1.5. Objetivos	13
1.5.1. General.....	13
1.5.2. Específicos	13
1.6. Alcance.....	13
1.7. Organización del documento.....	14
1.8. Difusión de Resultados.....	14
CAPÍTULO 2	15
RECONSTRUCCIÓN TRIDIMENSIONAL.....	15
2.1. Objetos de Forma Libre.....	16
2.2. Aplicaciones de la Reconstrucción 3-D.	17
2.2.1. Ingeniería inversa.	17
2.2.2. Diseño Industrial.....	17
2.2.3. Medicina.....	18
2.2.4. Inspección.....	19
2.2.5. Simulación	20
2.2.6. Entretenimiento.....	20
2.3. Adquisición de Datos Tridimensionales.....	21
2.3.1. Sistema Típico de Adquisición de Datos Tridimensionales	24
2.3.2. Imágenes de rango	24

2.4.	Registro.....	25
2.5.	Integración.....	27
2.5.1.	Ausencia de información.....	27
2.5.2.	Exceso de Información.....	28
2.5.3.	Ruido.....	29
2.6.	Ajuste de Superficies.....	30
CAPITULO 3.....		32
REVISION DE LA LITERATURA.....		32
3.1.	Corrección Global o Basada en Datos Dispersos.....	32
3.2.	Identificación y Corrección Local de Huecos.....	33
3.3.	Clasificación o selección automática de huecos a corregir.....	35
3.4.	Limitaciones Generales.....	35
CAPITULO 4.....		37
4.1.	Oclusiones.....	37
4.2.	Problemas de Registro.....	40
4.3.	Problemas Ópticos de la Superficie del Objeto.....	42
4.4.	Huecos Pertenecientes al Objeto.....	44
4.5.	Conclusiones.....	46
CAPÍTULO 5.....		48
DESCRIPCIÓN DE HUECOS EN SUPERFÍCIES 3-D.....		48
5.1.	Descripción y Representación de Curvas y Contornos.....	48
5.2.	Descriptores Locales: Geometría Diferencial de Curvas.....	50
5.2.1.	Torsión y Curvatura.....	54
5.3.	Descriptores Globales de Forma de la Curva.....	57
5.3.1.	Representación de la curva.....	58
5.3.2.	Remuestreo estructurado de los puntos de la curva.....	58
5.3.3.	Código de cadena para el conjunto estructurado de puntos.....	62
5.3.4.	Simetría y similitud de forma.....	66
5.4.	Resumen de los descriptores tratados.....	69
CAPÍTULO 6.....		71
CLASIFICACIÓN DE HUECOS EN SUPERFÍCIES 3-D.....		71

6.1. Método de Selección de Características	72
6.2. Métodos de Clasificación.....	73
6.3. Método de Medición del Error de los clasificadores.....	75
CAPÍTULO 7	76
EXPERIMENTOS Y RESULTADOS	76
7.1. Descripción del Experimento	76
7.2. Experimentos y Resultados	81
7.2.1. Análisis de los datos de trabajo	81
7.2.2. Selección de Características	83
7.2.3. Evaluación de los Clasificadores	84
7.3. Discusión del Experimento.....	86
CAPITULO 8	87
CONCLUSIONES Y TRABAJOS FUTUROS.....	87
8.1. Conclusiones.....	87
8.2. Trabajos Futuros.....	88
Bibliografía.....	89
ANEXOS.....	98

Lista De Figuras

Figura 1. Esquema secuencial de las etapas de reconstrucción 3D.....	11
Figura 2. Objeto de forma libre.	16
Figura 3. RepRap, proyecto GNU GPL para la construcción de una impresora 3D.	18
Figura 4. Ecografía 3D de un feto de 20 semanas.	19
Figura 5. Inspección 3D de una sección de un aspa de una turbina eólica. Capture3D ®.	19
Figura 6. Modelo 3D de una taza de café generado mediante un iPhone 4 y el software Trimensional®.	21
Figura 7. Taxonomía de los métodos activos de adquisición 3D.	22
Figura 8. Sensor de Rango basado en la proyección de línea Láser. Imagen tomada de [8].	23
Figura 9. Estructura de una imagen de rango. Figura tomada de [12].	25
Figura 10. Registro de vistas parciales. Imagen tomada de [12].	26
Figura 11. Pasos para la corrección de huecos en superficies.	28
Figura 12. Superficie con huecos reales.	28
Figura 13. En a) los puntos y la superficie originales P_i y S_p . b) los puntos simplificados R_i . c) superficie reconstruida con los datos simplificados S_r . d) diferencia entre la superficie original S_p y la simplificada S_r . Imagen extraída de [15].	29
Figura 14. Ajuste de superficies de una vista parcial de un modelo de la estatua de la libertad. A la izquierda, los puntos adquiridos. A la derecha el ajuste de una superficie de base radial. Nótese como todos los huecos son corregidos aunque en algunos casos la representación final no sea adecuada. Imagen extraída de [22].	31
Figura 15. Objeto prácticamente imposible de escanear mediante métodos ópticos. Imagen tomada de [46].	38
Figura 16. Representación del proceso de escaneo para un sensor óptico de luz estructurada en (de izquierda a derecha) a) una superficie plana, b) en presencia de una hendidura en la superficie y c) en presencia de una protuberancia en la superficie.	39
Figura 17. Hueco generado por auto-oclusión de la superficie.	40
Figura 18. Registro incorrecto de vistas parciales. La región al interior del círculo rojo, indica un hueco producto de este registro. Imagen tomada de [9].	41
Figura 19. Parámetro estándar para el llenado de huecos. Imagen tomada mediante el software RapidForm ®.	42
Figura 20. Huecos generados por problemas de especularidad de la superficie del objeto. Nótese que los huecos producto de problemas ópticos de la superficie del objeto son altamente irregulares debido a la distorsión del patrón proyectado.	43
Figura 21. Objetos con problemas de complejidad óptica. Imagen tomada de [28].	44
Figura 22. En la izquierda, un hueco perteneciente al objeto. En la derecha, un hueco producto de una oclusión. Nótese la diferencia en la suavidad de dichos huecos.	45

Figura 23. Similitud geométrica en huecos de objetos de forma libre.	45
Figura 24. Curvas digitales. a) curva digital de una imagen bidimensional con espaciamiento homogéneo. b) curva digital 3D con espaciamiento no homogéneo.	49
Figura 25. Descriptores de contornos de huecos en superficies 3D utilizados en este trabajo.	50
Figura 26. La longitud de arco $s(t)$ ayuda a definir los vectores tangente $T(t)$ y normal $N(t)$. Figura toma de [62].	51
Figura 27. Estimación de la curvatura mediante círculo osculador. Figura tomada de [67].	52
Figura 28. Comparación de diferentes métodos de estimación de propiedades geométricas de curvas en el espacio 3D para una curva helicoidal. Figura tomada de [59].	53
Figura 29. Comparación de diferentes métodos de estimación de propiedades geométricas de curvas en el espacio 3D para una curva hipocicloide. Figura tomada de [59].	54
Figura 30. Vectores tangente (rojo), normal(verde), binormal (azul) y plano osculador.	55
Figura 31. a) puntos originales. b) rojo: vector tangente, verde: vector normal, azul: vector binormal.	56
Figura 32. a) Torsión y b) curvatura para la curva 3D que aparece en la Figura 31.	57
Figura 33. a) vector de puntos del borde del hueco. b) vector ordenado. c) vector no ordenado. .	59
Figura 34. De izquierda a derecha a) Puntos Originales b) Puntos de la curva NURBS 75% c) Puntos de la curva NURBS 100%.	60
Figura 35. Datos estructurados en el espacio 3D.	62
Figura 36. Remuestreo estructurado mediante el algoritmo Bresenham 3D.	62
Figura 37. Código de cadena con conectividad 8.	63
Figura 38. Histograma del código de cadena.	63
Figura 39. Códigos de cadena para líneas 3D estructuradas. Imagen tomada de [75].	64
Figura 40. Código de cadena relativo.	64
Figura 41. a) Conjunto de puntos de la curva 3D. b) histograma del código de cadena absoluto de la curva.	65
Figura 42. Geometría Axial.	66
Figura 43. Simetría axial en el contorno de un hueco sobre una superficie 3D. Imagen tomada de [95].	67
Figura 44. Conjuntos de simetría.	67
Figura 45. Oblicuidad negativa y oblicuidad positiva.	68
Figura 46. Comportamiento del método de selección secuencial hacia adelante (curva), en el espacio de búsqueda (elipse).	72
Figura 47. El hiperplano o frontera debe estar tan lejos como se pueda de los datos de ambas clases.	73
Figura 48. Esquema general del algoritmo AdaBoost.	74
Figura 49. Árbol de decisión para un juego de golf.	75
Figura 50. Modelos utilizados para la evaluación del descriptor: similitud geométrica. De izquierda a derecha y de arriba a abajo, triángulo, cuadrado, rectángulo, círculo y elipse.	81
Figura 51. Clusterización de los datos para 4 clases.	82
Figura 52. Clusterización para 2 clases.	82

Figura 53. Clasificador propuesto.....	85
--	----

Lista De Tablas

Tabla 1. Características observables en los contornos de huecos en superficies 3D de objetos de forma libre.....	46
Tabla 2. Resumen de los descriptores propuestos.	69
Tabla 3. Datos de entrenamiento y validación del clasificador.....	76
Tabla 4. Subgrupos de características.	83
Tabla 5. Resultados de clasificación utilizando el primer grupo de características.	84
Tabla 6. Matriz de confusión para el clasificador SVM con el grupo 1 de características.	84
Tabla 7. Matriz de confusión para el clasificador SVM con el grupo 2 de características.	84
Tabla 8. Matriz de confusión para el clasificador AdaBoost con el grupo 1 de características.	84
Tabla 9. Matriz de confusión para el clasificador AdaBoost con el grupo 2 de características.	84
Tabla 10. Matriz de confusión para el clasificador C4.5 con el grupo 1 de características.....	84
Tabla 11. Matriz de confusión para el clasificador C4.5 con el grupo 2 de características.....	85
Tabla 12. Áreas de la curva ROC para los distintos clasificadores.	85

CAPÍTULO 1

INTRODUCCIÓN, OBJETIVOS Y METODOLOGÍA

Este capítulo tiene como objetivo presentar de forma general el contenido de la tesis para que en su posterior lectura, ya estén claros aspectos importantes tales como la motivación, la justificación y los objetivos de la misma. Al final se muestra como está organizado el resto del documento.

1.1. Motivación

El incremento constante de la capacidad de adquisición y resolución de los dispositivos de escaneo 3D y sumado a esto, la naturaleza parcial de la adquisición de los datos de la escena real, conlleva a la obtención de modelos digitales compuestos de millones de puntos y por ende mayor complejidad en el procesamiento de dichos datos. Aún con la actual precisión de estos dispositivos, los modelos digitales adquiridos pueden presentar problemas relacionados con oclusiones, zonas de difícil acceso para el escáner, reflectancia del material escaneado o registros indebidos de las vistas parciales que generan, entre otras anomalías, ausencia de información o huecos en el modelo. Debido a lo anterior, la comunidad científica del área ha desarrollado métodos capaces de identificar discontinuidades en superficies y reconstruir la zona faltante con niveles de precisión altos. Sin embargo, la mayoría de dichos métodos pueden ser enmarcados ya sea dentro de los métodos que reconstruyen de manera global la superficie y por lo tanto todos los huecos que se presenten en la misma, o métodos que identifican de manera local cada una de las anomalías pero requieren de la intervención del usuario para decidir que huecos deben ser corregidos, haciendo por lo general que el proceso sea tedioso o demande mucho tiempo. Aunque algún método [1] ha sido propuesto para solucionar el problema de la clasificación automática de huecos en superficies, este es aún muy limitado.

En este marco de ideas, la automatización del proceso de corrección de anomalías en superficies y del proceso mismo de reconstrucción tridimensional, será una tarea incompleta o muy restringida siempre y cuando no se desarrollen métodos robustos capaces de clasificar dichas anomalías como falsas o verdaderas. Se espera que el trabajo producto de esta propuesta impacte de manera relevante en la automatización del proceso de reconstrucción de superficies a través de imágenes de rango.

1.2. Definición del problema de investigación

La automatización de procesos es una tendencia natural del progreso tecnológico, ya sea para disminuir tiempos de producción o para mejorar la calidad del dicho proceso. La reconstrucción de

superficies a partir de imágenes de rango es aún un campo prolífero de investigación debido en gran parte a la necesidad evidente de mejorar y automatizar subprocesos o etapas propias del mismo. En este marco de ideas, la corrección de anomalías a nivel local relacionadas con ausencia de información o corrección de huecos (*hole filling*), como subproceso de la etapa de integración de la reconstrucción de superficies, requiere de la intervención de un usuario que seleccione las anomalías a corregir, considerando que, en una iso-superficie producto de la digitalización de un objeto de forma libre se podrán presentar discontinuidades o huecos propios de dicho objeto y no producto de oclusiones, errores en el sensado o un registro inapropiado, que no deberían ser corregidos. Gran parte de este problema radica en la posición de la comunidad académica al asumir que las superficies son continuas y cerradas de lo cual se podría inferir de manera inequívoca que las discontinuidades en las mismas son anomalías y por ende deben ser corregidas, en oposición a casos tales como las piezas industriales y arqueológicas, en las cuales se presentan de manera frecuente discontinuidades reales o pertenecientes a las mismas.

Por lo expuesto anteriormente, se hace necesario el desarrollo de métodos que permitan clasificar discontinuidades en superficies como falsas o verdaderas anomalías de manera automática o con menor intervención de usuario. Los métodos propuestos por la comunidad científica del área presentan limitaciones que mientras no sean superadas, incidirán de manera relevante en la imposibilidad de automatizar el proceso de reconstrucción de superficies.

1.3. Preguntas de investigación

- a) Presentarán los huecos generados producto de oclusiones o errores en etapas previas a la integración del modelo, alguna diferencia significativa con respecto a los huecos pertenecientes al objeto real?
- b) Será posible diferenciar la naturaleza del origen del hueco mediante el estudio de alguna propiedad de su borde?
- c) Podría brindar el contexto en el que se encuentra el hueco información acerca de la naturaleza de su origen?
- d) Que métrica estimada a partir de la “distorsión” (irregularidad) del borde del hueco permitiría clasificar de manera adecuada el mismo como una falsa anomalía?

1.4. Hipótesis de investigación

En esta propuesta, se plantea la hipótesis de que estudiando alguna característica geométrica del borde y de contorno del hueco, con el fin de estimar una medida de distorsión o de irregularidad, sería posible clasificar una discontinuidad en una superficie como falsa o verdadera. La hipótesis se sustenta en la definición de objetos de forma libre [2] en la cual se argumenta que las superficies de estos objetos son suaves y diferenciables, por lo cual se podría inferir que los bordes de los huecos y su zona circundante deben ser suaves igualmente. Por otra parte, existe la posibilidad de que en las discontinuidades producidas por errores de sensado o fallas en la adquisición, la distribución de los datos en los alrededores de las mismas se modifique con respecto a la distribución de datos del resto de la superficie. De igual manera, si el hueco o discontinuidad presente en la superficie es realmente perteneciente al objeto, la distribución de los datos alrededor del mismo presentará un comportamiento muy parecido al del resto de la superficie.

1.5. Objetivos

1.5.1. General

Proponer un método para la clasificación automática de huecos que representen anomalías relacionadas con ausencia de información en superficies 3-D de objetos de forma libre, a través de una medida de distorsión del contorno.

1.5.2. Específicos

- Realizar una clasificación relacionada con la naturaleza que genera la anomalía relacionada con ausencia de información, que permita analizar las características geométricas de cada tipo de contorno.
- Definir una métrica que aproxime la medida de distorsión en la geometría del contorno de las anomalías.
- Diseñar el método de clasificación automática de anomalías relacionadas con ausencia de información, basado en la métrica de aproximación de la distorsión del contorno.
- Validar el método propuesto para la clasificación automática de anomalías en superficies 3-D de objetos de forma libre, a partir de la estimación de una medida del error.

1.6. Alcance

El alcance del siguiente trabajo será el desarrollo de un método de clasificación de huecos, en superficies 3-D de objetos de forma libre. Las implementaciones computacionales del método, tendrán como fin único, validar la funcionalidad del mismo, ya que el objetivo de este trabajo no es desarrollar un software. La clasificación mencionada, determinará si el hueco pertenece al objeto real o si es una anomalía. Objetos de forma libre, hace referencia a la definición establecida por Campbell en[3]. De igual forma, la estructura sobre la cual se va a extraer información de la superficie tridimensional no es específica, por lo cual, se podrá trabajar ya sea con imágenes de rango, con nubes de puntos o con formatos estándar CAD.

A continuación, los resultados verificables de este trabajo:

- ❖ Conjuntos de trabajo y de prueba de superficies 3-D subdivididos en superficies con huecos reales y superficies con anomalías asociadas con ausencia de información que a su vez estarán clasificadas de acuerdo a la naturaleza del evento o causa que las generó.
- ❖ Tabla de características geométricas observables, para cada uno de los subconjuntos anteriores.
- ❖ Conjunto de ecuaciones de métrica de cada una las características de la tabla anterior.
- ❖ Resultados de la validación experimental de las métricas establecidas.

- ❖ Resultados de la experimentación realizada para obtener las características altamente diferenciables entre los subconjuntos de superficies 3-D.
- ❖ Diagrama de secuencia del método de clasificación automática de anomalías relacionadas con ausencia de información en superficies 3-D.
- ❖ Algoritmo del método de clasificación automática de anomalías relacionadas con ausencia de información en superficies 3-D.
- ❖ Resultados de la validación experimental del método realizado.

1.7. Organización del documento

Este documento de tesis ha sido organizado de la siguiente forma: En el capítulo 2 se presentan los principales conceptos sobre el proceso de reconstrucción tridimensional o reconstrucción 3D, necesarios para entender los posteriores capítulos. En el capítulo 3 se realiza un recorrido por algunos de los trabajos más importantes al respecto de reconstrucción 3D haciendo énfasis en el proceso de corrección de huecos. El capítulo 4 es una exploración de las causas y naturalezas que pueden generar huecos en las superficies digitales 3D. En el capítulo 5 se realiza una descripción de algunas de las propiedades medibles y relevantes en los contornos de los huecos en superficies digitales 3D. En el capítulo 6 se describen y comparan algunos de los métodos de clasificación comúnmente utilizados en la literatura, haciendo uso de las características y métricas definidas en el capítulo anterior. En el capítulo 7 se presentan los experimentos y resultados obtenidos en este trabajo. Por último, en el capítulo 8 se presentan las conclusiones y trabajos futuros.

1.8. Difusión de Resultados

A continuación se presentan las publicaciones en revistas científicas y en congresos, obtenidas al momento, las cuales han permitido la divulgación de este trabajo:

- Remuestreo Estructurado de Contornos de Huecos en Superficies 3D de Objetos de Forma Libre Utilizando Bresenham. Revista DYNA. Universidad Nacional de Colombia Sede Medellín. ISSN 0012-7353, Febrero 2012, Volumen 79, Edición 171, pp. 191 - 198. Autores: John Branch, Germán Sánchez y Pedro Atencio.
- A Metric for Automatic Hole Characterization. Proceedings of the 19th International Meshing Roundtable. October 3-6 2010, Chattanooga, Tennessee, USA. Springer-Verlag, pp. 195-208. Autores: John Branch, Germán Sánchez, Pedro Atencio.
- Método Automático para el Cálculo de Códigos de Cadena 3D sobre Contornos de Huecos en Superficies Tridimensionales de Objetos de Forma Libre. Simposio de Tratamiento de Señales, Imágenes y Visión Artificial (STSIVA) 2012. Trabajo en evaluación. Autores: John Branch, Pedro Atencio.

CAPÍTULO 2

RECONSTRUCCIÓN TRIDIMENSIONAL

La reconstrucción tridimensional (3-D) no es un proceso trivial, debido al amplio conjunto de subprocesos y etapas necesario para realizarla. Estos procesos van desde adquirir un conjunto discreto de puntos digitales 3-D del objeto real mediante algún sistema electrónico de sensado, hasta generar un modelo digital que aproxime lo mejor posible la superficie del objeto original, preservando a su vez, la forma, textura y volumen de dicho objeto.

Las etapas implícitas en este proceso son bien conocidas [4] y ampliamente abordadas en la literatura. Estas etapas son conocidas como: *Adquisición*, *Registro*, *Integración* y *Ajuste*. En la **¡Error! No se encuentra el origen de la referencia.** se pueden observar las etapas del proceso de RDS (*Reconstrucción De Superficies*).



Figura 1. Esquema secuencial de las etapas de reconstrucción 3D.

La etapa de adquisición consiste en la captura de datos de la superficie del objeto mediante algún sistema de sensado, del cual se obtienen un conjunto de vistas parciales (*imágenes de rango*) correspondientes a zonas del objeto vistas desde una posición específica del sensor. Una vez obtenidas las vistas parciales del objeto, es necesario dar un sistema de coordenadas común definidas como un conjunto finito de transformaciones espaciales (*escalado, rotación y traslación*). El anterior proceso es conocido como registro de vistas parciales. Aunque el registro se encarga de “juntar” las diferentes piezas que conforman nuestro objeto, es necesario corregir múltiples anomalías inherentes a la naturaleza del proceso de adquisición y de registro, específicamente la ausencia de información visible como huecos en el modelo, el ruido en los datos adquiridos y por último la redundancia de información generada cuando un grupo de vistas tienen datos de una misma zona del objeto. El proceso anterior es conocido como integración. Por último, la etapa de ajuste de superficies tiene como fin ajustar un modelo computacional sobre los datos integrados, por ejemplo una malla triangular. Además, puede ser necesario simplificar el modelo final dependiendo de la aplicación (*videojuegos, simulación, ingeniería inversa, etc.*) de dicho modelo. Esto también es realizado dentro de la etapa de ajuste.

En este capítulo serán abordados los principales tópicos sobre el proceso de reconstrucción 3D, necesarios para la comprensión de los posteriores capítulos. El mismo está organizado de la

siguiente manera: En la sección 2.1, se presenta la definición de objetos de forma libre. En la sección 2.2, se muestran algunas de las aplicaciones comunes de este proceso. En la sección 2.3, se describe el proceso de adquisición de datos 3D. En la sección 2.4, se describe la etapa de registro. En la sección 2.5, se describe la etapa de integración y finalmente en la sección 2.6, se describe la etapa de ajuste de superficies.

2.1. Objetos de Forma Libre.

Según Besl[2]: *“Una superficie de forma libre tiene una normal de superficie bien definida que es casi continua por completo, excepto en los vértices, bordes y cúspides”*. Los objetos de forma libre tienen una definición por lo general intuitiva. Otros nombres que denotan la misma característica son objetos esculpidos, de flujo libre, de partes suaves, entre otros. Un ejemplo de este tipo de objetos puede observarse en la Figura 2.

Una definición un poco más clara se encuentra en Campbell [3]: *“de forma libre, es una caracterización general de un objeto cuyas superficies no son de una clase más fácil de reconocer como, planar y/o superficie cuadrática natural”*.



Figura 2. Objeto de forma libre.

Algunos autores como Besl, Dorai y Myers [2][5][4] concuerdan en ejemplos tales como esculturas, cuerpos de carro, órganos vitales, rostros humanos, entre otros. La importancia de este tipo de objetos radica en que son elemento de interés de diferentes áreas de aplicación entre las que se encuentran la medicina, el diseño industrial y a visualización científica.

Debido a lo anterior, el conjunto conformado por estos objetos constituye la mayor parte de los objetos de interés en diversas áreas de la ciencia, y por esta razón las comunidades científicas de la computación han orientado gran parte de su trabajo al estudio del proceso de reconstrucción 3D a estos objetos.

2.2. Aplicaciones de la Reconstrucción 3-D.

El constante aumento de la capacidad de cómputo de las herramientas tecnológicas y el decremento en igual proporción de su costo y tamaño, ha permitido que esta tecnología se encuentre inmersa en un sinfín de actividades en la sociedad, las cuales van desde la ingeniería, el entretenimiento y la medicina hasta la química y la física teórica. La reconstrucción 3D cada vez se torna más como una tecnología ubicua que impacta directamente en la forma de vivir de las personas. A continuación algunas de las aplicaciones principales de la reconstrucción 3D.

2.2.1. Ingeniería inversa.

El objetivo de la RDS en ingeniería inversa es obtener modelos CAD a partir de objetos físicos. Es muy frecuente en la industria, trabajar con piezas creadas sin programa alguno de diseño asistido. Incluso, se pueden tener piezas de máquinas antiguas de las cuales no se cuenta siquiera con planos físicos de construcción. Por esta razón, la RDS brinda la oportunidad de generar un modelo digital a partir del escaneado del objeto original, ya sea para almacenar y tener registro de dicho objeto o para la realización de alguna mejora sobre el mismo.

2.2.2. Diseño Industrial

Si bien es cierto que los programas actuales de diseño asistido por computador, presentan muchas facilidades para trabajar objetos tridimensionales, diseñar de manera completa un modelo tridimensional utilizando los dispositivos comunes de entrada como el Mouse o tabletas digitalizadoras, puede tornarse en una tarea tediosa debido a la naturaleza bidimensional de dichos dispositivos de captura. Por esta razón es común en la industria, partir de un objeto modelado por un escultor y luego digitalizar dicho objeto para generar un modelo digital sobre el cual trabajar. Por otra parte, el advenimiento reciente de la tecnología de impresión tridimensional permite crear prototipos funcionales de forma rápida que parten de un modelo digital CAD que es enviado a una impresora 3D, la cual genera un modelo físico, comúnmente en plástico. Numerosos proyectos de carácter libre para crear impresoras 3D pueden encontrarse en la web (Figura 3). De igual forma, existen empresas especialistas en estos dispositivos, como es el caso de 3dsystems y Stratasys, entre otras.

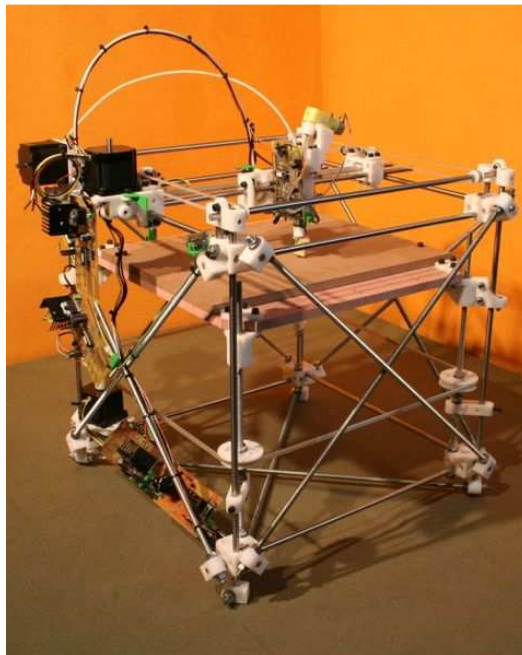


Figura 3. RepRap, proyecto GNU GPL para la construcción de una impresora 3D.

2.2.3. Medicina

Cada vez es más común utilizar las ventajas de la información tridimensional en la medicina sobreprocedimientos trabajados tiempo atrás de manera bidimensional. Algunos casos son la ecografiatridimensional (Figura 4), tomografía computarizada 3D y el diseño de prótesis personalizadas, entre otros. Avances recientes en simulación de operaciones permiten trabajar con modelos 3D del cuerpo humano para entrenar estudiantes de medicina sin necesidad de utilizar tejido orgánico real, lo cual constituye un gran aporte al área de la medicina. Por otra parte, un gran número de e modelos digitales 3D del cuerpo humano que se pueden encontrar en la web, permitiendo una mejor comprensión de la anatomía humana a estudiantes de medicina y al público en general.



Figura 4. Ecografía 3D de un feto de 20 semanas.

2.2.4. Inspección

La inspección de objetos basada en el análisis de imágenes bidimensionales mediante visión por computador, es una tecnología con mucho recorrido en la industria, así Batchelor y Whelan expresan que: *“Los sistemas de visión de máquina para la industria recibieron atención seria en los mediados de 1970, aunque la propuesta de utilizar sistemas de video para la inspección industrial fue hecha en primera instancia en 1930”*[6].

Sin embargo, las imágenes bidimensionales no nos brindan directamente información de volumen y forma del objeto, por lo cual, la RDS se presenta como una alternativa muy factible en la inspección industrial.

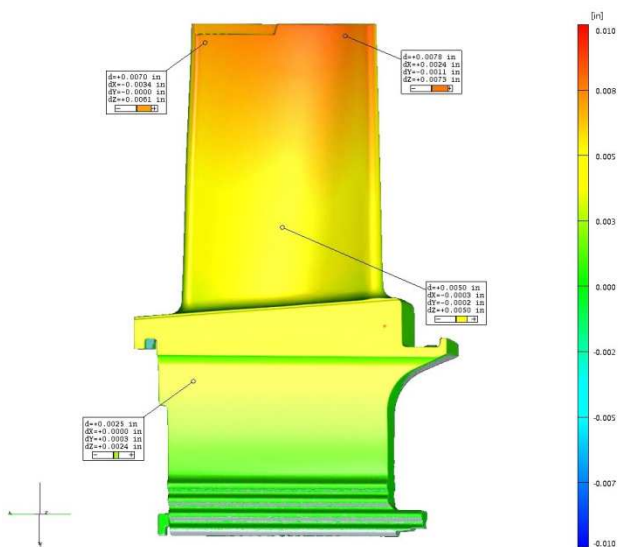


Figura 5. Inspección 3D de una sección de un asa de una turbina eólica. Capture3D®.

Inicialmente, este proceso tenía como limitante el sistema de sensado 3D que presentaba muchas limitaciones tales como tamaño y capacidad de cómputo, sin embargo, en la actualidad existen sistemas capaces de capturar información 3D a velocidades industriales y de trabajar en espacios reducidos. En la Figura 5 se observa un modelo 3D de un aspa de una turbina eólica sometido a un proceso de inspección para la detección puntos críticos.

2.2.5. Simulación

Los modelos digitales obtenidos a través de la RDS de objetos físicos, pueden ser utilizados para estimar propiedades del mismo mediante simulaciones físicas, por ejemplo, el calentamiento de la superficie de un traspordador espacial entrando a la atmósfera o la aerodinámica de un nuevo avión.

2.2.6. Entretenimiento

De acuerdo con D'Apuzzoen [7], la RDS tiene un campo acción muy fuerte en la industria del cine y del videojuego, en las que se necesitan modelos cada vez más realistas de personajes y mundos virtuales. Estos modelos son tomados del mundo real, ya sea, a través del escaneo de objetos esculpidos o de rostros de actores. El impacto de estas dos industrias en la RDS toma su mayor relevancia en el desarrollo de sistemas capaces de digitalizar información tridimensional en tiempo real.

Por otra parte, la reciente llegada al mercado de interfaces 3D para el control de juegos como el caso de MS Kinect, permitió el uso de la tecnología de reconstrucción 3D a bajo costo en una amplia gama de aplicaciones tanto comerciales, investigativas, así como proyectos libres, elevando el estado del arte de este tema a escalas difícilmente pronosticables hace unos pocos años. De igual forma, la creciente demanda de teléfonos inteligentes en el mercado ha permitido el desarrollo de aplicaciones que utilizan la RDS en dispositivos móviles (Figura 6).



Figura 6. Modelo 3D de una taza de café generado mediante un iPhone 4 y el software Trimensional®.

2.3. Adquisición de Datos Tridimensionales.

Los métodos de adquisición de información tridimensional han evolucionado y adquirido enorme robustez, precisión y aplicación en diferentes sectores del conocimiento. Esta tecnología ha superado la etapa de experimentación en laboratorios de la década de 1980 y de numerosas aplicaciones y demostraciones de los 90...Sin embargo, estos sistemas están en su periodo de infancia comparado con su contraparte bidimensional [8]. La evolución en el desarrollo de esta tecnología se debe en su mayor parte a los avances en las áreas de la física y la electrónica, en lo relacionado con el desarrollo de nuevos componentes, nuevas tecnologías y nuevos dispositivos más veloces y más precisos [9]. Así mismo, Curless considera que los dispositivos y métodos de adquisición existentes, pueden ser agrupados en dos grandes categorías: de sensado activo y de sensado pasivo [10]. El sensado pasivo es aquel que no tiene interacción con el objeto de estudio, por ejemplo, extracción de la forma a través del estudio de las sombras, la visión estereoscópica y flujo óptico. Por el contrario, los métodos de sensado activo hacen contacto directo sobre el objeto o proyectan algún tipo de energía sobre este. Aunque la naturaleza no invasiva de los métodos pasivos pudiera tomarse como una gran ventaja con respecto a los activos, el nivel de precisión y cantidad de datos alcanzado con los primeros, no alcanza los niveles requeridos por muchas aplicaciones. Debido a lo anterior, los métodos usados con mayor frecuencia en la industria y diversas comunidades científicas, son los activos. Una taxonomía de dichos métodos es propuesta por Brian Curless en [10], la cual se puede observar en Figura 7.

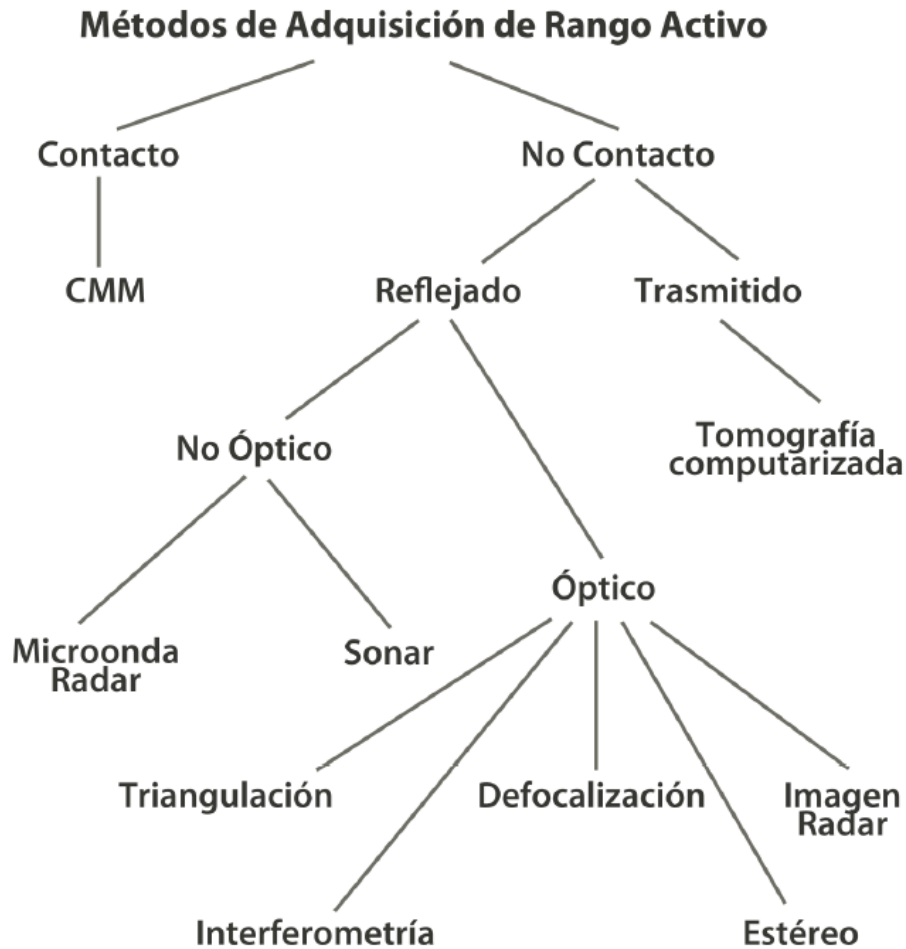


Figura 7. Taxonomía de los métodos activos de adquisición 3D.

La primera aproximación a la captura de datos tridimensionales no es reciente, ya que, anteriormente un operario tomaba medidas del objeto mediante alguna herramienta como el pie de rey y posteriormente las unificaba en un modelo completo del objeto con sus medidas en las tres dimensiones. Un primer aporte a la automatización de este proceso nace con la aparición de las primeras CMM o máquinas de medición por coordenadas referenciadas por John Branch en [9]. Estas máquinas son extremadamente precisas aunque presentan algunas limitantes como baja velocidad de captura, requieren de un operador humano, deben tocar la superficie (invasiva) y por lo general son difíciles de manipular. La llegada del Silicio a la electrónica a mediados de 1954 abrió las puertas a la era digital y a mediados de 1970 y principio de 1980 se lograron desarrollar dispositivos electro-ópticos que junto con la introducción del microprocesador, amplió el panorama de la adquisición de datos tridimensionales con la creación de los primeros sensores de rango basados en el principio de triangulación. Los primeros dispositivos desarrollados se basaban en el uso de fotodiodos de efecto lateral LEP y cámaras vidicon con los cuales se utilizaba información de la velocidad de la luz reflejada sobre un objeto para estimar distancias; la naturaleza de dichos sensores era análoga. No fue hasta la llegada del CCD o dispositivo de carga acoplada que se realizaron las primeras aplicaciones robustas de digitalización tridimensional. Los sensores de rango comúnmente utilizados, basan su funcionamiento en alguno de estos tres principios: dispositivos de punto a punto, de línea (Slit Scanner) y de superficie. La aplicación del Láser para

estas técnicas fue más que conveniente debido a que el Láser proyecta un patrón coherente, preciso y fácil de segmentar en una imagen. Los dispositivos punto a punto se basan en estimar el tiempo de retorno de un rayo Láser sobre un objeto para estimar la distancia a la que se encuentra. Este método es todavía muy aplicado en las ingenierías de construcción, para estimar alturas y comportamiento de superficies por ejemplo. La proyección de línea Láser sobre el objeto de estudio es sin duda alguna el método más aplicado de adquisición tridimensional en diversas áreas del conocimiento, por facilidades de implementación, sencillez del procesamiento de la imagen, costos de los dispositivos, entre otros. Su principio radica en el barrido de una línea Láser sobre el objeto y la captura simultánea en imágenes de dicha línea. Luego mediante algoritmos de procesamiento de imágenes se estiman valores de profundidad a través de la curvatura de la línea en la imagen (Figura 8).

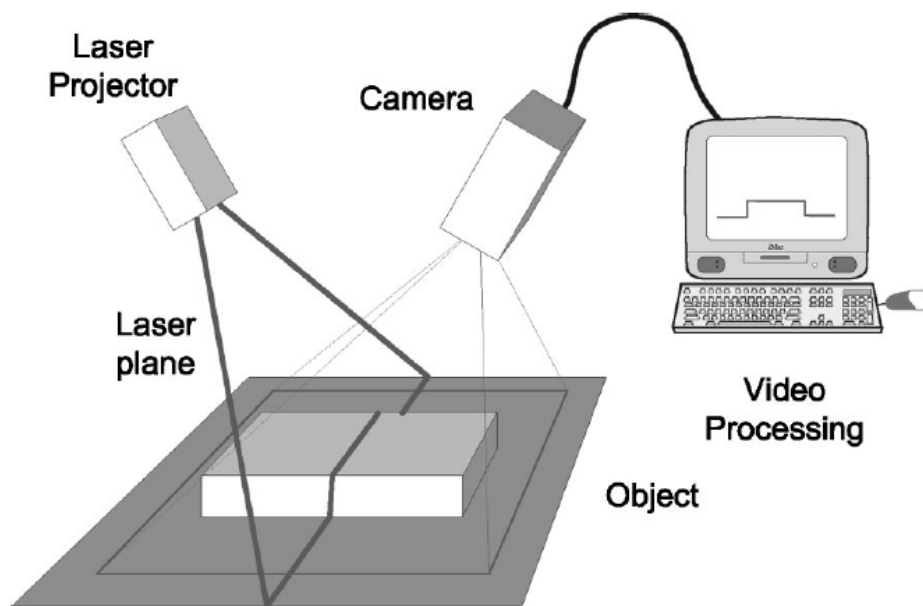


Figura 8. Sensor de Rango basado en la proyección de línea Láser. Imagen tomada de[8].

La proyección de superficies Láser es un subconjunto de los métodos basados en la proyección de patrones sobre la superficie, en los cuales se utiliza un patrón de luz que se proyecta sobre el objeto a digitalizar, a la vez que se captura la imagen de la escena mediante un dispositivo CCD. Debido a que el patrón que se proyecta es conocido, la evaluación de la deformación del patrón en la imagen, indica la forma de la superficie del objeto real. Este método tiene como ventaja que presenta la mayor velocidad de los métodos de digitalización tridimensional, ya que, en una sola captura o una sola imagen, se puede obtener la información tridimensional de una superficie y por esta razón es muy utilizado en la industria en inspección de piezas. Sin embargo, la precisión de los datos capturados depende directamente de la resolución del patrón proyectado. Watanabe, Komuro e Ishikawa [11] realizan una aplicación en la cual se aplica la proyección de una malla Láser para la captura en tiempo real de la deformación de los objetos.

2.3.1. Sistema Típico de Adquisición de Datos Tridimensionales

Por lo general, un sensor de rango activo y de no contacto, consta de un proyector ya sea de luz coherente (Láser) o luz no coherente, una cámara y una base móvil para el objeto o un montaje móvil para el proyector de luz y la cámara. El sistema de adquisición de imágenes (regularmente una cámara de video) permite adquirir secuencias de imágenes mientras que se realiza el escaneo. Luego, mediante sistemas de procesamiento de imágenes se segmenta la información de interés de cada una de las imágenes capturadas con el fin de extraer ya sea la línea Láser, el punto Láser o el patrón proyectado. Por último, se comprime dicha información extraída, mediante una estructura estándar llamada imagen de rango, la cual servirá para calcular posteriormente la información de profundidad o la tercera dimensión del objeto escaneado.

2.3.2. Imágenes de rango

Los sensores de rango activos y de no contacto (Figura 7) pueden convertir las imágenes de intensidad tomadas mediante la cámara durante el proceso de escaneo, en una estructura datos más compacta llamada *imagen de rango*, también conocida como *mapa de rango*, *datos de rango*, *mapa de profundidad* o *imagen de profundidad* [12]. Las imágenes de rango se almacenan en una matriz de $m \times n$ donde m es el número de filas de las imágenes o el alto de la resolución (todas las imágenes de un escaneo parcial, tienen la misma resolución) y n es el número de imágenes que se tomaron. El valor que se almacena en cada posición es el número de columna de la imagen n en el que la fila m toma una valor diferente de cero (Figura 9).

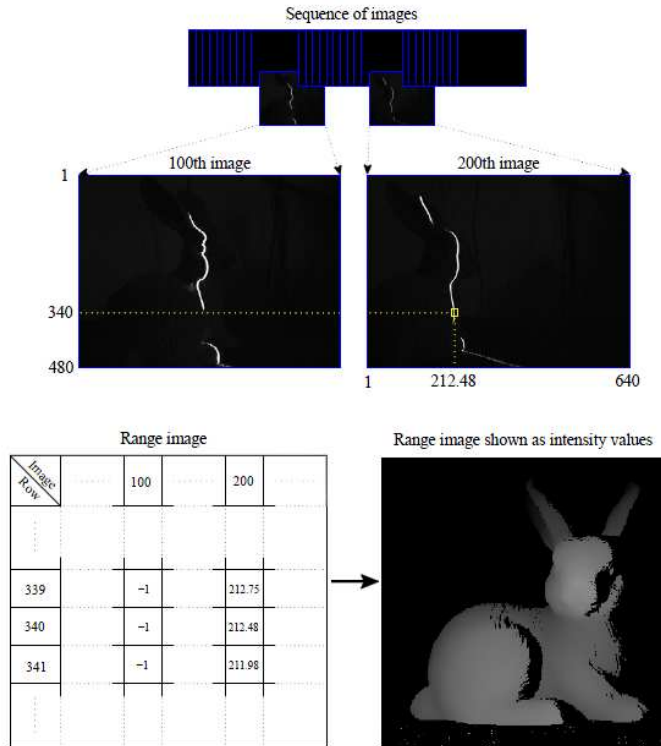


Figura 9. Estructura de una imagen de rango. Figura tomada de [12].

Debido a que una imagen de rango almacena los valores en una matriz, ciertos datos como densidad de los datos adquiridos y discontinuidades en la superficie pueden ser detectados. Esta propiedad es de gran utilidad para reconstruir superficies tridimensionales.

2.4. Registro.

Cuando se trabaja con imágenes de rango y cada una de estas representa la información de una vista parcial del objeto, tomadas en diferentes tiempos, diferentes posiciones con respecto al objeto o diferentes cámaras, se deben encontrar las transformaciones (rotación, traslación y escalamiento) que permitan unificar dichas vistas en un solo marco espacial de referencia. Según Blais Gerard [13], el problema del registro se puede definir como:

“Dadas N vistas de un objeto en una escena, cada una describiendo la estructura 3D del objeto como se ve desde un particular punto de vista, deseamos encontrar N transformaciones rígidas $T_1, T_2, T_3, \dots, T_N$, que especifiquen las verdaderas posiciones del dispositivo de escaneo con respecto a un único marco de referencia (escogido arbitrariamente y usualmente el cuadro de alguna de las vistas)”.

En resonancia con esta definición, la tarea de encontrar dichas transformaciones espaciales, requiere encontrar que puntos de o partes de una vista parcial corresponden con los puntos o partes de una o algunas de las vistas parciales. Esta situación es conocida como el problema de

correspondencia y constituye la etapa de mayor consumo de tiempo de ejecución del algoritmo [9]. Los métodos para registrar vistas parciales de un objeto pueden ser clasificados en dos categorías principales: *Sistemas de Ciclo Abierto (Open Loop Systems)* o *Sistemas de Ciclo Cerrado (Close Loop Systems)* [13]. Los primeros, asumen que si el sistema de adquisición está precisamente calibrado y se pueden obtener las diferentes transformaciones de traslación y rotación de dicho sistema durante el proceso de adquisición, se pueden utilizar estas transformaciones de manera fiable para registrar las vistas parciales. Es común utilizar este tipo de registro en sistemas de adquisición de caja cerrada. Por otra parte, los métodos de registro de ciclo cerrado, se basan en información presente en las vistas parciales e información del dispositivo de adquisición para determinar las transformaciones espaciales necesarias, por lo que el proceso se vuelve iterativo al actualizar y refinar de manera constante los parámetros de las transformaciones hasta que las vistas quedan debidamente registradas. Usualmente, los últimos métodos requieren de la intervención del usuario para seleccionar los puntos o regiones de interés en las imágenes de rango que permitirán al sistema ir optimizando o disminuyendo el error de las distancias entre las imágenes de rango. Cabe aclarar que estas regiones o puntos de interés están conformados por información común entre vistas parciales, es decir, regiones que se superpongan entre imágenes de rango. En la Figura 10 se ilustra el proceso de registro para un conjunto de dos vistas parciales mediante el algoritmo ICP (*Iterative Closest Point*).

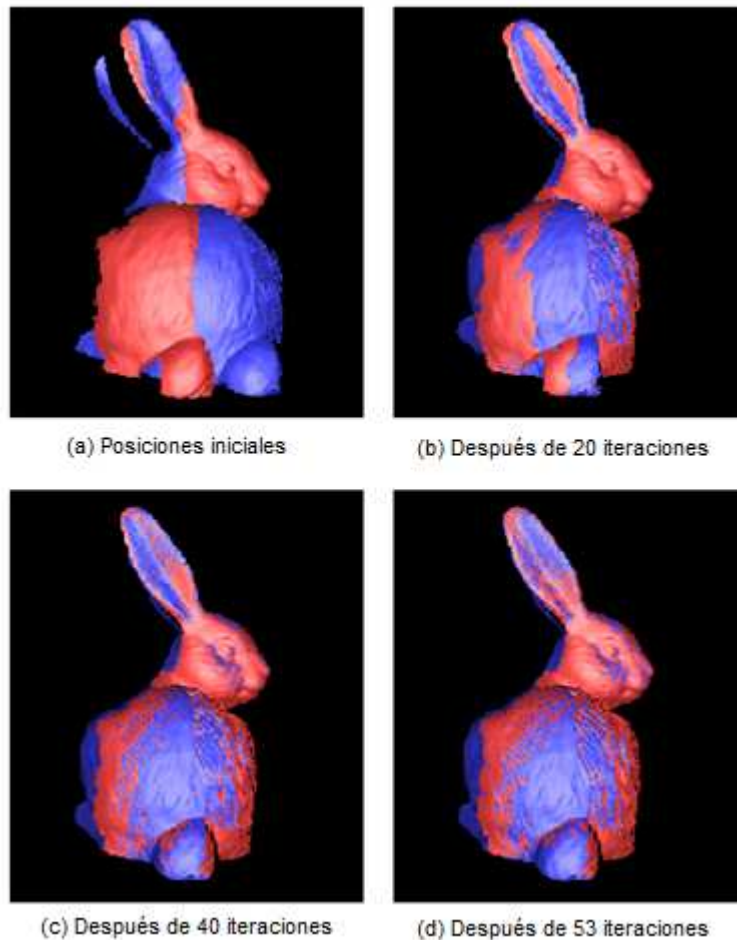


Figura 10. Registro de vistas parciales. Imagen tomada de [12].

2.5. Integración.

Una vez se obtiene el conjunto de vistas parciales alineadas o registradas, es necesario obtener un único conjunto de datos del objeto 3D. Para esto se requiere integrar dichas vistas. Curless en [10] define la etapa de integración de la siguiente manera:

“Dado un conjunto p de imágenes alineadas (registradas) y con ruido $\hat{f}_1, \hat{f}_2, \hat{f}_3, \dots, \hat{f}_n$, encontrar la superficie que mejor aproxime los puntos contenidos en las imágenes de rango”.

Esta etapa según su definición, puede ser asumida como el ajuste de superficies (sección 2.6). De hecho, algunos métodos de RDS propuestos obvian esta etapa como veremos posteriormente. Una definición un poco más detallada del objetivo de la integración es la propuesta por Branch en [9]:

“La integración es una etapa del proceso de reconstrucción 3-D, que consiste en obtener una representación computacional suave y continua del objeto. Una vez se han registrado las diferentes imágenes de rango utilizadas para el proceso de reconstrucción, se tiene un conjunto de datos en el espacio 3-D cuya conectividad y relación aún no está muy bien definida, debido a que los datos registrados en la etapa previa generan superficies parciales que poseen huecos, intersecciones poligonales, falsos bordes, entre otros.”

Esta definición hace énfasis en que en esta etapa, se deben corregir defectos comunes en las imágenes de rango, propios de la naturaleza de la adquisición de los datos y del proceso mismo de registro de vistas parciales como pueden ser: huecos producto de la ausencia de información, ruido en los datos y exceso de información. A continuación, una breve descripción de cada uno de estos defectos o anomalías.

2.5.1. Ausencia de información.

Problemas tales como oclusiones, zonas de limitado acceso para el dispositivo de adquisición, número limitado de vistas parciales, registros indebidos de vistas parciales, propiedades reflectivas del material del objeto escaneado, entre otros, pueden conllevar a que falte información en el conjunto de datos registrados, lo cual, a simple vista se presenta como huecos en el modelo. El principal objetivo de los métodos de corrección de huecos en superficies es reconstruir los datos del segmento de información faltante, que generalmente se infiere a través de un análisis del comportamiento del resto de información (interpolación). Según Liepa en [14]:

“Un algoritmo satisfactorio de llenado de huecos debería: ser automático, ejecutarse en tiempo razonable, rellenar el hueco de manera que la superficie reconstruida sea difícil de distinguir y ser capaz de corregir huecos arbitrarios en superficies arbitrarias (forma libre)”.



Figura 11. Pasos para la corrección de huecos en superficies.

Los algoritmos de corrección de huecos constan de tres pasos principales: *identificación*, *selección* y *rellenado* (Figura 11). La identificación de huecos consiste en encontrar sucesiones cerradas de aristas límites [14], basado en la idea de que una discontinuidad en la superficie puede ser definida como un conjunto cerrado de aristas límites. Una arista límite se define como una arista de un triángulo que no es compartida. Luego de haber detectado los huecos en la superficie, se debe seleccionar cuales de estos serán corregidos, teniendo en cuenta que en un objeto de forma libre puede haber huecos realmente pertenecientes al mismo (Figura 12).

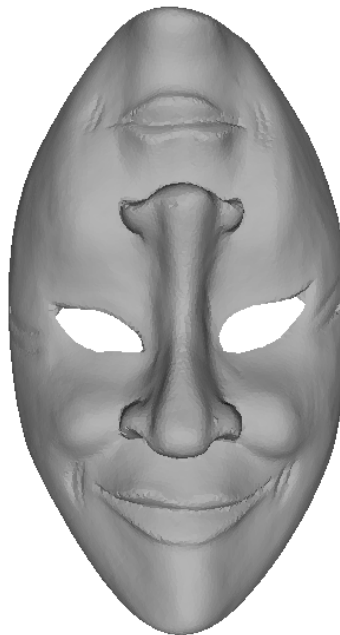


Figura 12. Superficie con huecos reales.

2.5.2. Exceso de Información.

Debido a la actual capacidad de resolución de los dispositivos de captura comerciales, se alcanzan en ocasiones modelos digitales compuestos por millones de puntos [15]. Esta situación conlleva a una mayor complejidad en el procesamiento de los datos y por ende en un mayor gasto

computacional. Aunque esta situación se presenta como una anomalía, la cantidad de datos necesaria varía según la aplicación que se esté trabajando, por lo cual, el exceso de información en algunos casos representará un problema y en otros será necesario. Por ejemplo, en la industria del entretenimiento, específicamente en cine y videojuegos, se requieren modelos digitales muy simplificados pero que conlleven a una representación visual agradable. Por el contrario, simulaciones físicas, se desea contar con la mayoría de datos posibles sobre el objeto de estudio con el fin de que dicha simulación se aproxime lo mayor posible a la realidad. Sin embargo, debido a la naturaleza parcial de la adquisición de los datos tridimensionales, habrá regiones comunes entre las vistas parciales que al ser registradas, generarán regiones con redundancia de información, o una densidad de datos elevada en dichas zonas. Por esta razón en el proceso de RDS incluye la simplificación de datos en de la superficie [16]. Pauly et ál.[17] define el problema de la simplificación de puntos como:

“Dada una superficie S definida por un conjunto de puntos $P_n = p_1, p_2, p_3, \dots, p_n$, simplificar P_n es obtener un conjunto de puntos $P_m = p_1, p_2, p_3, \dots, p_m$ ($m < n$), tal que la distancia \mathcal{E} de la correspondiente superficie S' definida por P_m a la superficie original S sea mínima”.

Esta definición puede ser observada gráficamente en la Figura 13.

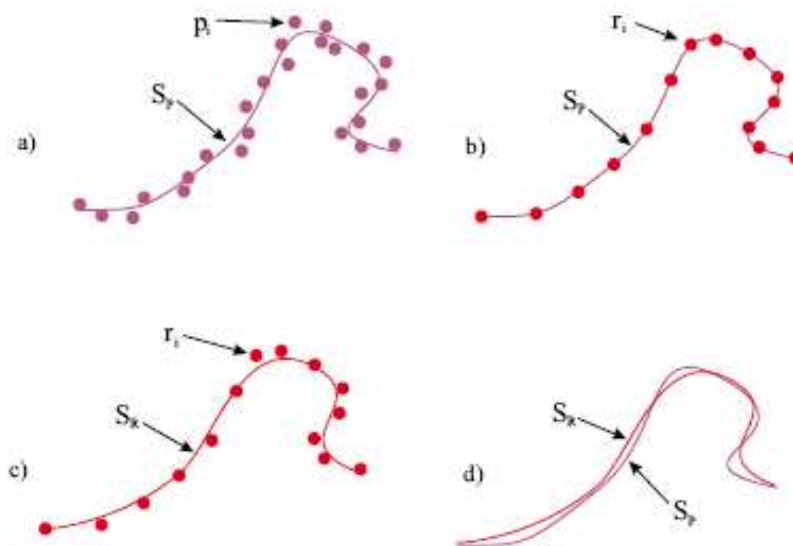


Figura 13. En a) los puntos y la superficie originales P_i y S_p . b) los puntos simplificados R_i . c) superficie reconstruida con los datos simplificados S_r . d) diferencia entre la superficie original S_p y la simplificada S_r . Imagen extraída de [15].

2.5.3. Ruido.

El ruido puede ser definido como información no deseada que contamina la imagen. El ruido aparece en las imágenes, procedente de una gran variedad de fuentes [18]. Aunque la información

del modelo digital obtenido a partir de la RDS es de carácter tridimensional, debemos recordar que los sensores de rango en su mayoría, se basan en la captura de imágenes bidimensionales desde diferentes puntos de vista del objeto, por lo cual, el ruido definido anteriormente y presente en dichas imágenes, influirá de manera directa sobre el modelo digital final. Esta anomalía siempre estará presente en el proceso de RDS como le expresa Sunet.al[19]:

“El ruido es ubicuo en los datos medidos, necesariamente los escáneres 3D contienen algún tipo de ruido”.

El conjunto de métodos y técnicas de eliminación del ruido, por lo general realizan un suavizado de la superficie o *surface smoothing*, por lo que, el tratamiento del ruido busca entre otros objetivos el diseño de superficies curvadas de alta calidad y la generación de mallas más regulares [20].

2.6. Ajuste de Superficies.

En esta última etapa del proceso de RDS se busca encontrar una superficie matemática que se ajuste al conjunto de puntos o muestras discretas producto de la adquisición, registro e integración de las vistas parciales, es decir, el conjunto de datos del objeto completo y libre de anomalías. Sin embargo, en algunos métodos de RDS, se realiza el ajuste de superficies directamente sobre los datos registrados suponiendo que este proceso implícitamente corrige anomalías como se había comentado anteriormente. De manera formal, el ajuste de superficies puede ser definido según Hrádek [21] como:

“Sea S una superficie bidimensional perteneciente a un objeto O , incluida en el espacio tridimensional Euclídiano R^3 , y el conjunto de puntos $p_i \in P \subset R^3, i = 1, 2, 3, \dots, n$, un conjunto de muestras discretas de la superficie S . El ajuste consiste entonces en encontrar una superficie S' que aproxime S usando los datos del conjunto P , siendo la superficie reconstruida S' topológicamente equivalente a la superficie S del objeto original”.

Según Hádrek[21], Los métodos de ajuste de superficies pueden dividirse en dos grupos: métodos de aproximación o métodos de interpolación. En los primeros, la superficie se reconstruye cercana a los puntos. Los segundo, ajustan la superficie para que pase por todos los puntos. Los métodos de aproximación son comúnmente utilizados ya que al no ajustarse de manera exacta a los puntos, genera buenos resultados visuales cuando los datos tienen ruido y huecos. La comunidad del área ha desarrollado una cantidad considerable de trabajos en el tema de ajuste de superficies y de igual manera muchos métodos de representación de superficies entre los que se destacan: *mallas triangulares, las funciones de base radial* (Figura 14) y *las superficies NURBS*.



Figura 14. Ajuste de superficies de una vista parcial de un modelo de la estatua de la libertad. A la izquierda, los puntos adquiridos. A la derecha el ajuste de una superficie de base radial. Nótese como todos los huecos son corregidos aunque en algunos casos la representación final no sea adecuada. Imagen extraída de [22].

CAPITULO 3

REVISION DE LA LITERATURA

A continuación se realiza una revisión de algunos de los trabajos más relevantes desarrollados hasta el momento con respecto a la corrección de anomalías relacionadas con ausencia de información, en el proceso de reconstrucción de superficies. Los métodos propuestos por la comunidad científica para dicha tarea pueden ser agrupados según dos criterios: *Corrección global* y *Corrección Local*. En la primera clasificación tenemos los trabajos que realizan la corrección de anomalías en la superficie de manera global al ajustar una superficie matemática al conjunto de datos de las vistas registradas, por lo cual, la corrección de huecos pasa a ser una propiedad de dichos métodos más que un objetivo específico. El segundo grupo es conformado por los métodos que reparan de manera localizada los huecos y que constan de tres etapas: *identificación de los huecos*, *selección de los huecos a reparar* y *rellenado de huecos*. Sin embargo en algunos trabajos, la etapa de selección no se realiza y por lo tanto las discontinuidades son reparadas de manera indiscriminada.

3.1. Corrección Global o Basada en Datos Dispersos

En esta primera clasificación, encontramos muchos de los trabajos clásicos de reconstrucción de superficie, realizados desde mitad de los 90`s hasta principio del 2000 ([23],[24], [25], [22], [26], [27]) y algunos muy recientes ([28], [10], [29]). La corrección global del objeto, considera las superficies como continuas o cerradas, por lo cual no contempla fase alguna de detección ni clasificación de huecos, por lo tanto al repararse por completo dicha superficie, el llenado de huecos se torna en una propiedad más no en un objetivo de estos métodos. Por otra parte, presentan como gran desventaja, que producen resultados incorrectos al reparar superficies complejas en las cuales se presenten bordes o falsas discontinuidades, ya sea debido a que pueden extender los límites de la superficie o porque se reparan huecos que realmente pertenecen al objeto y por ende no deben ser corregidos. Un trabajo de referencia en lo que a reconstrucción de superficies se refiere, es el realizado por Hughes Hoppe ([23], [24]), en el cual el autor presenta un método para representar superficies desde puntos arbitrarios, partiendo de la construcción inicial de una malla triangular con una alta densidad de datos mediante la cual se estima la categoría topológica de la superficie y por último la simplificación de la malla a través de una técnica denominada mallas progresivas (*progressive meshes*). El conjunto de datos inicial se considera como una representación discreta de una superficie de naturaleza continua, por lo cual la malla generada no contendrá huecos. La reconstrucción de superficies basada en funciones de base radial (FBR) es comúnmente utilizada en los métodos corrección global. Por ejemplo, Carr et. al ([22], [25]) propone un método para la reconstrucción de superficies a partir de nubes de puntos. Para ello estima planos parciales utilizando subconjuntos de la nube de puntos con el fin de aproximar las normales de la superficie y determinar cuáles puntos pertenecen o no a esta. Por

último teniendo los dos subconjuntos de puntos (que pertenecen y no pertenecen a la superficie) se ajusta una función implícita, se interpola la función implícita a través del uso de una función de base radial y se reconstruye la superficie a través de esta última. En los resultados obtenidos se muestra que este método realiza reparación de la superficie o llenado de huecos, además de la posibilidad de realizar suavizado. De manera parecida, en Ohtake y Belyaev[27] desarrollan un procedimiento adaptativo de ajuste de funciones de base radial que genera los centros de la función de manera aleatoria con base en características tales como la densidad de los datos y la geometría de la superficie logrando reproducir detalles finos dentro de la superficie real. En Sharfet ál.[30], se propone un método de reconstrucción de regiones faltantes basado en el copiado y pegado de porciones de datos del resto de la superficie en el hueco o región faltante y luego validar la concordancia de los datos añadidos con respecto al entorno con el fin de encontrar las zonas existentes de la superficie del objeto que pueden ser utilizadas para rellenar la región faltante. Este método tiene un buen comportamiento siempre y cuando la región a reconstruir tenga propiedades geométricas presentes en otras partes o regiones del objeto. Esta técnica denominada *inpainting* permite generar detalles finos en superficies considerando la restricción anterior. Otros trabajos que aplican dicha técnica se pueden observar en ([31][32][33]), los cuales utilizan información de objetos almacenados en una base de datos para reconstruir regiones faltantes en una superficie. Los autores estiman una función de forma que define que indica a qué tipo de objeto puede pertenecer la superficie, con lo cual se buscan en la base de datos los modelos que cumplan dicho criterio y por último utilizan información de los modelos hallados para reconstruir las regiones faltantes de la superficie. Una limitación fuerte en estos trabajos radica en la complejidad y precisión del proceso de búsqueda de los objetos de la base de datos. Además, la gran cantidad de tipos de objetos de forma libre existentes obligan a utilizar una base de datos muy extensa. Park y Avinash [28], afrontan el problema de la corrección global de la superficie desde los datos adquiridos, utilizando para ello una nueva técnica denominada *multipeak range imaging*, que permite almacenar múltiples candidatos para cada punto de la superficie del objeto. Luego mediante una serie de restricciones se seleccionan los puntos más factibles de pertenecer a la superficie del objeto escaneado. Este método presenta algunas propiedades muy deseables en la RDS como robustez ante el ruido y ante características tales como reflectancia y opacidad, por ejemplo. Por otro lado, este método aumenta la complejidad del proceso de adquisición debido a su naturaleza múltiple.

3.2. Identificación y Corrección Local de Huecos.

Los métodos de corrección local se presentan como la alternativa correcta cuando se desea corregir de manera específica las anomalías sin distorsionar el resto de la información presente en la superficie. Estos métodos parten de la detección del hueco en la superficie, para lo cual la aproximación más sencilla radica en considerar que un hueco en una superficie es una sucesión cerrada de aristas límites o aristas que solo pertenecen a un triángulo. Entonces el procedimiento de detección de huecos parte de encontrar una arista límite en la superficie [14]. Una vez realizado lo anterior se realiza una búsqueda en los triángulos vecinos al triángulo de la arista límite, buscando en estos otras aristas límites hasta encontrar un camino cerrado el cual describe el borde del hueco. Sin embargo, aunque este método de detección es aplicado de manera frecuente, una aproximación diferente ha sido propuesta por Dong et ál.[34], el cual desarrolla un método sencillo de corrección de huecos en imágenes de rango que parte de la identificación de los mismos a

través del estudio del histograma de imágenes bidimensionales de la superficie real del objeto adquirido, donde, valores muy cercanos a 0 en imágenes de grises, representan píxeles pertenecientes a un hueco. Una vez se detectan los píxeles pertenecientes al hueco se debe realizar un mapeo que permita establecer a que valores tridimensionales pertenecen al borde del mismo. La mayor limitante de este método radica en que solo trabaja con huecos realmente pertenecientes a la imagen, ya que, discontinuidades producto de errores en el proceso de RDS, no tienen representación alguna en las imágenes bidimensionales de tomadas de la superficie del objeto real.

Una de las primeras aproximaciones a la corrección local de anomalías relacionadas con ausencia de información en superficies, es la propuesta por Brian Curless y Marc Levoy [35] donde se presenta un método de carácter volumétrico para generar modelos tridimensionales a partir de imágenes de rango, donde primero se obtiene una representación volumétrica de la imagen de rango y se clasifican todos los puntos de esta representación en alguno de los estados: *unseen*, *empty* o *near the surface*. Los huecos en la superficie son indicados por las fronteras entre regiones *unseen* y regiones *empty*. Luego se rellenan dichas fronteras para corregir los huecos. Davis et ál.[36] propone una técnica parecida, que utiliza difusión volumétrica para unir o cerrar superficies, la cual es aplicada para cerrar de manera suave discontinuidades en superficies. Peter Liepa [14] desarrolla un algoritmo para triangular el hueco, basado en la estimación de los triángulos de mínima área, además de asignar un peso a los triángulos del borde del hueco con el fin de evitar problemas cuando dicho borde cuenta con triángulos adyacentes a la superficie. Uno de los primeros trabajos en utilizar interpolación para estimar los datos faltantes en el hueco, es el propuesto por Manuel Oliveira y Jianning Wang [37], en el cual se utiliza información geométrica de los píxeles vecinos al borde del hueco. Estos datos se utilizan para ajustar una superficie usando una aproximación a través de cuadrados de menor movimiento o *Moving Least Squares(MLS)*. La preservación de detalles y características tales como esquinas y bordes es una propiedad deseada en los métodos de llenado de huecos. En [38] se presenta una técnica de corrección local de huecos que preserva esquinas o zonas de cambio angular abrupto, a través del estudio de los ángulos de los triángulos que conforman el borde del hueco y el uso posterior de dicha información para determinar si el hueco se encuentra en una zona con esquinas o no y repararlo de acuerdo al caso. El ajuste de la malla generada para llenar el hueco en la superficie, puede verse como un problema de optimización. En [39] se realiza un algoritmo basado en optimización para generar la mejor representación de la superficie que llena el hueco. Para esto se estima una malla inicial basada en la técnica *Advance Front Meshing* y por último se optimiza dicha malla utilizando para ello la ecuación de Poisson de manera iterativa. En las ciencias de la computación es común encontrarse con que la premisa “divide y vencerás” está presente en las soluciones más eficientes a muchos problemas tales como búsqueda, ordenamiento, análisis sintáctico, etc. En [40] se realiza un método basado en la premisa anterior, que divide un hueco complejo en n huecos sencillos a través del estudio de los vértices del mismo. Cuando se encuentran zonas del hueco con una alta curvatura, se separan o dividen estas regiones del resto del hueco. Por último se realiza triangulación simple de Delaunay para llenar cada uno de los sub-huecos simples producto de la división del hueco original. Como se comentó anteriormente, las superficies matemáticas utilizadas para la RDS son diversas, entre las que tenemos, las mallas triangulares, voxels, NURBS, funciones de base radial, entre otras. Kumar Amitesh et ál.[41] desarrolla un algoritmo en el cual se reconstruye la información faltante en huecos en superficies a través del ajuste de una malla de NURBS. Los autores hacen énfasis en que el algoritmo solo funciona con huecos de topología simple y que la aplicación del mismo a huecos de topología compleja queda como trabajo futuro. Los métodos de corrección local revisados hasta este punto o corrigen todos los huecos

una vez identificados, o relegan la tarea de selección de los huecos a corregir al usuario de sistema. El trabajo más reciente encontrado en esta revisión es el realizado por Li et ál.[42] en el que se utilizan características relevantes de trabajos anteriores como “divide y vencerás” y se realiza una aproximación para resolver una de las limitaciones comúnmente encontradas en los métodos propuestos: El desaprovechamiento de la información presente en islas o superficies internas del hueco. El algoritmo comienza con la división del hueco en sub-huecos más simples tomando como líneas de división las curvas características del hueco. Estas curvas características son reconstruidas a través de una técnica denominada *Polynomial Blending*. Por último se genera un parche híbrido de Lagrange-Bezier para llenar cada sub-hueco. La generación de este parche es un proceso menos complejo que ajustar funciones de base radial. Aunque el método genera resultados correctos para reconstruir la información faltante en el hueco, sigue presentando la limitación de la selección manual de los huecos a corregir.

3.3. Clasificación o selección automática de huecos a corregir

Un trabajo líder en el área de la clasificación o selección automática de los huecos a corregir en superficies es el realizado por Branch et ál.[1][9][43] en el cual mediante el estudio de la torsión espacial del borde del hueco y la utilización de un umbral simple, se estima con niveles de precisión aceptables si un hueco es producto de un error en alguna de las etapas del proceso de RDS o si pertenece realmente al objeto original. Sin embargo, el autor aclara que este umbral fue determinado de forma imprecisa y sugiere un mejor estudio para su posterior uso en un clasificador. Por otra parte, debido a que la medida de torsión es espacial, el algoritmo presenta problemas cuando el hueco se presenta en superficies relativamente planas. Sanchez et ál.[44][45] propone el uso de la entropía de la torsión y de la curvatura como métrica de distorsión del contorno. Experimentos iniciales en este trabajo, suponen que la clasificación del hueco mediante estos valores se limita a un único umbral. Aunque algunas métricas han sido propuestas anteriormente, aun se necesita determinar la naturaleza y metodología del método de clasificación.

3.4. Limitaciones Generales.

- Debido a su naturaleza, los métodos de corrección global o basada en datos dispersos, reparan de manera indiscriminada los huecos en la superficie, aun cuando algunos de estos pueden pertenecer al objeto real, y aunque por lo general llegan a modelos visualmente correctos, en muchos campos de aplicación no es deseable añadir información extra a la que puede encontrarse en el objeto real.
- La selección de los huecos a reparar en el caso de los métodos de corrección local es una tarea realizada por un usuario y aunque algún método ha sido propuesto para realizar dicha tarea de manera automática, este aún presenta muchas limitaciones.
- Los métodos propuestos para corrección de huecos, ya sean de corrección global o local, necesitan trabajar sobre representaciones totales o parciales de la superficie del objeto, es

decir, no utilizan información subyacente en los datos discretos registrados sino de superficies aproximadas a los mismos.

- La mayoría de métodos y técnicas propuestos, trabajan con huecos simples, aun cuando es común encontrar huecos que presentan islas o información interna que podría utilizarse para aproximar de mejor forma la información faltante.

- Los métodos de clasificación de huecos en superficies 3D encontrados en la literatura no son claros respecto a los valores o parámetros de clasificación.

CAPITULO 4

NATURALEZA Y CARACTERÍSTICAS DE LOS HUECOS EN SUPERFICIES 3-D

La ausencia de información en superficies tridimensionales, comúnmente reflejada en huecos en el modelo digital, tiene amplio conjunto de causas o naturalezas, relacionadas en gran parte con las propiedades del objeto de estudio y con los subprocesos mismos del escaneo 3D, por ejemplo, problemas de sensado y registro, como se verá posteriormente. Es de suponer que cada una de estas causas generará características diferentes en los huecos que producen. El propósito de este capítulo será realizar una breve descripción sobre las condiciones y causas que generan ausencia de información en modelos 3D más comunes en el proceso de reconstrucción 3D. En este trabajo se dividen a modo práctico, las diferentes causas o naturalezas de la ausencia de información en los siguientes grupos: *Oclusiones*, *Problemas de Registro*, *Problemas ópticos de Superficie del Objeto* y *Huecos pertenecientes al objeto*. El primer grupo contempla los huecos producidos por obstrucciones del sensor por zonas del mismo objeto y los producidos por limitaciones de alcance del sensor en regiones de difícil acceso. El grupo *problemas de registro* hace referencia a errores en la alineación de las vistas obtenidas y de igual forma a la ausencia de una o más de estas, producto del proceso de adquisición. Los huecos generados por problemas de complejidad óptica de la superficie del objeto escaneado se agrupan en *problemas de superficie del objeto*, y por último, los huecos que se encuentran en el objeto real y de igual forma en el modelo digital obtenido, son los de *naturaleza pertenecientes al objeto real*.

A continuación, se detallan algunas de las diferentes causas que generan ausencia de información en superficies 3D, según la agrupación realizada anteriormente.

4.1. Oclusiones

Muchos autores concuerdan en que los huecos en superficies 3D por lo general son el resultado de oclusiones y limitaciones de accesibilidad, lo cual queda expresado según Marschner en[36]: “*la causa fundamental de los huecos es la oclusión...Sin embargo, los huecos también pueden ser causados por baja reflectancia, limitaciones de posicionamiento del escáner o simplemente vistas perdidas*”.

Desde un punto de vista general, en el dominio de la reconstrucción 3D, una oclusión puede ser entendida como una obstrucción total o parcial de una vista del objeto, producida por una región perteneciente al mismo. Esta situación se presenta en gran parte, debido a la presencia de irregularidades en la superficie de objetos geoméricamente complejos (Figura15), las cuales

generan hendiduras profundas y/o protuberancias las cuales pueden producir regiones de difícil acceso para el sensor.



Figura 15. Objeto prácticamente imposible de escanear mediante métodos ópticos. Imagen tomada de [46].

La presencia de huecos en superficies 3D producto de oclusiones es muy común en el proceso reconstrucción 3D de objetos de forma libre debido a las innumerables configuraciones de superficie posible en dichos objetos. La mayor parte de la investigación realizada en corrección de huecos en superficies 3D supone que estos son producto de oclusiones o como algunos autores denominan, auto-occlusiones de la superficie.

Una limitación común en los dispositivos ópticos de adquisición 3D, radica en la necesidad de utilizar un ángulo fijo de triangulación entre el sensor y el emisor de luz estructurada. Este hecho ocasiona que no se puedan evitar todas las posibles configuraciones de oclusiones, como se observa en la **¡Error! No se encuentra el origen de la referencia..**

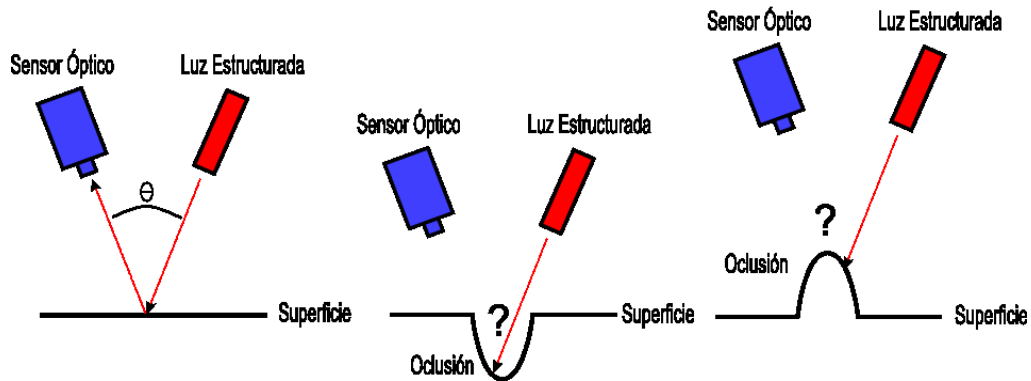


Figura 16. Representación del proceso de escaneo para un sensor óptico de luz estructurada en (de izquierda a derecha) a) una superficie plana, b) en presencia de una hendidura en la superficie y c) en presencia de una protuberancia en la superficie.

Cabe mencionar que en la anterior figura, se presenta el caso de la protuberancia en la superficie a modo de ilustración de la limitación en el ángulo entre el sensor y el emisor, ya que, de modo práctico esta oclusión puede ser corregida mediante otra vista parcial, bien sea a través de una rotación del par sensor-emisor respecto al eje del objeto o de una rotación del objeto en su mismo eje. Lo anterior difiere del caso de la hendidura, para la cual, independiente a la rotación del objeto o del sensor, siempre habrá zonas ocluidas.



Figura 17. Hueco generado por auto-oclusión de la superficie.

Desde el punto de vista de un experto humano, los huecos producto de oclusiones son fácilmente diferenciables siempre y cuando se tenga información global del objeto, es decir, cuando el experto puede observar el contorno del hueco y compararlo con información acerca de conocimiento previo del objeto, posición del hueco en el objeto, etc. Por ejemplo, en la **¡Error! No se encuentra el origen de la referencia.** se puede observar un hueco producido por oclusión; Nótese que resulta trivial inferir que el mismo es producido por oclusión del dedo pulgar del pie, lo cual es posible debido a la capacidad de la inteligencia humana de inferir dicha información una vez se conoce el método de adquisición y contrastar este proceso respecto al conocimiento del objeto. Sin embargo, estimar que un hueco es producto de oclusión contando únicamente con los datos del contorno no resulta un proceso trivial ni siquiera para el experto humano, como se verá en los experimentos realizados en el capítulo 5 de este documento.

4.2. Problemas de Registro

La etapa de registro del proceso de reconstrucción tridimensional, la cual fue tratada en la sección 2.4, implica tener que alinear las diferentes vistas parciales producto del proceso parcial de adquisición 3D. Este proceso generalmente genera regiones incorrectamente alineadas, lo que puede generar ausencia de información. Esto se debe en gran parte debido a que los algoritmos actuales de registro requieren que el usuario seleccione de forma manual una serie de puntos en común entre cada par de vistas parciales. La selección incorrecta de estos puntos, genera alineaciones inadecuadas entre estas vistas lo que genera espacios que no se tocan. Esto puede ser apreciado en la Figura 18.

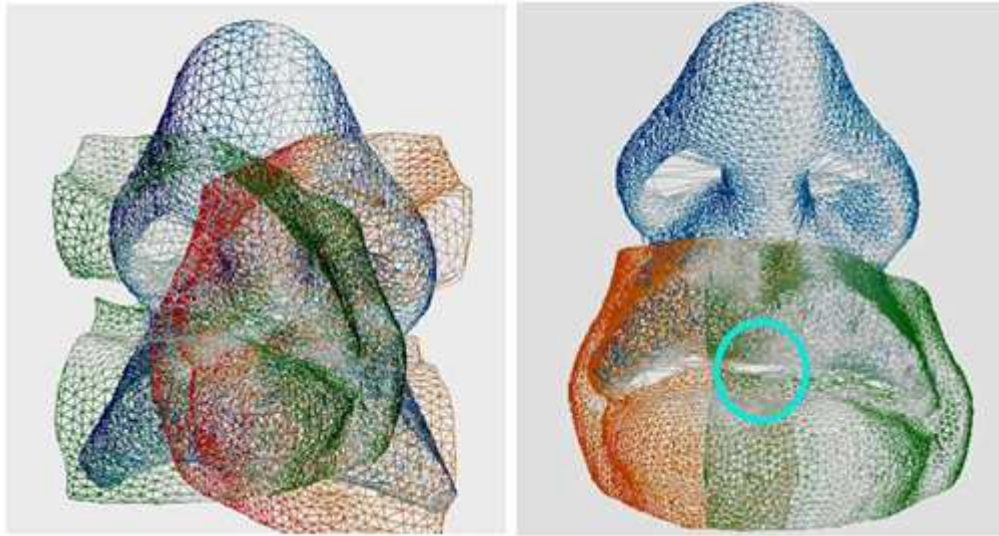


Figura 18. Registro incorrecto de vistas parciales. La región al interior del círculo rojo, indica un hueco producto de este registro. Imagen tomada de [9].

De lo anterior, podemos plantear la siguiente definición: *Un hueco producto de errores de registro se puede entender como un ciclo cerrado de aristas compuestas por vértices limítrofes de múltiples vistas parciales.*

Según la anterior definición, determinar si un hueco es producto de un problema de registro resulta de fácil solución, ya que cada vértice del mismo puede ser relacionado con una vista parcial específica, y en caso tal de que exista más de una vista parcial para el conjunto de puntos que conforman el hueco, entonces podremos concluir que el hueco es producto de esta naturaleza. Este proceso sería posible siempre y cuando la información de la vista parcial a la que pertenece cada vértice del modelo esté disponible. Por ejemplo, si a los conjuntos de puntos registrados $p_0, p_1, p_2, \dots, p_n$ y $v_0, v_1, v_2, \dots, v_n$ para dos vistas parciales p, v le es aplicada una operación de simplificación y se genera un nuevo conjunto de puntos simplificados $r_0, r_1, r_2, \dots, r_n$ la información de la vista parcial a la que pertenece cada punto de este nuevo conjunto se habrá perdido o a lo sumo, será necesario aplicar algún proceso de clusterización para determinar dicha información. Esto puede ser tratado de dos formas. La primera es realizando el proceso de identificación del hueco en el mismo proceso de registro antes de realizar cualquier operación que implique modificación del conjunto de puntos originales y mediante algún proceso de indexación, almacenar información respecto a la posición relativa del hueco en el objeto para que este sea corregido en el futuro proceso de integración. La segunda es utilizando alguna estructura de información para los puntos de las vistas parciales que permita conservar la información de la vista parcial a la que pertenece cada punto independientemente de los procesos aplicados a los mismos luego del proceso de registro, con lo cual, la información acerca de las vistas parciales de cada vértice del modelo puede ser utilizada en el proceso de integración.

Debido a que el propósito de este trabajo es determinar la naturaleza de los huecos utilizando únicamente información del contorno del mismo, no es viable aplicar la definición anterior. Sin

embargo, existe una característica común en los huecos producto de errores en el proceso de registro: *El tamaño del hueco*.



Figura 19. Parámetros estándar para el llenado de huecos. Imagen tomada mediante el software RapidForm®.

La constante investigación en el proceso registro para imágenes de rango ha generado algoritmos cada vez más precisos que se basan en características geométricas locales de la superficie para identificar regiones en común entre las vistas parciales [47][48], lo cual se ve reflejado directamente en la disminución del error de alineación entre vistas parciales, lo que a su vez implica una disminución del tamaño de los huecos. La mayoría de programas comerciales y libres para reconstrucción 3D, entre ellos, RapidForm y Meshlab, permiten establecer un valor de tamaño, específicamente el número de aristas, para filtrar los huecos que se desean corregir (ver Figura 19).

4.3. Problemas Ópticos de la Superficie del Objeto

Según lo visto en la sección 2.3, la mayoría de sensores de para la adquisición 3D, se basan en métodos ópticos activos, los cuales operan basados en triangulación óptica para obtener los datos de profundidad, debido a que presentan una mayor precisión y su hardware es relativamente simple [28]. Una vez el sensor proyecta un patrón óptico sobre el objeto real, una cámara captura una imagen bidimensional de la escena y mediante la deformación del patrón en dicha imagen, se puede establecer una relación de profundidad mediante ecuaciones de triangulación (ver Figura 8). Ahora, resulta evidente que si el patrón proyectado es distorsionado por alguna propiedad óptica del objeto, esto generará anomalías que podrían causar huecos en la superficie reconstruida (ver Figura 20).

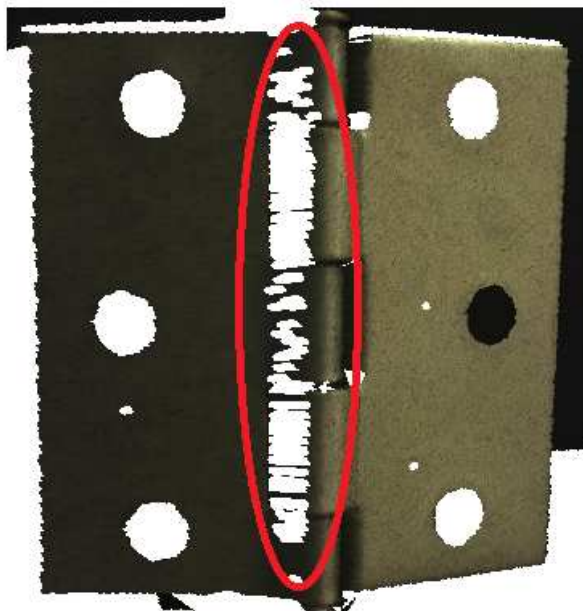


Figura 20. Huecos generados por problemas de especularidad de la superficie del objeto. Nótese que los huecos producto de problemas ópticos de la superficie del objeto son altamente irregulares debido a la distorsión del patrón proyectado.

A saber, las propiedades de la superficie que generan estos problemas son: *Reflectancia* y *Especularidad*. En la Figura 21 se pueden apreciar tres objetos con distintas propiedades ópticas. El primer objeto (busto) representa el caso ideal en que el patrón proyectado que se muestra en la parte inferior, es capturado por la cámara de manera correcta. El segundo objeto (plato de metal) presenta especularidad lo que genera reflexiones que distorsionan el patrón proyectado. El tercer objeto (gorila) presenta una superficie de baja reflectancia lo que implica que el patrón proyectado pierde luminosidad causando una imagen borrosa.

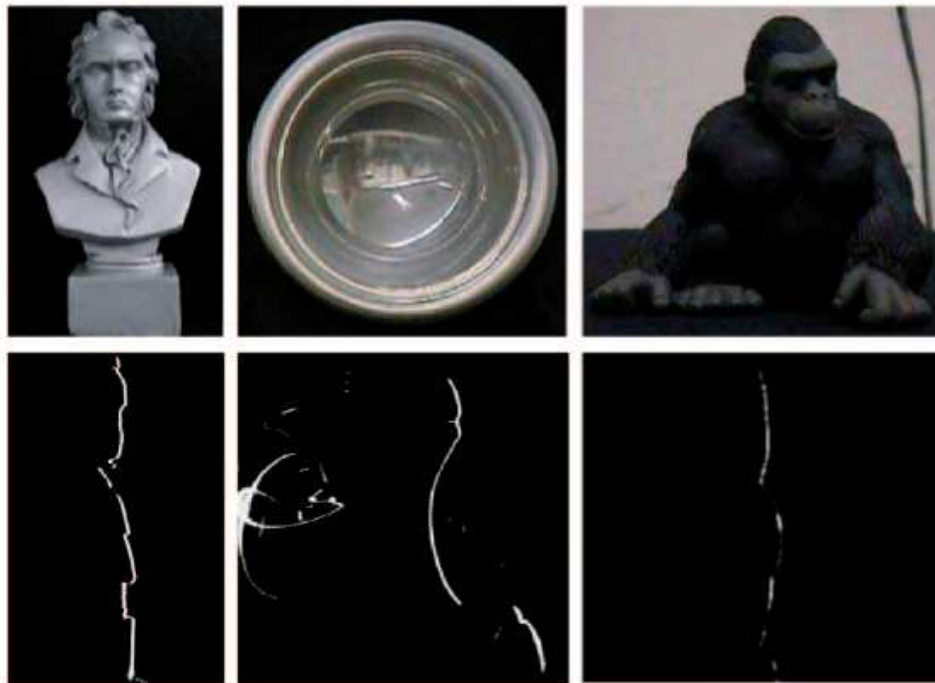


Figura 21. Objetos con problemas de complejidad óptica. Imagen tomada de [28].

Esta naturaleza no es de interés para la comunidad científica en el tema de corrección de huecos, ya que bajo condiciones prácticas, el tratamiento de los errores producidos por las propiedades ópticas del objeto es estudiado desde el proceso mismo de la adquisición, ya sea mejorando físicamente la superficie a través de la aplicación de sprays antireflectivos en los casos en que esto resulte viable, de métodos novedosos que determinen cuales valores de una imagen de rango no son producto de ruido generado por esta naturaleza [49][28][12][50][51], o de mejoras en el hardware de adquisición [52].

4.4. Huecos Pertencientes al Objeto

Las causas que generan huecos en superficies 3D descritas anteriormente, deben ser tratadas como anomalías en el modelo reconstruido, es decir, deben ser corregidas mediante algún método de llenado de huecos. Por el contrario, los huecos pertenecientes al objeto real deben mantenerse para preservar la fidelidad de la reconstrucción. Según [9] los huecos pertenecientes a la superficie del objeto real, difieren de los huecos falsos o anomalías, en la distorsión del contorno, ya que si se parte de la definición de objeto de forma libre propuesta por Besl en [2] y descrita en la sección 2.1, los huecos reales o pertenecientes al objeto deberían ser suaves.

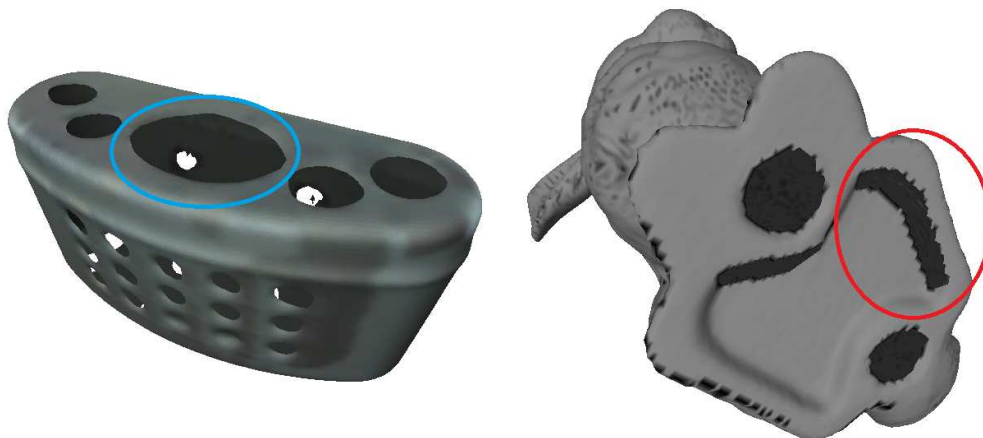


Figura 22. En la izquierda, un hueco perteneciente al objeto. En la derecha, un hueco producto de una oclusión. Nótese la diferencia en la suavidad de dichos huecos.

Por otra parte, la simetría parece estar presente en la mayoría de huecos pertenecientes al objeto, lo cual no es extraño si se considera la simetría como propiedad que refleja orden y estética en la naturaleza[53]. Si se observa la Figura 22 se puede notar que los huecos pertenecientes al objeto presentan una simetría mayor que el hueco señalado en el conejo de Stanford. De esta forma es trivial determinar que los dos huecos circulares del conejo de Stanford pertenecen realmente al objeto.



Figura 23. Similitud geométrica en huecos de objetos de forma libre.

Otra característica geométrica relacionada con la simetría en los huecos pertenecientes a superficies de objetos de forma libre radica en la similitud con ciertas figuras geométricas preestablecidas como círculos, elipses, cuadrados, estrellas, etc. (ver Figura 23). Esta característica podría ser utilizada para determinar si un hueco es real o no si su similitud geométrica respecto a una base de datos figuras geométricas preestablecidas es alta o una base de datos de huecos de los cuales se conoce con certeza que aparecerán en los modelos a escanear.

Tabla 1. Características observables en los contornos de huecos en superficies 3D de objetos de forma libre.

Característica	Descripción	Observación
Distorsión	Que tan irregular es el contorno del hueco.	
Tamaño	Cuántas aristas tiene el contorno de un hueco.	
Ruido	El nivel de dispersión de los datos de las regiones cercanas al hueco.	No aplica para este trabajo.
Número de vistas parciales del contorno	Cuántas vistas parciales confluyen en el contorno de un determinado hueco.	No aplica para este trabajo.
Simetría	Que tan simétrico es el contorno del hueco respecto a un eje.	
Similitud Geométrica	Que tanto se parece el contorno del hueco a una figura determinada.	

En la Tabla 1 se muestran las características observables en los contornos de los huecos en superficies 3D de objetos de forma libre que serán evaluadas a continuación. Las características ruido y Número de vistas parciales del contorno no son objeto de estudio de este trabajo debido a que su medición requiere de información extra a parte de los vértices que conforman el contorno del hueco.

4.5. Conclusiones

La distorsión del contorno, el tamaño del hueco, la simetría y la similitud geométrica son características comunes y en algunos casos discriminantes a la hora de determinar la naturaleza de un hueco en una superficie 3D de un objeto de forma libre. Los descriptores para estas características serán propuestos en capítulo 5 y evaluados en el capítulo 6. Sin embargo, determinar la naturaleza específica que causa la anomalía parece una tarea que requiere del análisis del hueco en diversas etapas del proceso de reconstrucción y de información extra a los datos del contorno del hueco.

Los huecos producto de errores de registro podrían ser validados respecto al tamaño del mismo, característica que utilizada actualmente en el proceso de llenado de huecos por ciertos programas comerciales y libres de reconstrucción 3D.

Según la definición de huecos producto de errores de registro expuesta en la sección 4.2, el número de vistas parciales presentes en un conjunto de vértices de un hueco es una métrica viable para determinar si el mismo es producto de un error de registro. De igual forma, obtener una medida del nivel de ruido de las regiones cercanas al contorno del hueco podría ser utilizada para clasificar el hueco como producto de problemas ópticos de la superficie. Cabe aclarar que debido a que las anteriores mediciones (número de vistas parciales y ruido) no serán de interés de este trabajo ya que requieren información extra al conjunto de puntos que conforman el contorno.

Los huecos relacionados con problemas ópticos de la superficie no representan un grupo de mayor interés debido a que la investigación en el tema se ha enfocado en resolver el problema desde el proceso mismo de la adquisición.

CAPÍTULO 5

DESCRIPCIÓN DE HUECOS EN SUPERFÍCIES 3-D

Describir el contorno de un hueco en una superficie 3D puede ser entendido como obtener un conjunto de mediciones de una curva discreta-digital en el espacio R^3 , con el cual se pueda caracterizar dicha curva. Esto implica resolver un problema no trivial que radica en determinar cuál es el conjunto de características medibles y relevantes para la caracterización de la curva y por otra parte, cual es la representación más adecuada para obtener información de la curva. Por esta razón, la descripción realizada en este capítulo se basará en implementar las características observadas en el capítulo anterior, haciendo uso de la revisión literaria de los trabajos más relevantes respecto a medición de características en curvas en el espacio tridimensional.

Los descriptores presentados a continuación trabajan con los puntos 3D que conforman el contorno del hueco. Para la extracción de estos datos se utiliza el método propuesto por Liepa en [14], para lo cual se recorren el modelo en busca de aristas límite, o aristas pertenecientes a un solo triángulo. Una vez se encuentra una arista de este tipo, se recorren recursivamente las aristas límite vecinas, buscando un ciclo cerrado. Este ciclo cerrado de aristas límite describe un hueco en la superficie del modelo. Los vértices de dichas aristas conforman el conjunto de puntos 3D del contorno del hueco que será utilizado a continuación.

5.1. Descripción y Representación de Curvas y Contornos.

La literatura existente respecto a la descripción y representación de curvas y contornos es muy amplia en el área de la visión por computador de imágenes bidimensionales. Este tema ha sido de gran interés en la comunidad científica y académica debido al papel fundamental que juegan los contornos o bordes en el análisis humano de una escena visual, por ejemplo, al determinar fronteras entre objetos, reconocer siluetas en condiciones de poca iluminación y determinar la profundidad de una escena, entre otras funciones.

Determinar el conjunto de descriptores que se deben utilizar para una determinada curva es un problema abierto en la literatura, debido a la arbitrariedad de las formas que estos pueden tomar, por lo que generalmente este conjunto es estimado de forma experimental y dependiente de la aplicación. Este problema ha sido estudiado en diversas ramas de la ciencia tales como el reconocimiento de patrones, aprendizaje estadístico e inteligencia artificial.

Por otra parte, determinar que característica o medición puede ser utilizada como un descriptor del objeto (en este caso una curva 3D) requiere validar ciertas condiciones como lo expresa Pajares en [18]:

“Por lo general, los descriptores deben ser independientes del tamaño, la localización y orientación del objeto y deben contener suficiente información de discriminación para distinguir un objeto de otro”.

Por el contrario, una representación de una curva, no estima un valor o medición de la misma, sino que codifica la información de la misma en una forma distinta a la original, por ejemplo, una imagen digital puede ser representada en diferentes espacios de color (*RGB, HSL, YIQ*), cada uno de estos apto para medir diferente información en una determinada aplicación.

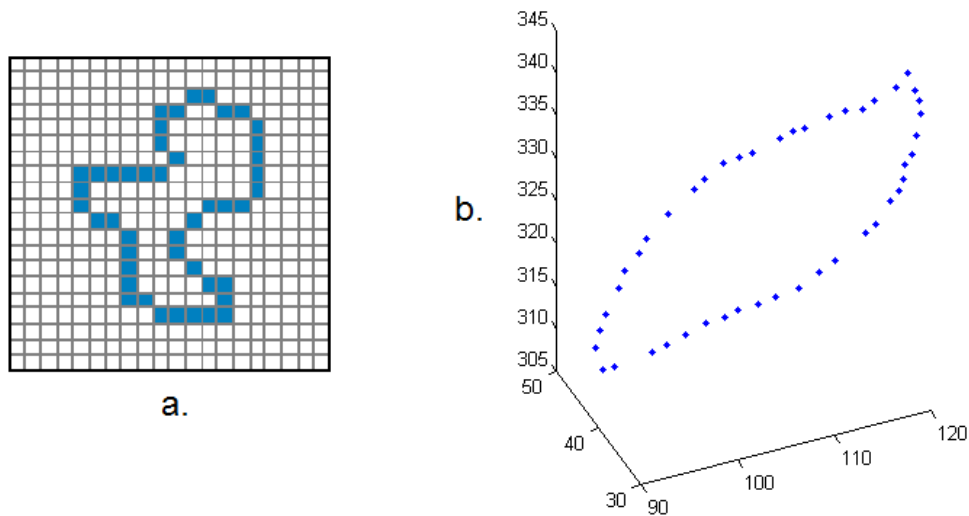


Figura 24. Curvas digitales. a) curva digital de una imagen bidimensional con espaciamiento homogéneo. b) curva digital 3D con espaciamiento no homogéneo.

Entre los descriptores y representaciones comúnmente utilizados en procesamiento de imágenes bidimensionales encontramos los códigos de cadena, firmas, descriptores de Fourier, momentos y funciones de energía. Sin embargo, la naturaleza no homogénea del espaciamiento de los puntos (ver Figura 24) de una curva en una superficie 3D limita la aplicabilidad de las técnicas de descripción de curvas ampliamente estudiadas en el procesamiento bidimensional de imágenes, otras palabras, la mayor parte de las técnicas de representación y descripción de bordes en imágenes 2D no son directamente aplicables al dominio de las curvas 3D.

La mayor parte de la literatura sobre descripción de curvas en el espacio 3D hace énfasis en la utilización de la geometría diferencial de curvas (GDC) que se encarga del estudio de las propiedades de estas mediante métodos que utilizan cálculo diferencial e integral.

Según Sánchez en [44] las dos características principales de una curva son su torsión y curvatura, ambas características de la geometría diferencial de curvas, por otra parte, Branch en ([1][9]) aclara que un hueco puede ser descrito y clasificado mediante sus propiedades geométricas,

curvatura y torsión. Estas características brindan información local exclusivamente, por lo cual deben ser combinadas con características que brinden información global del objeto de estudio tales como simetría, forma y parecido a figuras geométricas preestablecidas.

Los descriptores de huecos tratados en este capítulo son agrupados en dos subconjuntos principales: *Descriptores locales* (Sección 5.2.) y *Descriptores globales* (Sección 5.3.). En la Figura 25 se pueden observar los descriptores utilizados en este trabajo.

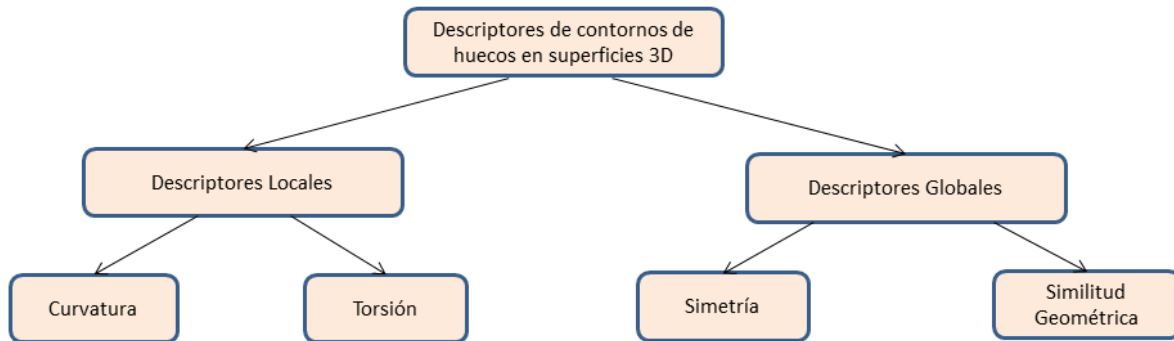


Figura 25. Descriptores de contornos de huecos en superficies 3D utilizados en este trabajo.

5.2. Descriptores Locales: Geometría Diferencial de Curvas

Las características tratadas en la geometría diferencial de curvas constituyen un conjunto importante de estudio en áreas de aplicación tales como visión por computador [54] y computación gráfica [55]. De particular interés para el tratamiento geométrico de la curva tenemos la torsión y la curvatura; propiedades intrínsecas en las curvas continuas. Según Nguyen [56][57], en el espacio 3D, la torsión y la curvatura son las propiedades más importantes que permiten estudiar el comportamiento de una curva espacial. Para esto, numerosos métodos [56][57][58][59][60][61] han sido propuestos los cuales se pueden separar en dos grandes grupos. Los primeros son los métodos basados en curvas analíticas y los segundos son los métodos que trabajan directamente sobre la curva discreta.

Los métodos basados en curvas analíticas ajustan curvas de este tipo al conjunto discreto de puntos originales (por naturaleza discretos) y estiman la torsión y curvatura mediante derivaciones de las funciones de la curva ajustada [56][62][63][64]. Según Yi et al. [61], estos métodos utilizan por lo general curvas spline [65][66], curvas polinomiales [63], círculos [67][68] y curvas paramétricas [69][62][55]. Por el contrario, los métodos que trabajan directamente sobre la curva discreta, evitan el realizar un ajuste curvas y realizan los cálculos directamente sobre los puntos originales producto del proceso de adquisición [70][71][60][57][56][72][64].

En [63] se utiliza una curva polinomial, usualmente una curva cuadrática de la forma:

$$f(x) = A_0 + A_1x + A_2x^2(1)$$

Esta es ajustada a un conjunto de puntos vecinos $P_j | k - m \leq j \leq k + m$ respecto al punto que se desea evaluar P_k . Posteriormente la curvatura y la torsión son estimadas utilizando los coeficientes A_i , resueltos mediante MMC (*Método de los Mínimos Cuadrados*). Lewinner et ál. logra mejorar estos resultados [62][55] al proponer un método para la estimación de la curvatura de una curva paramétrica cuadrática, basado en un ajuste de mínimos cuadrados ponderados y la aproximación local de la longitud de arco (ver Figura 26), con mejores resultados respecto a métodos anteriores en cuanto robustez ante ruido y diferentes condiciones de muestreo. La forma de la curva paramétrica cuadrática viene dada por la ecuación:

$$x(s) = x_k + x'_k s + \frac{1}{2} x''_k s^2, (2)$$

$$y(s) = y_k + y'_k s + \frac{1}{2} y''_k s^2 (3)$$

Donde s es la longitud de arco de la curva. Las derivadas son resueltas mediante el método de los mínimos cuadrados.

En [55] los mismos autores utilizan una curva paramétrica cúbica de la misma forma que el método anteriormente propuesto.

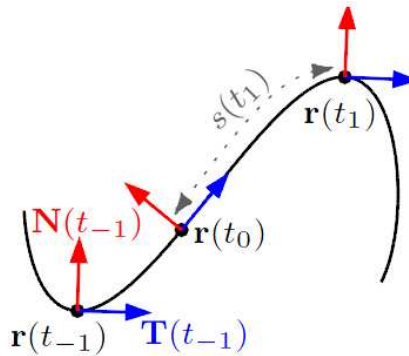


Figura 26. La longitud de arco $s(t)$ ayuda a definir los vectores tangente $T(t)$ y normal $N(t)$. Figura toma de [62].

En [67] se utiliza el concepto de círculo osculador (ver Figura 27) que define la curvatura de una curva en un punto P_s como $k(s) = \frac{1}{r(s)}$, donde $r(s)$ es el radio de dicho círculo. Los autores definen entonces la curvatura k en el punto P_k según la siguiente ecuación:

$$k(P_k) = \frac{1}{r(P_k)} = \frac{4S_k}{|P_{k-m}P_k| |P_kP_{k+m}| |P_{k-m}P_{k+m}|} \quad (4)$$

Donde S_k es el área del triángulo (a , b , c) que se observa en la Figura 27.

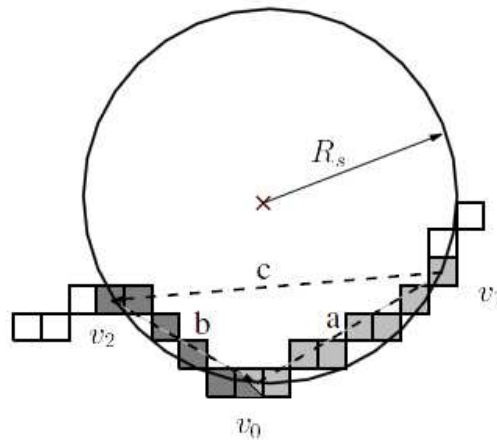


Figura 27. Estimación de la curvatura mediante círculo osculador. Figura tomada de [67].

Aunque en la literatura sobre medición de propiedades de geometría diferencial de curvas en visión por computador y computación gráfica los métodos basados en curvas analíticas representan un conjunto numeroso, esta aproximación tiene las siguientes limitaciones generales:

1. Determinar el tipo de curva analítica a utilizar es dependiente de la aplicación y por lo tanto un problema abierto, es decir, no existe método alguno para determinar que curva analítica (*splines*, *B-Splines*, *NURBS*, *polinomial*, *etc.*) se debe utilizar ante una determinada curva, lo cual influye en el resultado de las mediciones de las características.[59].
2. Es casi imposible que un determinado tipo de curva analítica se ajuste de forma correcta a todos los diferentes tipos de curvas en el espacio 3D.[61].
3. Es bien sabido que las curvas analíticas como las NURBS no presentan buenos resultados ajustándose a figuras geométricas bien definidas como cuadrados, triángulos, estrellas, etc.[73][61].
4. El número de nuevos puntos generados mediante el ajuste de la curva analítica o resolución de la curva analítica, es un valor experimental que depende del tipo de curva y de la aplicación, por lo cual no es definido.
5. La naturaleza de las curvas analíticas, tiende a generar suavización de los puntos de la curva, lo cual resulta desfavorable a la hora de obtener una medida de distorsión de la misma.

Los métodos que trabajan directamente sobre la curva discreta están constituidos en mayor parte por trabajos recientes, por lo general, de mediados de la década del 2000 hasta el presente. Esto métodos tienen como ventaja respecto a los anteriores, que las estimaciones de las propiedades geométricas de la curva se calculan directamente de los puntos discretos originales y no de puntos interpolados que en cierta forma ingresan ruido. Por otra parte el costo computacional disminuye

al disminuir directamente el número de puntos necesarios para el cálculo de las propiedades geométricas.

Estos últimos pueden estudiarse en dos grupos principales: los métodos que trabajan mediante arreglos 3D de resolución definida [70][71][60][57][56][72][64], los cuales funcionan de forma correcta para curvas discretas sobre superficies 3D con resolución fija, lo cual es una limitante. Y por otra parte están los métodos que trabajan sobre el concepto de derivadas discretas los cuales presentan las ventajas de los métodos anteriores pero no presentan la limitante de la resolución [61][59], para ello los autores evaluaron su método respecto a los métodos propuestos en [63][62][55] obteniendo claramente un mejor resultado debido a que el mismo trabaja sobre puntos discretos y es independiente de cualquier modelo analítico. Como se puede observar en la Figura 28 y Figura 29, el método anteriormente expuesto (Discrete Geometry en color rojo) es el que presenta menor error de la raíz cuadrada media evaluando la curvatura estimada ante condiciones de densidad de puntos y ruido variables.

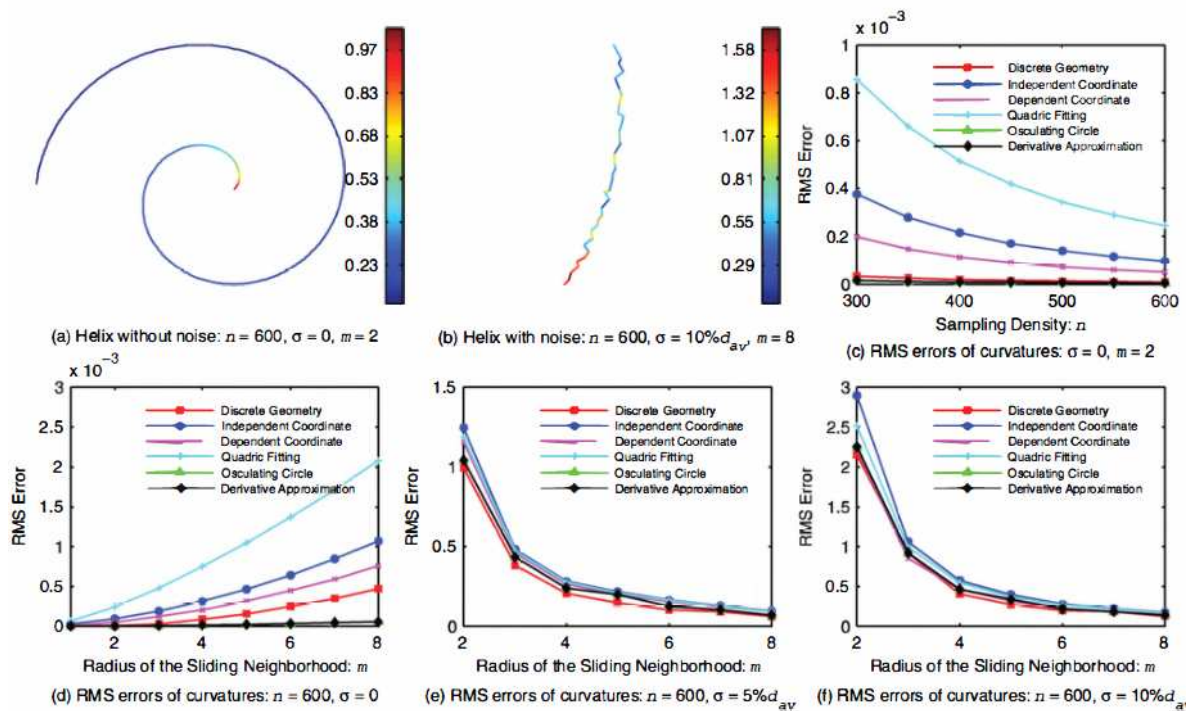


Figura 28. Comparación de diferentes métodos de estimación de propiedades geométricas de curvas en el espacio 3D para una curva helicoidal. Figura tomada de [59]

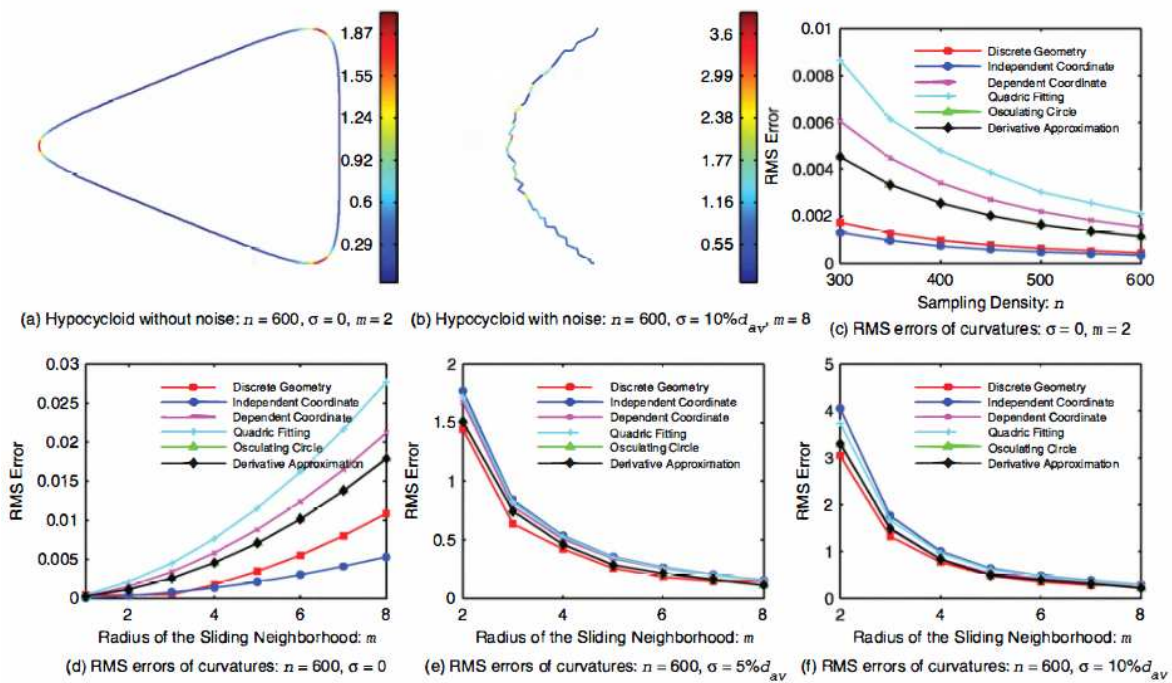


Figura 29. Comparación de diferentes métodos de estimación de propiedades geométricas de curvas en el espacio 3D para una curva hipocicloide. Figura tomada de [59].

En el presente trabajo, las propiedades de geometría diferencial, Torsión y Curvatura, son calculadas utilizando el método propuesto en [61][59] debido a las ventajas mencionadas anteriormente.

5.2.1. Torsión y Curvatura.

Como se explicó en la sección 5.1, las características principales de una curva son su torsión y curvatura, por lo cual, estas dos características serán utilizadas como descriptores para la posterior etapa de clasificación.

Desde un punto de vista geométrico, la curvatura es la tasa de cambio del vector tangente con respecto a la longitud de la curva, es decir, mide la magnitud del cambio de la curva respecto a un plano. Por otra parte, la torsión mide alguna magnitud que indica la rotación del plano osculador que es aquél que contiene a los vectores normal y tangente, con respecto a la longitud de la curva (ver Figura 30).

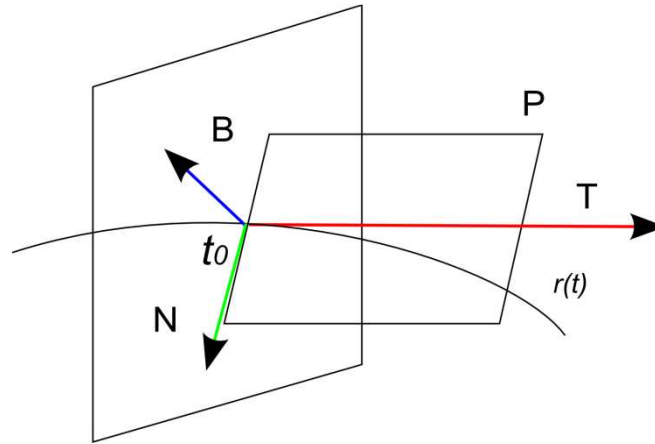


Figura 30. Vectores tangente (rojo), normal(verde), binormal (azul) y plano osculador.

La torsión y curvatura pueden ser calculadas mediante el método propuesto en [61][59] a través del siguiente conjunto de ecuaciones.

La derivada de una curva en un punto específico P_k puede ser definida como la pendiente de la tangente que pasa por dicho punto, donde la tangente es la línea recta que pasa por P_k y que minimiza la suma de los cuadrados de las distancias sobre el eje y entre esta y los puntos vecinos $\{p_j = (x_j, y_j) | k - m \leq j \leq k + m\}$.

Definición 1. Sea $y_i = fd(x_i)$ una función discreta. La derivada discreta de dicha función $y'_k = fd'(x_k)$ viene dada por:

$$y'_k = fd'(x_k) = \frac{\sum_{j=k-m}^{k+m} (x_j - x_k)(y_j - y_k)}{\sum_{j=k-m}^{k+m} (x_j - x_k)^2} \quad (5)$$

En geometría diferencial, es necesario parametrizar la curva para poder calcular sus propiedades. En este trabajo se utiliza el método de parametrización propuesto en [74] llamada parametrización del largo del segmento (*Chord Length Parametrization*).

Definición 2. Sea t_i el parámetro acumulativo del largo del segmento en el punto p_i definido como:

$$t_i = \frac{\sum_{j=1}^{i-1} |p_{j+1} - p_j|}{\sum_{j=1}^{n-1} |p_{j+1} - p_j|} \quad (2 \leq i \leq n) \quad (6)$$

Con $t_1 = 0$.

Ahora que se puede obtener la curva parametrizada $\alpha_d(t_i)$, es posible obtener las propiedades geométricas de esta a partir de las siguientes ecuaciones.

Definición 3. Sea $\alpha_d(t_i)$ una curva discreta parametrizada mediante el largo del segmento, se puede decir que:

$$(1) t_d(t_i) = \frac{\alpha'_d(t_i)}{|\alpha'_d(t_i)|}, \quad (7) \text{ es el vector tangente unitario de la curva discreta } \alpha_d(t_i).$$

$$(2) k_d(t_i) = \frac{|t'_d(t_i)|}{|\alpha'_d(t_i)|}, \quad (8) \text{ es la curvatura de la curva discreta } \alpha_d(t_i).$$

$$(3) \beta_d(t_i) = \frac{t'_d(t_i)}{k_d(t_i)|\alpha'_d(t_i)|}, \quad (9) \text{ es el vector normal principal de la curva discreta } \alpha_d(t_i).$$

$$(4) b_d(t_i) = t'_d(t_i) \times \beta_d(t_i), \quad (10) \text{ es el vector binormal de la curva discreta } \alpha_d(t_i).$$

$$(5) \tau_d(t_i) = \frac{b'_d(t_i) \cdot \beta_d(t_i)}{|\alpha'_d(t_i)|}, \quad (11) \text{ es la torsión de la curva discreta } \alpha_d(t_i).$$

En la definición 3, la derivada de la curva paramétrica $\alpha'_d(t_i)$ puede ser calculada mediante la definición 1. Por ejemplo, $\alpha'_d(t_i) = (x'_d(t_i), y'_d(t_i), z'_d(t_i))$, donde:

$$x'_d(t_i) = \sum_{j=i-m}^{i+m} (t_j - t_i)(x_j - x_i) / \sum_{j=i-m}^{i+m} (t_j - t_i)^2 \quad (12)$$

$$y'_d(t_i) = \sum_{j=i-m}^{i+m} (t_j - t_i)(y_j - y_i) / \sum_{j=i-m}^{i+m} (t_j - t_i)^2 \quad (13)$$

$$z'_d(t_i) = \sum_{j=i-m}^{i+m} (t_j - t_i)(z_j - z_i) / \sum_{j=i-m}^{i+m} (t_j - t_i)^2 \quad (14)$$

En la Figura 31 se pueden observar los vectores tangente, normal y binormal estimados mediante las ecuaciones anteriores.

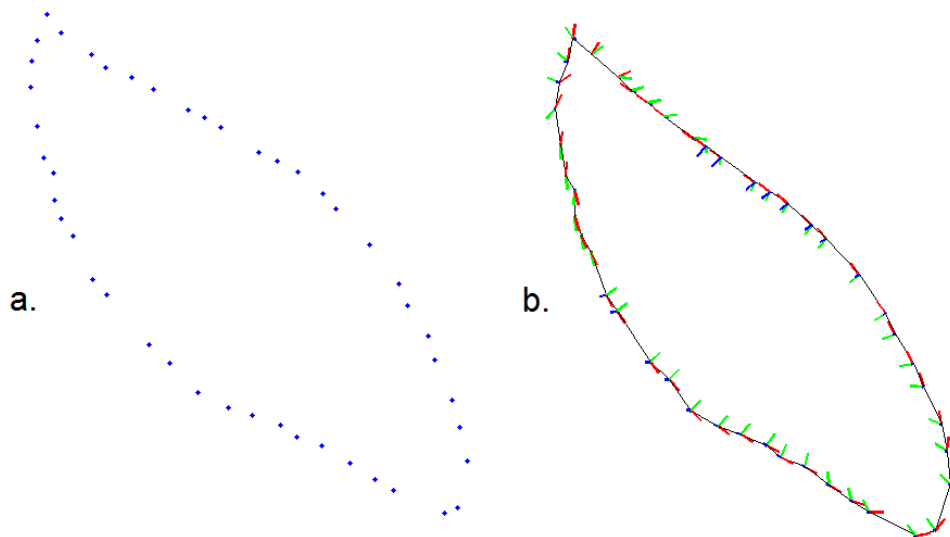


Figura 31. a) puntos originales. b) rojo: vector tangente, verde: vector normal, azul: vector binormal.

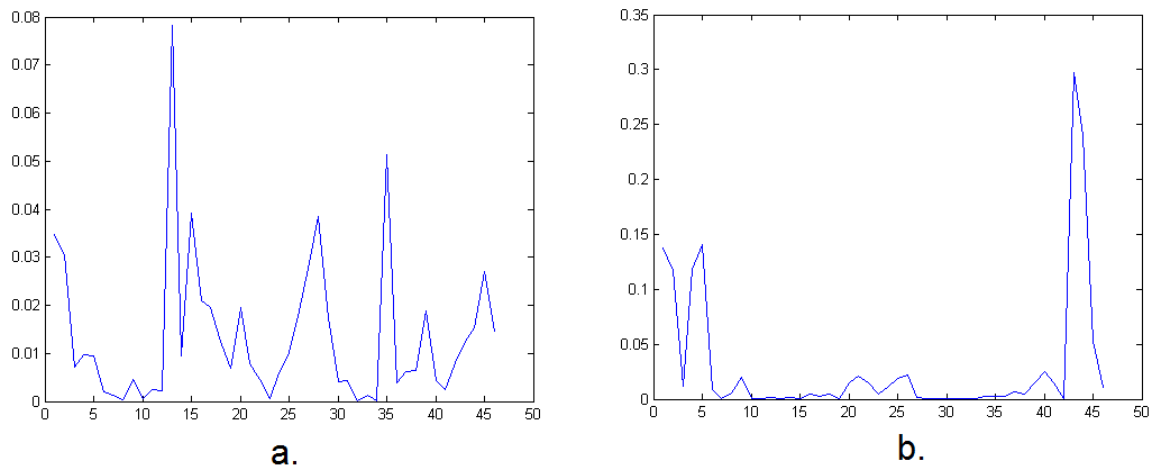


Figura 32. a) Torsión y b) curvatura para la curva 3D que aparece en la Figura 31.

En la Figura 32 se puede observar el comportamiento de la torsión y la curvatura de la curva en el espacio 3D de la Figura 31, donde el eje y representa el valor de la diferencia absoluta entre la torsión - curvatura del punto p_i respecto al punto p_{i+1} y el eje x cada uno de los puntos de la curva 3D, solo a modo de ilustración.

Finalmente, se calcula la varianza de las torsiones y curvaturas estimadas para cada punto de la curva mediante:

$$S = \frac{\sum_{i=0}^n (x_i - \bar{x})^2}{n} \quad (15)$$

Esta varianza representa una medida de distorsión de la curva debido a que según la definición de objeto de forma libre expuesta en [2], las superficies de estos objetos son suaves y diferenciables, por lo que un hueco perteneciente al mismo debería presentar torsiones y curvaturas también suaves, lo que indica que la varianza de estas características es directamente proporcional al nivel de distorsión presente en el mismo.

5.3. Descriptores Globales de Forma de la Curva.

Los trabajos encontrados en la literatura sobre clasificación de huecos en superficies 3D se limitan a enfrentar este problema a través de mediciones geométricas locales de la curva, específicamente torsión y curvatura [9][43] mediante el ajuste de curvas analíticas sobre el conjunto de puntos discretos de la curva. En algunos casos la clasificación ha sido mejorada mediante mediciones de la densidad y comportamiento de las regiones de la superficie que circundan el contorno del hueco [44][45], pero esto implica contar siempre con información completa de la superficie del objeto, es decir, no funcionaría en casos en los que se cuenta con los datos del contorno del hueco únicamente y por otra parte, el costo computacional de obtener información del comportamiento de las zonas de la superficie del objeto cercanas al hueco es

mayor que el análisis de los datos del contorno. En [44][45] la medición del comportamiento del entorno del contorno del hueco se realiza mediante clusterización, lo cual implica un alto costo computacional y una restricción a la hora de seleccionar el número correcto de clústeres.

En este trabajo se propone utilizar descriptores geométricos globales de la forma del objeto, es decir, descriptores que permitan obtener información del comportamiento general del hueco, lo cual no tiene la limitante de necesitar información extra y por otra parte permite realizar ciertas operaciones de clasificación inicial o gruesa de igual forma que lo haría un humano, por ejemplo, comparar el contorno del hueco respecto a figuras conocidas, donde se plantea el supuesto de que si el hueco es muy parecido a una figura conocida, se puede tomar una decisión de clasificación antes de llegar a las mediciones locales de curvatura y torsión.

5.3.1. Representación de la curva.

La representación de la curva del contorno del hueco como una secuencia de puntos $p_0, p_1, p_2, \dots, p_n$ para una curva de n puntos, no permite obtener mayor información que la que brindan los descriptores de geometría diferencial. Por esta razón es necesario representar la curva mediante algún otro modelo que brinde mayor información. Herbert Freeman en su famoso artículo de revisión bibliográfica *Computer Processing of Line-Drawing Images* [75], realiza un recorrido por las diferentes formas de representación de líneas en imágenes digitales tanto en el espacio bidimensional como en el tridimensional. En este trabajo se puede apreciar que la constante es trabajar con representaciones que codifiquen de alguna manera, información sobre los cambios de dirección de los puntos de la línea. Resalta entre estas representaciones los códigos de cadena, debido a que son fácilmente extensibles al espacio 3D [75], además cumple con las condiciones propuestas por el mismo Freeman[75] para cualquier representación de curvas, las cuales son:

- Debe preservar de manera fiable la información de interés.
- Debe permitir un almacenamiento compacto y ser conveniente para la visualización.
- Debe facilitar cualquier procesamiento requerido.

Los códigos de cadena son ampliamente utilizados en el procesamiento de imágenes digitales 2D. Sin embargo, hace algunos años esta forma de representación de curvas ha tomado auge en el procesamiento de imágenes 3D y en el área del reconocimiento de patrones [76][77][78][79][80][81][82][83][84].

Obtener esta representación para una serie de puntos, requiere que estos cumplan con la característica de resolución constante en el espacio 3D (Figura 24 a.). Por lo que resulta necesario realizar un proceso de remuestreo estructurado de los puntos de la curva previo al cálculo de los códigos de cadena.

5.3.2. Remuestreo estructurado de los puntos de la curva.

El método que se presenta a continuación trabaja con un vector ordenado de puntos (ver Figura 33) en coordenadas euclídeas x, y, z pertenecientes al contorno del hueco en la superficie del objeto

de forma libre. Cabe aclarar que hasta este paso ha sido necesario identificar el hueco en el objeto y extraer las coordenadas de los vértices del borde del mismo.

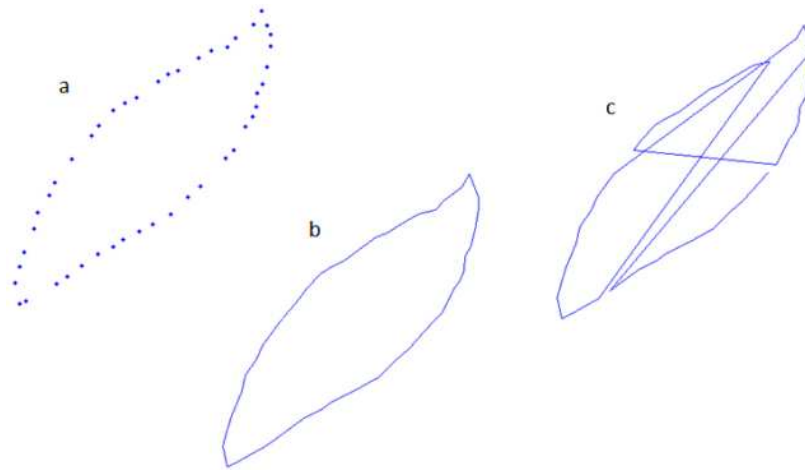


Figura 33. a) vector de puntos del borde del hueco. b) vector ordenado. c) vector no ordenado.

A continuación se describe cada una de las etapas del método de remuestreo estructurado propuesto.

5.3.2.1. Ajuste de puntos mediante una curva NURBS

A diferencia de la medida de distorsión propuesta en la sección 5.2.1, la idea de los descriptores globales es obtener información geométrica global del contorno del hueco, por lo que la distorsión original de dicho contorno constituye una desventaja para este caso al introducir ruido en los cambios de dirección de los puntos del contorno del hueco. En otras palabras, para mejorar la medición de las propiedades globales de forma del contorno del hueco, es necesario suavizar los datos iniciales. Esto se puede lograr mediante el ajuste de una curva paramétrica a los puntos originales del borde. En este trabajo se hizo uso de las curvas NURBS para hallar dicha relación entre puntos, debido a que estas han sido ampliamente adoptadas para los sistemas de modelado computacional debido a su generalidad según expresa Lengyel en [85]. Por otra parte según Piegl [86] las curvas NURBS se han convertido en el estándar industrial de facto y pueden ser utilizadas para el llenado de huecos en superficies 3D presentando como ventajas generalidad en la solución y control geométrico local de los datos, entre otras. Cabe aclarar que el tipo y orden de la curva paramétrica depende de la aplicación del método, y que estimar esto implica un trabajo netamente experimental, esto a la limitante mencionada en una ocasión anterior respecto a que es casi imposible que un tipo de curva analítica en específico se ajuste de forma correcta a todas las posibles formas que puede tomar una curva en el espacio 3D. Para mayor información respecto a las ventajas y limitaciones de las curvas NURBS son tratadas en [73].

Las NURBS (*Non Uniform Rational B-Splines*) son funciones polinomiales a trozo (*piecewise*) evaluadas vectorialmente según la siguiente expresión matemática[73]:

$$S = \frac{\sum_{i=0}^n w_i P_i N_{i,p}(u)}{\sum_{i=0}^n w_i N_{i,p}(u)} \quad (16)$$

Donde C es el vector de puntos final ajustados mediante la curva NURBS, w_i son los pesos, P_i son los puntos de control o los puntos originales del contorno del hueco, y $N_{i,p}(u)$ son las funciones base de las B-Spline normalizadas de grado p , definidas recursivamente según:

$$N_{i,0}(u) = \begin{cases} 1 & \Leftrightarrow k_i \leq u \leq k_{i+1} \\ 0 & \Leftrightarrow \text{otherwise} \end{cases} \quad (17)$$

$$N_{i,p}(u) = \frac{u - k_i}{k_{i+p} - k_i} N_{i,p-1}(u) + \frac{k_{i+1} - u}{k_{i+1} - k_{i+2}} N_{i+1,p-1}(u) \quad (18)$$

Donde k es el vector de nudos (*knots vector*). Generalmente este vector toma la forma $K = \{\alpha, \alpha, \dots, \alpha, k_{p+1}, \dots, k_{m-p-1}, \dots, \beta, \beta, \dots, \beta\}$, en la práctica, comúnmente $\alpha = 0$ y $\beta = 1$ según [73].

Para resolver la función de la curva NURBS se utilizó el algoritmo clásico CoxdeBoor [87][88] cuyo resultado puede observarse en la Figura 34.

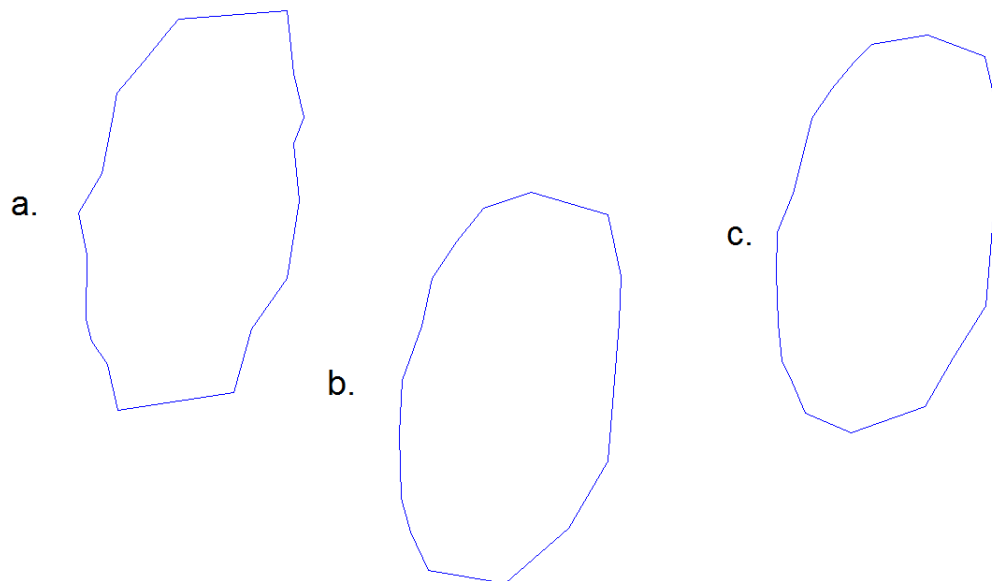


Figura 34. De izquierda a derecha a) Puntos Originales b) Puntos de la curva NURBS 75% c) Puntos de la curva NURBS 100%.

Es claramente observable que la suavidad lograda mediante el ajuste de la curva NURBS es mayor que en los puntos originales y esta aumenta directamente respecto al nivel de resolución o nivel de detalle de la curva. Este tipo de curvas paramétricas permite tanto reducir como aumentar el número de puntos final de la curva. El objetivo de este proceso es obtener un nuevo vector de puntos $V' = \{v'_1, v'_2, \dots, v'_n\}$ del cual conocemos la relación entre cada secuencia v'_i, v'_{i+1} y se ajusta de forma más adecuada al comportamiento suave y uniforme de los objetos de forma libre. Determinar el número final de puntos de la curva NURBS dependerá de la aplicación que se esté trabajando.

5.3.2.2. Remuestreo estructurado mediante el algoritmo de Bresenham 3-D.

Una vez han sido suavizados los datos del contorno, es necesario realizar el remuestreo estructurado con el fin obtener un nuevo conjunto de puntos $p'_0, p'_1, p'_2, \dots, p'_n$ tal que tengan una resolución constante y una distribución espacial homogénea i.e. que cada uno de estos puntos pueda ser accesible mediante coordenadas enteras i, j, k .

Para este proceso de estructuración se utilizó el algoritmo de discretización de rectas propuesto por Jack Bresenham [89] en 1965 (ver Algoritmo 1), el cual es un estándar de facto en la industria de la computación gráfica, debido a su bajo costo computacional al trabajar únicamente con operaciones enteras. En este caso se utiliza la implementación de Bresenham 3D propuesta por Kaufman en [90].

```

procedure linea(x0,x1,y0,y1,col:integer);
var x,y,dx,dy,e,ix,iy:integer;
begin
  dx := x1 - x0; dy := y1 - y0;
  ix := 2*dx; iy := 2*dy;
  y := y0; e := iy - dx;
  for x := x0 to x1 do begin
    putpixel(x,y,col);
    e := e+iy;
    if e>0 then do begin
      y := y+1;
      e := e-ix;
    end;
  end;
end;

```

Algoritmo 1. Algoritmo de discretización de rectas 2D de Bresenham. Tomado de [91].

El algoritmo de Bresenham toma dos puntos en el espacio y genera un nuevo conjunto de puntos que aproxima de la mejor forma una línea recta entre ambos, bajo una resolución espacial establecida. Este nuevo conjunto tiene la propiedad de moverse a intervalos espaciales regulares, de tal forma que para cada punto p_i de mi nuevo conjunto, p_{i-1} y p_{i+1} pertenecen a los 26 vecinos cercanos (ver Figura 35).

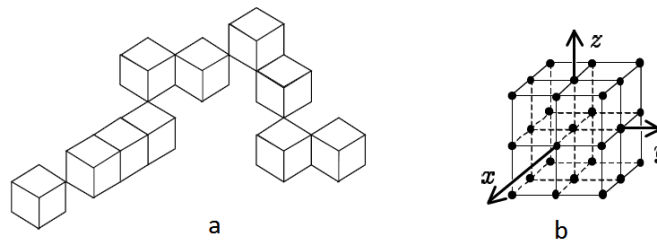


Figura 35. Datos estructurados en el espacio 3D.

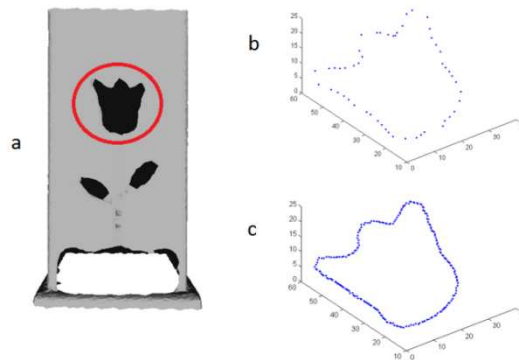


Figura 36. Remuestreo estructurado mediante el algoritmo Bresenham 3D.

El resultado del proceso de remuestreo estructurado mediante el método propuesto, puede observarse en la Figura 36.

5.3.3. Código de cadena para el conjunto estructurado de puntos.

Los códigos de cadena son una representación de curvas discretas como un conjunto de segmentos de longitud y dirección especificadas [18]. Para ello, se suele utilizar un cuadrícula de conectividad 4 u 8 para el caso de imágenes bidimensionales. En la Figura 37 se puede observar el código de cadena de conectividad 8 para una curva bidimensional.

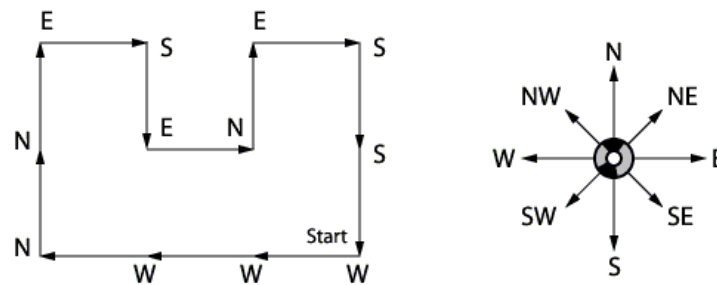


Figura 37. Código de cadena con conectividad 8.

El resultado de esta representación es un histograma que muestra la frecuencia con la que aparece una determinada dirección. En el ejemplo de la Figura 37, la secuencia de direcciones sería: [W WW N E S E N E S S] y su respectivo histograma de la Figura 38.

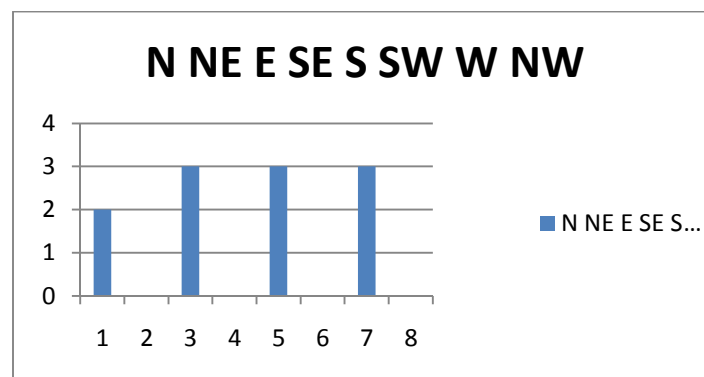


Figura 38. Histograma del código de cadena.

En el caso de las curvas 3D estructuradas, se trabaja con conectividad 26 o grillas de 3x3x3 (ver Figura 39).

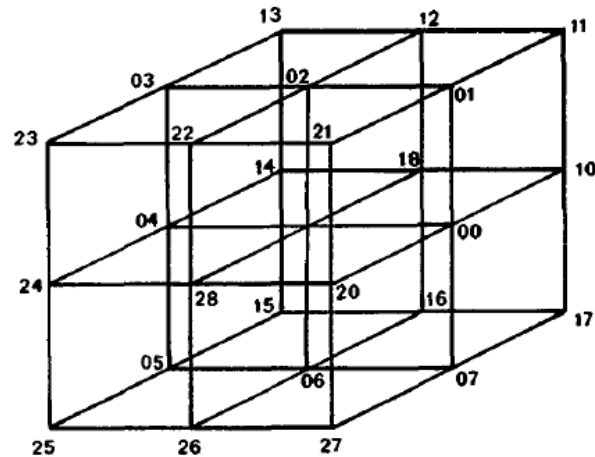


Figura 39. Códigos de cadena para líneas 3D estructuradas. Imagen tomada de [75].

En la figura anterior se observa el concepto de código de cadena 3D propuesta por Freeman en [75]. Cada una de las intersecciones de las líneas del cubo, indican una nueva dirección respecto a un punto situado en el centro del mismo. Obsérvese que las direcciones están organizadas en el sentido de las manecillas del reloj respecto los valores positivos de todos los ejes x, y, z , lo cual permite el mejor entendimiento del histograma del código de cadena.

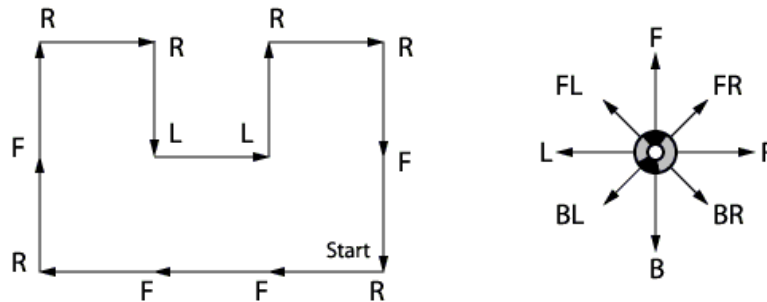
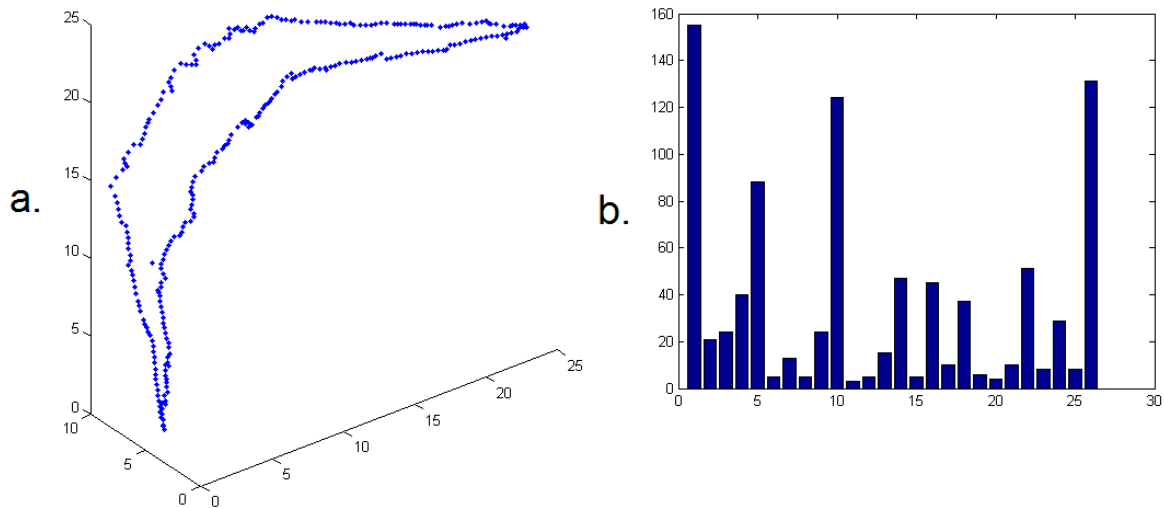


Figura 40. Código de cadena relativo.

Los códigos de cadena se pueden ser calculados mediante dos aproximaciones generales: *códigos de cadena absolutos* y *códigos de cadena relativos*. Los primeros (ver Figura 37), parten del hecho de que el sistema de coordenadas de la curva es constante y que esta no será sometida a transformaciones espaciales comunes. En estos, el conjunto de direcciones representan el conjunto absoluto de puntos de referencia. Esta primera aproximación tiene como limitante que el código de cadena entre dos curvas variará significativamente si alguna es sometida a una transformación espacial. Por el contrario, los códigos de cadena relativos (ver Figura 40) utilizan como sistema de referencia la dirección del punto anterior de la curva, por lo que el punto inicial de la curva p_0 se inicializa en cualquier dirección, se resuelve la dirección de los demás puntos y por último la dirección del punto p_0 se resuelve respecto al punto p_{n-1} para una curva de n puntos. Esta aproximación permite representar cualquier curva discreta estructurada independiente a su escala, traslación y rotación.

Una de las funciones principales de la representación de líneas mediante códigos de cadena es el reconocimiento de patrones, lo cual resulta de gran interés para el método clasificación de este trabajo. Un método para obtener una medida de disimilitud de forma para curvas 3D es propuesto en [92] en que se utiliza una representación de códigos de cadena de 5 direcciones para representar cualquier curva discreta en el espacio 3D y se mide que tan parecida es una curva a otra, encontrando subcadenas comunes entre los códigos de cadena de ambas curvas. En [77] se utilizan los códigos de cadena para representar estructuras comunes en computación gráfica tales como árboles, venas, plantas, etc, con lo que se logra reducir el espacio de almacenamiento en aplicaciones 3D. Wagen knecht presenta en [83] un método para combinar información sobre clasificación de objetos 3D con regiones 2D interactivas que permitan seleccionar regiones de interés en la escena. Nuevas formas de representación de códigos de cadena más compactas y flexibles son presentadas en [76][93].

El método implementado en este trabajo asume que los datos de entrada pertenecen a una curva en el espacio 3D debidamente estructurada según el método propuesto en la sección 5.3.2.



En la Figura 41 se puede observar una curva discreta en el espacio 3D y su respectivo histograma del código de cadena implementado. Este histograma muestra la cantidad de ocurrencias de cada una de las 26 posibles direcciones de la cadena. Cabe aclarar que esta representación, al ser un histograma de direcciones, conlleva a la pérdida total de la información local de la curva, pero por el contrario, permite extraer información del comportamiento general o global de la misma, lo cual es de interés para la extracción de las características que serán descritas a continuación.

5.3.4. Simetría y similitud de forma.

Para el estudio de curvas en el espacio 3D, aparte de las propiedades medibles mediante geometría diferencial, son pocas las características geométricas relevantes. Un estudio realizado en las etapas iniciales de este trabajo muestra que las características geométricas simetría axial y el parecido o similitud con alguna forma geométrica común, son propiedades relevantes a la hora de discriminar si un hueco es real o una anomalía. A continuación se presentan estos dos descriptores basados en el estudio del histograma de los códigos de cadena presentados anteriormente.

5.3.4.1. Simetría

La simetría axial puede ser definida como la simetría alrededor de un eje, de forma tal que si un objeto es dividido por un plano que contiene dicho eje, las dos partes resultantes son similares en la forma como se puede apreciar en la Figura 42.

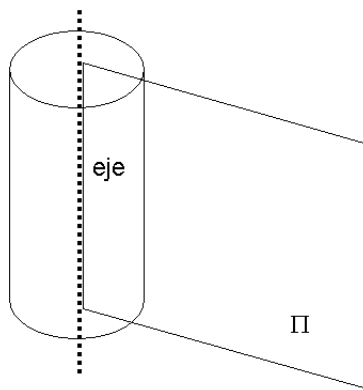


Figura 42. Geometría Axial.

Claramente, una limitante para implementar una medida de simetría axial, radica en la selección del eje de simetría. Una aproximación sería tomar como eje las líneas conformadas por los puntos p_i, p_j donde $1 \leq i \leq n$ y $1 \leq j \leq n$ para una curva con n puntos, utilizar alguna métrica de simetría para cada caso y por último realizar alguna operación estadística para obtener un solo resultado. Claramente esto conlleva a una limitante en la complejidad computacional del algoritmo. Algo parecido es realizado en [94] (ver Figura 43).

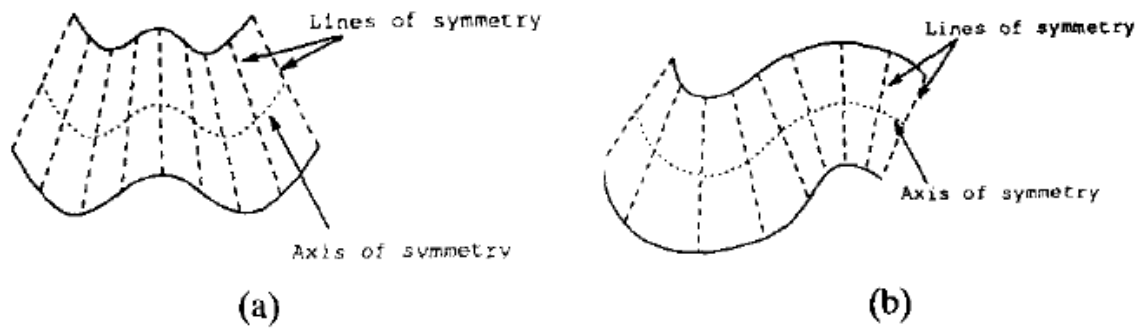


Figura 43. Simetría axial en el contorno de un hueco sobre una superficie 3D. Imagen tomada de [95].

Debido a lo anterior, en este trabajo se propone un método para medir la simetría axial basado en el estudio del histograma del código de cadena absoluto de la curva.

Si entendemos el histograma de un código de cadena como la frecuencia en que se presentan las direcciones de la curva, es posible realizar alguna medida de simetría sobre un subconjunto de direcciones en el espacio. Para esto se definen los siguientes subconjuntos de dirección como se muestra en la Figura 44 sobre los cuales se mide la simetría.

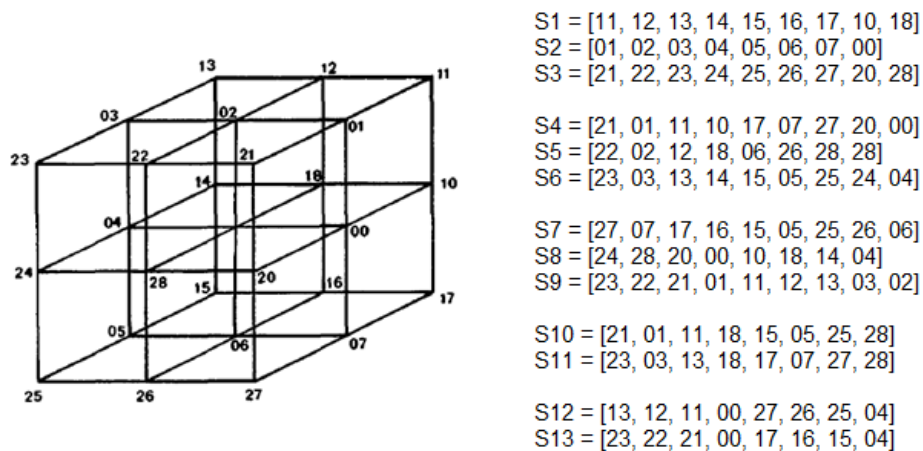


Figura 44. Conjuntos de simetría.

Nótese que cada subconjunto de simetría o de dirección, representa un plano en el espacio 3D. Entonces, cada subconjunto puede ser visto como una proyección del contorno 3D sobre un plano, lo cual simplifica el estudio de la simetría a una forma bidimensional.

El orden de las direcciones en cada subconjunto de simetría, viene dado por el sentido de giro de las manecillas del reloj o el sentido inverso. Lo anterior es necesario para que las direcciones contrarias, permanezcan opuestas en el histograma.

Para cada uno de estos conjuntos se estima una medida de simetría a través de la oblicuidad del histograma definida como:

$$u = \frac{\sum_{i=1}^n (Y_i - \bar{Y})^3}{(N-1)s^3} \quad (19)$$

Donde \bar{Y} es el promedio de los datos y N el número de datos o direcciones del histograma.

La oblicuidad es una medida de la asimetría de la distribución de probabilidad de una variable real, en otras palabras mide la simetría o más bien la falta de simetría. Un histograma simétrico será aquel que es idéntico para el lado derecho como para el lado izquierdo respecto al punto central. La oblicuidad puede tomar valores negativos o positivos y su interpretación es la siguiente: si el valor es positivo, el histograma tiene una mayor distribución hacia la derecha. Si el valor es negativo, el histograma tiene una mayor distribución hacia la izquierda. Si el valor es cero, indica que el histograma es totalmente simétrico.

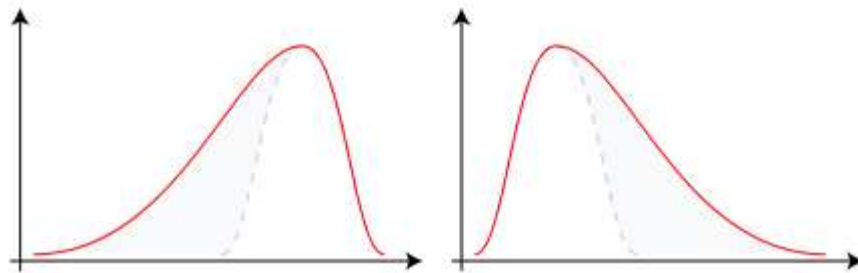


Figura 45. Oblicuidad negativa y oblicuidad positiva.

Para este caso la medida de simetría del contorno del hueco basada en el histograma de su código de cadena h viene dada por:

$$S(h) = \operatorname{argmin}(u(S_i)), i = \{1, 2, 3, \dots, 13\} \quad (20)$$

Para un número n de espacios de simetría S_i . En otras palabras, la medida de simetría del contorno del hueco es calculada como la menor oblicuidad presente en los espacios de los subconjuntos de simetría.

5.3.4.2. Similitud de forma

Debido a que el histograma del código de cadena relativo brinda información acerca del comportamiento global de la curva independiente de las transformaciones espaciales que a ella se apliquen, medir la similitud de una curva respecto a otra, se reduce a un problema de comparación de histogramas. En este trabajo se utiliza un método de comparación de histogramas [96] basado en distancia euclídea, para evaluar la similitud de color entre dos frutas. Si bien no es el caso de evaluación del color, se asume que el método de comparación es independiente a la aplicación.

La comparación de histogramas viene dada por las siguientes ecuaciones:

$$E_{MJ}^X = \sqrt{\frac{1}{n} \sum_{i=0}^{255} M(i)^2 - J(i)}, \quad X = \{K_i; i = H, S\} \quad (21)$$

$$\varepsilon(M, C) = \frac{E_{MC}^H + E_{MC}^S}{2} \quad (22)$$

Donde E_{MJ}^X es el error medio cuadrado de los histogramas M and J. $\varepsilon(M, C)$ es el promedio de los errores entre histogramas H y S de una curva M y un conjunto de curvas modelo C.

Esta medida puede ser utilizada de dos formas. La primera es para reconocer una curva que se sabe pertenece a una clase dentro de un conjunto de curvas modelo, donde la curva evaluada pertenecerá a la clase con la cual presente menor diferencia. La segunda forma es para determinar si la curva es o no una figura geométrica establecida. Para lo que se evalúa la curva respecto a unas curvas modelos, pero se estima un umbral experimental que restringe el mínimo de similitud necesaria para considerar la curva como una figura geométrica. Esta última forma es la utilizada en este trabajo.

5.4. Resumen de los descriptores tratados.

En la siguiente tabla se resumen los descriptores propuestos en este trabajo y se realiza una breve descripción de los mismos.

Tabla 2. Resumen de los descriptores propuestos.

Descriptor	Característica
Varianza Torsión	Propiedad que aproxima la distorsión general de la curva.
Varianza Curvatura	Propiedad que aproxima la distorsión general de la curva.

Simetría Axial	Propiedad que estima que tan simétrica es la curva respecto a un determinado subconjunto de dirección del código de cadena.
Comparación de histogramas de códigos de cadena.	Propiedad que establece una relación de similitud entre los histogramas de los códigos de cadena de dos curvas.
Tamaño	Propiedad que indica cuantas aristas tiene el contorno del hueco.

CAPÍTULO 6

CLASIFICACIÓN DE HUECOS EN SUPERFICIES 3-D

Determinar la clase a la que pertenece un contorno de un hueco, puede ser entendido como clasificar el mismo, utilizando para ello una serie de características medibles sobre este y un conocimiento a priori de las clases a las que se quiere llegar. Esto es tema de investigación de una disciplina científica conocida como *Reconocimiento de Patrones*, que según Pajares en [18] define esta como “*la disciplina científica cuyo objetivo es la clasificación de objetos en un cierto número de categorías o clases*”.

Dentro de la literatura se mencionan 4 enfoques principales para abordar el problema del reconocimiento de patrones: *Redes Neuronales*, *Ajuste de Plantillas*, *Reconocimiento de Patrones Sintáctico* y *Reconocimiento Estadístico de patrones* [97]. A su vez, se denomina a un clasificador como supervisado o de aprendizaje inducido cuando se tiene un conjunto de clases y de objetos ejemplo los cuales están debidamente etiquetados con alguna clase apropiada. Por el contrario, se denomina un clasificador como no-supervisado cuando este debe determinar que objetos del conjunto de ejemplo, deben ser agrupados unos con otros. Dicho de otra forma, el clasificador debe determinar las clases mismas. Según lo anterior, la calidad final del clasificador construido, ya sea supervisado o no-supervisado, dependerá en mayor parte de la calidad de los datos ejemplo, i.e. de la calidad de las características medidas en los objetos ejemplo.

En la construcción de cualquier clasificador, es necesario afrontar las siguientes etapas generales [98]:

- **Selección y extracción de características:** Radica en obtener y seleccionar el subconjunto de características o descriptores relevantes a través de técnicas tales como el análisis de componentes principales.
- **Selección y aprendizaje del método de clasificación:** Este algoritmo es seleccionado a partir del tipo de información que se desea clasificar. Para cada algoritmo, existen una serie de mecanismos para realizar el entrenamiento.
- **Medición y análisis del Error del clasificador:** Este análisis es realizado con un subconjunto de los datos clasificados a priori en el caso de la clasificación supervisada. En el caso de la clasificación no supervisada, un experto le da un sentido físico a la clasificación realizada.

La correcta selección de los métodos que se utilizarán en cada una de las etapas del proceso de clasificación de acuerdo a las necesidades y requerimientos específicos del problema, determinarán en gran parte la calidad del clasificador final. La selección de dichos métodos será el propósito de este capítulo.

6.1. Método de Selección de Características

La selección de características es un tema que ha sido ampliamente estudiado en estadística y reconocimiento de patrones [99]. Su importancia radica en que afecta directamente el desempeño del clasificador final, así como en su complejidad y precisión[100]. Existen diversos métodos comúnmente utilizados para la selección de características, tales como selección secuencial hacia adelante (SFS)[97], búsqueda ortogonal hacia adelante [101], selección a través de la identificación de variables críticas de componentes principales [102] Ranking por el criterio de la separabilidad de las clases[103], y combinación con componentes principales[104].

Según Mery en [98], el método de selección secuencial hacia adelante presenta los mejores resultados, sobre todo por su baja complejidad, lo que permite implementarlo en sistemas de inspección en tiempo real. Este algoritmo funciona agregando características de forma secuencial que maximicen la separabilidad de clases cuando se sumen al resto de características que se han ido agregando (ver Figura 46).



Figura 46. Comportamiento del método de selección secuencial hacia adelante (curva), en el espacio de búsqueda (elipse).

Otra característica atractiva de este método de selección de características para este trabajo, radica en que el mismo alcanza su mejor comportamiento cuando el espacio de características óptimas es pequeño, y debido a que para clasificar el contorno del hueco se utilizarán 5 características, es de suponer que el comportamiento de este método será el mejor.

6.2. Métodos de Clasificación

Según lo comentado en la introducción de este capítulo, el problema de clasificación de este trabajo puede entenderse en el dominio del aprendizaje supervisado de máquina o de los métodos de clasificación supervisada. Dos algoritmos principales y prevalentes de aprendizaje supervisado comúnmente utilizados en el análisis de patrones en visión por computador son las máquinas de soporte vectorial (SVM por sus siglas en inglés) y los algoritmos boosting, ambos temas de reciente investigación en la comunidad científica de aprendizaje supervisado[105]. Por otra parte, la utilización de árboles de decisión para la clasificación supervisada, es una práctica bastante conocida en la comunidad de aprendizaje de máquina [100][106], debido a su robustez, tiempo de ejecución y al hecho de que el conocimiento generado, es fácilmente interpretable por el usuario.

Las máquinas de soporte vectorial, de gran utilidad para el aprendizaje de máquina con muestras relativamente pequeñas, se fundamentan en la teoría estadística. Este método presenta mejoras significativas respecto a los métodos convencionales de aprendizaje supervisado, entre ellos las redes neuronales, y ha sido ampliamente utilizado en aplicaciones de clasificación de documentos, reconocimiento de letra manuscrita, clasificación de imágenes y procesamiento biológico de información.

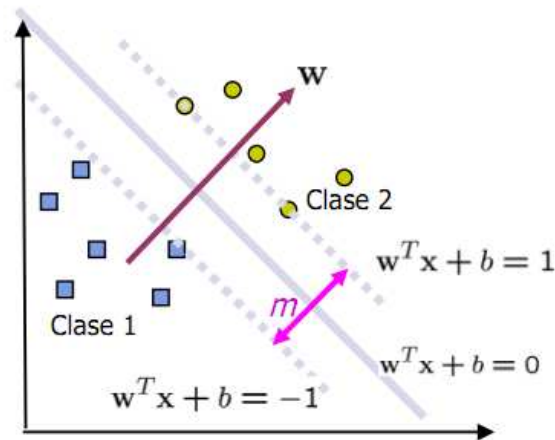


Figura 47. El hiperplano o frontera debe estar tan lejos como se pueda de los datos de ambas clases.

Una máquina de soporte vectorial basa su funcionamiento mapeando los datos de entrada de clasificación a un espacio de dimensión mayor, y encuentra el hiperplano que maximice el margen m entre las clases en este espacio (ver Figura 47). Los vectores de soporte entonces, son los puntos de entrada que solucionan el hiperplano óptimo. Debido a la facilidad de este método para trabajar con imágenes, aplicaciones tales como las de OCR lo utilizan.

Los algoritmos boosting tienen como idea principal, combinar reglas y clasificadores simples para formar un único clasificador mejor que los demás. El principio del algoritmo boosting es reunir y combinar una serie de reglas para producir una regla más fuerte[105]. Schapire [107] propone el primer algoritmo boosting de tiempo polinomial y más tarde, este mismo propone el primer algoritmo boosting aplicable en práctica conocido como el algoritmo AdaBoost [108]. Este algoritmo es muy popular en aplicaciones de procesamiento de imágenes y visión por computador. La

implementación del algoritmo AdaBoost es relativamente sencilla y su esquema general puede ser observado en la Figura 48.

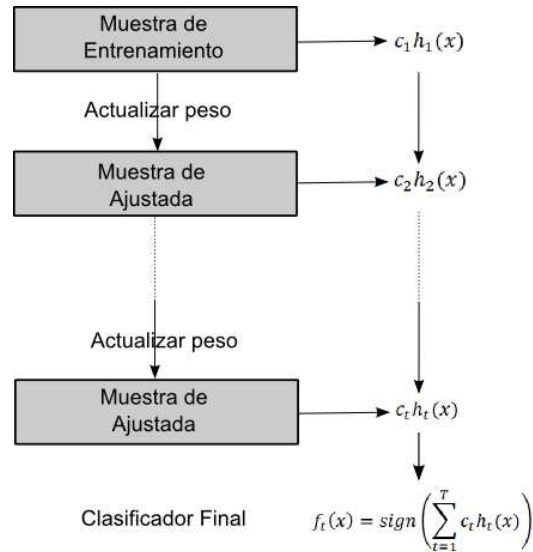


Figura 48. Esquema general del algoritmo AdaBoost.

Para generar árboles de decisión en aprendizaje supervisado de máquina, existen diversos algoritmos, entre estos, los algoritmos ID3, CHAID y MARS. El algoritmo más sofisticado y de mayor aplicación en aprendizaje supervisado mediante árboles de decisión, es el algoritmo C4.5 [109], el cual es una mejora del algoritmo ID3 [110]. Este algoritmo comienza eligiendo una característica como nodo raíz, se generan tantos hijo como subgrupos de datos correspondientes a valores que puede tomar el nodo padre y se evalúa la entropía de estos subgrupos. Si la entropía de los hijos es menor que la del padre, entonces se ha ganado información respecto a la clasificación. La diferencia principal entre los algoritmos C4.5 e ID3 radica en que el primero permite generar árboles más reducidos y con menor probabilidad de sobreajuste. Este algoritmo es robusto, preciso, rápido y produce una estructura de conocimiento comprensible. En la Figura 49 se observa un árbol de decisión para determinar si se debe jugar o no un partido de golf de acuerdo a las condiciones climáticas.

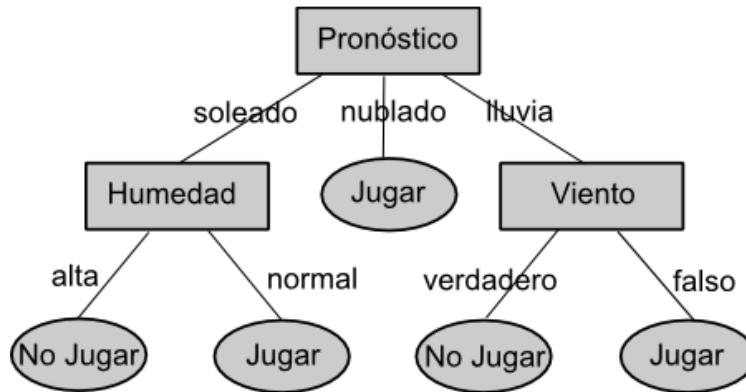


Figura 49. Árbol de decisión para un juego de golf.

Nótese que la interpretación de este tipo de modelos es directa, donde los nodos padre son las variables de clasificación, las ramas son los valores que puede tomar cada nodo y los nodos hoja representan la clasificación o clase estimada. La implementación de árboles se limita entonces a condicionales respecto a las variables (nodos) y los valores que toma cada una (ramas), por lo que es una alternativa ampliamente utilizada en sistemas de inspección industrial. Según la Figura 49, podemos decir que si el pronóstico del tiempo es soleado y la humedad es normal, entonces se puede jugar golf.

6.3. Método de Medición del Error de los clasificadores

Para el aprendizaje de máquina supervisado, generalmente se cuenta con un grupo de datos de entrenamiento y otro de validación. Esto implica contar con una cantidad correcta de datos para ambos grupos, lo cual no siempre es posible en problemas de visión por computador. Para estos casos, un método comúnmente utilizado para validar el rendimiento del clasificador, es la validación cruzada [111]. La validación cruzada divide el número de muestras totales de trabajo en F subgrupos elegidos de manera aleatoria. $F-1$ grupos son utilizados para entrenar el método de clasificación y el grupo restante es utilizado para validar en clasificador entrenado. Este proceso se repite F veces y por último se promedian los rendimientos individuales del clasificador y este valor se toma como el rendimiento general del mismo. Lo anterior se presenta como una alternativa de validación atractiva para las condiciones de muestreo en este trabajo.

CAPÍTULO 7

EXPERIMENTOS Y RESULTADOS

El presente capítulo tiene como fin describir el experimento realizado para evaluar la clasificación automática de anomalías relacionadas con ausencia de información en superficies tridimensionales de objetos de forma libre, según las características y métodos de clasificación descritos en capítulos anteriores.

7.1. Descripción del Experimento

Para este experimento, se utilizó una base de datos de 128 contornos para los cuales se obtuvo cada una de las características descritas en el capítulo 5 y les fue relacionada una clasificación real (**Naturaleza**) realizada por un experto humano. Estos datos pueden ser observados en la Tabla 3. Los contornos de trabajo fueron extraídos de 7 objetos escaneados mediante un sensor de rango Minolta 9i y de diversos repositorios de modelos 3D accesibles en internet.

Si bien en internet es posible encontrar una gran cantidad de repositorios de modelos 3D, generalmente estos se encuentran integrados y ajustados. Esto presenta una limitación fuerte para la investigación en corrección de anomalías en modelos 3D, no solo para el caso de las relacionadas con ausencia de información, sino las relacionadas con ruido y redundancia en los datos. Como se mencionará en la sección 8.2, sería importante realizar un aporte en esta área a través de la creación de repositorios con modelos que contengan diversas anomalías.

Tabla 3. Datos de entrenamiento y validación del clasificador.

Hueco	Varianza Torsión	Varianza Curvatura	Simetría	Tamaño	Similitud geométrica	Naturaleza	Clasificación
1	0.008680776	0.069176241	0.167976322	26	0.001820214	óptico	anomalía
2	0.217611721	0.016588704	0.630555041	95	0.000395752	real	real
3	0.046903158	0.209705969	0.080187537	26	0.000879049	óptico	anomalía
4	0.38365377	0.020635165	0.382173119	105	0.001481801	real	real
5	0.270226559	4.753.384.591	0.298678365	38	0.003644443	óptico	anomalía
6	0.017235624	0.105062673	0.324090608	24	0.00374767	óptico	anomalía
7	0.215287874	0.087572367	0.448778922	101	0.003041784	real	real

Capítulo 7

8	0.056789102	0.19045993	0.167976322	27	0.000514801	óptico	anomalía
9	0.237197097	0.098671502	0.310991816	36	0.001976234	óptico	anomalía
10	0.294979265	0.017299694	0.168956397	106	0.001206602	real	real
11	0.113373585	0.065576344	0.158826622	42	0.002688037	oclusión	anomalía
12	0.215213213	0.017355155	0.538273072	102	7,51E-01	real	real
13	0.364616861	0.022111946	0.134482683	527	0.002727486	oclusión	anomalía
14	0.211790358	0.01754559	0.352789206	95	0.001908096	real	real
15	0.007702532	0.007924243	0.059625679	23	0.002566171	registro	anomalía
16	8,09E+00	0.029800297	0.030855199	12	0.000459402	registro	anomalía
17	0.003074343	0.038543803	0.142455765	13	0.002675443	registro	anomalía
18	0.002848426	0.025326509	0.210818511	14	0.000948471	registro	anomalía
19	0.00033654	0.067874856	0.461516788	13	0.002740554	registro	anomalía
20	0.008329519	0.003492777	0.168976097	150	0.001358689	registro	anomalía
21	0.000214053	0.004432123	0.243600074	20	0.001631239	real	real
22	4,42E-01	0.004056598	0.133582683	19	0.003707782	real	real
23	7,98E+00	0.004413982	0.088059571	16	0.000548847	real	real
24	0.00019677	0.005234232	0.349544323	17	0.00231939	real	real
25	0.000293617	0.004689753	0.295744287	20	0.003864676	real	real
26	7,95E+00	0.004278906	0.506993026	21	0.003234326	real	real
27	0.000515855	0.005122299	0.205492466	19	0.002117705	real	real
28	2,25E+00	0.005369801	0.159891267	14	0.003822072	real	real
29	0.000132511	0.005819323	0.325262845	16	0.000216303	real	real
30	0.000230231	0.007108701	0.150396402	17	0.001955915	real	real
31	0.000169291	0.005167026	0.1162913	16	0.003699188	real	real
32	0.001819024	0.001281596	0.159851177	34	0.003079444	real	real
33	0.001521707	0.001283958	0.170157744	34	0.000548466	real	real
34	1,15E+00	0.004100314	0.053671766	21	0.003304726	real	real
35	7,88E+00	0.005851692	0.365786594	16	0.002583906	real	real
36	0.00014166	0.006391873	0.099436891	17	0.003106869	real	real
37	0.00135704	0.006076893	0.260790909	18	0.000492437	real	real
38	0.000485866	0.004119632	0.270621114	21	0.003292553	real	real
39	0.00020051	0.004227359	0.027710208	18	0.000973988	real	real

Capítulo 7

40	0.000370479	0.005648579	0.054936601	16	0.003838724	real	real
41	0.000174899	0.005250353	0.173977063	16	0.002391729	real	real
42	0.007265017	0.011120252	0.078797642	23	0.000360304	real	real
43	1,11E+00	0.004931399	0.135186786	18	0.002557139	real	real
44	0.000631901	0.001962919	0.205937967	78	0.000118614	real	real
45	0.001660744	0.001850045	0.038274069	32	0.003817317	real	real
46	5,65E-01	0.004843908	0.283210432	19	0.003699188	real	real
47	0.064784423	0.010417663	0.120462676	5	0.000264963	registro	anomalía
48	0.001075837	0.002094114	0.63979137	26	0.001837501	real	real
49	0.010135936	0.002801554	0.351470969	156	0.001925798	registro	anomalía
50	0.007265017	0.011120252	0.078797642	23	0.000360304	real	real
51	0.394078711	0.014539845	0.077998912	129	0.001599417	real	real
52	0.325144842	0.020917706	0.459697369	180	0.002682804	oclusión	anomalía
53	5,47E-01	1.757.985.322	0.992156742	14	0.001796983	registro	anomalía
54	0.125909137	0.020804693	0.19451395	59	0.000437826	oclusión	anomalía
55	0.398344145	0.036610024	0.901972565	73	0.006829158	oclusión	anomalía
56	0.223890604	0.013086687	0.451215767	196	0.001812491	real	real
57	0.299461697	0.01729453	0.181164435	131	0.002172161	real	real
58	0.28171681	0.02623149	0.057537201	207	0.003382504	oclusión	anomalía
59	3.710.036.546	9,51E+00	0.110824609	6	0.001153088	registro	anomalía
60	0.170660004	0.002296835	0.08049386	358	0.001854491	real	real
61	6,63E-01	0.248999944	0.468608887	6	0.002660004	registro	anomalía
62	0.128394727	0.020084144	0.048053427	60	0.003106745	oclusión	anomalía
63	0.120320726	0.0072576	0.367142674	398	0.002708447	registro	anomalía
64	0.193678928	0.008468553	0.240027433	523	0.002369912	real	real
65	0.138420175	0.216870458	0.061042159	55	0.004292196	oclusión	anomalía
66	0.177074308	0.003990934	0.301485658	521	0.00145841	registro	anomalía
67	0.132452917	0.003035113	0.120824609	1345	0.003855915	oclusión	anomalía
68	0.185569157	0.005810974	0.893497844	441	0.010238675	real	real
69	0.130916707	0.022861947	0.142943124	49	0.000394371	oclusión	anomalía
70	0.152103959	0.006182014	0.756787469	572	0.000532696	registro	anomalía
71	0.093287269	0.003728445	0.507305102	477	0.004624762	registro	anomalía

Capítulo 7

72	0.003014778	0.005945394	0.219468338	14	0.003474065	registro	anomalía
73	0.000406202	0.010401577	0.324090608	22	0.002911513	registro	anomalía
74	0.001378499	0.000419377	0.656302096	12	0.000422668	oclusión	anomalía
75	1,14E+00	0.002582507	0.292654526	21	0.0031379	real	real
76	0.001094203	0.002176228	0.424513302	52	0.00019237	real	real
77	0.003978312	0.000306289	1.336.068.865	172	0.002251649	real	real
78	0.005268591	0.000242463	1.463.677.196	208	0.001256671	real	real
79	0.002580369	0.000759205	0.033084563	85	0.003360776	real	real
80	0.005373612	0.007366336	0.434742204	39	0.003627589	oclusión	anomalía
81	2,47E+00	0.002957134	0.195844616	19	0.002312719	real	real
82	0.00734222	0.004701685	0.407286021	40	0.003831871	oclusión	anomalía
83	0.00013666	0.139351769	0.070868952	11	0.003290152	oclusión	anomalía
84	0.0037589	0.001350651	0.776044742	60	0.003930079	oclusión	anomalía
85	0.01380486	0.003413937	0.092170108	104	0.001178681	oclusión	anomalía
86	0.013014279	0.000799443	0.218150264	168	0.006783138	oclusión	anomalía
87	2,36E+00	0.312831404	0.080929236	10	0.00077901	oclusión	anomalía
88	3,51E-02	0.008244215	0.175074371	18	2,09E+00	oclusión	anomalía
89	0.002658675	0.013535694	0.111527428	17	0.000751123	oclusión	anomalía
90	0.000656888	0.019330692	0.210818511	16	0.001191573	oclusión	anomalía
91	0.000485504	0.06331162	0.331154873	17	0.000281037	oclusión	anomalía
92	0.002767891	0.034177644	0.12345679	20	0.000597332	oclusión	anomalía
93	0.000326474	0.002045832	0.0170056	26	0.003784579	oclusión	anomalía
94	0.001232373	0.008157657	0.733871127	12	0.00189229	oclusión	anomalía
95	0.002420788	0.000599865	0.008113829	44	0.00202019	real	real
96	0.002301943	0.001571692	0.230405448	42	0.00178377	real	real
97	1,49E+00	0.078892849	0.437926247	4	0.001990801	registro	anomalía
98	0.002121974	0.022869597	0.009816786	17	0.001314278	registro	anomalía
99	0.023921207	8,42E-06	0.695808407	5	0.002758949	registro	anomalía
100	6,26E+00	0.0303265	0.212156794	8	0.000632963	registro	anomalía
101	0.002764907	0.0322501	0.023975672	24	0.002924913	registro	anomalía
102	0.000279091	0.053522081	0.272698357	10	0.001603648	oclusión	anomalía
103	0.000871888	0.153062209	0.513632	10	0.000491787	registro	anomalía

104	0.001537514	0.014340577	0.027972954	16	0.002955677	oclusión	anomalía
105	0.000771654	0.18474778	0.076881148	25	0.002318586	registro	anomalía
106	0.022178978	0.02073655	0.182902151	22	0.001854703	registro	anomalía
107	0.022625906	0.057718026	0.32721903	24	0.001623798	registro	anomalía
108	0.017219059	0.021825958	0.606091527	21	0.000368533	registro	anomalía
109	0.006224185	0.031950869	0.182191476	35	0.001895408	oclusión	anomalía
110	0.006412627	0.057447273	0.23318745	21	0.002636959	registro	anomalía
111	0.014624239	0.004312084	0.103931142	36	0.001929165	oclusión	anomalía
112	0.011878083	0.005040545	0.219933046	119	0.00587006	real	real
113	0.014264153	0.008738864	0.066320393	78	0.005203576	real	real
114	0.003694637	0.016310261	0.190336085	20	0.00107054	real	real
115	0.004972847	0.012198953	0.119613764	39	0.003195105	oclusión	anomalía
116	0.019662473	0.019247666	0.200038627	36	2,39E+00	oclusión	anomalía
117	0.005990793	0.006762147	0.769194647	27	0.003801704	oclusión	anomalía
118	0.000549214	0.016466186	0.209633526	16	0.00382235	oclusión	anomalía
119	0.015608742	0.008462455	0.09321851	43	0.002073321	oclusión	anomalía
120	0.011207481	0.001341257	0.335743725	69	0.002329607	real	real
121	0.015696263	0.008634249	0.056732523	12	0.003838724	registro	anomalía
122	0.007066667	0.009153458	0.461516788	26	0.001136612	real	real
123	0.039369827	0.016221695	0.149317088	38	0.006092665	oclusión	anomalía
124	8,96E-03	0.000919417	0.971579543	5	0.001497704	registro	anomalía
125	1,22E+00	0.434641707	0.597509017	7	0.001109694	registro	anomalía
126	0.000409788	0.024051153	0.153840681	16	0.003864676	oclusión	anomalía
127	5,58E+00	0.04335892	0.665034292	10	0.001736575	registro	anomalía
128	0.003085217	0.002329318	0.053178848	9	0.000773802	registro	anomalía

La columna **Naturaleza** en la tabla 3, hace referencia a la clasificación relacionada a la naturaleza original que genera el hueco, según lo tratado en el capítulo 4 de este documento: Oclusiones (*oclusión*), problemas de registro (*registro*), problemas de ópticos de la superficie (*óptico*) y hueco perteneciente al objeto (*real*). La columna **Clasificación** en esta misma tabla, se indica la clasificación de interés en este trabajo: Anomalías relacionadas con ausencia de información (*anomalía*) y huecos reales o pertenecientes al objeto (*real*).

Para la característica similitud geométrica, se utilizó un conjunto de 5 contornos de figuras geométricas básicas modelo que se pueden observar en la Figura 50.

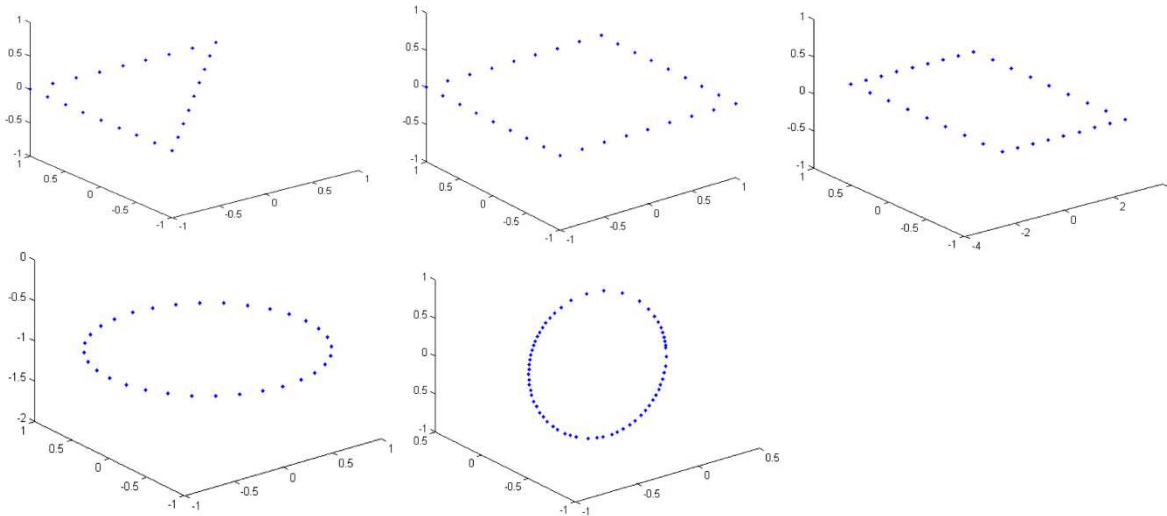


Figura 50. Modelos utilizados para la evaluación del descriptor: similitud geométrica. De izquierda a derecha y de arriba a abajo, triángulo, cuadrado, rectángulo, círculo y elipse.

Los experimentos fueron ejecutados en una máquina con sistema operativo MS Windows 7 sin ninguna condición especial de hardware. Los descriptores locales y globales del contorno fueron implementados en MATLAB R2010a. Para la selección de características, clasificación y validación se utilizó el software WEKA 3.6 el cual está desarrollado en el lenguaje de programación Java y distribuido bajo licencia GNU-GPL. Para la generación del árbol de decisión se utilizó el algoritmo J48, el cual es una implementación en Java del algoritmo C4.5. Para la generación de la máquina de soporte de vectores se utilizó la librería libsvm para JAVA y para el algoritmo AdaBoost se utilizó la implementación que trae WEKA, llamada AdaBoostM1.

7.2. Experimentos y Resultados

Los experimentos fueron realizados en tres etapas principales. En la primera etapa se realizó un estudio del comportamiento de los datos según una técnica de clusterización con el fin de evaluar que tan separables son las clases finales. En la segunda etapa se determinan los conjuntos de características relevantes según una estrategia de selección. Y en la tercera y última etapa se entrenan y evalúan los clasificadores según los conjuntos de características definidos.

7.2.1. Análisis de los datos de trabajo

Mediante una clusterización k-means de los 128 registros para las clases **Naturaleza** y **Clasificación** se obtuvieron las siguientes gráficas.

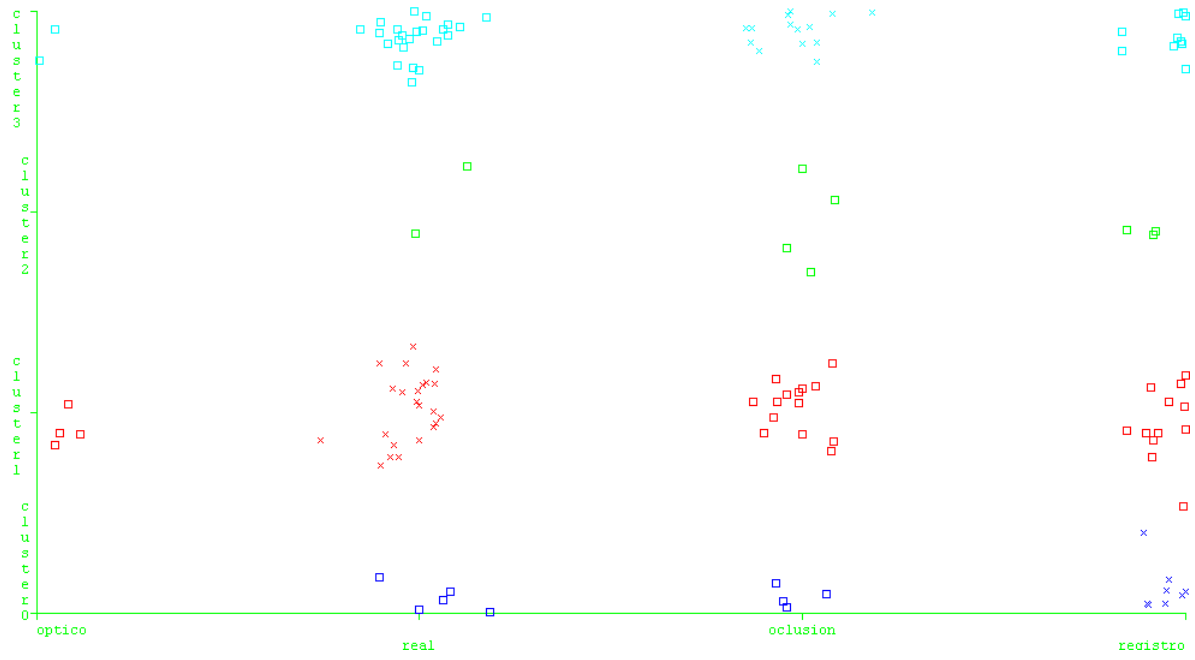


Figura 51. Clusterización de los datos para 4 clases.

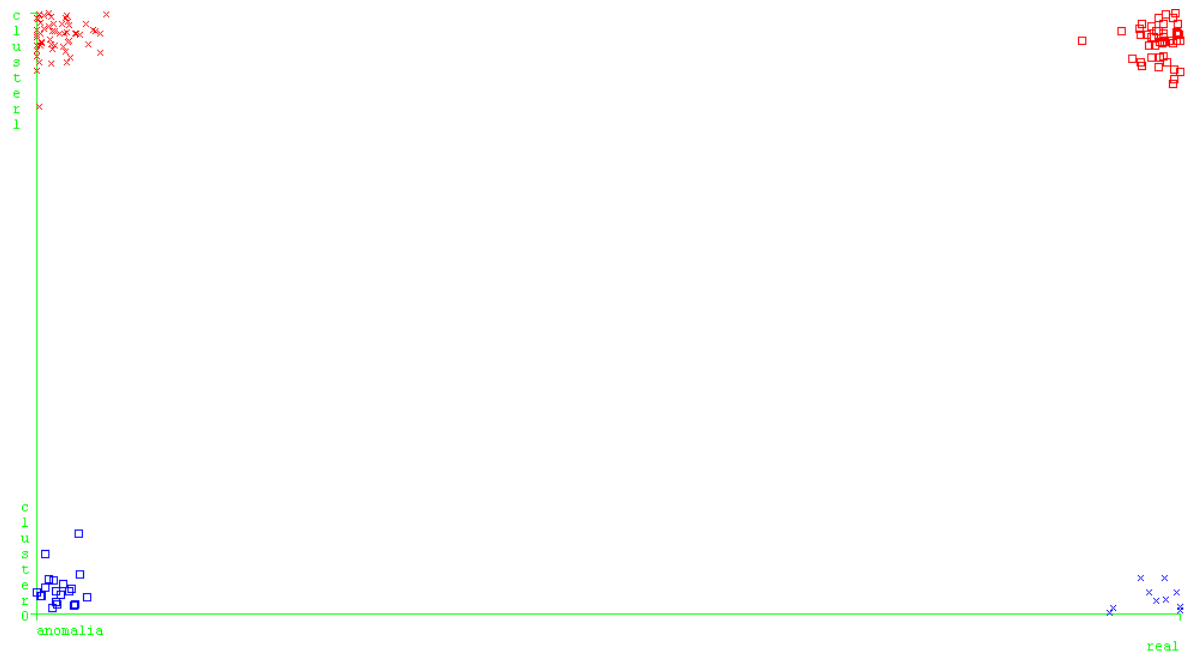


Figura 52. Clusterización para 2 clases.

En las figuras anteriores, cada color representa una clase de las que aparecen en el eje x. En las mismas se puede observar que las clases no se encuentran debidamente separadas según las características utilizadas en este trabajo y que por el contrario, presentan un nivel de solapamiento alto. Lo anterior indica que la relación entre las características y las clases finales no presenta un comportamiento lineal. Esto justifica el hecho de utilizar técnicas de clasificación supervisada no lineales.

Para las clusterizaciones en 4 clases y 2 clases se obtuvo un 64.84% y 49.21% de instancias clasificadas incorrectamente de forma respectiva, lo cual indica que las naturalezas que producen huecos en superficies 3D de objetos de forma libre no son separables. Sin embargo, el resultado obtenido para el caso de 2 clases muestra que la separabilidad de las clases aumenta cuando se trata de clasificar un hueco solo como anomalía o hueco real. Los resultados totales de las clusterizaciones, pueden ser corroborados en los anexos 1 y 2 respectivamente.

Según lo establecido en el alcance de este trabajo, la clasificación de interés es la que separa los datos de forma binaria en las clases *anomalía* y *real*.

7.2.2. Selección de Características

Esta selección fue realizada mediante el método de selección secuencial hacia adelante, el cual determinó que las clases de relevancia para la clasificación binaria de los datos de trabajo son la varianza de la curvatura y el tamaño del hueco. Esta prueba arrojó que se evaluaron 52 subgrupos de combinaciones entre los 5 descriptores propuestos, y que el mejor subgrupo (varianza de la curvatura y tamaño) obtuvo un mérito de 0.302. Debido a que el mérito del grupo de interés obtenido es menor que el 50%, para el entrenamiento y evaluación de los clasificadores se utilizaron dos grupos de variables según se muestra en la Tabla 4.

Tabla 4. Subgrupos de características.

grupo 1	grupo 2
Varianza_Torsion	Varianza_Curvatura
Varianza_Curvatura	Tamaño
Tamaño	
Simetría	
Similitud Geométrica	

Los resultados de este experimento pueden corroborarse en el anexo 3.

7.2.3. Evaluación de los Clasificadores

Para esto se entrenaron 3 clasificadores (ver sección 6.3) cada uno respecto a dos grupos de características según se definieron en la Tabla 4. Cada clasificador fue evaluado mediante validación cruzada con 50 selecciones aleatorias de datos.

En la Tabla 5 se puede observar los resultados del comportamiento en la predicción de clases para los tres clasificadores.

Tabla 5. Resultados de clasificación utilizando el primer grupo de características.

Clasificador	Instancias clasificadas correctamente. Grupo de características 1.	Instancias clasificadas correctamente. Grupo de características 2.
SVM	73.4375%	71.875%
AdaBoost	76.5625%	75%
C4.5	86.7188%	75.7813%

Las respectivas matrices de confusión de cada clasificador para el conjunto de 128 datos de trabajo se pueden observar en las siguientes tablas:

Tabla 6. Matriz de confusión para el clasificador SVM con el grupo 1 de características.

a	b	← clasificado como
59	16	A
18	35	B

Tabla 7. Matriz de confusión para el clasificador SVM con el grupo 2 de características.

a	b	← clasificado como
59	16	a
20	33	b

Tabla 8. Matriz de confusión para el clasificador AdaBoost con el grupo 1 de características.

a	b	← clasificado como
51	24	a
6	47	b

Tabla 9. Matriz de confusión para el clasificador AdaBoost con el grupo 2 de características.

a	b	← clasificado como
51	24	a
8	45	b

Tabla 10. Matriz de confusión para el clasificador C4.5 con el grupo 1 de características.

a	b	← clasificado como
70	5	a
12	41	b

Tabla 11. Matriz de confusión para el clasificador C4.5 con el grupo 2 de características.

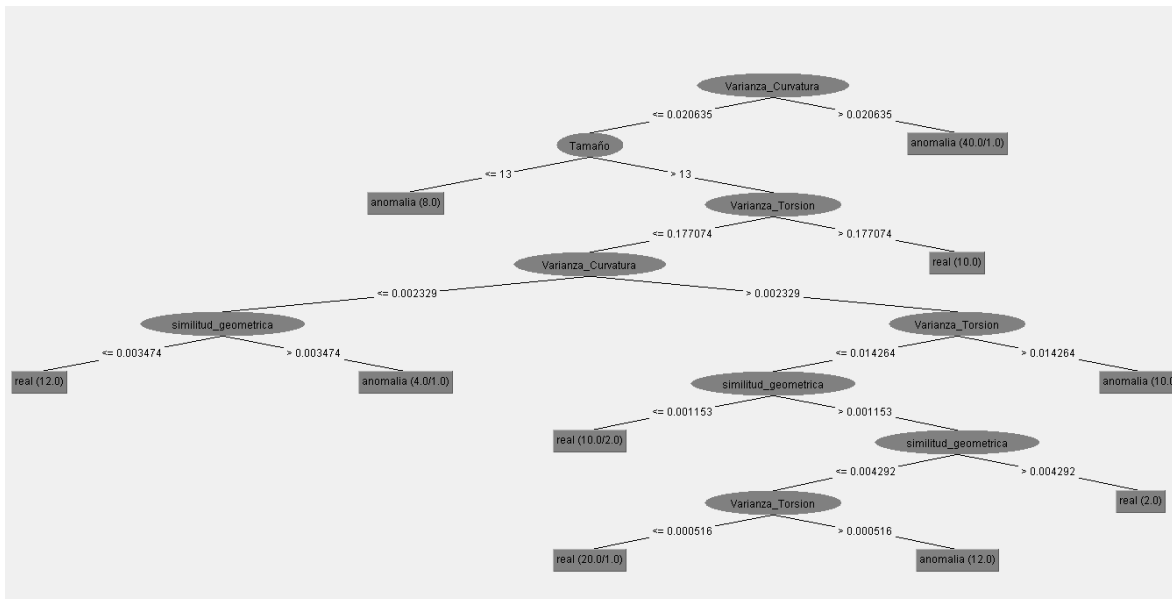
a	b	← clasificado como
47	28	a
3	50	b

Las áreas de las curvas ROC para los clasificadores se pueden observar en la Tabla 12. Esta área indica la exactitud global del clasificador, siendo el máximo 1.

Tabla 12. Áreas de la curva ROC para los distintos clasificadores.

Clasificador	Área ROC. Grupo de características 1.	Área ROC. Grupo de características 2.
SVM	0.724	0.705
AdaBoost	0.822	0.719
C4.5	0.848	0.659

Los resultados completos de estos experimentos, pueden ser verificados en los anexos 4, 5, 6, 7, 8 y 9.

**Figura 53.** Clasificador propuesto.

En la Figura 53 se puede observar el árbol de clasificación obtenido mediante el algoritmo ID3, para el conjunto de características locales y globales mencionadas anteriormente. Nótese como la curvatura y el tamaño son las características de menor nivel en el árbol, lo cual quiere decir que estas presentan mayor poder de discriminación para determinar si el hueco es real o una anomalía.

7.3. Discusión del Experimento

Un punto importante a destacar es el hecho de que según el proceso de selección de características se encontró que la varianza de la curvatura y el tamaño del hueco son las que mayor peso tienen para describir un hueco en la superficie de un objeto de forma libre. Esto se debe a que la primera característica mide que tanto varía el vector tangente respecto a la longitud de la curva del contorno, lo que indica una medida de que tan distorsionado es el mismo. Por otra parte, el tamaño del hueco según lo expuesto en el capítulo 4, resulta una característica discriminadora debido a que la constante investigación en métodos de adquisición y registro de datos, asegura que los errores generados son cada vez menores, implicando esto que el tamaño de los huecos producto de error en la adquisición o en el registro será cada vez menor. Aunque el nivel de *simetría* axial es alta para muchos huecos reales, la variedad de la forma que pueden presentar estos huecos en objetos de forma libre es infinita, lo que implica que el nivel de simetría no sea discriminante. Sin embargo, para aplicaciones industriales, donde la forma de huecos es bien definida, la simetría podría presentarse como una característica discriminadora. Debido al limitado grupo de modelos de huecos utilizados en este trabajo para la medición de la característica *similitud geométrica*, esta no resultó discriminante en el conjunto de datos utilizados. Es de suponer que si para una determinada aplicación, los huecos reales son bien definidos esta característica podría mejorar el nivel de clasificación entre huecos falsos y reales. Por otra parte, debido a que la medida de similitud geométrica está basada en la distancia euclídea entre histogramas, no se tiene en cuenta

Los experimentos realizados muestran que el mejor resultado es obtenido utilizando un árbol de decisión construido mediante el algoritmo C4.5, el cual alcanza una fiabilidad cercana al 87% (ver Figura 53) y una exactitud cercana al 85% con el grupo 1 de características. Sin embargo, la fiabilidad del mismo decreció sustancialmente del grupo 1 al grupo 2 de características, lo que indica un posible sobreajuste a los datos de entrada. Esto podría entenderse como que este clasificador no es apropiado para generalizar modelos, pero por otra parte, su velocidad de procesamiento (ver anexos) sugiere que puede tener mucha aplicación en condiciones prácticas de clasificación.

Los clasificadores SVM y AdaBoost presentan comportamientos parecidos en este trabajo, ambos con niveles de fiabilidad cercanos al 75%, aunque con mayor costo computacional que el clasificador C4.5 (ver anexos).

Otro hecho que cabe destacar es que según la tabla de áreas de la curva ROC (ver Tabla 12) la variabilidad de la exactitud del método de clasificación SVM es mucho menor que en los métodos AdaBoost y C4.5, lo que sugiere un mayor nivel de generalización del modelo de clasificación.

CAPITULO 8

CONCLUSIONES Y TRABAJOS FUTUROS

8.1. Conclusiones

En este trabajo se presentó una aproximación a la clasificación automática de anomalías relacionadas con ausencia de información en superficies 3-D de objetos de forma libre. Del mismo, se obtuvieron las siguientes conclusiones:

- Bajo condiciones generales, realizar la clasificación automática de un hueco como anomalía o hueco real, desde los mismos puntos del contorno es posible. En este trabajo se obtuvo un nivel de fiabilidad cercana al 87 por ciento.
- La varianza de la curvatura como medida de distorsión o irregularidad del contorno, y el tamaño del mismo se presentan como factores cruciales a la hora de determinar si el hueco pertenece o no al objeto real.
- Las naturalezas específicas que generan anomalías relacionadas con ausencia de información en superficies 3-D de objetos de forma libre, a saber, oclusiones, problemas ópticos y problemas de registro, no son diferenciables mediante las propiedades continuas de la curva ni de su forma. Sin embargo, es posible diferenciar si un hueco pertenece a la superficie del objeto real o si es producto de errores, mediante el estudio de las propiedades del borde.
- La presencia de ruido en las regiones circundantes al contorno del hueco parece ser una propiedad común cuando la naturaleza que lo causa está relacionada con propiedades ópticas del objeto. Sin embargo, este problema es tratado en la literatura desde el mismo proceso de adquisición, ya sea mejorando las condiciones del objeto a través del uso de sprays antireflectivos o filtrando los datos recién adquiridos.
- El estudio de las vistas parciales a las que pertenecen los puntos de un contorno de un hueco, podría brindar información relevante para determinar si el mismo es producto de un error de registro.
- Los descriptores simetría y la similitud geométrica fueron propuestos para evaluar la forma del contorno de un hueco en el espacio 3-D. Aunque para este estudio ambas características no discriminan la clase a la que pertenece un hueco, es posible que bajo condiciones industriales en las que los huecos tengan formas predefinidas, estas características sean relevantes.

Entre los aportes más destacables de este trabajo se encuentran:

1. La aproximación a la solución del problema de clasificación de huecos en superficies 3D desde la teoría de clasificación y reconocimiento de patrones.
2. La evaluación y análisis de diversas metodologías de clasificación para el problema en específico.

3. La proposición y desarrollo de un algoritmo para la obtención automática de códigos de cadena de curvas en el espacio 3D, bajo condiciones de espacialidad no homogénea de los datos.
4. La proposición e implementación de dos descriptores globales de contornos cerrados en el espacio 3D: *simetría y similitud geométrica*, basados en el análisis de códigos de cadena 3D.
5. La proposición e implementación de un método completamente automático para la clasificación de huecos en superficies 3D.
6. La utilización de descriptores novedosos torsión y curvatura que no utilizan aproximación mediante curvas paramétricas.

El clasificador propuesto se presenta como una alternativa inicial para realizar la clasificación automática de anomalías relacionadas con ausencia de información en superficies 3-D de objetos de forma libre. Sin embargo, este es susceptible de mejoramiento desde la descripción del contorno más que del método de clasificación.

8.2. Trabajos Futuros

De este trabajo se pueden obtener muchos tópicos que podrían ser puntos de partida para trabajos futuros:

- Probar los descriptores y el clasificador obtenidos, en objetos de naturaleza distinta tales como objetos industriales.
- Obtener y validar un descriptor del ruido en las regiones cercanas al contorno del hueco, lo cual sería muy útil para caracterizar huecos pertenecientes a problemas ópticos de la superficie del objeto real.
- Desarrollar una nueva estructura de imágenes de rango que permita almacenar la información de la vista parcial, lo cual sería muy útil para la caracterización de huecos producto de errores de registro.
- Probar diferentes técnicas de clasificación o combinaciones estas con los descriptores propuestos en este trabajo, por ejemplo, redes neuronales genéticas.
- Realizar una implementación completa de los descriptores propuestos y del clasificador utilizando tecnologías libres.
- Utilizar un mayor número de muestras para el entrenamiento y evaluación del clasificador propuesto.
- Apoyar la investigación en corrección de anomalías relacionadas con ausencia de información en superficies 3D, a través de la creación de un repositorio con modelos 3D que presenten este tipo de anomalías, en lo posible clasificados según la naturaleza que la genera.

Bibliografía

- [1] J. Branch, P. Boulanger, and F. Prieto, "Automatic hole-filling of triangular meshes using local radial basis function," *3DPVT '06: Proceedings of the Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pp. 727 - 734, 2006.
- [2] P. J. Besl, *The Free-Form Surface Matching Problem In Machine Vision Three Dimensional Scenes*. San Diego: Academic Press, 1990.
- [3] R. J. Campbell and F. J. Patrick, "A Survey of Free-Form Object Representation and Recognition Techniques," *Computer Vision and Image Understanding*, vol. 81, pp. 166-210, 2001.
- [4] A. Myers, "Introductory literature review surface reconstruction from three Dimensional Range Data," 1999.
- [5] C. Dorai and A. Jain, "COSMOS—A representation scheme for 3D free-form objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 1115 - 1130, 1997.
- [6] B. Batchelor and P. Whelan, *Intelligent Vision Systems for Industry*, First. Springer, 1997, p. 478.
- [7] N. D'Apuzzo, "Recent Advances In 3d Full Body Scanning With Applications To Fashion And Apparel," *Optical 3-D Measurement Techniques IX*, 2009.
- [8] F. Blais, "Review of 20 years of range sensor development," *Journal of Electronic Imaging*, vol. 13, no. 1, 2004.
- [9] J. Branch, "Reconstrucción de Objetos de Forma Libre a Partir de Imágenes de Rango Empleando una Red NURBS," Universidad Nacional de Colombia Sede Medellin, 2006.
- [10] B. Curless, "New Methods for Surface Reconstruction from Range Image," 1997.
- [11] Y. Watanabe, T. Komuro, and M. Ishikawa, "955-fps Real-time Shape Measurement of a Moving/Deforming Object using High-speed Vision for Numerous-point Analysis," *IEEE International Conference on Robotics and Automation*, pp. 10-14, 2007.

- [12] J. Park, "3D Modeling of Real-World Objects Using Range and Intensity Images," *Innovations in Machine Intelligence and Robot Perception*, 2004.
- [13] B. Gerard, "Registering multiview data to create 3d computer objects," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 17, 1995.
- [14] P. Liepa, "Filling holes in meshes," *SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pp. 200 - 205, 2003.
- [15] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. Silva, "Point Set Surfaces," *VIS '01 Proceedings of the conference on Visualization '01*, 2001.
- [16] P.-F. Lee and B.-S. Jong, "Point-based simplification algorithm," *WSEAS Transactions on Computer Research*, vol. 3, no. 1, 2008.
- [17] M. Pauly, M. Gross, and L. Kobbelt, "Efficient simplification of point-sampled surfaces," *VIS '02: Proceedings of the conference on Visualization '02*, pp. 163-170, 2002.
- [18] G. Pajares and J. De la Cruz, *Visión por Computador: Imágenes Digitales y Aplicaciones*, Primera. Madrid, España: RA-MA Editorial, 2002, p. 267.
- [19] S. Xianfang, P. L. Rosin, R. R. Martin, and F. C. Langbein, "Noise in 3D Laser Range Scanner Data," *Shape Modeling and Applications, 2008. SMI 2008. IEEE International Conference on*, no. 4, pp. 37 - 45, 2008.
- [20] G. Sanchez, "Metodología para la Corrección de Anomalías de Objetos Tridimensionales en Datos de Rango Empleando Inferencia Estadística-Bayesiana," Universidad Nacional de Colombia Sede Medellin, 2009.
- [21] J. Hrádek, "Methods of surface reconstruction from scattered data," 2003.
- [22] J. Carr, R. K. Beatson, B. C. MacCallum, W. R. Fright, T. J. McLennan, and T. J. Mitchell, "Smooth surface reconstruction from noisy range data," in *GRAPHITE '03 Proceedings of the 1st international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, 2003.
- [23] H. Hoppe, "Surface Reconstruction From Unorganized Points," Washington University, 1994.
- [24] H. Hoppe, "Progressive Meshes," *SIGGRAPH '96*, 1996.
- [25] J. Carr et al., "Reconstruction and representation of 3D objects with radial basis functions," in *SIGGRAPH '01 Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001.

- [26] B. S. Morse, T. S. Yoo, P. Rheingans, D. T. Chen, and K. R. Subramanian, "Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions," in *SIGGRAPH '05 ACM SIGGRAPH 2005 Courses*, 2005.
- [27] Y. Ohtake, A. Belyaev, and H.-P. Seidel, "3D scattered data interpolation and approximation with multilevel compactly supported RBFs," *Graphical Models - Special issue on SMI 2003*, vol. 67, no. 3, 2005.
- [28] J. Park and A. C. Kak, "3D Modeling of Optically Challenging Objects," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, 2008.
- [29] T. Ju, "Robust repair of polygonal models," *ACM Transactions on Graphics*, vol. 23, no. 3, 2004.
- [30] A. Sharf, M. Alexa, and D. Cohen-Or, "Context-based surface completion," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 878 - 887, 2004.
- [31] M. Pauly, "Example-Based 3D Scan Completion," *Eurographics Symposium on Geometry Processing*, 2005.
- [32] V. Kraevoy and A. Sheffer, "Template-based mesh completion," *SGP '05: Proceedings of the third Eurographics symposium on Geometry processing*, 2005.
- [33] G. H. Bendels, R. Schnabel, and R. Klein, "Fragment-based Surface Inpainting," *Eurographics Symposium on Geometry Processing*, 2005.
- [34] J. Dong, S. Ma, L. Li, and Z. Yu, "Hole Filling On Three-Dimensional Surface Texture," *IEEE International Conference on Multimedia and Expo*, 2007.
- [35] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *SIGGRAPH '96 Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.
- [36] S. Marschner, J. Davis, M. Garr, and M. Levoy, "Filling holes in complex surfaces using volumetric diffusion," *SIGGRAPH 2001*, 2001.
- [37] M. Oliveira and W. Jianning, "A Hole-Filling Strategy for Reconstruction of Smooth Surfaces in Range Images," *XVI Brazilian Symposium on Computer Graphics and Image Processing*, 2003.
- [38] T. Guo, J. Li, J. Weng, and Y. Zhuang, "Filling Holes in Meshes and Recovering Sharp Edges," *Systems, Man and Cybernetics, 2006. SMC '06. IEEE International Conference on*, 2006.

- [39] W. Zhao, S. Gao, and H. Lin, "A robust hole-filling Algorithm for triangular mesh.," *Visual Computing*, vol. 23, pp. 987–997, 2007.
- [40] G. Li, X.-Z. Ye, and S.-Y. Zhang, "An algorithm for filling complex holes in reverse engineering," *Engineering with Computers*, vol. 24, no. 2, 2008.
- [41] A. Kumar and A. Shih, "A Hole-filling Algorithm Using Non-uniform Rational B-splines," in *Proceedings of the 16th International Meshing Roundtable*, 2008.
- [42] Z. Li, D. S. Meek, and D. J. Walton, "Polynomial blending in a mesh hole-filling application," *Computer-Aided Design*, vol. 42, pp. 340 - 349, 2010.
- [43] J. Branch, F. Prieto, and P. Boulanger, "A Hole-Filling Algorithm for Triangular Meshes using Local Radial Basis Function," *Proceedings of the 15th International Meshing Roundtable*, 2006.
- [44] G. Sanchez, J. Branch, and P. Atencio, "A Metric for Automatic Hole Characterization," in *Proceedings of the 19th International Meshing Roundtable*, 2010.
- [45] G. Sanchez and J. Branch, "Toward an Automatic Hole Characterization for Surface Correction," in *Advances in Visual Computing 6th International Symposium, ISVC 2010*, 2010, pp. 602-611.
- [46] M. Levoy, "Why is 3D Scanning Hard?," 2002. [Online]. Available: http://graphics.stanford.edu/talks/3Dscanning-3dpvt02/3Dscanning-3dpvt02_files/v3_document.htm. [Accessed: 2011].
- [47] G. Sanchez, A. Maldonado, and J. Branch, "Algoritmo Evolutivo Paralelo para la correspondencia de puntos en el registro de vistas de Rango parciales de Objetos Tridimensionales," *XII simposio de Tratamiento de señales, Imagenes y vision Artificial*, 2007.
- [48] A. Maldonado, "Registro de Imágenes de Rango Empleando Características Geométricas," Universidad Nacional de Colombia, Sede Medellin., 2011.
- [49] E. Trucco and R. B. Fisher, "Acquisition of Consistent Range Data Using Local Calibration," *Proc. IEEE Int'l Conf. Robotics and Automation*, 1994.
- [50] B. Curless and M. Levoy, "Better Optical Triangulation through Spacetime Analysis," *Proc. Fifth IEEE Int'l Conf. Computer Vision*, 1995.
- [51] S. Nayar, K. Ikeuchi, and T. Kanade, "Recovering Shape in the Presence of Interreflections," *Proc. IEEE Int'l Conf. Robotics and Automation (ICRA '91)*, 1991.

- [52] J. Clark, E. Trucco, and L. B. Wolf, "Using Light Polarization in Laser Scanning," *Image and Vision Computing*, vol. 15, no. 1, 1997.
- [53] M. Du Sautoy, *Symmetry: A Journey into the Patterns of Nature*. 2008, p. 376.
- [54] A. Carmona, N. L. Fernandez, R. Medina, and F. Madrid, "A method for dominant points detection and matching 2d object identification," *Lecture Notes in Computer Science*, vol. 32, no. 11, pp. 424-431, 2004.
- [55] T. Lewiner, J. Gomes, H. Lopes, and M. Craizer, "Curvature and torsion estimators based on parametric curve fitting," *Computers and Graphics*, vol. 29, no. 5, 2005.
- [56] T. Nguyen and I. Debled-Renneson, "Curvature and torsion estimators for 3D curves," in *4th International Symposium on Advances in Visual Computing*, 2008.
- [57] T. Nguyen and I. Debled-Renneson, "ON THE LOCAL PROPERTIES OF DIGITAL CURVES," *International Journal of Shape Modeling*, vol. 14, no. 6, pp. 105-125, 2008.
- [58] S. Hermann and R. Klette, "A Comparative Study on 2D Curvature Estimators," *Computing: Theory and Applications, 2007. ICCTA '07. International Conference on*, vol. 5, no. 7, pp. 584 - 589, 2007.
- [59] Y. An, C. Shao, W. Xiaoliang, and Z. Li, "Geometric Properties Computation for Discrete Curves Based on Discrete Derivatives," *IEEE International Conference on Intelligent Control and Information Processing*, 2010.
- [60] S. Hermann and R. Klette, "Multigrid Analysis of Curvature Estimators," 2003.
- [61] Y. An, C. Shao, W. Xiaoliang, and Z. Li, "Geometric properties estimation from discrete curves using discrete derivatives," *Computers & Graphics*, vol. 35, pp. 916-930, 2011.
- [62] T. Lewiner, J. Gomes, H. Lopes, and M. Craizer, "Arc-length Based Curvature Estimator," *Proceedings of the 17th Brazilian Symposium on Computer Graphics and Image Processing*, vol. 250-257, 2004.
- [63] F. Cazals and M. Pouget, "Estimating differential quantities using polynomial fitting of osculating jets," *SGP '03 Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, 2003.
- [64] S. Gumhold, "Designing optimal curves in 2D," *Proceedings of the CEIG*, pp. 61-76, 2004.

- [65] F. Lu and E. Millios, "Optimal local spline approximation of planar shape," *Proceedings of the 1991 International Conference on Acoustics, Speech, and Signal Processing*, vol. 24, pp. 69-72, 1991.
- [66] G. Medioni and Y. Yasumoto, "Corner detection and curve representation using cubic B-splines," *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 267-278, 1987.
- [67] D. Couerjolly and S. Svensson, "Estimation of curvature along curves with application to fibres in 3D images of paper," *SCIA '03 Proceedings of the 13th Scandinavian conference on Image analysis*, 2003.
- [68] V. Pratt, "Direct least-squares fitting of algebraic surfaces," *SIGGRAPH '87 Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, vol. 21, no. 4, 1987.
- [69] M. Marji, "On the detection of dominant points on digital planar curves," Wayne State University, Detroit, Michigan, 2003.
- [70] F. Feschet and L. Tougne, "Optimal Time Computation of the Tangent of a Discrete Curve: Application to the Curvature," *DISCRETE GEOMETRY FOR COMPUTER IMAGERY Lecture Notes in Computer Science*, vol. 1568, 1999.
- [71] M. Worring and A. Smeulders, "Digital Curvature Estimation," *CVGIP: Image Understanding*, vol. 58, no. 3, 1993.
- [72] B. Kerautret and J. Lachaud, "Curvature estimation along noisy digital contours by approximate global optimization," *Pattern Recognition*, vol. 42, no. 10, 2009.
- [73] L. Piegl, "On NURBS: A Survey," *IEEE Computer Graphics and Applications*, vol. 11, no. 1, 1991.
- [74] W. Ma and J. Kruth, "Parametrization of randomly measured points for least squares fitting of B-spline curves and surfaces," *Computer Aided Design*, vol. 27, no. 9, 1995.
- [75] H. Freeman, "Computer Processing of Line-Drawing Images," *ACM Computing Surveys (CSUR)*, vol. 6, no. 1, 1974.
- [76] E. Bribiesca, "3D-curve representation by means of a binary chain code," *Mathematical and Computer Modelling*, vol. 40, 2004.
- [77] E. Bribiesca, "A method for representing 3D tree objects using chain coding," *Journal of Visual Communication & Image Representation*, vol. 19, pp. 184 - 198, 2008.

-
- [78] F. Y. Shih and W.-T. Wong, "A one-pass algorithm for local symmetry of contours from chain codes," *Pattern Recognition*, vol. 32, no. 7, 1999.
- [79] J. M. Iñesta, M. Buendía, and M. A. Sartic, "Local symmetries of digital contours from their chain codes," *Pattern Recognition*, vol. 29, no. 10, 1996.
- [80] K. Ozkan, C. Topal, and C. Akinlar, "Corner Detection via Trilateral Filtering of Chain Codes," *Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium on*, pp. 168-172, 2011.
- [81] ong K. Liua, W. Weia, P. J. Wanga, and B. Žalik, "Compressed vertex chain codes," *Pattern Recognition*, vol. 40, no. 11, 2007.
- [82] E. Bribiesca, "A chain code for representing 3D curves," *Pattern Recognition*, vol. 33, 2000.
- [83] G. Wagenknech, "A contour tracing and coding algorithm for generating 2D contour codes from 3D classified objects," *Pattern Recognition*, vol. 40, 2007.
- [84] E. Bribiesca, "Scanning-curves representation for the coverage of surfaces using chain coding," *Computers & Graphics*, vol. 27, 2003.
- [85] E. Lengyel, *Mathematics for 3D game programming and computer graphics*, 2nd ed. Charles River Media, 2004.
- [86] L. Piegl and W. Tiller, "Filling n-sided Regions with NURBS patches," *The Visual Computer*, vol. 15, no. 2, pp. 77-89, 1999.
- [87] C. De Boor, "On Calculating With B-Splines," *Jornal of Approximation Theory*, vol. 6, no. 1, pp. 50-62, 1972.
- [88] M. G. Cox, "The Numerical Evaluation of B-Splines," *Journal of the Institute of Mathematics and its Applications*, vol. 10, no. 2, pp. 134-179, 1972.
- [89] J. Bresenham, "Algorithm for computer control of a digital plotter," *IBM SYSTEMS JOURNAL*, vol. 4, no. 1, 1965.
- [90] A. Kaufman and E. Shimony, "3D Scan-Conversion Algorithms for Voxel-Based Graphics," in *I3D '86 Proceedings of the 1986 workshop on Interactive 3D graphics*, 1987.
- [91] C. DelRieux, "Introducción a la Computación Gráfica: Notas del Curso." 2009.
- [92] E. Bribiesca and W. Aguilar, "A Measure of Shape Dissimilarity for 3D Curves," *INTERNATIONAL JOURNAL OF CONTEMPORARY MATHEMATICAL SCIENCES*, vol. 1, no. 15, 2006.

- [93] E. Bribiesca and C. Velarde, "A Formal Language Approach for a 3D Curve Representation," *Computers and Mathematics with Applications*, vol. 42, 2001.
- [94] P. Saint-Marc, H. Rom, and G. Medioni, "B-spline contour representation and symmetry detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 11, 1997.
- [95] P. Saint-Marc, H. Rom, and G. Medioni, "B-Spline Contour Representation and Symmetry Detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, no. 11, 1993.
- [96] P. Atencio, G. Sanchez, and J. Branch, "Automatic visual model for classification and measurement of quality of fruit: case mangifera indica l.," *Dyna*, vol. 76, no. 160, 2009.
- [97] A. K. Jain and P. W. Duin, "Statistical pattern recognition: a review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4-37, 2000.
- [98] D. Mery et al., "Quality Classification of Corn Tortillas using Computer Vision," *Journal of Food Engineering*, vol. 101, no. 4, 2010.
- [99] P. A. Devijver and J. Kittler, *Pattern Recognition: A Statistical Approach*. 1982.
- [100] M. Hall, "Correlation-based Feature Selection for Machine Learning," University of Waikato, 1999.
- [101] H. Wei and BillingsS, "Feature subset selection and ranking for data dimensionality reduction," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 1, pp. 162-166, 2007.
- [102] K. Mao, "Identifying critical variables of principal components for unsupervised feature selection," *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 35, no. 2, pp. 339-344, 2005.
- [103] MathWorks, "Matlab Toolbox of Bioinformatics," 2007.
- [104] R. Duda, P. Hart, and D. Stork, *Pattern Classification (2nd Edition)*. 2001.
- [105] N. Zheng and J. Xue, *Statistical Learning and Pattern Analysis for Image and Video Processing*. Springer, 2009, p. 356.
- [106] V. Cherkassky and F. Mulier, *Learning from Data: Concepts, Theory, and Methods, 2nd Edition*. Wiley, 2007.

- [107] M. Schapire, "The strength of weak learnability," *Machine Learning*, vol. 5, no. 2, 1990.
- [108] Y. Freund and M. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, 1997.
- [109] J. R. Quinlan, *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc. San Francisco, CA, USA ©1993, 1993.
- [110] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81-106, 1986.
- [111] T. J. Mitchell, *Machine Learning*. McGraw-Hill, Boston, 1997.

ANEXOS

Anexo 1: Resultados para la clusterización en 4 clases de los datos de trabajo.

=== Run information ===

Scheme:weka.clusterers.SimpleKMeans -N 4 -A "weka.core.EuclideanDistance -R first-last" -I 500 -S 10

Relation: huecos-weka.filters.unsupervised.attribute.Remove-R1,8

Instances: 128

Attributes: 6

Varianza_Torsion

Varianza_Curvatura

Simetria

Tamaño

similitud_geometrica

Ignored:

Clase

Test mode:Classes to clusters evaluation on training data

=== Model and evaluation on training set ===

kMeans

=====

Number of iterations: 8

Anexos

Within cluster sum of squared errors: 6.117338784763609

Missing values globally replaced with mean/mode

Cluster centroids:

Attribute	Full Data	Cluster#			
		0	1	2	3
(128)	(17)	(53)	(9)	(49)	
=====					
Varianza_Torsion	0.0821	0.0382	0.1235	0.1865	0.0334
Varianza_Curvatura	0.54	1.0761	0.0314	0.0102	1.0014
Simetria	0.2857	0.7848	0.1853	0.3945	0.2011
Tamaño	80.1406	79.8824	49.0189	497	37.3265
similitud_geometrica	0.0023	0.0015	0.0011	0.0046	0.0034

Time taken to build model (full training data) : 0.06 seconds

=== Model and evaluation on training set ===

Clustered Instances

0	17 (13%)
1	53 (41%)
2	9 (7%)
3	49 (38%)

Class attribute: Clase

Classes to Clusters:

```
0 1 2 3 <-- assigned to cluster
```

```
0 4 0 2 | optico
```

```
5 23 2 23 | real
```

```
4 14 4 14 | oclusion
```

```
8 12 3 10 | registro
```

```
Cluster 0 <-- registro
```

```
Cluster 1 <-- real
```

```
Cluster 2 <-- No class
```

```
Cluster 3 <-- oclusion
```

```
Incorrectlyclusteredinstances :    83.0    64.8438 %
```

Anexo 2: Resultados para la clusterización en 2 clases de los datos de trabajo.

=== Run information ===

```
Scheme:weka.clusterers.SimpleKMeans -N 2 -A "weka.core.EuclideanDistance -R first-last" -
I 500 -S 10
```

```
Relation:    huecos-weka.filters.unsupervised.attribute.Remove-R1,7
```

```
Instances:   128
```

```
Attributes:  6
```

Anexos

Varianza_Torsion

Varianza_Curvatura

Simetria

Tamaño

similitud_geometrica

Ignored:

Clase2

Test mode:Classes to clusters evaluation on training data

=== Model and evaluation on training set ===

kMeans

=====

Number of iterations: 4

Within cluster sum of squared errors: 8.543553231782326

Missing values globally replaced with mean/mode

Cluster centroids:

Attribute	Cluster#		
	Full Data	0	1
(128)	(30)	(98)	
=====			
Varianza_Torsion	0.0821	0.0704	0.0857
Varianza_Curvatura	0.54	0.6294	0.5126
Simetria	0.2857	0.6728	0.1672

Anexos

Tamaño	80.1406	99.1667	74.3163
similitud_geometrica	0.0023	0.0025	0.0022

Time taken to build model (full training data) : 0.01 seconds

=== Model and evaluation on training set ===

Clustered Instances

0 30 (23%)

1 98 (77%)

Class attribute: Clase2

Classes to Clusters:

0 1<-- assigned to cluster

20 55 | anomalia

10 43 | real

Cluster 0 <-- real

Cluster 1 <-- anomalia

Incorrectlyclusteredinstances : 63.0 49.2188 %

Anexo 3: Resultados de la selección de características para dos clases.

=== Run information ===

Evaluator: weka.attributeSelection.CfsSubsetEval

Search:weka.attributeSelection.LinearForwardSelection -D 1 -N 10 -I -K 50 -T 0

Relation: huecos-weka.filters.unsupervised.attribute.Remove-R1,7

Instances: 128

Attributes: 6

Varianza_Torsion

Varianza_Curvatura

Simetria

Tamaño

similitud_geometrica

Clase2

Evaluation mode:evaluate on all training data

=== Attribute Selection on all input data ===

Search Method:

Linear Forward Selection.

Start set: no attributes

Forward selection method: floating forward selection

Stale search after 10 node expansions

Linear Forward Selection Type: fixed-set

Number of top-ranked attributes that are used: 6

Total number of subsets evaluated: 52

Merit of best subset found: 0.302

Attribute Subset Evaluator (supervised, Class (nominal): 6 Clase2):

CFS Subset Evaluator

Including locally predictive attributes

Selected attributes: 2,4 : 2

Varianza_Curvatura

Tamaño

Anexo 4: Resultados de la clasificación mediante SVM para el conjunto de características 1.

=== Summary ===

Correctly Classified Instances	94	73.4375 %
Incorrectly Classified Instances	34	26.5625 %
Kappa statistic	0.4495	
Mean absolute error	0.2656	
Root mean squared error	0.5154	
Relative absolute error	54.6582 %	
Root relative squared error	104.5314 %	
Total Number of Instances	128	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
---------	---------	-----------	--------	-----------	----------	-------

Anexos

	0.787	0.34	0.766	0.787	0.776	0.724	anomalía
	0.66	0.213	0.686	0.66	0.673	0.724	real
Weighted Avg.	0.734	0.287	0.733	0.734	0.734	0.724	

=== Confusion Matrix ===

a b<-- classified as

59 16 | a = anomalía

18 35 | b = real

Anexo 5: Resultados de la clasificación mediante SVM para el conjunto de características 2.

=== Summary ===

Correctly Classified Instances	92	71.875 %
Incorrectly Classified Instances	36	28.125 %
Kappa statistic	0.4139	
Mean absolute error	0.2813	
Root mean squared error	0.5303	
Relative absolute error	57.8734 %	
Root relative squared error	107.5619 %	
Total Number of Instances	128	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.787	0.377	0.747	0.787	0.766	0.705	anomalía

Anexos

	0.623	0.213	0.673	0.623	0.647	0.705	real
Weighted Avg.	0.719	0.309	0.716	0.719	0.717	0.705	

=== Confusion Matrix ===

a b<-- classified as

59 16 | a = anomalia

20 33 | b = real

Anexo 6: Resultados de la clasificación mediante AdaBoost para el conjunto de características 1.

=== Summary ===

Correctly Classified Instances	98	76.5625 %
Incorrectly Classified Instances	30	23.4375 %
Kappa statistic	0.5399	
Mean absolute error	0.3082	
Root mean squared error	0.3959	
Relative absolute error	63.4281 %	
Root relative squared error	80.2942 %	
Total Number of Instances	128	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.68	0.113	0.895	0.68	0.773	0.822	anomalia
0.887	0.32	0.662	0.887	0.758	0.822	real

Anexos

Weighted Avg. 0.766 0.199 0.798 0.766 0.767 0.822

=== Confusion Matrix ===

a b<-- classified as

51 24 | a = anomalia

6 47 | b = real

Anexo 7: Resultados de la clasificación mediante AdaBoost para el conjunto de características 2.

=== Summary ===

Correctly Classified Instances	96	75	%
Incorrectly Classified Instances	32	25	%
Kappa statistic	0.5066		
Mean absolute error	0.3453		
Root mean squared error	0.4287		
Relative absolute error	71.0572	%	
Root relative squared error	86.9569	%	
Total Number of Instances	128		

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.68	0.151	0.864	0.68	0.761	0.719	anomalia
	0.849	0.32	0.652	0.849	0.738	0.719	real
Weighted Avg.	0.75	0.221	0.777	0.75	0.751	0.719	

```
=== Confusion Matrix ===
```

```
a b<-- classified as
```

```
51 24 | a = anomalia
```

```
8 45 | b = real
```

Anexo 8: Resultados de la clasificación mediante C4.5 para el conjunto de características 1.

```
=== Summary ===
```

Correctly Classified Instances	111	86.7188 %
Incorrectly Classified Instances	17	13.2813 %
Kappa statistic	0.7209	
Mean absolute error	0.1661	
Root mean squared error	0.3509	
Relative absolute error	34.1757 %	
Root relative squared error	71.1765 %	
Total Number of Instances	128	

```
=== Detailed Accuracy By Class ===
```

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.933	0.226	0.854	0.933	0.892	0.848	anomalia
	0.774	0.067	0.891	0.774	0.828	0.848	real
Weighted Avg.	0.867	0.16	0.869	0.867	0.865	0.848	

Anexos

```
=== Confusion Matrix ===
```

```
a b<-- classified as
```

```
70 5 | a = anomalia
```

```
12 41 | b = real
```

Anexo 9: Resultados de la clasificación mediante C4.5 para el conjunto de características 2.

```
=== Summary ===
```

Correctly Classified Instances	97	75.7813 %
Incorrectly Classified Instances	31	24.2188 %
Kappa statistic	0.5332	
Mean absolute error	0.3124	
Root mean squared error	0.4097	
Relative absolute error	64.28 %	
Root relative squared error	83.0884 %	
Total Number of Instances	128	

```
=== Detailed Accuracy By Class ===
```

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
	0.627	0.057	0.94	0.627	0.752	0.659	anomalia
	0.943	0.373	0.641	0.943	0.763	0.659	real
Weighted Avg.	0.758	0.188	0.816	0.758	0.757	0.659	

```
=== Confusion Matrix ===
```

a b<-- classified as

47 28 | a = anomalia

3 50 | b = real