



UNIVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

EVALUACIÓN DE ALGORITMOS DE MATCHING PARCIAL EN OBJETOS 3D UTILIZANDO UN ESCÁNER 3D ÓPTICO

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN
CIENCIAS, MENCIÓN COMPUTACIÓN
MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL
EN COMPUTACIÓN

RAFAEL ERNESTO MERUANE NARANJO

PROFESOR GUÍA:
BENJAMIN BUSTOS CARDENAS

MIEMBROS DE LA COMISIÓN:
NANCY HITSCHFELD KAHLER
DOMINGO MERY QUIROZ
PABLO GUERRERO PÉREZ

Este trabajo ha sido parcialmente financiado por proyecto Fondecyt 1110111 y
proyecto Fondecyt 1140783

SANTIAGO DE CHILE
2015

**RESUMEN DE LA TESIS PARA OPTAR AL TÍTULO
DE: INGENIERO CIVIL EN COMPUTACIÓN Y AL GRADO
DE MAGÍSTER EN CIENCIAS, MENCIÓN COMPUTACIÓN
POR: RAFAEL ERNESTO MERUANE NARANJO
FECHA: 30/06/2015
PROFESOR GUIA: BENJAMIN BUSTOS CARDENAS**

EVALUACIÓN DE ALGORITMOS DE MATCHING PARCIAL EN OBJETOS 3D UTILIZANDO UN ESCÁNER 3D ÓPTICO

El problema de que un computador determine la similitud entre objetos tridimensionales es un problema que tiene aplicaciones directas en diversas áreas como la medicina, la ingeniería y la arqueología, entre otras. Si bien, dentro de los últimos años se han desarrollado varias técnicas que permiten estimar la similitud entre objetos 3D [1], estas técnicas presentan serios problemas al momento de realizar consultas cuando la información tridimensional es parcial. Actualmente existen muy pocas técnicas que permiten el cálculo de similitud entre objetos 3D cuando la información es parcial [2]. Por otra parte, la evaluación de estas técnicas se ha realizado utilizando información simulada, que no representa efectivamente cómo es la información tridimensional que una máquina puede obtener del mundo real, por lo que se tiene poca información sobre si estas técnicas son realmente útiles en aplicaciones reales. Uno de los objetivos principales de esta tesis consiste en determinar si existe alguna técnica que permita determinar la similitud entre objetos tridimensionales cuando la información es parcial y proveniente de objetos reales, y evaluar su eficacia.

Se evaluaron algoritmos de matching parcial en objetos 3D utilizando la información proveniente de un escáner 3D óptico creado especialmente para este fin. El trabajo constó de tres fases: (1) la creación de un escáner 3D óptico utilizando un tablet Android y un proyector digital (2) la creación de una colección de referencia de objetos 3D que incluye mallas 3D parciales (3) la evaluación de algoritmos de matching parcial en objetos 3D. El trabajo de tesis es pionero en el ámbito de la evaluación de algoritmos de matching parcial en objetos 3D utilizando mallas 3D provenientes de objetos de la vida real, sin intervención alguna. Los resultados de este proyecto permitieron obtener métricas más realistas de *precision vs recall* que las existentes.

Agradecimientos

Agradezco a la Universidad de Chile y a todas las personas que han aportado a su noble tarea. Tarea que en palabras de su ex rector Juan Gómez Millas describe como “Nosotros, por medio de la investigación, la enseñanza y el aprendizaje, nos ocupamos de descubrir verdades, transmitir las y formar, al nivel más alto que nos es posible, un grupo seleccionado de la juventud en el amor activo y en el trato de las ciencias, las tecnologías y las artes.”. En particular, agradezco a mi profesor guía Dr. Benjamin Bustos y al Dr. Ivan Sipiran, quienes me apoyaron durante este proceso.

Agradezco a mi padre Teodoro Meruane, quien además de transmitirme valores a través de su ejemplo, fomentó mi motivación por el aprendizaje. Agradezco a mi madre Bibiana Naranjo, quien me enseñó cosas fundamentales de la vida y siempre estuvo presente para entregarme su apoyo y amor incondicional. Agradezco también a mis hermanos mayores Viviana, Carolina, Manuel, Claudia, Gabriel y Margarita, quienes han sido y siguen siendo un ejemplo para mí. Agradezco a mi padrino Julio Retamal, quien también ha sido un referente en mi vida.

Por último, agradezco a todos los amigos y personas que se han cruzado en mi camino, de todos he aprendido algo.

Tabla de contenido

| | |
|--|-----------|
| Resumen | I |
| Agradecimientos | II |
| Índice de figuras | IX |
| 1. Introducción | 1 |
| 1.1. Introducción | 1 |
| 1.2. Aplicaciones actuales | 4 |
| 1.3. Terminología | 7 |
| 1.4. Objetivos | 8 |
| 1.4.1. Objetivo general | 8 |
| 1.4.2. Objetivos específicos | 8 |
| 1.5. Metodología | 9 |
| 2. Marco teórico escáner 3D | 11 |
| 2.1. Introducción | 11 |
| 2.2. Metodologías para digitalizar un objeto | 11 |
| 2.2.1. Escáneres por contacto | 11 |
| 2.2.2. Escáneres sin contacto | 13 |
| 2.3. Metodologías ópticas | 15 |
| 2.3.1. Metodologías ópticas pasivas | 15 |
| 2.3.2. Metodologías ópticas activas | 19 |
| 2.4. Conclusiones | 24 |
| 3. Escáner 3D óptico basado en patrones de luz estructurada | 25 |
| 3.1. Introducción | 25 |
| 3.2. Modelo de una cámara | 26 |
| 3.2.1. Cámara estenopeica (pinhole camera model) | 26 |

| | | |
|-----------|--|-----------|
| 3.2.2. | Modelo de una cámara digital CCD/CMOS | 28 |
| 3.3. | Calibración de la cámara | 30 |
| 3.4. | Calibración del proyector | 33 |
| 3.5. | Calibración extrínseca | 36 |
| 3.6. | Reconstrucción por triangulación | 36 |
| 3.6.1. | El criterio de minimización | 37 |
| 3.6.2. | Triangulación lineal (<i>Linear-LS</i>) | 38 |
| 3.6.3. | Triangulación lineal iterativa (<i>Iterative-LS</i>) | 39 |
| 3.7. | Correspondencia | 40 |
| 3.7.1. | Generación del <i>codeword</i> | 40 |
| 3.7.2. | Recuperación del <i>codeword</i> | 42 |
| 3.7.3. | Correspondencia de puntos | 43 |
| 3.8. | Registro de vistas parciales | 44 |
| 3.9. | Superficies a partir de una nube de puntos | 45 |
| 3.10. | Dificultades | 46 |
| 3.11. | Resumen | 47 |
| 4. | Algoritmos de matching parcial | 49 |
| 4.1. | Introducción | 49 |
| 4.2. | Colecciones | 51 |
| 4.3. | Evaluación del algoritmos de búsqueda | 51 |
| 4.4. | <i>Partial Shape Retrieval</i> | 52 |
| 4.4.1. | <i>Range scan-based 3D model retrieval by incorporating 2D-3D alignment</i> | 53 |
| 4.4.2. | <i>Range scan-based 3D model retrieval by viewpoint entropy-based adaptive view clustering</i> | 54 |
| 4.4.3. | <i>Partial shape retrieval using data-aware partitioning</i> | 55 |
| 4.4.4. | <i>Partial shape retrieval using Bag of Features</i> | 56 |
| 4.4.5. | <i>Partial shape retrieval with spin images and signature quadratic form distance</i> | 57 |
| 5. | Evaluación de algoritmos | 61 |
| 5.1. | Colección 3D | 61 |
| 5.2. | Algoritmos evaluados | 63 |
| 5.2.1. | Spin images | 64 |
| 5.2.2. | Fast point feature histogram | 65 |
| 5.3. | Selección de puntos de interés | 66 |
| 5.4. | Experimentos | 66 |

Tabla de contenido

| | |
|--|-----------|
| 5.5. Resultados | 67 |
| 6. Conclusiones | 70 |
| 6.1. Conclusiones | 70 |
| 6.2. Trabajo futuro | 71 |
| A. Vistas parciales y colección de objetos | 73 |
| A.1. Vistas parciales de un objeto de la colección | 73 |
| A.2. Objetos 3D de la colección creada | 75 |
| Bibliografía | 82 |

Índice de figuras

| | | |
|------|--|----|
| 1.1. | A la izquierda, la sección de una malla 3D obtenida a partir de la digitalización de una muñeca, a la derecha, la sección de una malla 3D sintética que representa a una taza. | 3 |
| 1.2. | Diseño del escáner 3D | 10 |
| 2.1. | Metodologías para la obtención del modelo tridimensional de un objeto propuesta por B. Curless [5] | 12 |
| 2.2. | Bright-STRATO series CMM, cortesía de Mitutoyo America Corporation [6] | 13 |
| 2.3. | FaroArm Platinum, FARO Technologies Inc. [7] | 14 |
| 2.4. | Ilustración de <i>Industrial computerized axial tomography (CAT)</i> , Digital Image Processing, fig. 2.14, pag. 48 [8] | 15 |
| 2.5. | Ejemplo de reconstrucción de una imagen real de rostro utilizando <i>Shape from Shading</i> , Shape from Shading: a well-posed problem?, fig 6, pag 8 [9] | 16 |
| 2.6. | Imágenes tomadas a distinta distancia focal del hoyo de una placa de cerámica, de 70 micrómetros de diámetro, recubierto con una pasta de tungsteno. Shape from focus system, fig 6, pag 308 [10] | 17 |
| 2.7. | Ejemplo de reconstrucción de <i>Shape from Focus</i> utilizando las imágenes de la figura 2.6. La reconstrucción tridimensional muestra que el hoyo no fue cubierto lo suficiente por la pasta de tungsteno. Shape from focus system, fig 6, pag 308, [10] | 18 |
| 2.8. | Ejemplo de reconstrucción de la forma de un objeto utilizando <i>Shape from Silhouette</i> , A portable three-dimensional digitizer, fig 6, pag 202 [11] | 19 |
| 2.9. | Seis de cien fotografías tomadas a la escena de una mano, A theory of shape by space carving, fig. 9, pag 212 [12] | 20 |

| | |
|---|----|
| 2.10. Reconstrucción de una mano utilizando <i>Shape by Space Carving</i> , A theory of shape by space carving, fig. 10, pag 212 [12] | 21 |
| 2.11. Ilustración del problema de la correspondencia en <i>Shape from Stereo</i> , Shape and the stereo correspondence problem, fig 7, pag 7 [13] | 22 |
| 2.12. Esquema básico de obtención de la forma de un objeto mediante patrones de luz estructurada | 24 |
| 3.1. Escáner 3D creado para la evaluación de algoritmos de matching parcial. | 25 |
| 3.2. Ejemplificación del funcionamiento de una cámara estenopeica ideal . | 27 |
| 3.3. Diagrama geométrico de una cámara estenopeica ideal | 28 |
| 3.4. A la izquierda el patrón utilizado para calibrar la cámara, a la derecha dos tomas realizadas al patrón. | 31 |
| 3.5. Deformación provocada por los lentes de la cámara del tablet Asus TF300. Arriba la deformación radial, al medio la deformación tangencial y abajo el modelo completo. | 32 |
| 3.6. Proceso de calibración del proyector, proyectando un patrón conocido sobre un tablero con marcadores. | 33 |
| 3.7. Detección de los círculos en el proceso de calibración del proyector. Los puntos rojos corresponden a la imagen del proyector y los puntos azules corresponden a la imagen de la cámara (el patrón recuperado por la cámara). | 34 |
| 3.8. Deformación provocada por los lentes del proyector Epson Powerlite W12. Arriba la deformación radial, al medio la deformación tangencial y abajo el modelo completo. | 35 |
| 3.9. Ilustración del proceso de triangulación, dada una correspondencia de dos puntos: un punto x en el plano del proyector y un punto x' en el plano de la cámara. | 37 |
| 3.10. Secuencia de proyecciones utilizando código binario en un proyector de 8x8 pixeles | 42 |
| 3.11. Secuencia de proyecciones utilizando código de gray en un proyector de 8x8 pixeles | 42 |
| 3.12. A la izquierda, la imagen de un patrón proyectado sobre el rostro de una persona. El recuadro rojo indica la sección que se desea analizar. A la derecha, en azul se muestra la intensidad de luz de cada pixel y en rojo se muestra la intensidad de luz de cada pixel obtenida al proyectar el patrón en negativo | 43 |

| | |
|---|----|
| 3.13. Desplazamiento horizontal de las columnas de pixeles proyectadas y desplazamiento vertical de las filas de pixeles proyectadas al caer sobre el objeto a digitalizar. | 44 |
| 3.14. Ilustración del proceso de registro entre dos nubes de puntos, cada nube de puntos tiene un color distinto. | 45 |
| 3.15. A la izquierda, una sección de una nube de puntos sin triangular, a la derecha, la superficie 3D de la misma sección luego de la triangulación. | 46 |
| 3.16. Imprecisión en toda la línea de proyectores EPSON 3LCD, consistente en el desplazamiento de los pixeles azules con respecto al resto. | 47 |
| 3.17. Problemas en la captura del patrón proyectado cuando la escena a digitalizar tiene variaciones muy altas en su reflectancia. | 47 |
| 3.18. A la izquierda, objeto pintado con una tonalidad mate medianamente reflectiva para facilitar el proceso de digitalización. A la derecha, la nube de puntos obtenida. | 48 |
| | |
| 4.1. Ilustración del proceso de búsqueda por similitud utilizando una malla parcial como consulta. Arriba se muestra la malla parcial y abajo la lista resultados ordenados en base a alguna medida de similitud. | 50 |
| 4.2. Process to obtain the dataset. SHREC'13 Track: Large-Scale Partial Shape Retrieval Using Simulated Range Images [14] | 52 |
| 4.3. Flujo del algoritmo de recuperación <i>Range scan-based 3D model retrieval by incorporating 2D-3D alignment</i> , cortesía de [2] | 53 |
| 4.4. Proceso de generación del <i>Silhouette Feature View</i> de una vista parcial | 54 |
| 4.5. Flujo del algoritmo de recuperación <i>SBR-VC</i> , cortesía de [2] | 55 |
| 4.6. <i>Clustering</i> fijo vs <i>clustering</i> adaptativo. Signature quadratic form distance, Beecks et al. 2010 [15] | 58 |
| 4.7. (a) Dos <i>feature signatures</i> con sus respectivos centroides y pesos. (b) la estructura de la matriz de similitud. Beecks et al. 2010 [15] | 59 |
| | |
| 5.1. Dos nubes parciales obtenidas de la digitalización de un celular con un dispositivo <i>Microsoft Kinect</i> . <i>A large-scale hierarchical multi-view rgb-d object dataset</i> [16] | 62 |
| 5.2. Metodología para la captura de vistas parciales. Este mismo proceso se repite realizando las tomas desde arriba hacia abajo. | 63 |
| 5.3. Ejemplo de un objeto con sus respectivos puntos de interés, utilizando un detector de puntos de interés. | 64 |
| 5.4. <i>Spin Images</i> del modelo 3D de un patito de goma. [17] | 65 |
| 5.5. Comparación de los distintos métodos. | 68 |

| | |
|--|----|
| 5.6. Comparación del método de selección densa de puntos de interés y descriptor local <i>Fast Point Feature Histogram</i> con niveles de selección de puntos de interés cada 6mm, 8mm y 10mm. | 69 |
| A.1. Grupo <i>machinegun</i> | 75 |
| A.2. Grupo <i>svu</i> | 76 |
| A.3. Grupo <i>car</i> | 77 |
| A.4. Grupo <i>doll</i> | 78 |
| A.5. Grupo <i>shoe</i> | 79 |
| A.6. Grupo <i>bipedal dinosaur</i> | 80 |
| A.7. Grupo <i>quadrupedal dinosaur</i> | 81 |

Capítulo 1

Introducción

1.1. Introducción

Hoy en día la información multimedia está presente en todo ámbito de nuestra vida. La información tridimensional, en particular, se está volviendo parte esencial de diferentes aplicaciones, tales como, entretenimiento, medicina, seguridad y arte por nombrar algunas. Es importante, por lo tanto, estudiar el proceso de obtención de información tridimensional de un objeto, analizar la información obtenida y poder sacar conclusiones a partir de tal información.

El área de *3D Object Retrieval* (recuperación de objetos tridimensionales) se encarga de resolver el problema de recuperar un lista ordenada de objetos 3D (de una base de datos multimedia) de acuerdo a alguna medida de similitud dado un objeto 3D de consulta. La mayoría de las técnicas de recuperación de objetos 3D se basan en la forma completa del objeto 3D y en algunos casos presentan problemas cuando la consulta es solamente la parte de un objeto. El problema de recuperar los objetos 3D que contengan sólo una parte visualmente similar a otro objeto 3D es conocido como *matching* parcial. Este último es un desafío bastante interesante y continúa siendo un problema abierto del área.

Los algoritmos de matching global como *Depth buffer (DB)* [18], *Voxel (VX)* [18], *Complex (CP)* [19], *Rays-SH (RS)* [20, 19], *Silhouette (SL)* [18], *3DDFT (DF)* [21], *Shading-SH (SH)* [19], *Ray based (RA)* [22], *Rotation invariant (RI)* [23], *Harmonics 3D (H3)* [24], *Shape distribution (SD)* [25], *Ray moment (RM)* [20], *Cords based (CO)* [26], *Moments (MO)* [26], *Volume (VL)* [18] y *Shape spectrum (SS)* [27] han sido extensamente analizados [28, 29, 1]. No obstante los algoritmos de matching

parcial como *Salient geometric features* [30], *Salient spectral geometric features* [31], *Visual vocabulary signature* [32, 33], *Shape google* [34] y la metodología de *Signature Quadratic Form Distance* [15] son algoritmos del estado del arte cuya evaluación aún continúa vigente. Por otra parte, Dutagaci et al. [35] discutió las técnicas de matching parcial en donde el objeto de consulta es un *range image*¹ y los resultados de la búsqueda son los objetos 3D similares a los presentes en la escena.

Para comparar diferentes algoritmos de recuperación de objetos 3D se requiere de bases de datos de *benchmark* [3]. Dentro de las bases de datos de *benchmark* existentes, destacan, la *Princeton Shape Benchmark* [36], que consta de cerca de 1800 objetos recolectados desde Internet que representan objetos de la vida real, tales como vehículos, construcciones, animales o plantas; la *Konstanz 3D database* [28]; el *TOSCA dataset* [37]; y las bases de datos del *Shape Retrieval Contest (SHREC)*. Las bases de datos nombradas anteriormente contienen mayormente objetos 3D creados por personas (no son objetos provenientes de digitalizaciones de objetos físicos) y sólo contienen objetos 3D completos, sin incluir mallas parciales² de los objetos 3D para poder realizar consultas de matching parcial.

En la figura 1.1 se muestra un ejemplo de una malla obtenida a partir de la digitalización de una muñeca y una malla sintética creada por una persona. Se aprecia una clara diferencia entre las dos mallas. La malla digitalizada contiene un hoyo al interior de los labios de la muñeca, debido a que tal sección no se pudo digitalizar (la presencia de hoyos es común en las digitalizaciones de objetos reales), mientras que la malla sintética no contiene ningún hoyo (es un modelo perfecto). Por otra parte se aprecian claras diferencias en la densidad de puntos, la distribución de puntos y la estructura de la triangulación.

Uno de los principales problemas al trabajar con objetos 3D es que son escasos, cuesta conseguirlos, y si bien pueden construirse con software CAD (diseño asistido por computadora) es un proceso lento y tedioso. Para poder realizar mediciones de calidad en el área de 3D Object Retrieval, es necesario contar con bases de datos multimedia de miles de objetos 3D, lo cual es una limitante dado la escasez de objetos tridimensionales de acceso público. Contar con una manera de crear objetos 3D que representen objetos de la vida real de manera rápida puede ser un gran aporte a la investigación en el área de *3D object Retrieval*.

¹Un *range image* es una imagen que representa una escena tridimensional

²El término de mallas parciales se utilizará en este documento refiriéndose a las mallas geométricas que representan una vista parcial de un objeto 3D

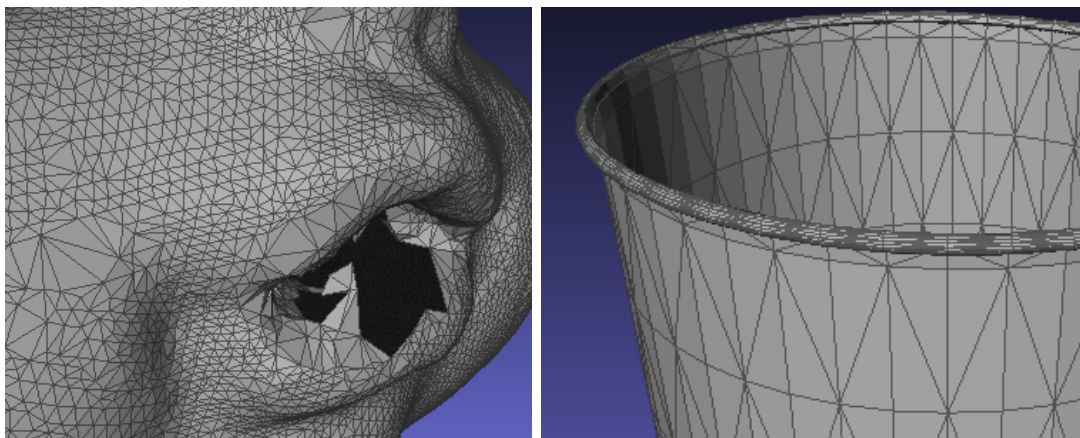


Figura 1.1: A la izquierda, la sección de una malla 3D obtenida a partir de la digitalización de una muñeca, a la derecha, la sección de una malla 3D sintética que representa a una taza.

En este trabajo de tesis se creó un escáner 3D óptico portátil capaz de generar mallas tridimensionales a partir de un objeto físico. La técnica utilizada para digitalizar los objetos fue la de patrones de luz estructurada [38, 39, 40], debido a su alta precisión. Se construyó una colección de referencia con los objetos creados por el escáner, que contiene tanto mallas parciales como objetos 3D completos, y se evaluó la efectividad de algoritmos de matching parcial en objetos 3D utilizando la colección de referencia creada mediante el escáner. En particular, se evaluó la metodología de *Signature Quadratic Form Distance* (SQFD) [4].

La importancia de este proyecto radica en que, actualmente, las mediciones de algoritmos de búsqueda de matching parcial en objetos 3D utilizan mallas tridimensionales artificiales para realizar consultas a una base de datos de benchmark, lo cual puede provocar que las mediciones obtenidas no sean del todo adecuadas. Estas mallas artificiales pueden ser obtenidas a partir de objetos de la misma base de datos modificados manualmente o mallas conseguidas de Internet. Se podrían obtener mediciones mucho más cercanas a la realidad si los algoritmos de búsqueda se evaluaran utilizando mallas geométricas obtenidas de objetos reales.

La evaluación de las técnicas del estado del arte en matching parcial en objetos 3D utilizando objetos reales puede ayudar a ajustar parámetros para distintas aplicaciones. De esta forma, se puede ayudar a acercar las técnicas del estado del arte a

aplicaciones concretas.

Adicionalmente, se podrían realizar consultas a una base de datos multimedia en tiempo real, tomando una “foto 3D” a un objeto y, luego, realizando una consulta a la base de datos con la “foto 3D” tomada. El proceso descrito anteriormente es una forma novedosa de consulta, que podría apoyar a la investigación en algoritmos de búsqueda de objetos 3D por similitud, al permitir una evaluación a tiempo real de la calidad de los algoritmos.

1.2. Aplicaciones actuales

Un escáner 3D óptico es un dispositivo que permite digitalizar objetos del mundo real en modelos tridimensionales mediante el uso de la luz. Entre las aplicaciones de un escáner 3D óptico destacan:

Entretenimiento: Los escáneres 3D son comúnmente usados para crear modelos 3D para videojuegos y películas, debido a que es más fácil digitalizar un objeto ya existente que tener que crear un objeto 3D manualmente con software de diseño.

Topografía de celdas solares [41]: Se ha utilizado un escáner 3D óptico para la medición topográfica de celdas solares durante el proceso de manufacturación, permitiendo analizar más de 2000 paneles por hora y obteniendo mediciones con una precisión de 0.5 mm.

Inspección forense [42]: El uso de un escáner 3D en la investigación de un escena del crimen puede ayudar a recopilar una mayor cantidad de información que con el uso de fotografías. Este tipo de escáner 3D también puede ser usado para investigaciones de accidentes automovilísticos o para reconstrucción de choques automovilísticos.

Patrimonio cultural: El uso de escáneres 3D en conjunto con impresoras 3D permite replicar obras culturales sin deteriorarlas, a diferencia de los métodos de duplicación con moldes de yeso, que en muchos casos pueden ser demasiado invasivos para obras frágiles [43]. Un caso famoso de digitalización de patrimonio cultural es el *The digital michelangelo project* [44], que consistió en una digitalización tridimensional de alta calidad del David de Miguel Ángel. También se han creado representaciones 3D altamente precisas de tablas antiguas con escritura cuneiforme [45], con el objetivo de preservación y análisis.

Medición de la espalda de pacientes con escoliosis [46]: La medición precisa de la topología de la espalda de un paciente puede ayudar al médico a medir la deformación de la espina dorsal. El posterior registro de estas mediciones tridimensionales permite observar los cambios de la espina dorsal a lo largo del tiempo. Por otra parte, se reduce la dependencia a las radiografías y, por consiguiente, se reduce la exposición del paciente a la radiación.

Análisis de ondas superficiales en el agua [47]: Se puede obtener una representación tridimensional de la superficie de un líquido, como el agua, para el análisis de las ondas superficiales.

Dentro de las aplicaciones que han utilizado búsqueda por similitud de objetos tridimensionales, algunas mencionadas en [48], se encuentran:

Detección de objetos peligrosos en equipajes [49]: Mediante la obtención de modelos tridimensionales por tomografía computarizada (CT) del equipaje de un pasajero, es posible determinar si esos modelos tridimensionales corresponden a objetos peligrosos, como armas de fuego o armas blancas.

Reconocimiento facial 3D [50]: Existe una extensa investigación en el área de matching de modelos tridimensionales de la forma de la cara humana para aplicaciones de reconocimiento facial.

Comparación geométrica de zapatos [51]: Dado un escaneo 3D de un pie, el autor propone una metodología geométrica para encontrar el zapato con mejor calce.

Análisis de la forma de los hipocampos en niños [52]: En numerosos estudios se muestra que la epilepsia en niños solo se da en casos donde el hipocampo de los niños es significativamente más grande que el de los niños sanos. No obstante, se ha encontrado un número significativo de niños con un hipocampo grande que no ha desarrollado la enfermedad. La solución médica para estas situaciones es, dependiendo del volumen del hipocampo, removerlo completamente mediante cirugía cerebral. Las representaciones 3D de los hipocampos son obtenidas mediante *MRI* o *CT scanning*.

La observación de niños con un gran hipocampo que no tienen epilepsia lleva a la hipótesis de que la forma de la deformación del hipocampo puede indicar la presencia de la enfermedad. Usando una base de datos de representaciones 3D de hipocampos, las deformaciones que provocan epilepsia pueden ser mejor

estudiadas y la cirugía puede ser mejorada removiendo sólo las partes afectadas del hipocampo. Por otra parte, se puede usar la base de datos para buscar casos similares y realizar la decisión de cirugía basada en el resultado de la búsqueda.

Clasificación del polen mediante su representación 3D [53]: Utilizando representaciones 3D de polen, obtenidas mediante un *confocal laser scanning microscope*, se propone un esquema para reconocer de manera totalmente automática los 26 tipos principales de polen aéreos.

Análisis y recuperación de objetos 3D en los museos [54]: Dentro de los objetivos de SCULPTEUR se encuentra el extender las facilidades digitales de navegación y recuperación de objetos 3D de los 5 principales museos de Europa.

Reconocimiento de orejas humanas en 3D [55]: La oreja humana es una medida biométrica en la que los investigadores han puesto atención últimamente.

Clasificación de objetos 3D para apoyar la investigación craneofacial [56]: Se presenta una metodología para representar y clasificar objetos 3D para ser usada en estudios de dismorfología craneofacial.

Clasificación y recuperación de proteínas 3D [57, 58, 59]: Se proponen esquemas para clasificar proteínas basadas en su representación tridimensional y por sus propiedades físico-químicas. Como resultado de estas investigaciones se proponen variados usos médicos. Un ejemplo de uso, propuesto por los autores, es el de una proteína que tiene una función crucial para el tratamiento de una enfermedad terminal, pero que tiene severos efectos adversos. Si se lograra encontrar una proteína de la misma familia, con estructura y funcionalidad similar, pero sin los efectos adversos, se podrían obtener beneficios inmensos para el tratamiento de ciertas enfermedades.

Recuperación de objetos 3D sólidos para usos en ingeniería [60]: Para aplicaciones ingenieriles, un nuevo diseño puede crearse eficientemente a partir de un modelo anterior de las mismas características. Es importante, por lo tanto, contar con buenas maneras de recuperar objetos 3D similares al que se desea construir.

Arqueología [61]: Un problema recurrente en esta área es la unión de fragmentos dispersos de alguna pieza arqueológica. Los autores proponen una metodología para reconstruir automáticamente una pieza a partir de sus fragmentos 3D.

Similitud en secuencias de video 3D [62]: Se pueden realizar búsqueda por similitud en secuencias de video tridimensionales.

Robótica: Existe una inmensa cantidad de trabajos en los cuales se utiliza visión robótica 3D para reconocer objetos.

Se observa, por lo tanto, la importancia de evaluar y comparar distintos enfoques que puedan aparecer en el ámbito de búsqueda por similitud de objetos tridimensionales para que, de este modo, se pueda contar con algoritmos de búsqueda eficaces y eficientes.

1.3. Terminología

Multimedia: Multimedia se utiliza para referirse a cualquier objeto que utiliza múltiples medios de expresión para presentar o comunicar información. Los medios pueden ser variados, pudiendo ser imágenes, texto, sonido, video, mallas 3D, etc.

Malla geométrica 3D: Una malla es un conjunto de caras poligonales (cuadriláteros y triángulos) que representa de manera discreta una superficie en el espacio, tal que si t_i y t_j son dos elementos de una malla T , entonces: $t_i \cap t_j$ es un vértice común, una arista común, o el conjunto vacío. Una malla tiene asociada un conjunto de elementos topológicos tales como: vértices, aristas y caras poligonales. Las mallas utilizadas en este trabajo de tesis están compuestas únicamente por triángulos (son las mallas más comunes para este tipo de aplicaciones).

Objeto 3D: En este trabajo de tesis se utilizará el concepto de ‘objeto 3D’ como la malla geométrica 3D que representa la superficie de un objeto completo.

Malla parcial: Se utilizará el concepto de malla parcial como la malla geométrica 3D que representa la superficie de una sección de un objeto.

Efectividad o Eficacia: Mide la calidad de las respuestas retornadas por un algoritmo de búsqueda, es decir, su habilidad para recuperar objetos relevantes y evitar los objetos no relevantes para una consulta dada.

Eficiencia: Cantidad de recursos necesarios para lograr el efecto que se desea de un algoritmo de búsqueda. Se mide en tiempo de CPU y tiempo de E/S. Existen estructuras de datos que permiten agilizar búsquedas, tales como índices multidimensionales [63] o índices métricos [64].

1.4. Objetivos

Colección de referencia: Es una base de datos que contiene cientos de objetos multimedia. Estas bases de datos se utilizan para evaluar algoritmos de búsqueda, ya sea en eficacia y/o eficiencia.

***Object retrieval* (Recuperación de objetos):** La recuperación de objetos se refiere a la obtención de objetos de una colección en base a algún criterio.

Matching global: Se refiere a los algoritmos que recuperan objetos de la base de datos que luzcan similares al objeto de consulta. La comparación se realiza utilizando la forma completa del objeto.

Matching parcial: Dada una malla parcial como consulta, el *matching* parcial busca encontrar los objetos de la base de datos que contengan partes similares.

1.4. Objetivos

1.4.1. Objetivo general

El objetivo general de este trabajo de tesis es la generación de una colección de referencia abierta a la comunidad, mediante un escáner 3D óptico, para la evaluación de algoritmos de matching parcial utilizando datos reales. Los objetivos específicos logrados en este trabajo se describen en la siguiente sección.

1.4.2. Objetivos específicos

Creación de un escáner 3D: Creación de un escáner 3D óptico a partir de un tablet Android y un proyector. El escáner creado es capaz de digitalizar en tres dimensiones cualquier objeto que refleje luz de manera uniforme (cualquier objeto de superficie lambertiana).

Generación de mallas geométricas 3D a partir de nubes de puntos parciales: Triangulación de nubes de puntos con el fin de crear superficies suaves, minimizando el error contenido en éstas y registro (unión) de vistas parciales para crear objetos 3D completos.

Creación de una colección de referencia de objetos 3D: Creación de un benchmark para evaluar algoritmos de búsqueda parcial. La colección creada es de acceso público y permite que cualquier investigador la utilice para evaluar sus algoritmos.

Evaluación de algoritmos de matching parcial: Evaluación de la metodología de matching parcial de descriptores locales con *clustering* local y distancia *Signature Quadratic Form Distance* (SQFD) [4], utilizando distintos algoritmos generadores de descriptores.

1.5. Metodología

En primera instancia se analizó el estado del arte de las distintas metodologías para crear representaciones tridimensionales de un objeto. A grandes rasgos, es posible separar las metodologías en dos grandes grupos. Las metodologías que requieren contacto físico con el objeto (sección 2.2.1), que dados sus altos costos están fuera del alcance de este trabajo, y las que no lo requieren (sección 2.2.2), donde destacan las metodologías ópticas (sección 2.3), en las cuales se enfocó esta investigación dado su bajo costo y versatilidad. Entre las metodologías estudiadas están las metodologías ópticas pasivas (sección 2.3.1), que no requieren intervención de la iluminación sobre el objeto, y las metodologías ópticas activas, que requieren modificar las condiciones de iluminación sobre el objeto. Dentro del grupo anterior destacan: la metodología de digitalización por láser (sección 2.3.2), consistente en la proyección de una línea láser que barre un objeto mientras un sensor captura como se distorsiona la línea, y las metodologías de patrones de luz estructurada (sección 2.3.2), donde se proyecta uno o varios patrones luminosos sobre el objeto a digitalizar. Luego de analizar las metodologías existentes, se implementó la técnica de patrones de luz estructurada, en particular, la de *gray codes* (sección 3), dada su alta precisión y relativamente bajo tiempo de digitalización.

El diseño del dispositivo se puede observar en la Figura 1.2. Consta de un Tablet Android y un proyector, conectados entre sí por un cable HDMI. En la figura, el Tablet está posicionado a la izquierda del proyector, no obstante, podría estar arriba, abajo, o a la derecha de éste. En la sección 3 se discute con mayor profundidad el diseño del escáner.

Las nubes de puntos generadas por el escáner fueron procesadas para crear mallas 3D, utilizando algoritmos de triangulación ya existentes (sección 3.6), donde se repararon las mallas, relleno agujeros (dentro de lo posible) y eliminando puntos redundantes. Cada una de las 480 digitalizaciones produjo una nube de puntos parcial. Para crear los 30 objetos 3D completos fue necesario juntar varias nubes de puntos parciales, para lo cual se utilizó un algoritmo de registro basado en ICP (*Iterative Closest Point*) (sección 3.8). Cabe notar que cada nube de puntos parcial

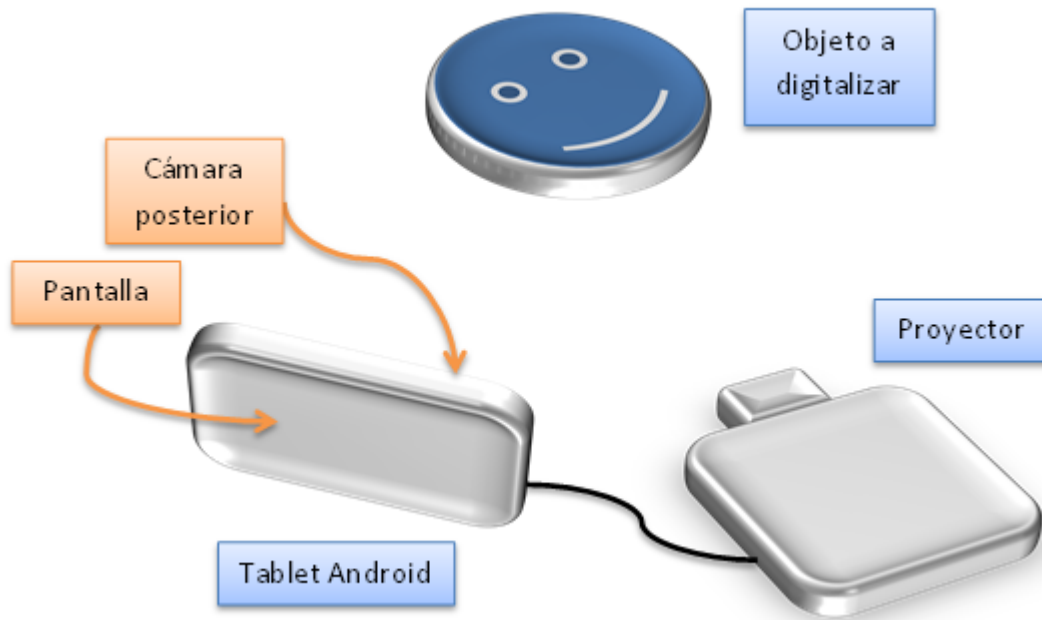


Figura 1.2: Diseño del escáner 3D

tiene su propio sistema de coordenadas, por lo que juntar nubes de puntos para crear un objeto 3D puede resultar costoso en términos de cómputo.

Se creó una colección de referencia con 480 mallas parciales de consulta y 30 objetos, agrupados en 7 clases: *machinegun*, *suv*, *car*, *doll*, *shoe*, *bipedaldinosaur* y *quadrupedaldinosaur*. La colección es abierta a la comunidad y cualquier investigador la puede usar para evaluar sus algoritmos. En la sección 5.1 se discute en mayor profundidad los detalles sobre esta colección. Por último, utilizando la colección creada se evaluó la metodología de descriptores locales, con *clustering* local y distancia *Signature Quadratic Form Distance* (SQFD) sobre distintos descriptores, en términos de eficacia. El proceso de evaluación se detalla en la sección 5.2.

Capítulo 2

Marco teórico escáner 3D

2.1. Introducción

La temática discutida en este capítulo se enfocará en las metodologías para la creación de un escáner 3D, con el fin de generar una colección de referencia de objetos 3D que contenga mallas parciales, para finalmente evaluar algoritmos de matching parcial del estado del arte. Se comenzará con el marco teórico relacionado a las metodologías para digitalizar un objeto y posteriormente se analizará el marco teórico relacionado a la evaluación de algoritmos de búsqueda.

2.2. Metodologías para digitalizar un objeto

En la actualidad existe una serie de metodologías con las cuales se puede crear un modelo tridimensional de un objeto. Cada metodología tiene sus ventajas y desventajas. Las metodologías existentes se pueden organizar en una taxonomía [5], tal como se muestra en la figura 2.1.

2.2.1. Escáneres por contacto

Las metodologías por contacto requieren contacto físico con el objeto y son usadas por lo general en aplicaciones donde la precisión es esencial. El tiempo necesario para digitalizar un objeto mediante estas metodologías es bastante alto, al igual que el costo del equipo necesario. Dentro de ellas se encuentran *Slicing*, *Coordinate measuring machine (CMM)* y *Jointed arms*.

2.2. Metodologías para digitalizar un objeto

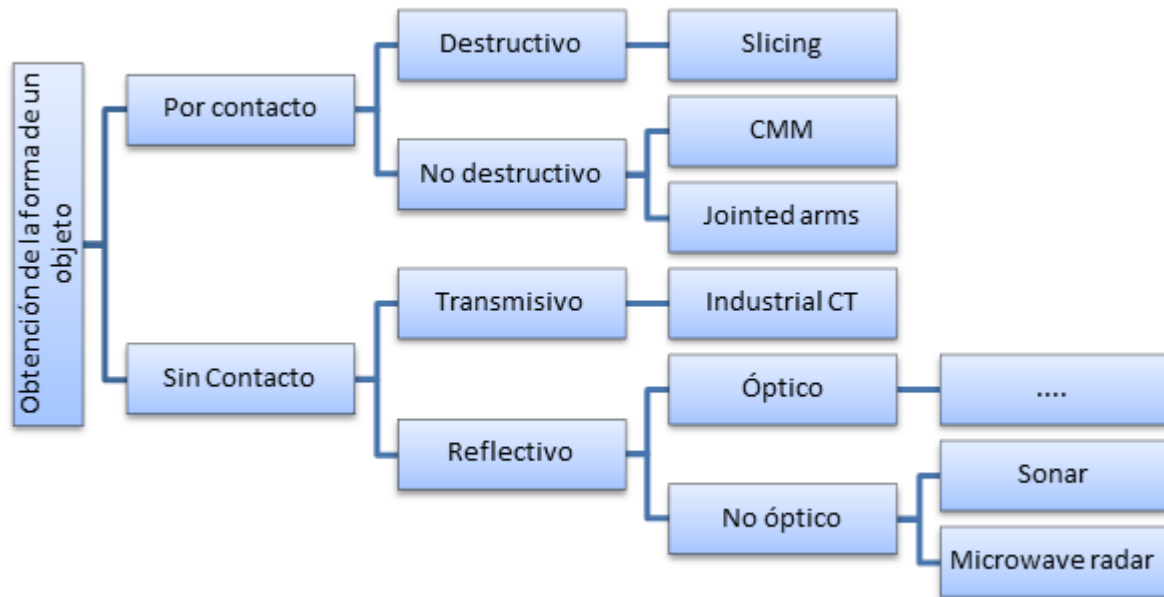


Figura 2.1: Metodologías para la obtención del modelo tridimensional de un objeto propuesta por B. Curless [5]

Slicing

Se corta el objeto en tajadas y luego se toma una imagen de cada tajada. La precisión de la digitalización dependerá del grosor de cada tajada y de la resolución de las imágenes capturadas. Estas metodologías son usadas en áreas como la medicina y la biología.

Coordinate measuring machine (CMM)

Es una máquina que contiene un sonda de contacto que puede moverse en los ejes X, Y, Z. La sonda de contacto generalmente es manejada por un operador. Si bien son efectivas, los métodos de contacto con los objetos pueden dañar objetos frágiles y requieren de largos períodos para crear modelos tridimensionales precisos. Se puede ver un ejemplo de este dispositivo en la figura 2.2.



Figura 2.2: Bright-STRATO series CMM, cortesía de Mitutoyo America Corporation [6]

Jointed arms

Es una máquina que contiene una sonda de contacto en la punta de un brazo robótico. El brazo robótico tiene tantos grados de libertad como juntas rotatorias tenga. Se puede ver un ejemplo de este dispositivo en la figura 2.3.

2.2.2. Escáneres sin contacto

Metodologías que no requieren un contacto físico con el objeto. Estos *scanners* permiten superar las limitantes de los *scanners* por contacto (la necesidad de un contacto físico con el objeto). Dentro de ellas se encuentran *Industrial Computed Tomography (Industrial CT)* (tomografía industrial computarizada), *Sonar*, *Microwave radar* (radar de microondas) y las metodologías ópticas.

Industrial Computed Tomography (Industrial CT)

Es un proceso que utiliza rayos X para producir representaciones tridimensionales de un objeto tanto interna como externamente. Se usa principalmente en análisis científicos. Entre los campos de interés específicos se encuentran detección de fallas, mediciones geométricas de piezas no accesibles, inspección de ensamblados y análisis estadísticos de las propiedades de un material, tales como su distribución de densidad [65]. Se puede observar un ejemplo de este proceso en la figura 2.4.



Figura 2.3: FaroArm Platinum, FARO Technologies Inc. [7]

Sonar

Es un mecanismo que utiliza la propagación del sonido o del ultrasonido generalmente bajo un líquido. El sonar emite pulsos de sonido y escucha los ecos a través de un transductor.

Microwave radar

Es un mecanismo que utiliza pulsos electromagnéticos (generalmente microondas). El radar recibe las ondas que rebotan en la superficie de un objeto generalmente a través de la ayuda de una antena con forma paraboloide.

Óptico

Estas técnicas utilizan la luz visible para generar un modelo tridimensional de un objeto. Tienen las fortalezas de no necesitar contacto físico con el objeto, son seguras, rápidas y menos costosas que las técnicas por contacto, no obstante, tienen las limitantes de sólo poder adquirir datos de las partes visibles de un objeto, ser sensibles a las características de la superficie de un objeto (transparencia, brillo, color y opacidad) y ser susceptible a interreflexiones [5].

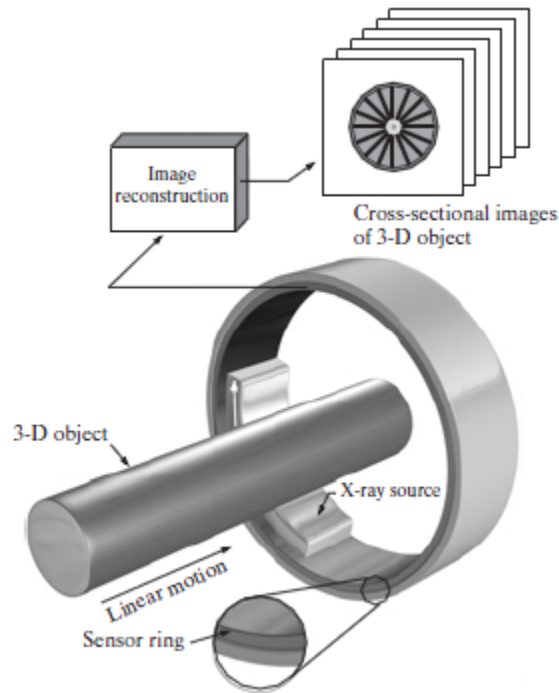


Figura 2.4: Ilustración de *Industrial computerized axial tomography (CAT)*, Digital Image Processing, fig. 2.14, pag. 48 [8]

2.3. Metodologías ópticas

Los escáneres ópticos se pueden separar en dos categorías principales, los activos y los pasivos.

2.3.1. Metodologías ópticas pasivas

Los scanners ópticos pasivos no requieren control directo de ninguna fuente de iluminación. Por el contrario, dependen totalmente de la luz ambiente. Dentro de estas metodologías se encuentran *Stereo vision* [66], *Shape from shading* [67], *Shape from focus and defocus* [68, 69], *Shape from silhouette* [70] y *Shape by Space Carving* [12].

Shape from Shading

Shape from Shading es una metodología para determinar la forma de la superficie de un objeto a partir de una imagen de éste [71]. *Shape from Shading* recupera la

forma de un objeto a partir de la variación gradual de la sombra en la imagen. Para resolver el problema de *Shape from Shading* es necesario comprender cómo se creó la imagen. El modelo Lambertiano define la formación de una imagen donde el valor en escala de gris de cada pixel depende de la dirección de la fuente de luz y de la dirección de la normal de la superficie. Dada una imagen en escala de grises, el objetivo es obtener la forma de la superficie para cada pixel de la imagen. La reconstrucción de la forma a partir de las sombras se puede reducir a resolver una ecuación diferencial parcial no lineal de primer orden. No obstante, la mayoría de las imágenes no siguen el modelo Lambertiano, lo que hace que *Shape from Shading* sea raramente usado en mediciones reales [67]. Se han hecho mejoras al esquema tradicional de *Shape from Shading*, modificando las ecuaciones diferenciales parciales utilizadas en el proceso de reconstrucción [9]. Un ejemplo de reconstrucción utilizando *Shape from Shading* puede observarse en la figura 2.5.

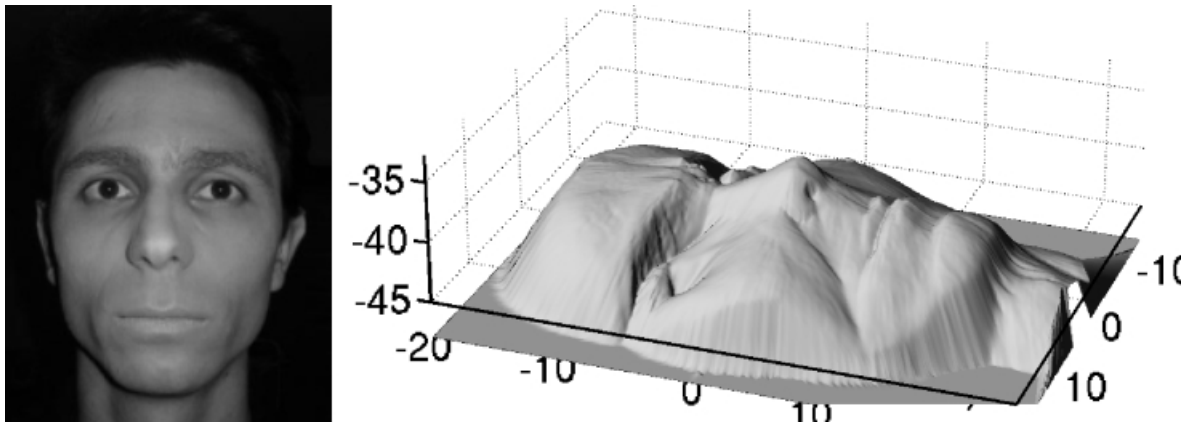


Figura 2.5: Ejemplo de reconstrucción de una imagen real de rostro utilizando *Shape from Shading*, *Shape from Shading: a well-posed problem?*, fig 6, pag 8 [9]

Shape from Focus/Defocus

En las imágenes provenientes de sistemas ópticos que utilizan lentes convexas, los objetos a cierta distancia del lente estarán enfocados, mientras que los objetos a otras distancias se verán borrosos o desenfocados en una magnitud que dependerá de la distancia del objeto al lente. En *Shape from Focus* se varía un parámetro de una cámara, como la distancia focal, hasta encontrar sectores enfocados en la imagen. El proceso se repite para distintas distancias focales. La distancia del objeto al lente se puede deducir a partir de una fórmula del lente [72].

2.3. Metodologías ópticas

En *Shape from Defocus* se considera la magnitud del desenfoque para determinar la distancia del lente a un objeto [73], requiriendo de este modo, una menor cantidad de imágenes para reconstruir la forma del objeto. *Shape from Defocus* requiere de alrededor de 3 imágenes para reconstruir la forma de un objeto, mientras que *Shape from Focus* requiere de una cantidad bastante mayor (alrededor de 10 imágenes) [69].

Para que tanto *Shape from Focus* como *Shape from Defocus* puedan funcionar adecuadamente es necesario conocer exactamente los parámetros ópticos del sistema. El cómputo necesario para la reconstrucción suele ser costoso. Un ejemplo de *Shape from Focus*, donde se analiza una pequeña pieza de cerámica de un microcircuito electrónico, puede observarse en las figuras 2.6 y 2.7.

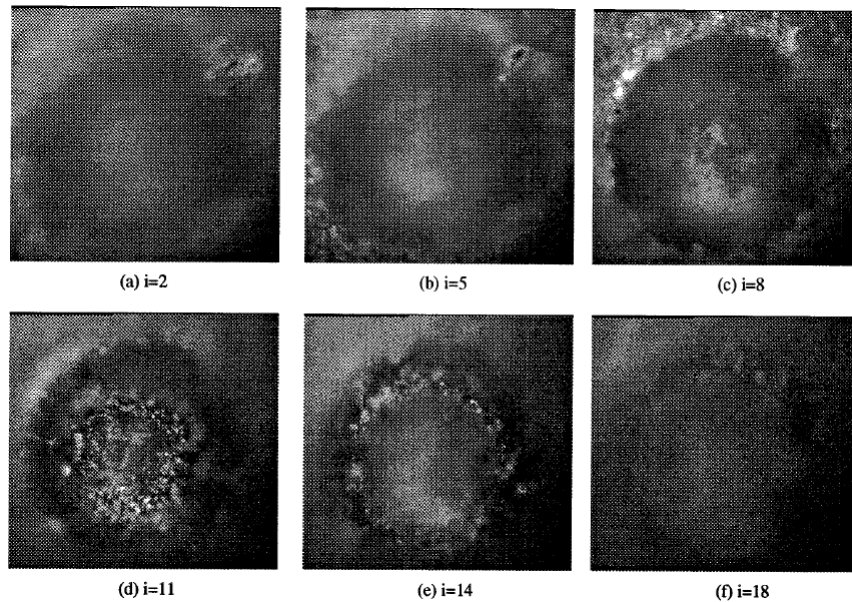


Figura 2.6: Imágenes tomadas a distinta distancia focal del hoyo de una placa de cerámica, de 70 micrómetros de diámetro, recubierto con una pasta de tungsteno. *Shape from focus system*, fig 6, pag 308 [10]

Shape from Silhouette

Bajo una situación controlada, donde se tenga un fondo plano o conocido, los bordes externos de un objeto pueden ser detectados de manera confiable. Como resultado de esto se han propuesto una serie de metodologías para obtener la forma de

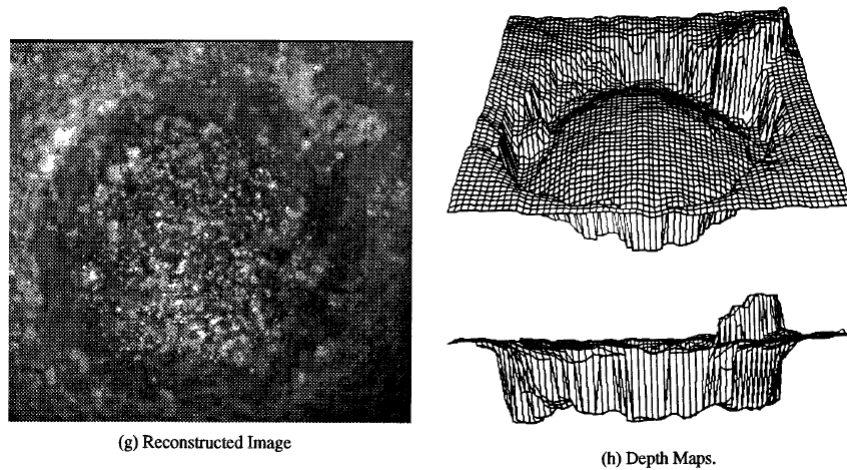


Figura 2.7: Ejemplo de reconstrucción de *Shape from Focus* utilizando las imágenes de la figura 2.6. La reconstrucción tridimensional muestra que el hoyo no fue cubierto lo suficiente por la pasta de tungsteno. Shape from focus system, fig 6, pag 308, [10]

un objeto a través de su silueta. La familia de metodologías se denominó *Shape from Silhouette*. Laurentini [70] consideró una cantidad finita de cámaras observando una escena. Se definió el *Visual hull* como la mejor aproximación de un objeto que puede ser obtenida con la metodología de la intersección de volúmenes.

Últimamente se ha utilizado esta metodología en *free-viewpoint video* [74], permitiendo un ajuste dinámico del *viewpoint* [75, 76]. Un ejemplo de la reconstrucción de la forma de un objeto utilizando esta metodología se puede observar en la figura 2.8.

Shape by Space Carving

Considera el problema de reconstruir la forma 3D de un objeto desconocido, a partir de múltiples fotografías tomadas desde posiciones conocidas pero determinadas arbitrariamente. Se prueba la existencia del *photo hull*, y se procesa a partir de las fotografías de la escena [12]. Un ejemplo de reconstrucción utilizando esta metodología se puede observar en las figuras 2.9 y 2.10

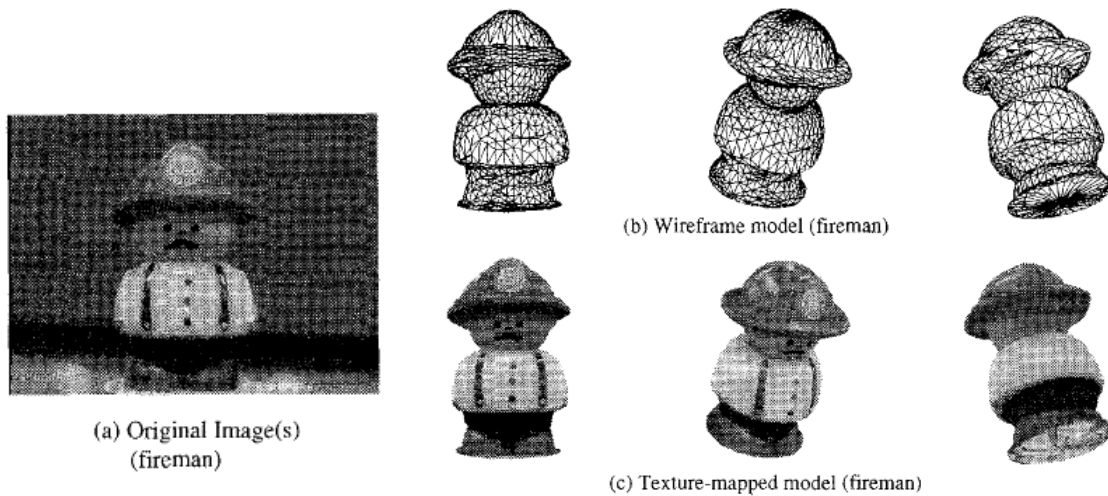


Figura 2.8: Ejemplo de reconstrucción de la forma de un objeto utilizando *Shape from Silhouette*, A portable three-dimensional digitizer, fig 6, pag 202 [11]

Shape from Stereo

Es una de las metodologías pasivas más utilizadas. Imitando al sistema visual de los humanos se intenta reconstruir la forma de un objeto utilizando dos (o más) imágenes [77]. Comparada con los sistemas activos, es una metodología mucho más económica en términos de implementación inicial. No obstante, encontrar correspondencia entre las dos imágenes se ha transformado en un problema particularmente difícil de resolver en las últimas décadas [66]. El desarrollo de algoritmos que resuelvan el problema de la correspondencia continúa abierto [78]. Una ilustración del problema de la correspondencia puede observarse en la figura 2.11.

Algunas técnicas como *Trinocular* [79], *multi-view stereo* [80], basadas en la correlación [81, 8] y técnicas de multi-resolución [82, 83] han sido introducidas con el fin de mejorar la precisión de los sistemas estereoscópicos.

2.3.2. Metodologías ópticas activas

Dentro de éstas, se encuentran las metodologías de *Photometric* [84], *Laser stripe scanning* [44] y *Structured Light Patterns* [38, 39, 40].

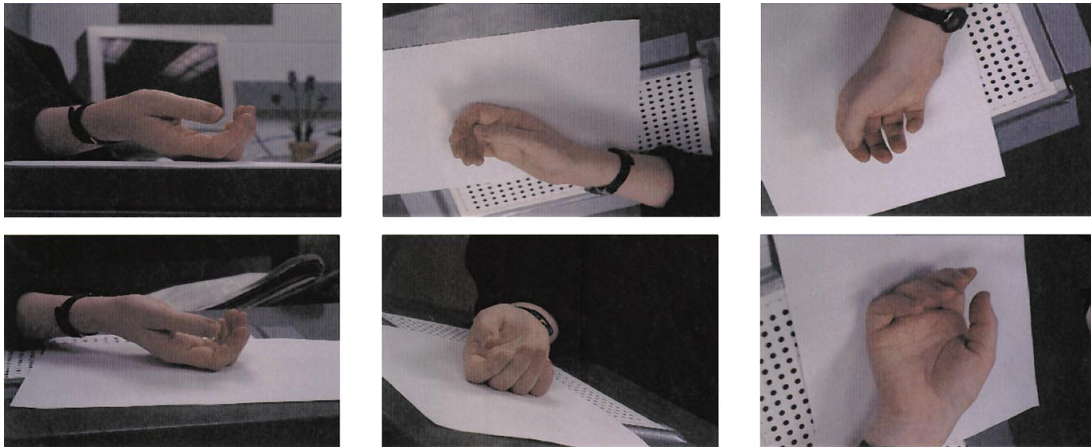


Figura 2.9: Seis de cien fotografías tomadas a la escena de una mano, *A theory of shape by space carving*, fig. 9, pag 212 [12]

Photometric stereo

Photometric stereo es una variante de *Shape from Shading* [84]. Estima la forma de la superficie de un objeto usando una secuencia de imágenes de la misma superficie tomada desde el mismo punto de vista, pero bajo una iluminación proveniente de distintas ubicaciones [85, 86, 87]. Se ha probado que es posible utilizar *Photometric stereo* a tiempo real utilizando tres fuentes de iluminación con colores distintos [88].

Laser Stripe Scanning

Inicialmente se proyectaba un punto láser sobre la superficie de un objeto y se utilizaba una relación de triangulación para extraer información sobre la superficie del objeto [89]. La proyección de un punto sobre un objeto se puede considerar como el caso más básico de patrón de luz estructurada [90]. Posteriormente se desarrollaron técnicas de digitalización mediante líneas láser. En estas técnicas una línea láser realizaba un barrido sobre la superficie de un objeto. Una cámara capturaba la luz reflejada y se reconstruía la forma del objeto por triangulación. En la actualidad existe una serie de digitalizadores comerciales que utilizan la metodología de líneas láser debido a su alta precisión, entre ellos se encuentran: *Handyscan 3D* [91], *Polhemus FastSCAN* [92] y *NextEngine 3D Laser Scanner* [93] dentro de otros.

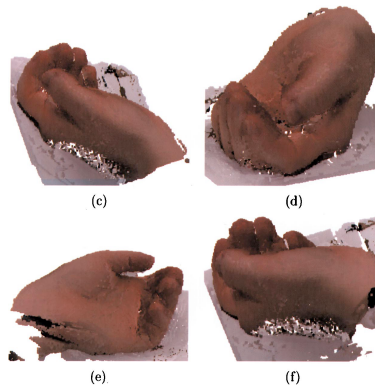


Figura 2.10: Reconstrucción de una mano utilizando *Shape by Space Carving*, A theory of shape by space carving, fig. 10, pag 212 [12]

Structured Light Patterns

El principal problema de los sistemas pasivos de visión estereoscópica, es que el proceso de encontrar correspondencia se complica en presencia de objetos sin textura [94]. Es por lo anterior que estos sistemas están limitados a reconstruir objetos con texturas densas [38]. Los métodos basados en luz estructurada solucionan esta problemática creando correspondencia para cada posición en la imagen. Un sistema de luz estructurada se puede entender como un sistema de visión estereoscópica, en donde una cámara ha sido reemplazada por un proyector [90]. El patrón proyectado otorga la ilusión de textura en el objeto [95]. Por lo anterior, la reconstrucción de la forma del objeto se puede realizar observando las diferencias entre el patrón proyectado y el patrón observado, deformado por la forma del objeto. Las técnicas de digitalización mediante patrones de luz estructurada están consideradas dentro de las más confiables [40].

Los método de digitalización mediante patrones de luz estructurada se pueden clasificar del siguiente modo [40]:

Métodos de codificación discretos Todos los métodos donde el patrón proyectado es discreto.

Multiplexación espacial Son el grupo de técnicas donde el identificador de una posición específica en el patrón (*codeword*) es obtenido a partir de sus puntos aledaños. Se pueden clasificar dentro de esta categoría a las secuencias de *De Bruijn* [96], los métodos no formales de codificación y los patrones *M-array* [97].

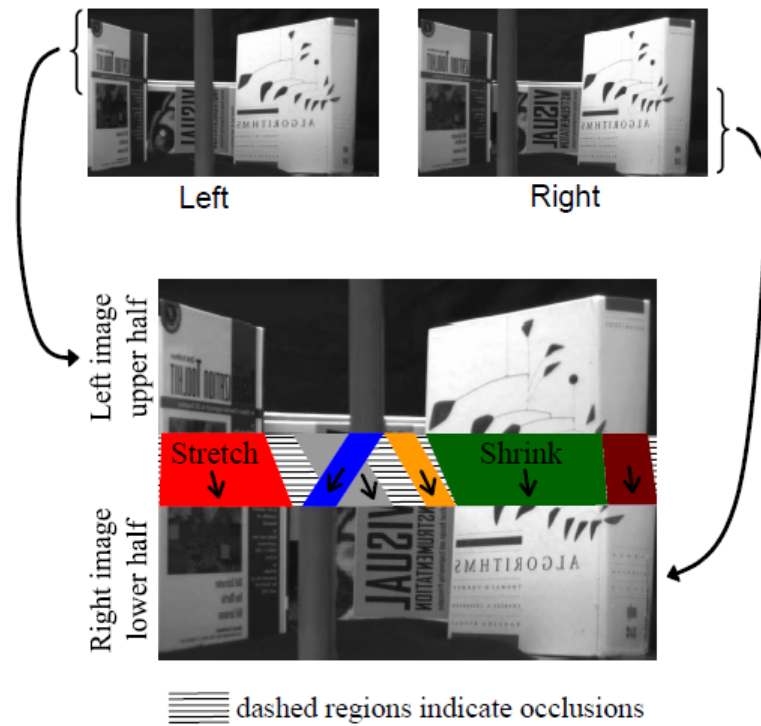


Figura 2.11: Ilustración del problema de la correspondencia en *Shape from Stereo*, Shape and the stereo correspondence problem, fig 7, pag 7 [13]

La técnica de *M-array* [97] consiste en la generación de un arreglo binario pseudoaleatorio de dimensión $r \times v$, en donde cada submatriz de dimensión $n \times m$ aparece exactamente una sola vez. Un ejemplo de utilización de esta metodología es el dispositivo *Microsoft Kinect* [98], el cual proyecta el arreglo binario utilizando luz infrarroja.

Multiplexación temporal En esta categoría se encuentran todas las técnicas en donde el identificador de una posición específica (*codeword*) en el patrón es obtenido a partir de una secuencia de proyecciones en la superficie del objeto. Por lo tanto, el *codeword* no estará completamente formado hasta que toda la serie de patrones haya sido proyectada. Se pueden clasificar dentro de este grupo a las técnicas de códigos binarios, códigos n-arios, códigos híbridos y los métodos de desplazamiento de patrones discretos.

Métodos de codificación continuos Aquí se encuentran todas las técnicas en cu-

Los patrones se muestran variaciones continuas en la intensidad de la luz o en el color proyectado, ya sea en un eje o en los dos ejes.

Desplazamiento de fase (*Phase shifting*) Al proyectar una onda sinusoidal bidimensional en una superficie, cada punto a lo largo de la línea paralela al eje de codificación puede ser identificada por un valor único de fase. Para cualquier objeto no plano esta proyección se deformará provocando una desviación de la fase. La desviación de la fase otorga información sobre la forma de la superficie del objeto. Es posible recuperar la forma del objeto mediante la observación entre la onda proyectada y el patrón recuperado. No obstante, para poder recuperar la desviación de la fase, es necesario desplazar la onda sinusoidal y volver a proyectarla, varias veces. Dentro de esta categoría se encuentran las técnicas de desplazamiento simple y las técnicas de desplazamiento múltiple, en donde se proyectan ondas de distinta frecuencia.

Multiplexación de frecuencia La metodología de multiplexación de frecuencia permite recuperar la desviación de la fase realizando un análisis en el dominio de frecuencia en vez de un análisis en el dominio espacial.

Gradientes continuos En estas metodologías el identificador de una posición específica (*codeword*) está dado únicamente por el valor del pixel. Como todos los pixeles deben ser distintos, estas metodologías están limitadas a la cantidad de pixeles distintos que pueda emitir el proyector. Por otra parte estas metodologías son altamente sensibles al ruido y tienen poca sensibilidad para determinar la forma de la superficie del objeto.

El esquema más simple que se puede montar consiste en un generador de patrones de luz estructurada, que puede ser un proyector digital o un dispositivo óptico/mecánico y una cámara. Un ejemplo del montaje mínimo para utilizar esta metodología se puede observar en la figura 2.12. Las imágenes capturadas por el sensor óptico deben ser procesadas para determinar la forma de objeto en base al patrón recuperado.

Limitaciones Una de las limitaciones de los esquemas de luz estructurada son las interreflexiones ópticas en objetos cóncavos y la dispersión bajo superficie (*Subsurface scattering*). Para disminuir los efectos indeseados nombrados anteriormente se han propuesto varios esquemas [99].

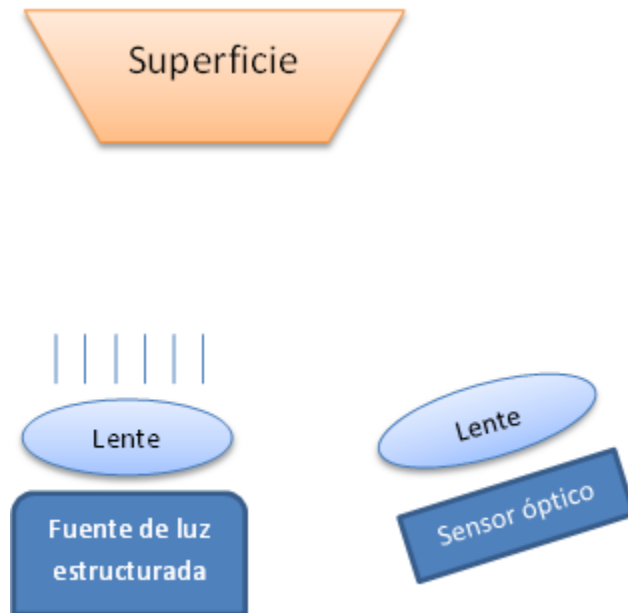


Figura 2.12: Esquema básico de obtención de la forma de un objeto mediante patrones de luz estructurada

2.4. Conclusiones

Dentro de las metodologías estudiadas, que van desde las metodologías que requieren contacto físico con el objeto, como las que no lo requieren, podemos encontrar que la metodología que tiene mayores ventajas es la de *Structured Light Patterns*, debido a su bajo costo, alta resolución y versatilidad. El resto de la metodologías necesita de equipamiento personalizado o entrega una resolución muy baja.

Capítulo 3

Escáner 3D óptico basado en patrones de luz estructurada

3.1. Introducción

Se construyó un escáner 3D basado en patrones de luz estructurada con el objetivo de obtener las vistas parciales 3D para la evaluación de algoritmos de matching parcial. El escáner consta de un tablet Android y un proyector digital conectado al tablet mediante HDMI, ver figura 3.1. En este capítulo se describe la teoría detrás del escáner, el proceso de registro de las nubes de puntos parciales y la creación de superficies a partir de una nube de puntos.

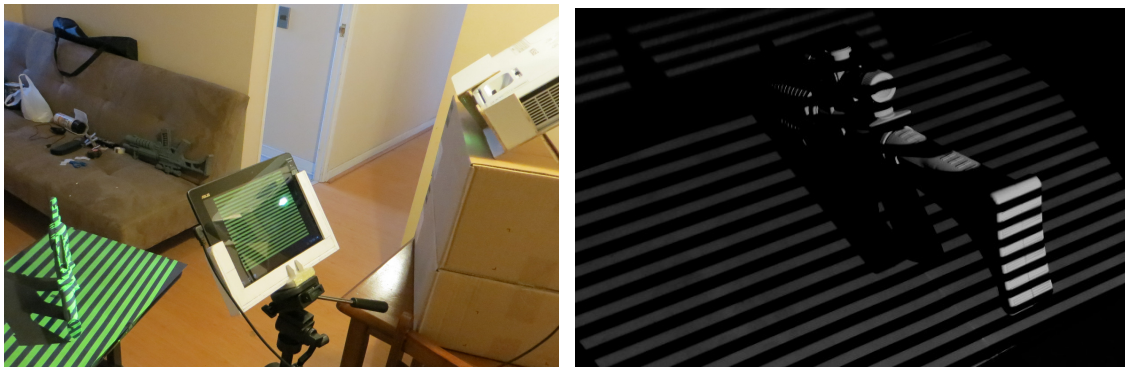


Figura 3.1: Escáner 3D creado para la evaluación de algoritmos de matching parcial.

Se utilizó un tablet android Asus TF300, con una cámara con auto-enfoque de 8MP con capacidad de tomar ráfagas de imágenes con una resolución de 1920x1080

3.2. Modelo de una cámara

(imágenes adquiridas en el proceso de digitalización), procesador quadcore NVIDIA Tegra 3, 16GB de almacenamiento, 1GB de memoria RAM y salida HDMI con una resolución de imagen de 1280x720. Adicionalmente se usó un proyector EPSON Powerlite W12, con enfoque manual, una intensidad lumínica de 2800 lumens, resolución de 1280x800 y entrada HDMI. El tablet se conectó con el proyector mediante un cable microHDMI-HDMI.

3.2. Modelo de una cámara

3.2.1. Cámara estenopeica (pinhole camera model)

Una cámara estenopeica ideal es una cámara fotográfica que no utiliza lentes para formar una imagen. El funcionamiento de esta cámara consiste en hacer pasar la luz por un pequeño agujero hacia una caja oscura.

Las cámaras modernas se pueden modelar, a grandes rasgos, utilizando el modelo el modelo de una cámara estenopeica ideal. El modelo de una cámara estenopeica ideal no contempla distorsiones geométricas o el difuminado provocado por objetos desenfocados, efectos causados por el uso de lentes, no obstante, provee una descripción razonable de cómo una cámara representa una escena 3D en un plano bidimensional.

Un ejemplo de una cámara estenopeica ideal se puede observar en la figura 3.2. La distancia entre el plano real y el agujero (*pinhole*) es la distancia focal f .

Generalmente se asume que la imagen que entrega una cámara estenopeica es la imagen proveniente del plano virtual, por simplicidad matemática, aunque en efecto, la imagen que realmente se obtiene se una cámara estenopeica es la imagen proveniente del plano real, la cual es la misma imagen que la del plano virtual pero invertida.

Un diagrama geométrico del funcionamiento de una cámara estenopeica ideal se puede observar en la figura 3.3. El plano virtual R es perpendicular al eje Z y se encuentra a una distancia f del centro óptico C .

La proyección de un punto del mundo 3D $(X, Y, Z)^T$, en el plano virtual en un

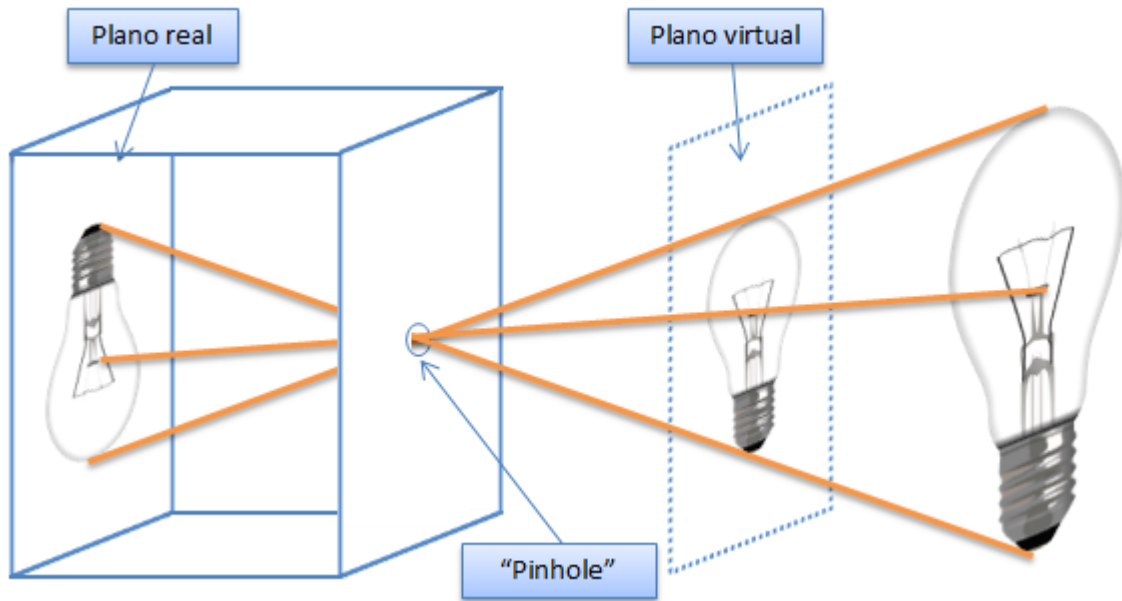


Figura 3.2: Ejemplificación del funcionamiento de una cámara estenopeica ideal

posición $(u, v)^T$ puede ser expresada, por teorema de Tales como:

$$u = \frac{Xf}{Z}, v = \frac{Yf}{Z} \quad (3.1)$$

donde f denota la distancia focal. Para evitar la operación no lineal de división, la relación anterior se puede reformular usando el esquema de la geometría proyectiva, la cual se puede expresar en una representación matricial del siguiente modo:

$$w \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (3.2)$$

donde $w = Z$ es el factor de escala homogéneo. Cabe destacar que en la ecuación 3.2 se considera que el origen del sistema de coordenadas coincide con el centro óptico C , que el eje óptico coincide con el eje Z y que el origen del sistema de coordenadas del plano virtual (plano de la imagen) coincide con el eje Z . Un modelo más general se presentará en la siguiente sección.

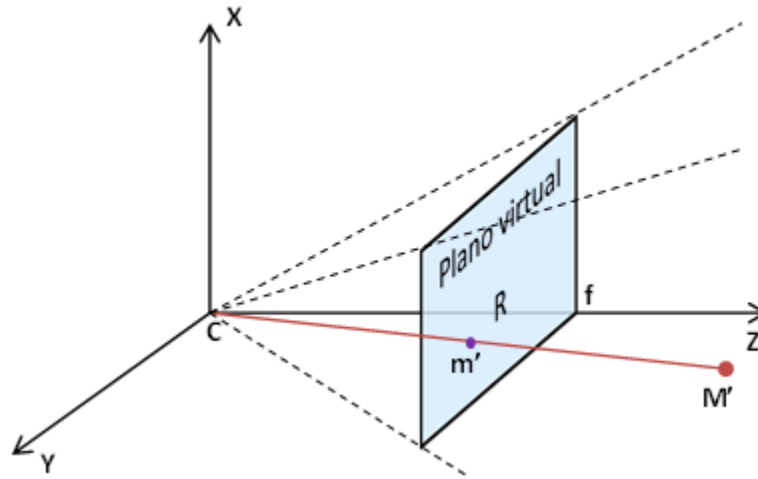


Figura 3.3: Diagrama geométrico de una cámara estenopeica ideal

3.2.2. Modelo de una cámara digital CCD/CMOS

El modelo anterior se puede extender para modelar una cámara digital CCD/CMOS. Una cámara digital es muy similar a una cámara estenopeica con las diferencias de que en el plano real se encuentra un sensor digital CCD o CMOS, que esencialmente es una matriz de celdas fotosensibles, y en que la luz en vez de pasar por un pequeño agujero (*Pinhole*), pasa por uno o varios lentes. La ecuación general para modelar una cámara digital, sin considerar la deformación provocada por los lentes, es la siguiente:

$$w \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & \gamma & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (3.3)$$

O de un modo más general:

$$wm' = A[R|t]M' \quad (3.4)$$

La matriz resultante de la multiplicación $A[R|t]$ se denomina matriz de la cámara, y generalmente se denota con la letra P .

M' representa un punto en el espacio 3D, en coordenadas homogéneas.

3.2. Modelo de una cámara

$[R|t]$ corresponde a la matriz de rotación-traslación, comúnmente llamada matriz de los parámetros extrínsecos. Esta matriz representa la posición de la cámara dentro de una escena estática o la posición de un objeto frente a una cámara estática, es decir, $[R|t]$ transforma las coordenadas del mundo 3D (X, Y, Z) hacia un sistema de coordenadas fijo con respecto a la cámara.

La matriz A representa la matriz de la cámara, comúnmente conocida como la matriz de los parámetros intrínsecos. Esta matriz no depende de la escena, y una vez determinada, permanece constante mientras la distancia focal de la cámara no cambie, lo cual puede ocurrir en cámaras con zoom. Dentro de los parámetros intrínsecos se tiene $f_x = m_x \cdot f$ y $f_y = m_y \cdot f$, donde f es la distancia focal, m_x y m_y son los factores de escala para transformar unidades de distancia a píxeles, el punto principal (c_x, c_y) que por lo general se encuentra al centro de la imagen y el coeficiente de inclinación γ . La razón f_x/f_y es llamada la proporción de aspecto y es distinta de 1 si los píxeles de la cámara no son cuadrados. El coeficiente γ codifica el ángulo entre el eje X y el eje Y del sensor óptico presente en la cámara, permitiendo de este modo representar a una cámara que utilice píxeles romboidales.

La ecuación 3.3, cuando $z \neq 0$, es equivalente a:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + t$$
$$\begin{aligned} x' &= x/z \\ y' &= y/z \\ u &= f_x \cdot x' + c_x \\ v &= f_y \cdot y' + c_y \end{aligned}$$

Las cámaras, en realidad, producen una distorsión en las imágenes capturadas debido a sus lentes internos. Un modelo que aproxima de buena manera la distorsión producida por un lente es el modelo que introdujo Brown en 1966 [100], que considera una distorsión radial polinómica y una distorsión tangencial producida por un descentramiento, o un centrado imperfecto de los componentes del lente y otros defectos de fabricación en un lente compuesto.

El modelo anterior se puede extender para incluir estas distorsiones siguiendo el

3.3. Calibración de la cámara

modelo propuesto por Bouguet en 2004 [101], del siguiente modo:

$$\begin{aligned} \begin{pmatrix} x \\ y \\ z \end{pmatrix} &= R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + t \\ x' &= x/z \\ y' &= y/z \\ r^2 &= x'^2 + y'^2 \\ x'' &= x'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 x' y' + p_2 (r^2 + 2x'^2) \\ y'' &= y'(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + p_1 (r^2 + 2y'^2) + 2p_2 x' y' \\ u &= f_x \cdot x'' + c_x \\ v &= f_y \cdot y'' + c_y \end{aligned} \tag{3.5}$$

donde k_1 , k_2 y k_3 son los coeficientes de distorsión radial, mientras que p_1 y p_2 son los coeficientes de distorsión tangencial.

Dentro de otros enfoques para estimar la deformación provocada por los lentes de una cámara está el modelo propuesto por Zhang en 1999, que considera una distorsión radial simétrica de 4to orden sin componente tangencial [102] y el modelo propuesto por Heikkila en 1997, que considera tanto la distorsión radial como la distorsión tangencial [103].

3.3. Calibración de la cámara

El proceso de calibración de la cámara, conocido comúnmente como *camera resectioning*, consiste en encontrar los parámetros intrínsecos, extrínsecos y los parámetros de distorsión de una cámara en particular. Uno de los métodos más usados en la actualidad es el propuesto por Zhang en 2000, que consiste en la observación de un patrón plano desde al menos 2 perspectivas distintas [104].

La técnica propuesta por Zhang sólo requiere de una cámara que observe un patrón plano desde al menos dos perspectivas distintas. El patrón puede ser impreso en una impresora de escritorio y luego ser adjuntado a una superficie razonablemente plana. Tanto la cámara como el patrón pueden ser movidos a mano. La posición relativa entre la cámara y el patrón no se necesita conocer, lo único que se necesita conocer es el tamaño del patrón impreso, en alguna unidad de medida (generalmente

3.3. Calibración de la cámara

mm).

Si bien basta con observar el patrón desde dos perspectivas distintas, el autor recomienda utilizar una mayor cantidad de tomas para obtener una mejor calidad en la solución.

La implementación realizada utilizó el algoritmo de Bouguet [101] para calibrar la cámara. Se capturaron 20 tomas a un patrón de calibración compuesto por una grilla asimétrica de 44 círculos. Para cada toma es necesario encontrar las coordenadas del centro de cada círculo, las 44 coordenadas encontradas (en píxeles) para cada toma se confrontan con las 44 coordenadas del patrón original (en milímetros) para encontrar los parámetros internos de la cámara.

El patrón usado para calibrar la cámara y algunas tomas capturadas por la cámara posterior del tablet ASUS TF300, que fueron utilizadas para el proceso de calibración se pueden observar en la figura 3.4.

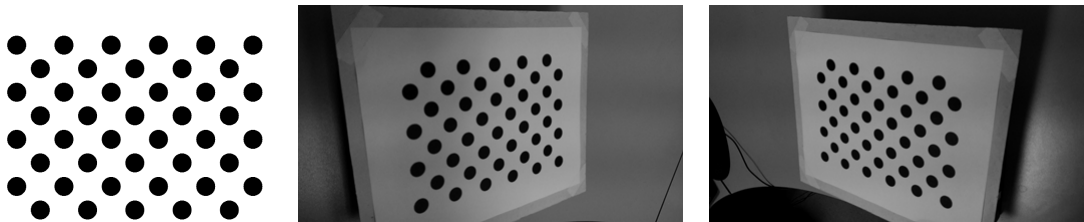


Figura 3.4: A la izquierda el patrón utilizado para calibrar la cámara, a la derecha dos tomas realizadas al patrón.

Para el caso particular de la cámara posterior presente en el tablet ASUS TF300, realizando tomas con una resolución de 1920x1080, se encontraron los siguientes parámetros intrínsecos: $f_x = 1571,18$, $f_y = 1571,18$, $c_x = 959,5$, $c_y = 539,5$ y $\gamma = 0$. Por otra parte se encontraron los siguientes parámetros de distorsión radial: $k_1 = 0,0519$, $k_2 = 0,164$ y $k_3 = -0,606$, y los siguientes parámetros de distorsión tangencial: $p_1 = -0,00156$ y $p_2 = -0,00286$.

Una representación de la deformación provocada por los lentes de la cámara puede observarse en la figura 3.5. Las imágenes fueron creadas utilizando MATLAB y el visualizador creado por Bouguet [101].

3.3. Calibración de la cámara

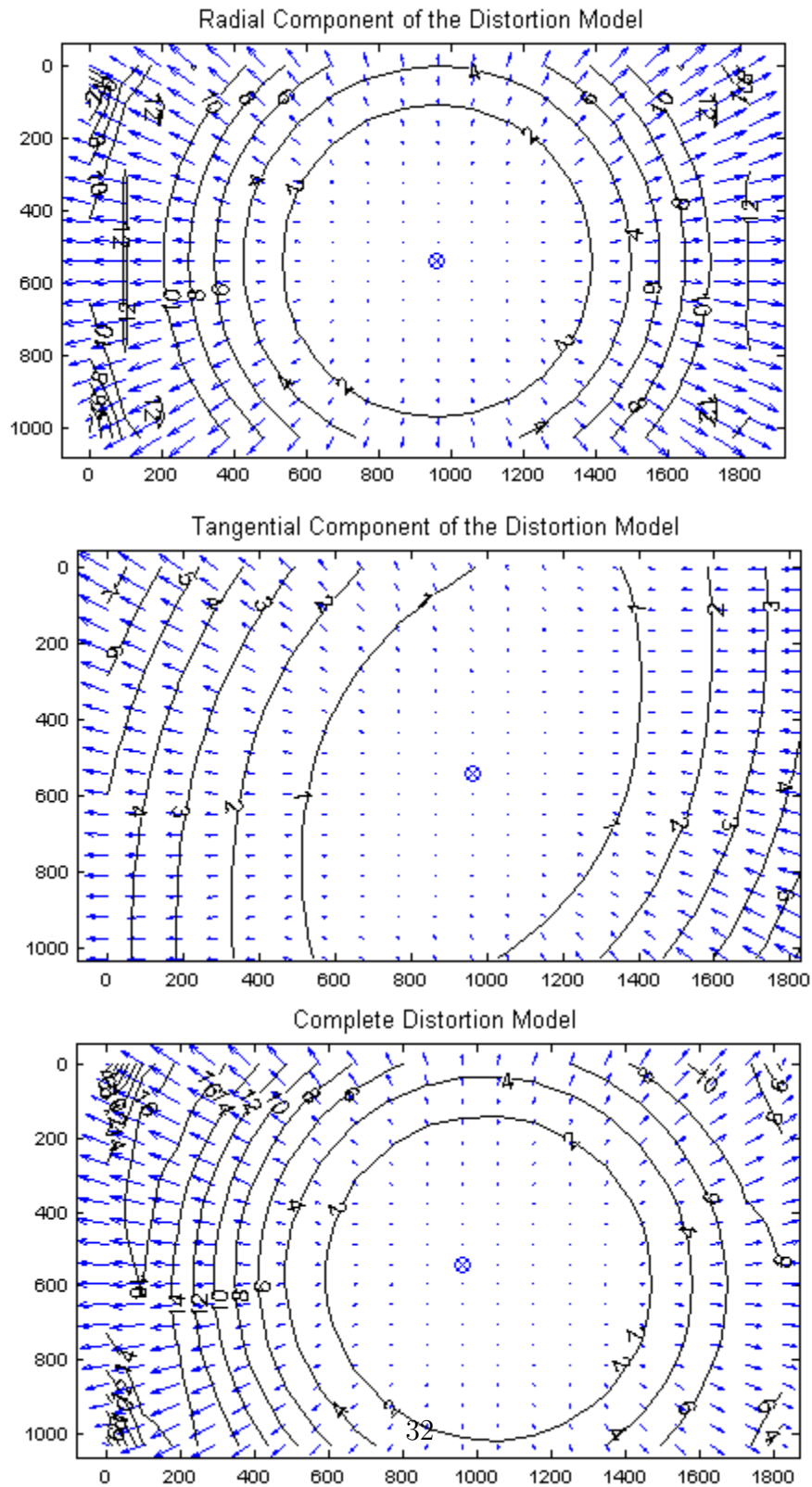


Figura 3.5: Deformación provocada por los lentes de la cámara del tablet Asus TF300. Arriba la deformación radial, al medio la deformación tangencial y abajo el modelo completo.

3.4. Calibración del proyector

Una vez calibrada la cámara, es posible calibrar los parámetros intrínsecos del proyector de manera análoga a la calibración de la cámara, pero considerando que el proyector es una cámara inversa.

La metodología elegida para la calibración del proyector consiste en proyectar un patrón conocido sobre un tablero con marcadores. Teniendo una cámara calibrada, los marcadores del tablero permiten determinar la ubicación del tablero en el espacio 3D. Conociendo la imagen del proyector, la posición del tablero en el espacio 3D, y la forma de la proyección en el tablero captada por la cámara, es posible calibrar los parámetros intrínsecos del proyector utilizando un procedimiento análogo a la calibración de la cámara. En la figura 3.6 se puede apreciar el proceso de calibración del proyector.



Figura 3.6: Proceso de calibración del proyector, proyectando un patrón conocido sobre un tablero con marcadores.

Para que los resultados del proceso de calibración sean estables, se utilizó la correspondencia de la proyección de un patrón de 56 círculos en 20 posiciones distintas del tablero. Un ejemplo de la detección de los círculos por el escáner es presentada en la figura 3.7.

Para el caso particular del proyector utilizado en este escáner, realizando proyecciones con una resolución de 1280x720, se encontraron los siguientes parámetros intrínsecos: $f_x = 1708,34$, $f_y = 1716,22$, $c_x = 627,94$, $c_y = 733,47$ y $\gamma = 0$. Por otra parte se encontraron los siguientes parámetros de distorsión radial: $k_1 = -0,096$, $k_2 = 0,581$ y $k_3 = -1,284$, y los siguientes parámetros de distorsión tangencial: $p_1 = 0,00858$ y $p_2 = 0,00163$.

3.4. Calibración del proyector

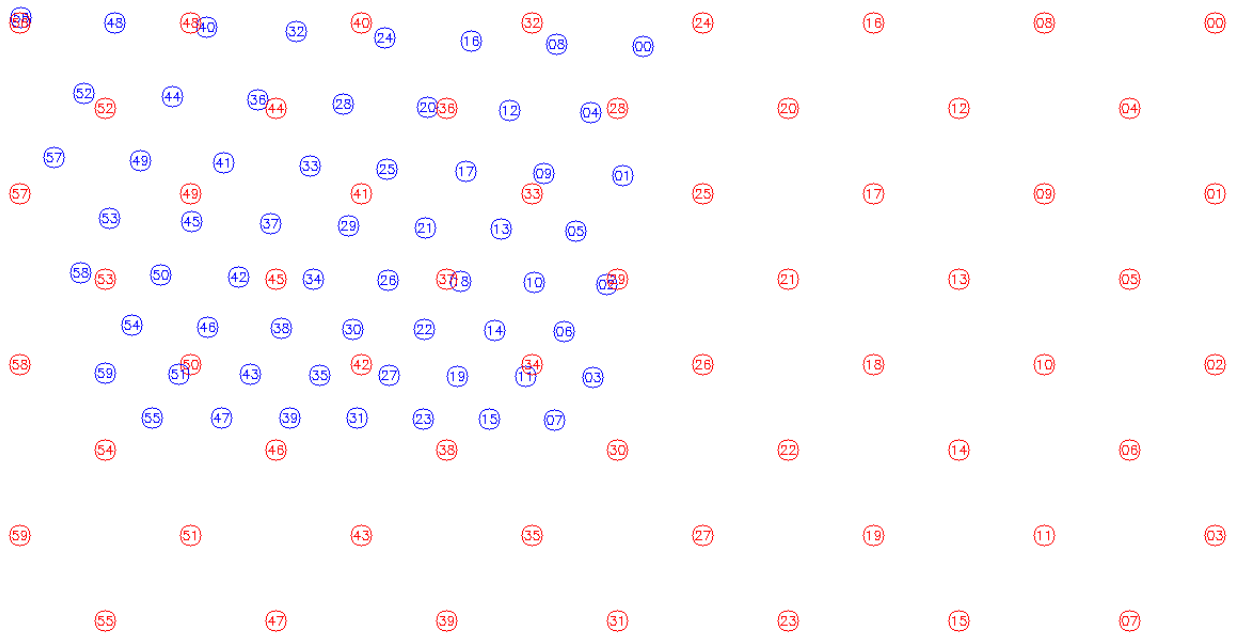


Figura 3.7: Detección de los círculos en el proceso de calibración del proyector. Los puntos rojos corresponden a la imagen del proyector y los puntos azules corresponden a la imagen de la cámara (el patrón recuperado por la cámara).

Una representación de la deformación provocada por los lentes del proyector puede observarse en la figura 3.5. Las imágenes fueron creadas utilizando MATLAB y el visualizador creado por Bouguet [101].

3.4. Calibración del proyector

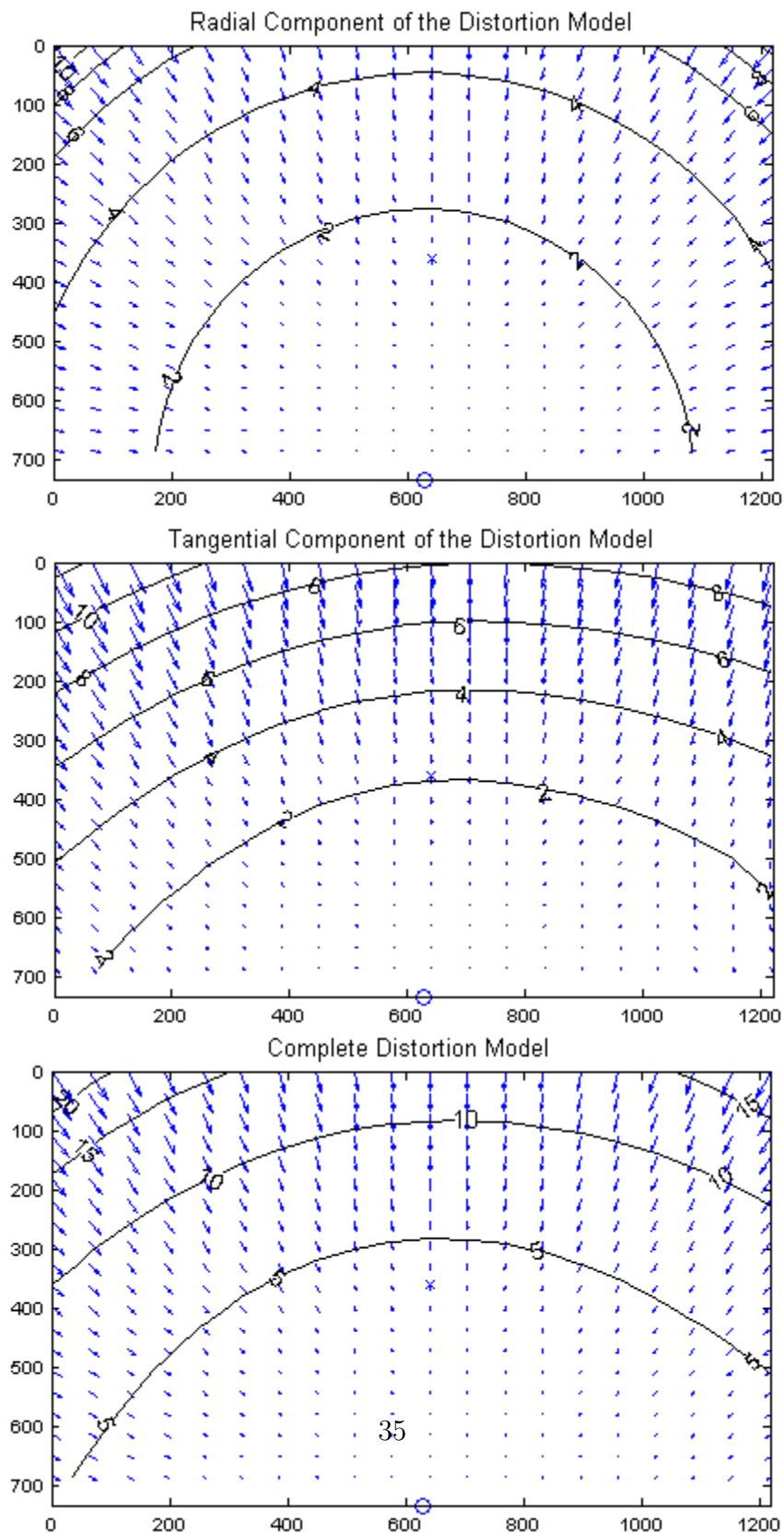


Figura 3.8: Deformación provocada por los lentes del proyector Epson Powerlite W12. Arriba la deformación radial, al medio la deformación tangencial y abajo el modelo completo.

3.5. Calibración extrínseca

Una vez calibrados los parámetros intrínsecos del sistema se pueden obtener los parámetros extrínsecos, que representan la posición relativa entre el centro del proyector y el centro de la cámara (traslación y rotación). En términos matemáticos, los parámetros extrínsecos del sistema se pueden agrupar en la siguiente matriz:

$$[R|t] = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{pmatrix}$$

donde R es la matriz de rotación y t es el vector de traslación. Cabe destacar que los parámetros de la matriz de rotación no son independientes y representan en realidad 3 grados de libertad.

A diferencia de los parámetros intrínsecos, los parámetros extrínsecos del sistema se deben recalcular cada vez que cambia la posición relativa entre la cámara y el proyector. Para poder obtener estos parámetros es necesario utilizar el tablero de calibración usado en la sección anterior.

3.6. Reconstrucción por triangulación

Una vez que se tienen determinados los parámetros intrínsecos y extrínsecos, tanto de la cámara como del proyector, se puede reconstruir en el espacio 3D cada punto que tenga correspondencia entre el plano del proyector y el plano de la cámara.

En la figura 3.15 se muestra el caso de un punto x proyectado desde el proyector, que llega a la superficie de un objeto y que la cámara lo observa como un punto x' en su plano correspondiente. La intersección de las proyecciones de x y x' determina dónde se encuentra el punto X en el espacio 3D.

Desafortunadamente, es poco probable que las proyecciones de x y x' efectivamente se intersecten en algún punto del espacio 3D, debido a que la medición de los puntos x y x' no es exacta. El ruido que presenta el sistema impide determinar la intersección de ambos rayos, por lo se deberá estimar la mejor solución para el punto X en el espacio 3D.

La mejor solución requiere de la definición y minimización de una función de costo adecuada. Es deseable encontrar un método de triangulación que sea invariante a

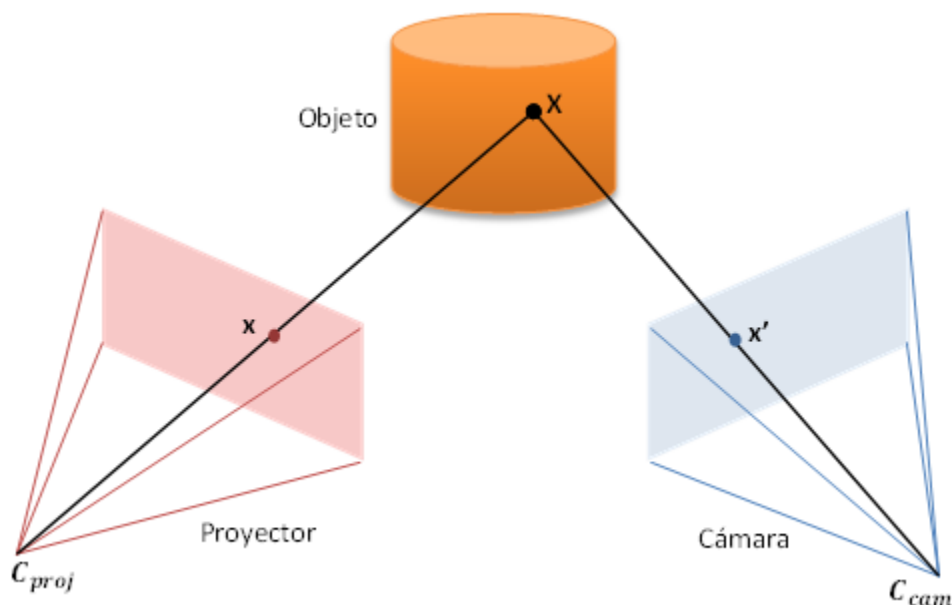


Figura 3.9: Ilustración del proceso de triangulación, dada una correspondencia de dos puntos: un punto x en el plano del proyector y un punto x' en el plano de la cámara.

transformaciones proyectivas del espacio [80].

Existe una serie de metodologías distintas para realizar el proceso de triangulación en presencia de ruido [105]. En este trabajo se analizarán dos: el método *Linear-LS* y el método *Iterative-LS*.

3.6.1. El criterio de minimización

Se suponen conocidas las matrices del proyector P , la matriz de la cámara P' y por lo tanto la matriz fundamental F , la cual se puede calcular a partir de las matrices P y P' [106]. Los dos rayos correspondientes a las proyecciones de los puntos correspondientes $x \leftrightarrow x'$ se intersectarán en algún punto del espacio 3D si y solo si satisfacen la restricción epipolar [107]:

$$x'^{\top} F x = 0 \tag{3.6}$$

Debido al ruido presente en el sistema, típicamente los puntos correspondientes $x \leftrightarrow x'$ no satisfarán la restricción epipolar 3.6. En efecto, los puntos que realmente

3.6. Reconstrucción por triangulación

satisfarán la relación 3.6 son los puntos $\hat{x} \leftrightarrow \hat{x}'$, lo cuales estarán muy cerca de los puntos correspondientes $x \leftrightarrow x'$, de modo que $\hat{x}^\top F \hat{x}' = 0$.

Para resolver el problema de la triangulación será necesario encontrar los puntos \hat{x} y \hat{x}' que minimicen la función (distancia de Sampson):

$$d(x, \hat{x})^2 + d(x', \hat{x}')^2 \quad (3.7)$$

Donde $d(*, *)$ es la distancia Euclidiana, sujetos a la restricción epipolar:

$$\hat{x}^\top F \hat{x}' = 0 \quad (3.8)$$

3.6.2. Triangulación lineal (*Linear-LS*)

Es uno de los métodos más comunes debido a su simplicidad, aunque no asegura una estimación óptima.

Para el proyector se tiene $x = PX$, mientras que para la cámara se tiene $x' = P'X$. Estas ecuaciones pueden combinarse para formar una ecuación del tipo $AX = 0$, la cual es una ecuación lineal en X .

En la ecuación del proyector $x = PX$, la variable x se puede escribir como $x = w(u, v, 1)^\top$, donde (u, v) son las coordenadas observadas y w es el factor de escala. Denotando con p_i^\top la i -ésima fila de la matriz P , esta ecuación puede ser reescrita del siguiente modo:

$$\begin{aligned} wu &= p_1^\top X \\ wv &= p_2^\top X \\ w &= p_3^\top X \end{aligned}$$

Reemplazando la tercera ecuación en la segunda y la primera es posible despejar el factor de escala w , quedando las siguientes ecuaciones:

$$\begin{aligned} up_3^\top X &= p_1^\top X \\ vp_3^\top X &= p_2^\top X \end{aligned} \quad (3.9)$$

3.6. Reconstrucción por triangulación

Para la vista de la cámara el procedimiento es análogo, denotando p_i^\top la i -ésima fila de la matriz P' :

$$\begin{aligned}u'p_3^\top X &= p_1^\top X \\v'p_3^\top X &= p_2^\top X\end{aligned}\tag{3.10}$$

Se tiene, por lo tanto, un total de 4 ecuaciones lineales, las cuales pueden ser escritas de la forma $AX = 0$, donde A es:

$$A = \begin{pmatrix} up_3^\top - p_1^\top \\ vp_3^\top - p_2^\top \\ u'p_3^\top - p_1^\top \\ v'p_3^\top - p_2^\top \end{pmatrix}$$

Estableciendo $X = (x, y, z, 1)^\top$, el sistema homogéneo, $AX = 0$, es reducido a un sistema lineal no homogéneo con 4 ecuaciones y 3 incógnitas. Para resolver este sistema sobredeterminado se utiliza la metodología de mínimos cuadrados, minimizando la norma euclídea del correspondiente vector residual.

Pueden aparecer dificultades si la verdadera solución de X tiene su última coordenada con valor 0 (un punto en el infinito), en tal caso no es legítimo establecer tal valor en 1 e inestabilidades numéricas podrían ocurrir.

3.6.3. Triangulación lineal iterativa (*Iterative-LS*)

El principal problema del método anterior es que el valor que está siendo minimizado, $\|AX\|$, no tiene ningún sentido geométrico y tampoco corresponde a la función de costo 3.7.

Al multiplicar cada una de las ecuaciones (filas de la matriz A) por algún peso, cambiará la solución del sistema. El objetivo del método iterativo *Iterative-LS* es cambiar los pesos de las ecuaciones adaptativamente, de manera que las ecuaciones ponderadas correspondan al error de medición de las coordenadas en la imagen.

El punto X encontrado utilizando el método *Linear-LS*, no satisfará exactamente la primera de las ecuaciones de 3.9, por el contrario se tendrá un error $\epsilon = up_3^\top X - p_1^\top X$. Sin embargo, lo que realmente se desea minimizar es la diferencia entre la coordenada medida en la imagen del proyector u y la proyección de X , que está dado por $p_1^\top X / p_3^\top X$. En particular queremos disminuir $\epsilon' = \epsilon / p_3^\top X = u - p_1^\top X / p_3^\top X$.

3.7. Correspondencia

Esto significa que si la ecuación se pondera por un factor $1/w$, donde $w = p_3^\top X$, entonces el error resultante será precisamente lo que se desea minimizar. El mismo factor $1/w$ es también válido de aplicar en la segunda ecuación de 3.9. Para la imagen de la cámara el factor correspondiente es $1/w'$, donde $w' = p_3'^\top X$. Los pesos de las ecuaciones no se pueden cambiar directamente de esta manera, ya que los pesos dependen del valor de X , el cual no se conoce hasta que se resuelva el sistema de ecuaciones. Por lo tanto, una manera de resolver este problema es adaptar iterativamente los pesos, partiendo con $w_0 = w'_0 = 1$, para luego resolver el sistema de ecuaciones y encontrar la solución X_0 (esta es la misma solución que se encontraría utilizando el procedimiento de *Linear-LS*).

El proceso se repite varias veces, en la i -ésima iteración se multiplican las ecuaciones de la imagen del proyector 3.9 por $1/w_i$, donde $w_i = p_3^\top X_{i-1}$, y se multiplican las ecuaciones de la imagen de la cámara por $1/w'_i$, donde $w'_i = p_3'^\top X_{i-1}$, donde X_{i-1} es la solución encontrada en la iteración anterior. Después de unas cuantas iteraciones el proceso convergerá, teniendo $X_i = X_{i-1}$.

La ventaja de este método sobre otros métodos de minimización iterativos por mínimos cuadrados, como la iteración de Levenberg-Marquardt (LMA) [108], es que es más simple de programar y el criterio de detención de la iteración (convergencia) es simple. La desventaja de este método es que bajo ciertas circunstancias puede no converger.

3.7. Correspondencia

3.7.1. Generación del *codeword*

Para poder realizar la correspondencia entre los puntos proyectados por el proyector y los puntos encontrados en la cámara, es necesario codificar de manera única cada pixel proyectado por el proyector. La codificación única de cada pixel se denomina el *codeword*. En el sistema propuesto por Posdamer y Altschuler [109] una secuencia de patrones que solo contiene franjas en blanco y negro es proyectada sobre un objeto. El número de franjas se incrementa en dos en cada patrón, siguiendo una estrategia de lo grueso a lo fino. La cantidad máxima de patrones proyectada, y por consiguiente el largo máximo del *codeword* es $\lceil \log_2 w \rceil + \lceil \log_2 h \rceil$ bits, donde w representa la resolución del proyector a lo ancho y h la resolución a lo alto.

Cada pixel proyectado en el patrón tiene una secuencia única de 0s y 1s, donde

3.7. Correspondencia

el 0 representa la ausencia de luz y el 1 representa la presencia de luz. Por lo tanto, el *codeword* se obtiene cuando la secuencia ha sido completada. La codificación se denomina temporal dado que los bits del *codeword* están multiplexados en el tiempo. Este esquema puede lograr una alta precisión en las mediciones [39], debido principalmente a dos factores: el codeword es simple, se requieren solo dos elementos para codificarlo (luz y no luz), con lo que es fácilmente distinguible, por otra parte, al seguir una estrategia de lo grueso a lo fino, la posición de cada pixel se codifica de manera más precisa mientras más patrones sean proyectados.

Inokuchi et al. [110] mejoraron este esquema utilizando el código Gray (nombrados así en honor del investigador Frank Gray) en vez de utilizar código binario. La ventaja del código Gray es que es más robusto al ruido. Un ejemplo de la codificación de Gray en 3 bits se muestra en la tabla 3.1. Es posible convertir un código binario en un código de Gray y viceversa de manera simple mediante el algoritmo 3.2.

| Decimal | Binario | Gray |
|---------|---------|------|
| 0 | 000 | 000 |
| 1 | 001 | 001 |
| 2 | 010 | 011 |
| 3 | 011 | 010 |
| 4 | 100 | 110 |
| 5 | 101 | 111 |
| 6 | 110 | 101 |
| 7 | 111 | 100 |

Cuadro 3.1: Numeración en código binario y numeración en código de Gray

Sato et al [111] propusieron un método complementario para la segmentación de las franjas, utilizando la proyección de cada patrón en positivo y negativo (con los colores invertidos). Luego, comparando la intensidad de luz en ambas representaciones es posible obtener una mejor segmentación de las franjas proyectadas. Utilizando este método se incrementa la cantidad de patrones proyectados a $2\lceil \log_2 w \rceil + 2\lceil \log_2 h \rceil + 2$. Las dos imágenes adicionales corresponden a la proyección de una imagen con todos los píxeles en blanco y otra con todos los píxeles en negro, para poder calibrar la reflectancia del objeto.

```

function BINARIOGRAY(B)
  G[1] ← B[1]
  n ← length(B)
  for i := 2 → n do
    G[i] ← B[i - 1]xorB[i]
  end for
  return G
end function

```

```

function GRAYBINARIO(G)
  B[1] ← G[1]
  n ← length(G)
  for i := 2 → n do
    B[i] ← B[i - 1]xorG[i]
  end for
  return B
end function

```

Cuadro 3.2: A la izquierda función para convertir un código binario a Gray y a la derecha función para convertir un código Gray a binario

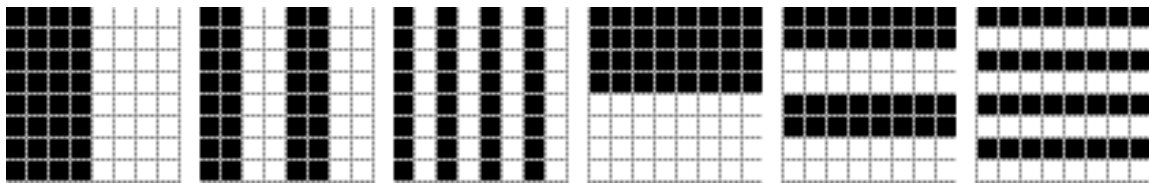


Figura 3.10: Secuencia de proyecciones utilizando código binario en un proyector de 8x8 píxeles

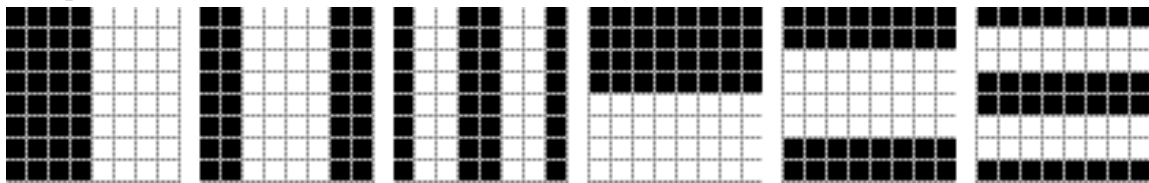


Figura 3.11: Secuencia de proyecciones utilizando código de gray en un proyector de 8x8 píxeles

3.7.2. Recuperación del *codeword*

Para cada patrón proyectado sobre el objeto a digitalizar se obtiene una fotografía. La fotografía muestra cómo se distorsiona el patrón proyectado al ser reflejado por el objeto desde el punto de vista de la cámara. Una vez proyectada la serie de patrones, es posible determinar el *codeword* de cada pixel, no obstante, el proceso puede presentar dificultades, debido a que muchas veces no es posible determinar si el pixel corresponde a un 1 o un 0. Para resolver esta problemática se utiliza la técnica propuesta por Sato et al [111], si la intensidad de luz para un pixel en particular es la misma al proyectar tanto el patrón en positivo como en negativo, significa que estamos en presencia de un borde. Los bordes no son ni 0s ni 1s, por lo que son

3.7. Correspondencia

pixeles cuya identificación es imposible y se descartan. Un ejemplo de esta técnica se muestra en la figura 3.12, la zonas de intersección de ambas curvas representan los bordes del patrón.

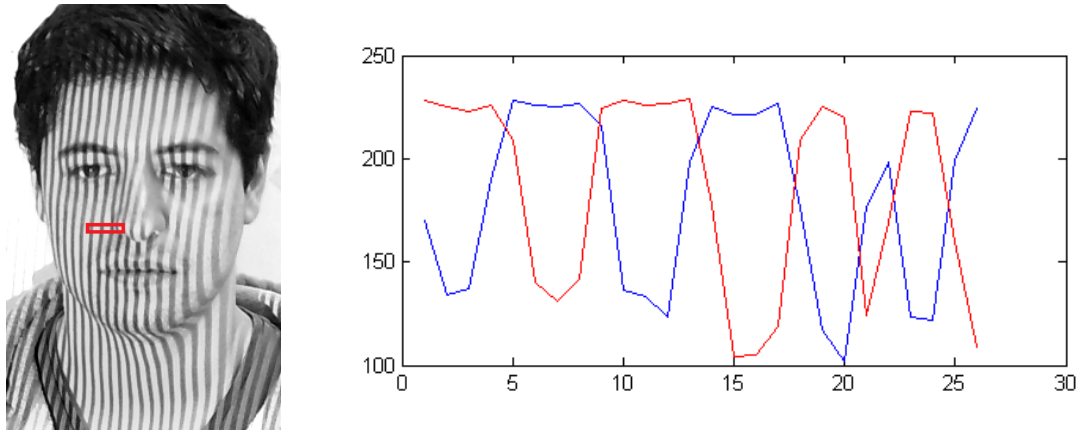


Figura 3.12: A la izquierda, la imagen de un patrón proyectado sobre el rostro de una persona. El recuadro rojo indica la sección que se desea analizar. A la derecha, en azul se muestra la intensidad de luz de cada pixel y en rojo se muestra la intensidad de luz de cada pixel obtenida al proyectar el patrón en negativo

3.7.3. Correspondencia de puntos

Una vez determinado el *codeword* de cada pixel recuperado por la cámara, es posible realizar la correspondencia entre los puntos proyectados y los puntos recuperados por la cámara. Al tener dos puntos correspondientes, mediante triangulación, es posible determinar la posición en el espacio 3D de ese punto. El proceso se repite para todos los pares de puntos correspondientes encontrados, con el fin de construir la nube de puntos para esa vista.

Si coloreamos virtualmente cada columna de pixeles del proyector con un color distinto, podemos determinar visualmente el desplazamiento de las columnas proyectadas al caer sobre el objeto, del mismo modo, podemos colorear virtualmente cada fila de pixeles del proyector con un color distinto. El resultado de esta prueba nos permite apreciar cómo el objeto desplaza los pixeles proyectados desde el punto de vista de la cámara. Un ejemplo de este proceso se puede apreciar en la figura 3.13.

3.8. Registro de vistas parciales

Tal como se aprecia en la figura 3.13, no solamente se obtienen pares de puntos correspondientes en el objeto que se desea digitalizar sino en todos los objetos presentes en la escena, tales como la mesa en la cual está puesto el objeto. Como la mesa no es de interés para este trabajo, se eliminó manualmente de la nube de puntos.

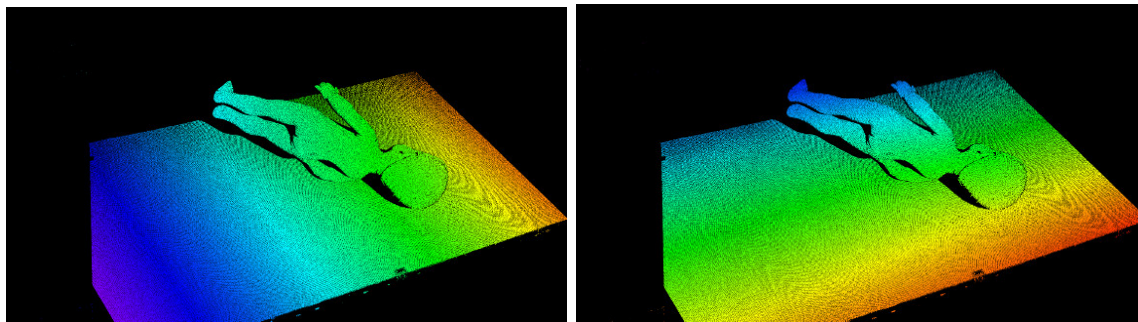


Figura 3.13: Desplazamiento horizontal de las columnas de pixeles proyectadas y desplazamiento vertical de las filas de pixeles proyectadas al caer sobre el objeto a digitalizar.

3.8. Registro de vistas parciales

De cada digitalización se obtiene una nube puntos parcial, correspondiente a la parte visible del objeto digitalizado. Para reconstruir el objeto completo, es necesario unir o *registrar* varias nubes de puntos parciales. El objetivo de este proceso es encontrar las posiciones relativas y orientaciones de cada una de las nubes parciales en un sistema global de coordenadas, de tal modo que las secciones superpuestas entre distintas nubes de puntos calcen perfectamente. Un ejemplo del proceso de registro entre dos nubes de puntos puede apreciarse en la figura 3.14, donde finalmente se obtiene una sola nube de puntos que contiene a ambas.

Para resolver esta problemática se utilizó una variante del algoritmo *Iterative closest point (ICP)* [112, 113], el cual, dada una transformación de cuerpo rígido inicial entre dos nubes de puntos, realiza un proceso iterativo minimizando la distancia entre las nubes de puntos mejorando la transformación de cuerpo rígido en cada iteración. La transformación inicial fue establecida manualmente, debido a que bajo ciertas circunstancias el proceso puede terminar convergiendo en un registro

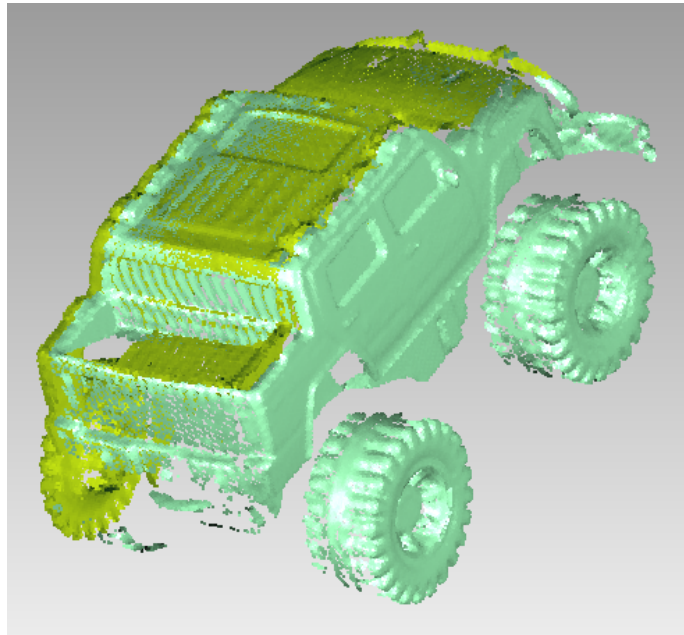


Figura 3.14: Ilustración del proceso de registro entre dos nubes de puntos, cada nube de puntos tiene un color distinto.

inadecuado si se realiza de manera no supervisada.

El proceso de registro se realiza siempre entre dos nubes de puntos. Para poder registrar una cantidad mayor a dos nubes de puntos, se debe realizar siempre entre pares de nubes y luego ir incorporando los resultados al sistema global de coordenadas.

3.9. Superficies a partir de una nube de puntos

Debido a pequeños errores en la determinación de la posición de cada punto en la nube de puntos, se obtienen pequeñas irregularidades al intentar crear una superficie a partir de la nube de puntos. Superficies planas pueden presentar puntos fuera del plano por fracciones de milímetros, los cuales son muy difíciles de eliminar por análisis estadísticos. Con el objetivo de eliminar este tipo de errores en la nube de puntos, los datos fueron suavizados usando el algoritmo *Moving Least Squares* [114].

3.10. Dificultades

Una vez procesada la nube de puntos, se trianguló mediante *Delaunay*. La triangulación de *Delaunay* maximiza el ángulo mínimo de los ángulos internos de cada triángulo, es decir, evita crear triángulos muy delgados. Un ejemplo de la triangulación de una sección de una nube de puntos se puede ver en la figura 3.15.

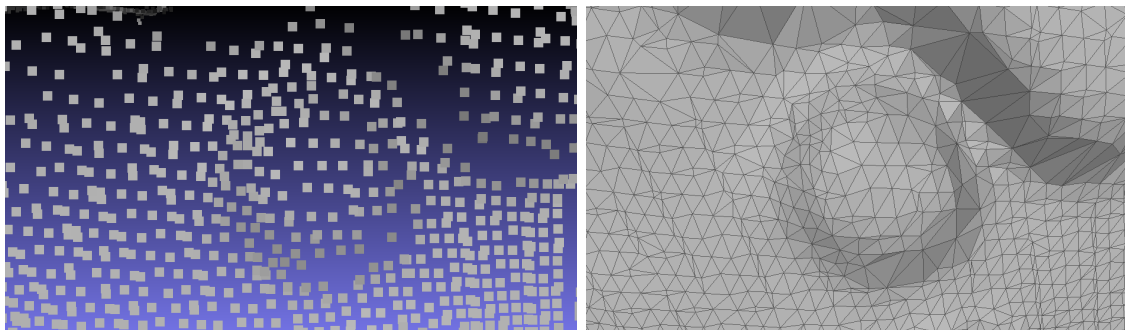


Figura 3.15: A la izquierda, una sección de una nube de puntos sin triangular, a la derecha, la superficie 3D de la misma sección luego de la triangulación.

3.10. Dificultades

La construcción del escáner 3D no estuvo exenta de problemas. El proyector EPSON usado para la construcción del escáner 3D presenta un defecto en la precisión de los pixeles proyectados. El defecto está presente en todos los proyectores EPSON 3LCD y consiste en que los pixeles azules están desplazados con respecto al resto. Esto implica que al proyectar un pixel en color blanco, éste presentará un aura azul en la esquina inferior derecha (exceso de azul) y un aura amarilla en la esquina superior izquierda (falta de azul en la mezcla de colores primarios). El problema principal de este defecto, es que la luz de un pixel llega al pixel aledaño, con lo que afectaría la precisión del escáner. Para solucionar este problema se evitó usar el color azul en los patrones proyectados, utilizando una mezcla de rojo con verde (amarillo) en vez de blanco. En la figura 3.16 se aprecia la proyección de 4 cuadrados de 6x6 pixeles, el primero utilizando sólo luz roja, el segundo utilizando sólo luz verde, el tercero utilizando sólo luz azul y el último utilizando la mezcla de los 3 colores primarios (blanco).

Otra dificultad fue la manipulación de la cámara del tablet Android. El tablet Asus TF300 impide bloquear el ajuste de exposición de luz, el cual es administrado automáticamente por el controlador de la cámara. A raíz de esto, algunas escenas

3.11. Resumen

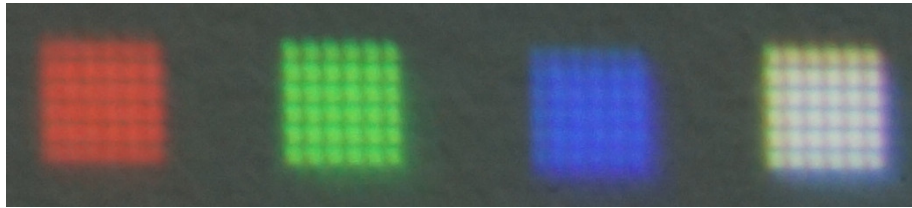


Figura 3.16: Imprecisión en toda la línea de proyectores EPSON 3LCD, consistente en el desplazamiento de los píxeles azules con respecto al resto.

pueden presentar problemas de saturación de luz, haciendo imposible la detección de los patrones proyectados. Para solucionar este problema, se pintaron todos los objetos a escanear con una tonalidad mate medianamente reflectiva, que permite que los patrones proyectados sean fácilmente distinguibles. Un ejemplo del problema antes mencionado se puede ver en la figura 3.17, donde la botella blanca refleja demasiada luz, haciendo muy difícil la detección del patrón. En general, se obtienen buenos resultados al digitalizar un objeto arbitrario. El proceso de pintado se realizó para asegurar un resultado óptimo en la colección creada.



Figura 3.17: Problemas en la captura del patrón proyectado cuando la escena a digitalizar tiene variaciones muy altas en su reflectancia.

3.11. Resumen

La construcción de un escáner 3D óptico basado en patrones de luz estructurada consta de varias fases. Podríamos resumir el trabajo realizado en las siguientes

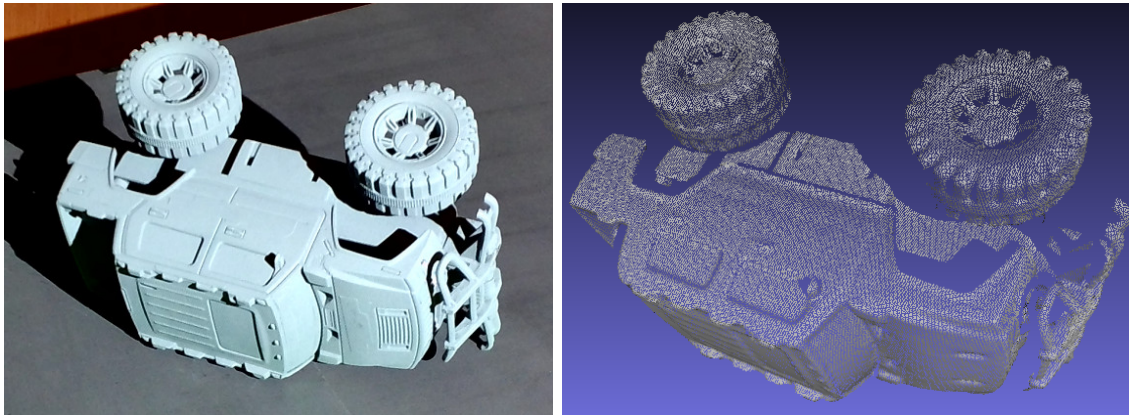


Figura 3.18: A la izquierda, objeto pintado con una tonalidad mate medianamente reflectiva para facilitar el proceso de digitalización. A la derecha, la nube de puntos obtenida.

etapas. Inicialmente se requiere la calibración del sistema para obtener tanto sus parámetros intrínsecos como extrínsecos. Posteriormente, se necesita llevar a cabo un procedimiento para establecer una correspondencia entre cada pixel proyectado y cada pixel recuperado por la cámara. Para establecer esta correspondencia se codifica la posición de cada pixel en un patrón de luz estructurada. Una vez determinada la correspondencia entre los pixeles proyectados y recuperados se puede obtener la posición de cada punto en el espacio 3D mediante triangulación. El conjunto de puntos 3D obtenidos, denominado nube de puntos, se utiliza posteriormente para generar mallas geométricas mediante triangulación de *Delaunay*. Por último, para generar un objeto 3D completo es necesario unir o ‘registrar’ varias vistas parciales, mediante algún algoritmo de alineamiento como ICP.

En particular, para el escáner creado, se proyectan un total de 22 patrones sobre cada objeto (10 franjas de horizontales, 10 de franjas verticales, uno totalmente blanco y otro totalmente negro). El proceso de adquisición toma alrededor de 20 segundos. Posteriormente, el proceso de generación de la nube de puntos se realiza también dentro del tablet y demora cerca de 30 segundos, con lo que en menos de 1 minuto, el tablet es capaz de generar y proyectar los patrones, tomar las fotografías, realizar la correspondencia y la triangulación de más de 1 millón de puntos. Los procesos posteriores de creación de la superficie a partir de la nube de puntos y registro de mallas parciales se realizan fuera del tablet.

Capítulo 4

Algoritmos de matching parcial

4.1. Introducción

Una de las tareas más importantes en el área de *multimedia retrieval* es implementar algoritmos de búsqueda efectivos y eficientes. Los objetos multimedia no pueden ser consultados en el sentido clásico, donde se considera la búsqueda exacta de contenido, ya que es muy poco probable que dos objetos multimedia sean idénticos, a menos que uno sea la copia exacta del otro. El problema, por ende, es el de encontrar los objetos más similares a un objeto de consulta dado.

Una posible manera de afrontar el problema de búsqueda por similitud es manualmente agregar etiquetas a cada objeto. Tal estrategia no solamente es poco práctica, ya que significaría etiquetar manualmente miles de objetos, sino que depende del criterio de la persona que asigna tal información, pudiendo no incorporar toda la información presente en el objeto o catalogarla de manera inadecuada. Una estrategia mucho más razonable es la llamada *content-based search* o búsqueda basada en el contenido. En esta metodología, es la misma información presente en el objeto la usada para determinar la similitud entre objetos. Un ejemplo, donde el objeto de consulta es una malla 3D parcial, se muestra en la figura 4.1.

Los objetos 3D representan información compleja y no existe una teoría formal para describir la forma de un objeto 3D, por lo que existe una gran cantidad de metodologías para determinar similitud entre objetos 3D. Una metodología bastante utilizada es extraer información numérica que represente la forma de un objeto 3D. La idea de extraer información numérica a partir de ciertos aspectos geométricos de un objeto 3D permite la comparación entre objetos por parte de un computador de



Figura 4.1: Ilustración del proceso de búsqueda por similitud utilizando una malla parcial como consulta. Arriba se muestra la malla parcial y abajo la lista resultados ordenados en base a alguna medida de similitud.

manera simple. Las similitudes determinadas de esta forma estarán sujetas a los modelos de similitud empleados y podrían no reflejar adecuadamente la similitud entre objetos bajo distintas circunstancias.

Hay dos aspectos fundamentales a considerar en la evaluación de un algoritmo de búsqueda: la efectividad (qué tan bueno es encontrando objetos similares) y la eficiencia (la cantidad de cómputo necesario para poder realizar tal labor). Un sistema se considera efectivo si dado un objeto de consulta, obtiene los objetos más relevantes de una colección entre los primeros resultados y los objetos menos relevantes al final del ranking. El sistema debe determinar, de alguna manera, cuáles son efectivamente los objetos similares y cuáles no. No obstante, cabe preguntarse cuál es el significado de la similitud. En el ejemplo de la figura 4.1 se aprecia que en el resultado de la consulta se encuentran objetos que, a pesar de tener una similitud global, presentan distintas formas. Aunque existan posibles diferencias geométricas entre los objetos, se pueden considerar similares porque pertenecen a cierta **clase semántica** (los objetos resultantes son todos metralletas).

4.2. Colecciones

Para mejorar el desempeño de los algoritmos de búsqueda en distintas aplicaciones, es importante evaluar y comparar las metodologías existentes con las nuevas metodologías que vayan apareciendo. Dado el aumento de dispositivos que obtienen información 3D disponibles en el mercado, existe una gran cantidad de información 3D dispersa en la web. También existen colecciones estructuradas de objetos 3D denominadas colecciones de *benchmark*, tales como la *Princeton Shape Benchmark* [36], que consta con cerca de 1800 objetos recolectados de Internet que representan objetos de la vida real, tales como vehículos, construcciones, animales o plantas; la *Konstanz 3D database* [28]; el *TOSCA dataset* [37]; y las bases de datos del *Shape Retrieval Contest (SHREC)*. Estas colecciones de referencia contienen objetos 3D agrupados en clases semánticas y permiten la evaluación de distintos algoritmos frente al mismo conjunto de datos. Todas las colecciones nombradas anteriormente solo contienen objeto completos, por lo que no sirven para evaluar algoritmos de matching parcial.

La colección *Large-Scale Partial Shape Retrieval Using Simulated Range Images* [14] es una de las colecciones de *benchmark* más grandes para evaluar algoritmos de matching parcial. La colección contiene 7200 vistas parciales simuladas con distintos niveles de complejidad. Esta colección representó un avance considerable en la evaluación de algoritmos de matching parcial en comparación con las colecciones anteriores. Cabe destacar que los objetos presentes en esta colección son todos artificiales y no provienen de un escáner 3D real. En la figura 4.2 se muestra el proceso de generación de vistas parciales.

4.3. Evaluación del algoritmos de búsqueda

Para evaluar la calidad de los algoritmos de búsqueda, la comunidad de *information retrieval* ha definido y propuesto varias métricas [3]. Dos métricas bastante conocidas para medir la eficacia son *precision* y *recall* [115]. *Precision* es la razón entre la cantidad de objetos relevantes recuperados frente al total de objetos recuperados, mientras que *recall* es la razón entre la cantidad de objetos relevantes recuperados frente al total de objetos relevantes en la colección. Se pueden obtener valores de *precision* para distintos niveles de *recall* para producir gráficos de *precision versus recall*.

Otra métrica ampliamente usada es *R-precision* [115], la cual está definida como el *precision* para obtener N objetos, donde N es el número de objetos relevantes

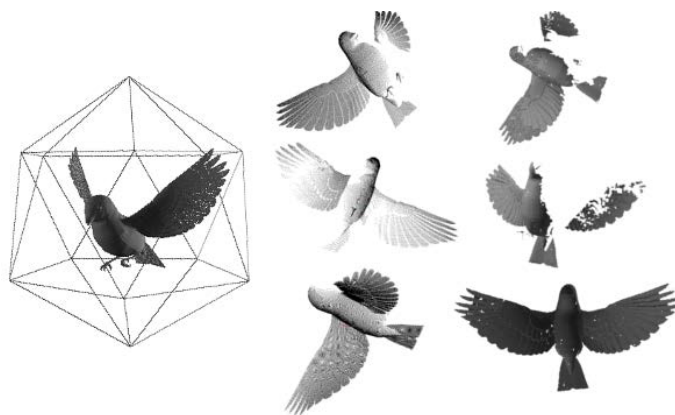


Figura 4.2: Process to obtain the dataset. SHREC'13 Track: Large-Scale Partial Shape Retrieval Using Simulated Range Images [14]

en la base de datos. *R-precision* otorga un sólo número para medir la calidad del algoritmo. Esta medida de eficacia es similar al *bull eye percentage* [27], definido como el *recall* al recuperar $2N$ objetos de la base de datos. Otra medida bastante utilizada dentro de la comunidad es *Mean Average Precision (MAP)*, que consiste en obtener el *Average Precision* promedio para una serie de consultas, donde el *Average Precision* es el promedio de los valores de *precision* para todos los objetos relevantes recuperados. Para poder realizar las mediciones anteriormente nombradas es necesario contar con una colección de referencia [3]. Las mediciones dependerán de la colección de referencia utilizada. El mismo algoritmo puede tener distintas métricas al ser evaluado en distintas colecciones de referencia. Tiene gran importancia contar con una colección de referencia robusta para obtener mediciones estables.

4.4. *Partial Shape Retrieval*

En el estado del arte existe una serie de métodos para el problema del *Partial Shape Retrieval* (recuperar una lista de objetos dado un objeto parcial), tales como:

- *Range scan-based 3D model retrieval by incorporating 2D-3D alignment*, Li et al. [116, 117]
- *Range scan-based 3D model retrieval by viewpoint entropy-based adaptive view clustering*, Li et al [118]
- *Partial shape retrieval using data-aware partitioning*, Sipiran et al. [119]

- *Partial shape retrieval using Bag of Features*, Sipiran et al. [2]
- *Partial shape retrieval with spin images and signature quadratic form distance*, Sipiran and Bustos [14]

4.4.1. *Range scan-based 3D model retrieval by incorporating 2D–3D alignment*

Este algoritmo es un versión modificada del algoritmo de recuperación 3D basado en *sketch* propuesto por Li et al. [116]. El algoritmo contempla un preprocesamiento (cálculo del *Shape Context* y cálculo del *View Context*) y una recuperación a tiempo real, la cual contiene dos pasos consecutivos: *Scan-Model alignment* y *Scan-Model matching*. El proceso puede observarse en la figura 4.3.

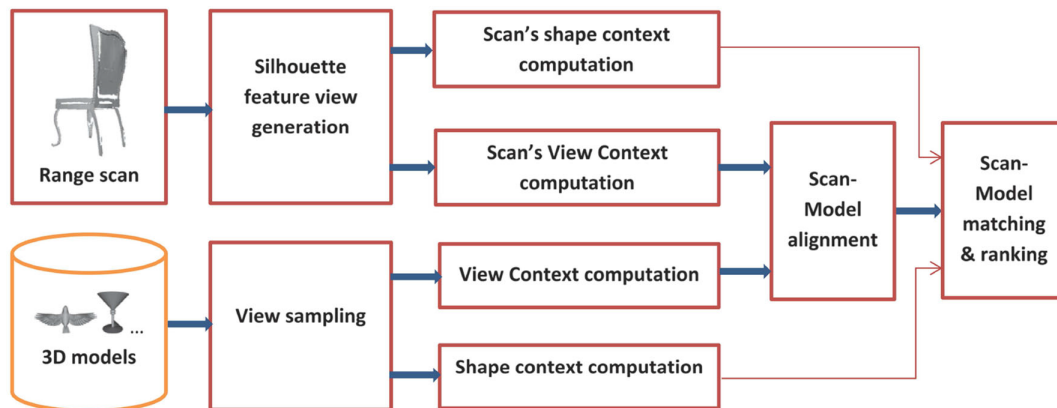


Figura 4.3: Flujo del algoritmo de recuperación *Range scan-based 3D model retrieval by incorporating 2D–3D alignment*, cortesía de [2]

En primera instancia el algoritmo precalcula los descriptores *View Context* [120] y los descriptores *Shape Context* [121] de un conjunto denso de 81 vistas generadas (*Sample Views*) para cada modelo en la colección 3D (proceso denominado *View Sampling*).

Para el objeto de consulta, primero se genera su *Silhouette Feature View*, proceso que puede observarse en la figura 4.4, y posteriormente se calcula su *View Context* y su descriptor *Shape Context*. Basados en el *View Context* del *Silhouette Feature View* y en el conjunto de 81 vistas generadas para cada modelo 3D, se realiza un

alineamiento *Scan-Model*, preseleccionando varias vistas candidatas del modelo, y finalmente se realiza un *matching Scan-Model* basado en los descriptores *Shape Context* entre el *Silhouette Feature View* y la vista candidata (*Sample View*) del modelo 3D.

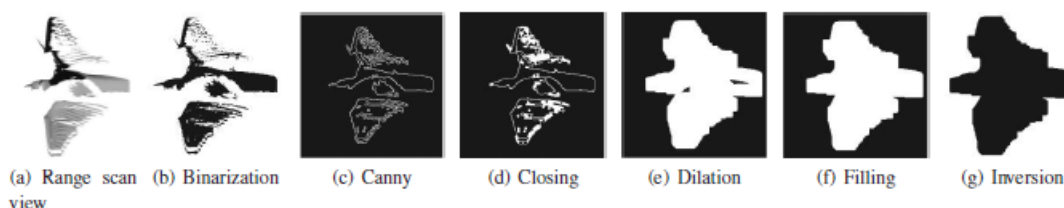
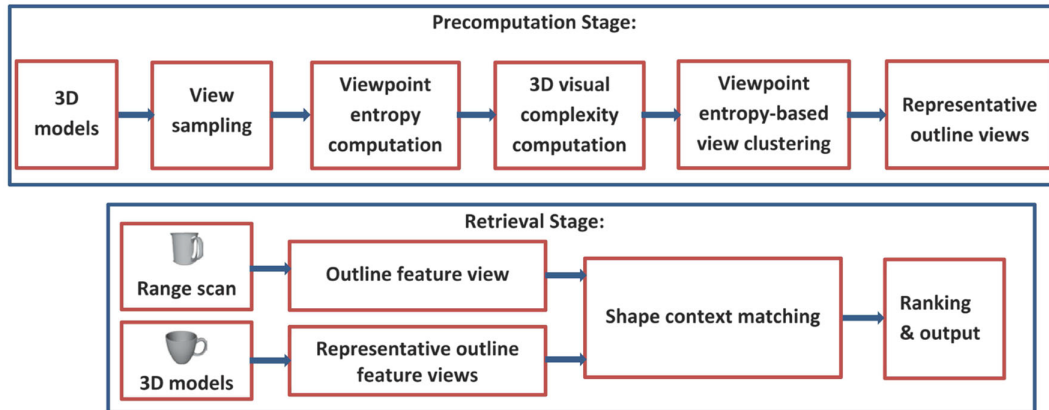


Figura 4.4: Proceso de generación del *Silhouette Feature View* de una vista parcial

4.4.2. *Range scan-based 3D model retrieval by viewpoint entropy-based adaptive view clustering*

Es una modificación de la técnica anterior, donde se seleccionan las vistas más representativas del proceso de *View Sampling*.

Por lo general, la cantidad de *Sample Views* necesarios para representar un modelo 3D es distinta para cada modelo, donde las formas más complejas requieren una mayor cantidad de *Sample Views* para ser descritas. Se propone un modelo de complejidad visual utilizando, en primera instancia, la distribución de entropía (basado en el método de Takahashi [122]) de 81 *Sample Views* de un modelo 3D. Posteriormente, ésta es utilizada para decidir de manera adaptativa el número de vistas representativas del modelo 3D, para realizar un *Fuzzy C-means View Clustering* de sus 81 *Sample Views*. Finalmente, durante la recuperación a tiempo real se realiza un *matching* basado en *Shape Context* [123] más preciso entre el *sketch* de consulta y las vistas representativas de cada modelo de la colección. La figura 4.5 muestra las dos etapas de este algoritmo de recuperación, el preprocesamiento y la recuperación a tiempo real.


 Figura 4.5: Flujo del algoritmo de recuperación *SBR-VC*, cortesía de [2]

4.4.3. *Partial shape retrieval using data-aware partitioning*

Método propuesto por Sipiran et al. [2] para formas genéricas basado en la detección de puntos de interés. La idea detrás de este método es representar una forma mediante un descriptor global $s \in \mathbb{R}^n$, y un conjunto de descriptores de secciones del objeto (particiones) $P = p^1, p^2, \dots \in \mathbb{R}^n$. Es decir, para dos objetos 3D O y Q , su representación sería la siguiente:

$$\begin{aligned}
 S_O &= \{(s_O, P_O) \mid s_O \in \mathbb{R}^n \text{ y } P_O = \{p_O^i\}_{i=1}^m, p_O^i \in \mathbb{R}^n\} \\
 S_Q &= \{(s_Q, P_Q) \mid s_Q \in \mathbb{R}^n \text{ y } P_Q = \{p_Q^i\}_{i=1}^k, p_Q^i \in \mathbb{R}^n\}
 \end{aligned}$$

donde O tiene m particiones y Q tiene k particiones. Las particiones son obtenidas al agrupar puntos de interés de Harris 3D [124] en el espacio Euclideo, usando un algoritmo de *clustering* adaptativo [119]. Posteriormente se calcula la esfera envolvente mínima de los puntos de interés para definir cada partición. Por último, las secciones de la malla contenidas en la esfera son consideradas parte de la partición. Las representaciones S_O y S_Q contienen tanto la descripción de la forma global como la descripción de las particiones. El descriptor DESIRE [125] es utilizado para describir tanto la forma global como las particiones.

Para determinar la similitud entre dos objetos 3D se define una función de distancia $d(S_O, S_Q)$ que considera la disimilitud de dos objetos mediante sus representaciones. Se utiliza una combinación lineal entre la disimilitud global-global y las disimilitud de las particiones del siguiente modo:

$$d(S_O, S_Q) = \mu \|s_O - s_Q\|_2 + (1 - \mu) d_{part}(P_O, P_Q)$$

donde $0 \leq \mu \leq 1$ es el factor de ponderación entre los términos involucrados y la función $d_{part}(P_O, P_Q)$ mide la distancia entre dos conjuntos de particiones. Para construir tal función se propone una función auxiliar f :

$$f(x) = \sum_{i,j} \|p_O^i - p_Q^j\|_2 \cdot x(i, j)$$

donde $x(\cdot)$ es un indicador booleano que indica si hay match entre p_O^i y p_Q^j . El objetivo es encontrar el óptimo x^* que minimice $f(x)$. Esto se puede entender como el problema de optimización de encontrar el minimizador $x^* = \operatorname{argmin}_x f(x)$, sujeto a $\sum_i x(i, j') = 1$ y $\sum_j x(i', j) = 1 \forall i, j$. Finalmente, la función de distancia entre particiones queda definida como:

$$d_{part}(P_O, P_Q) = \frac{f(x^*)}{\min(|P_O|, |P_Q|)}$$

donde la normalización permite trabajar con particiones de distinto tamaño.

4.4.4. Partial shape retrieval using Bag of Features

La metodología de *Bag of Features* utiliza la idea de la organización de documentos en base a un diccionario de palabras. El diccionario contiene características únicas y permite caracterizar a cada documento como una distribución de éstas. La metodología propuesta por Sipiran et al. [2] para aplicar *Bag of Features* al problema de recuperación parcial de objetos 3D contiene dos etapas. La primera, consiste en el cálculo de un diccionario de características usando los descriptores locales (*Spin Images*) de los objetos 3D de la colección (los objetos completos) y la segunda, que consiste en usar el diccionario para calcular los descriptores finales de cada uno de los objetos de la colección y de los objetos de consulta. Denominando D al conjunto de descriptores en \mathbb{R}^n y k a la cantidad de *clusters* que se desea encontrar, se procede a utilizar el algoritmo *k-means* sobre el conjunto D .

El conjunto M de los centroides representativos del diccionario se combina con el conjunto P de los descriptores locales de un objeto, para calcular un solo descriptor por objeto. Posteriormente se calcula la distribución de características para cada descriptor en P , tal que, $\theta(p_j) = (\theta_1(p_j), \dots, \theta_k(p_j))^T$, donde:

$$\theta_i(p_j) = c(p_j) \exp\left(\frac{-\|p_j - m_i\|_2}{2\sigma^2}\right)$$

$c(p_j)$ es una constante, tal que $\|\theta(p_j)\|_2 = 1$, m_i es el centroide del *cluster* C_i y σ es una constante. Cada *bin* del vector $\theta_i(p_j)$ representa la probabilidad de que el descriptor p_j pertenezca a un *cluster*. Usando la versión suave de cuantización, la cual ha mostrado ser más efectiva en el área de recuperación de formas [126] (a diferencia de la cuantización fuerte, donde los *bins* representan descriptores cercanos a los *clusters*), se tiene el descriptor final de un objeto, representado por el conjunto de descriptores P como:

$$f(P) = \sum_{p_j \in P} \theta(p_j)$$

La disimilitud entre dos objetos puede ser medida mediante distancia L_1 entre los vectores cuantizados. No obstante, en el contexto de recuperación parcial, la divergencia *Kullback-Leibler* ha mostrado ser efectiva como medida de distancia [127].

4.4.5. Partial shape retrieval with spin images and signature quadratic form distance

La *signature quadratic form distance* [4] (SQFD de ahora en adelante) es una medida de distancia flexible que ha mostrado ser efectiva en el dominio de recuperación de imágenes. El método consiste en el cálculo de una representación intermedia de cada objeto 3D usando un conjunto de descriptores locales (*Spin Images*), los cuales son calculados en sectores representativos de la superficie 3D. Este algoritmo es una versión modificada del método evaluado en [128].

En primera instancia, se calculan puntos de interés usando Harris 3D [124]. Estos puntos de interés son usados como puntos de referencia sobre los cuales se calcularán descriptores locales en su vecindad. Posteriormente, se aplica un algoritmo de *clustering* local [129], el cual se describe en el algoritmo 1. El algoritmo utiliza dos umbrales para definir las propiedades de *intra-cluster* e *inter-cluster* en el espacio, sin estipular la cantidad de *clusters* que se desea encontrar, ya que esta cantidad dependerá de la distribución de descriptores en el *feature space*. La figura 4.6 muestra la diferencia entre un *clustering* fijo y un *clustering* adaptativo.

De un modo más formal, un objeto P es representado por un conjunto de descriptores $F = f_1, f_2, \dots, f_k \in FS$, donde FS es el *feature space* de dimensión arbitraria y k es la cantidad de descriptores. Suponiendo la existencia de un *clustering* local C_1, C_2, \dots, C_n para F , se define el *feature signature* como el conjunto de tuplas $FS \times \mathbb{R}^+$ del siguiente modo:

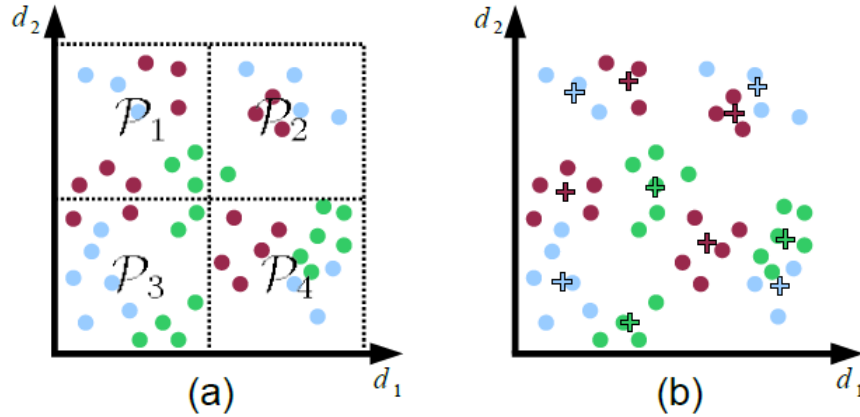


Figura 4.6: *Clustering* fijo vs *clustering* adaptativo. Signature quadratic form distance, Beecks et al. 2010 [15]

$$S^P = \{(c_i^P, w_i^P), i = 1, \dots, n\}$$

donde $c_i^P = \frac{\sum_{f \in C_i} f}{|C_i|}$ representa el centroide del i -ésimo *cluster* y $w_i^P = \frac{|C_i|}{k}$ representa su respectivo factor de ponderación. Cabe destacar que el tamaño de S^P depende de la partición local y puede variar de objeto en objeto.

Dados los *feature signatures* de dos objetos P y Q , $S^P = \{(c_i^P, w_i^P), i = 1, \dots, n\}$ y $S^Q = \{(c_i^Q, w_i^Q), i = 1, \dots, m\}$ y una función de similitud $f_S : FS \times FS \rightarrow \mathbb{R}$, se define la *Signature Quadratic Form Distance* entre S^P y S^Q como:

$$SQFD_{f_S}(S^P, S^Q) = \sqrt{(w^P | - w^Q) A_{f_S} (w^P | - w^Q)^T} \quad (4.1)$$

donde la notación $|.$ representa la concatenación de vectores de ponderación y $A_{f_S} \in \mathbb{R}^{(n+m) \times (n+m)}$ es la matriz de similitud, donde cada valor a_{ij} de la matriz está definido como:

$$a_{ij} = \begin{cases} f_S(c_i^P, c_j^P) & \text{si } i \leq n \text{ y } j \leq m \\ f_S(c_{i-n}^Q, c_j^P) & \text{si } i > n \text{ y } j \leq m \\ f_S(c_i^P, c_{j-m}^Q) & \text{si } i \leq n \text{ y } j > m \\ f_S(c_{i-n}^Q, c_{j-m}^Q) & \text{si } i > n \text{ y } j > m \end{cases}$$

donde f_S es una función de similitud. La figura 4.7 muestra la estructura de esta matriz de similitud. Se puede apreciar que la matriz de similitud contiene tanto

inter-dependencias (comparaciones entre centroides de distintos *feature signatures*) como intra-dependencias (comparaciones entre centroides de la misma *feature signature*). Los factores de ponderación indican la importancia de cada componente del *feature signature*.

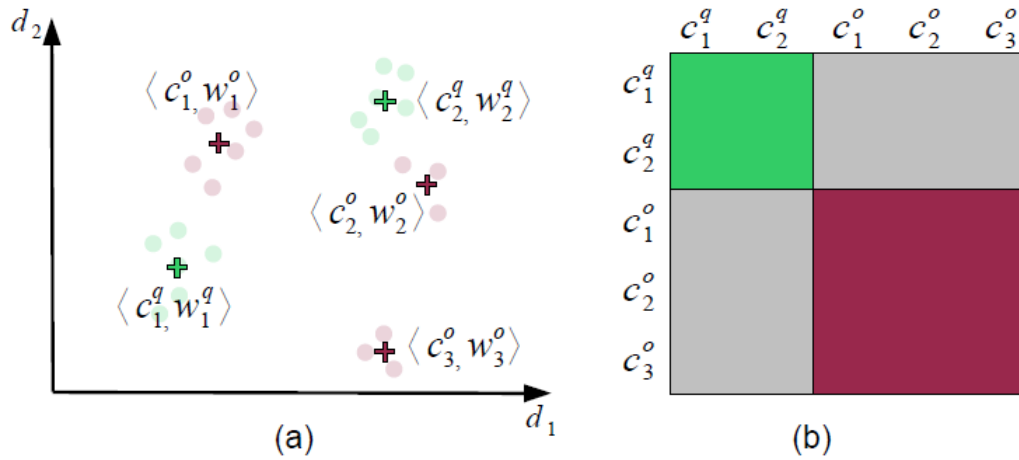


Figura 4.7: (a) Dos *feature signatures* con sus respectivos centroides y pesos. (b) la estructura de la matriz de similitud. Beecks et al. 2010 [15]

Beecks et al. [15] propone usar una de las siguientes funciones de similitud, basados en la distancia Euclidiana $d = L_2$:

- Minus: $f_-(c_i, c_j) = -d(c_i, c_j)$
- Gaussian: $f_g(c_i, c_j) = \exp(-\alpha d^2(c_i, c_j))$
- Heuristic: $f_h(c_i, c_j) = \frac{1}{\alpha + d(c_i, c_j)}$

Algorithm 1 Algoritmo de clustering local adaptativo

Require:

Conjunto de vectores P
Cantidad mínima de vectores por *cluster* N
Distancia *intra-cluster* R
Distancia *inter-cluster* T
Cantidad de iteraciones IT
Conjunto de *clusters* C

```
1:  $C \leftarrow \emptyset$ 
2: for  $j := 1 \rightarrow IT$  do
3:   for each  $p \in P$  do
4:     if  $C == \emptyset$  then
5:        $d = 2T$ 
6:     else
7:        $C^* = \arg \text{mindist}(p, C_i)_{C_i \in C}$ 
8:        $d = \text{distancia entre } p \text{ y } C^*$ 
9:     end if
10:    if  $d > T$  then
11:       $C_{\text{nuevo}} = \{p\}$ 
12:       $C \leftarrow C \cup C_{\text{nuevo}}$ 
13:       $P \leftarrow P - \{p\}$ 
14:    else if  $d \leq R$  then
15:       $C^* \leftarrow C^* \cup \{p\}$ 
16:       $P \leftarrow P - \{p\}$ 
17:    end if
18:  end for
19:  for  $i := 1 \rightarrow |C|$  do
20:    if  $|C^*| \geq N$  then
21:      Actualizar centroide para  $C^*$ 
22:    else
23:       $P \leftarrow P \cup C^*$ 
24:       $C \leftarrow C \setminus C^*$ 
25:    end if
26:  end for
27: end for
28: return  $C$ 
```

Capítulo 5

Evaluación de algoritmos

5.1. Colección 3D

Para evaluar el método de recuperación 3D basado en descriptores locales y SQFD se creó una colección de referencia (*benchmark*) pública, la cual puede ser vista y descargada desde cualquier parte del mundo a través de la url <http://dataset.dcc.uchile.cl/PRISMA/>. La idea detrás de esta colección de referencia, es que todos los objetos 3D presentes en ella provienen de digitalizaciones de objetos reales. Lo cual permite evaluar los algoritmos con una de las colecciones más realista que existe hasta el momento disponible.

Recientemente se creó una colección de objetos obtenidos a través de un dispositivo *Microsoft Kinect* [16]. Tal colección, además de contener información sobre la forma tridimensional de los objetos, contiene información sobre sus colores (RGB-D). No obstante, no contiene superficies (son solo nubes de puntos), los modelos tienen una baja resolución y un error considerable. En la figura 5.1 se observa un ejemplo de dos vista parciales de un objeto de la colección.

La colección consta de 7 clases de similitud. Los objetos dentro de una clase de similitud son lo más similares posibles y representan el mismo significado semántico. Por otra parte, se construyó la colección de modo tal que todas las clases de similitud sean lo más distintas posible entre ellas. Las clases de similitud son las siguientes:

- Machinegun
- Suv

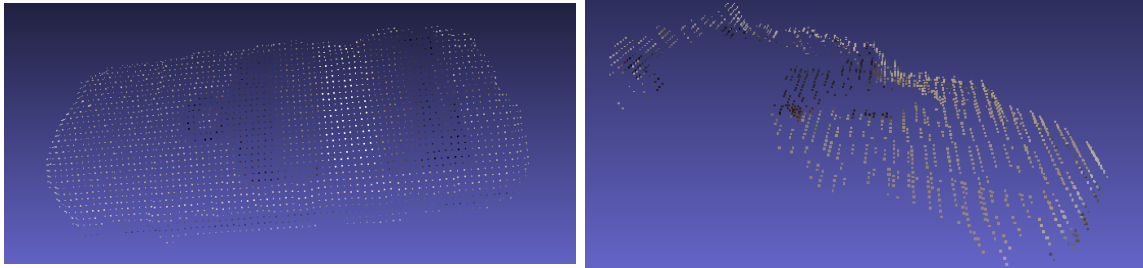


Figura 5.1: Dos nubes parciales obtenidas de la digitalización de un celular con un dispositivo *Microsoft Kinect*. *A large-scale hierarchical multi-view rgb-d object dataset* [16]

- Car
- Doll
- Shoe
- Bipedal Dinosaur
- Quadrapedal Dinosaur

Dentro de estas 7 clases de similitud se encuentran 30 objetos 3D. La sección de anexos A.2 incluye una imagen de cada uno de estos objetos digitalizados. Cada objeto 3D tiene un total de 16 vistas parciales de consulta (mallas parciales), con lo que la colección creada suma 480 vistas parciales de consultas en total. Cada vista parcial contiene información parcial distinta sobre el objeto 3D en cuestión. Un mayor detalle de como fueron creadas las vistas parciales se encuentra en la sección ??.

Cada objeto de la colección creada contiene 16 vistas parciales (mallas parciales), que fueron obtenidas al digitalizar al objeto desde distintos puntos de vista. Lo que se quiere buscar es que cada vista parcial sea única, es decir, que cada vista parcial sea distinta de las demás. Para lograr este objetivo se digitalizó cada objeto, realizando 8 tomas en su parte superior y 8 tomas en su parte inferior. En la figura 5.2 se muestra el proceso de digitalización de las 8 tomas de la parte inferior de un objeto.

El detalle de las 16 tomas realizadas a esta zapatilla se puede observar en la sección de anexos A.1.

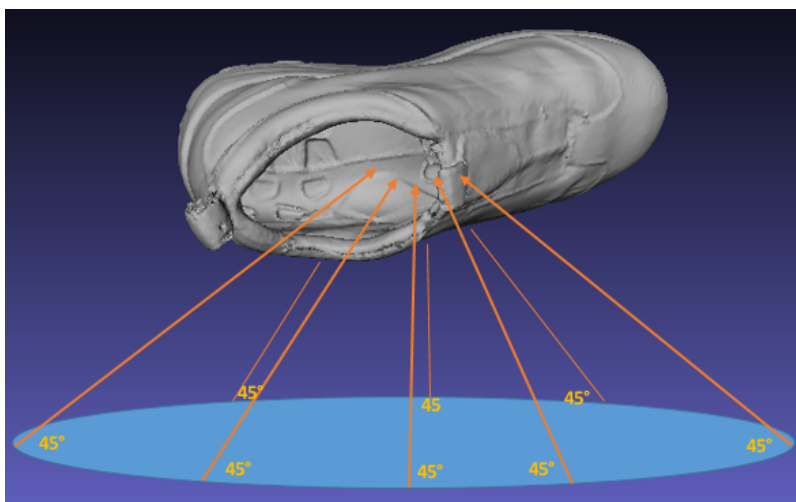


Figura 5.2: Metodología para la captura de vistas parciales. Este mismo proceso se repite realizando las tomas desde arriba hacia abajo.

5.2. Algoritmos evaluados

Una vez creada la colección 3D, se evaluó la efectividad de la metodología de descriptores locales, *clustering* local y distancia *SQFD* (descrita en la sección 4.4.5). Los descriptores locales utilizados fueron *Spin images* [130] y *Fast Point Feature Histogram* [131]. *Spin images* genera descriptores locales (vectores de alta dimensionalidad) que consideran todos los puntos del objeto, desde un sistema de coordenadas parcial definido en cada punto de interés. *Spin images* ha mostrado ser efectivo en reconocer objetos que contienen partes faltantes y oclusiones [17]. *Fast Point Feature Histogram*, por otra parte, genera descriptores locales que describen solamente la geometría local alrededor de un punto de interés.

Cabe hacerse la pregunta sobre cual es la mejor manera de describir un objeto 3D. ¿Basta con describir un objeto 3D por sus puntos más sobresalientes? Un objeto 3D con sectores planos o suaves, tiene como principal característica la ausencia de puntos sobresalientes. ¿Qué características son las que finalmente describen a un objeto?. La figura 5.3 muestra cuales serían los puntos de interés seleccionados con un selector de puntos de interés. En este caso estaríamos describiendo al objeto como un objeto con 4 esquinas, sin embargo, estamos omitiendo una de las informaciones más relevantes del objeto: que el objeto tiene un gran sector plano. ¿Qué es lo que define a una pizarra? ¿Es el hecho de que tenga cuatro esquinas o es el hecho de sea un objeto

5.2. Algoritmos evaluados

con un gran superficie plana?. Para dar solución a estas interrogantes se incluyó un experimento para evaluar la efectividad de los algoritmos utilizando selección densa de puntos de interés (La selección densa de puntos de interés consiste en seleccionar puntos de interés de manera uniforme con un cierto espaciado) versus la selección de puntos de interés en base a puntos sobresalientes.

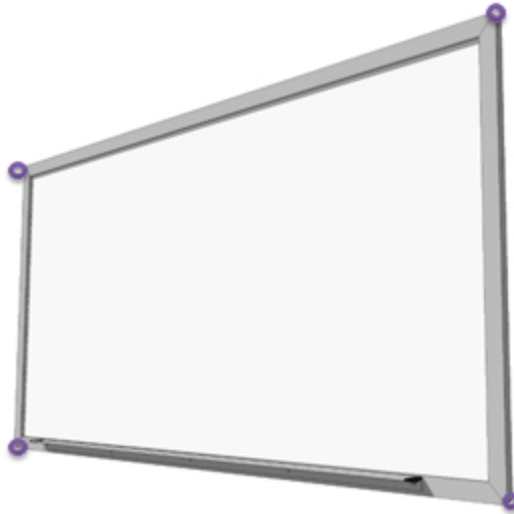


Figura 5.3: Ejemplo de un objeto con sus respectivos puntos de interés, utilizando un detector de puntos de interés.

5.2.1. Spin images

Considerando los puntos de la superficie de un objeto y su vector normal es posible obtener lo que se conoce como "puntos orientados". Un punto orientado define un sistema de coordenadas cilíndricas parcial centrado en el objeto, con el cual se define α como la distancia perpendicular a la línea del vector normal y la coordenada de elevación β como la distancia perpendicular (con signo) al plano tangente definido por el vector normal. La coordenada angular es omitida debido a que no puede ser definida sin ambigüedad en superficies planas.

El *Spin Image* de un punto orientado es creado utilizando un acumulador bidimensional (matriz) de índices α y β . Cada punto del objeto 3D es utilizado para incrementar algún *bin* del acumulador. Se utiliza una interpolación bilinear para

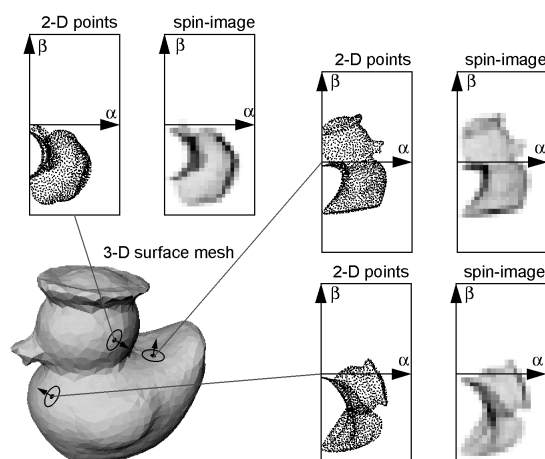


Figura 5.4: *Spin Images* del modelo 3D de un patito de goma. [17]

suavizar la contribución de cada punto en el acumulador. El resultado final del acumulador puede verse como una imagen, donde los sectores más oscuros representan *bins* que contienen muchos puntos proyectados. Un ejemplo de los *Spin Images* generados para puntos arbitrarios de un modelo 3D puede observarse en la figura 5.4.

Los *Spin Images* generados de dos superficies distintas que representen el mismo objeto serán similares debido a que están basados en la misma forma del objeto, aunque no serán exactamente iguales debido a eventuales variaciones en las superficies y el ruido.

5.2.2. Fast point feature histogram

Es una modificación del descriptor *Point Feature Histogram* [132, 133] que mediante algunas optimizaciones y reformulaciones de sus bases teóricas, es capaz de reducir el tiempo de cómputo de manera considerable, mientras mantiene la mayor parte del poder discriminativo de éste.

El objetivo de *Point Feature Histogram* es codificar las propiedades geométricas de la k -vecindad de un punto en un objeto 3D, al generalizar la curvatura promedio alrededor del punto, utilizando histogramas multidimensionales. La representación está basada en la relación de un punto con sus k puntos más cercanos y sus correspondientes vectores normales. De manera simple, el algoritmo intenta capturar de la mejor manera posible las variaciones en la superficie del objeto, tomando en cuenta

todas las interacciones entre las direcciones de las normales estimadas.

5.3. Selección de puntos de interés

Se probaron los descriptores locales anteriores usando dos metodologías distintas de selección de puntos de interés: usando *Harris 3D* [124] y usando selección densa de puntos de interés.

Harris 3D entrega un índice numérico que indica cuánto sobresale un punto con respecto a su vecindad. La selección de puntos de interés mediante *Harris 3D* consiste en la selección del 1 % de los puntos más sobresalientes del objeto. La selección densa de puntos de interés se realiza construyendo una grilla de vóxeles, con tamaños de cada voxel de 6, 8 y 10 mm (3 experimentos distintos) y seleccionando el punto más central que se encuentre dentro de cada voxel (en caso de que no hayan puntos dentro del voxel no se selecciona un punto de interés en ese sector). La selección densa de puntos de interés termina seleccionando entre el 1 % y el 2 % de los puntos del objeto, dependiendo del espaciado del voxel, en promedio.

5.4. Experimentos

Se realizaron los siguientes experimentos:

- Selector de puntos de interés *Harris 3D* y descriptor local *Spin images*
- Selector de puntos de interés *Harris 3D* y descriptor local *Fast Point Feature Histogram*
- Selección densa de puntos de interés y descriptor local *Spin images*
- Selección densa de puntos de interés y descriptor local *Fast Point Feature Histogram*

Se utilizaron los siguientes parámetros para la evaluación:

Selector de puntos de interés *Harris 3D* [124] se utilizó la detección adaptativa de la vecindad y una selección del 1 % de los puntos más representativos.

Spin images Se utilizó un ancho de la imagen de 8.

Clustering Se usó 0.1 como distancia *intra-cluster* y 0.2 como distancia *inter-cluster*. El número mínimo de vectores por *cluster* fue de 10. Estos parámetros se pueden apreciar en el algoritmo definido en la sección 4.4.5. La distancia *intra-cluster* es la distancia máxima que pueden tener los elementos dentro del mismo *cluster*, mientras que la distancia *inter-cluster* es la distancia mínima entre dos *clusters*.

Función de similitud f_S Se utilizó la función *Gaussian*, con $\alpha = 0,1$ como función de similitud para *SQFD*, debido a que con estos valores se obtienen mejores resultados [2].

5.5. Resultados

En esta sección se muestra la comparación de los resultados de cada experimento, al evaluarlos con la colección de 480 mallas parciales de consulta y 30 objetos 3D agrupados en 7 clases de similitud. Se incluyen gráficos de *precision-recall* y el valor de *Mean Average Precision (MAP)*, entre otros, descritos en la sección 4.3, para cada metodología evaluada.

En la tabla 5.1 se observa que en términos de MAP, el método de selección densa de puntos de interés y descriptor local *Fast Point Feature Histogram* (fpfh-u6mm) es el que obtiene mejores resultados, seguido por el método de selección densa de puntos de interés y descriptor local *Spin images* (sp-u6mm), seguidos por los métodos de selector de puntos de interés Harris 3D y descriptor local *Spin images* (sp-harris) y selector de puntos de interés Harris 3D y descriptor local *Fast Point Feature Histogram* (fpfh-harris).

| Método | NN | FT | ST | MAP |
|-------------|---------------|---------------|---------------|---------------|
| sp-harris | 0.1833 | 0.1784 | 0.2812 | 0.3086 |
| sp-u6mm | 0.3187 | 0.3397 | 0.3525 | 0.4154 |
| fpfh-harris | 0.2437 | 0.1894 | 0.1726 | 0.3042 |
| fpfh-u6m | 0.4458 | 0.4057 | 0.4421 | 0.4909 |

Cuadro 5.1: Resultados de la evaluación de distintos métodos con sus mejores parámetros

De estos resultados se puede desprender que el método de selección densa de

5.5. Resultados

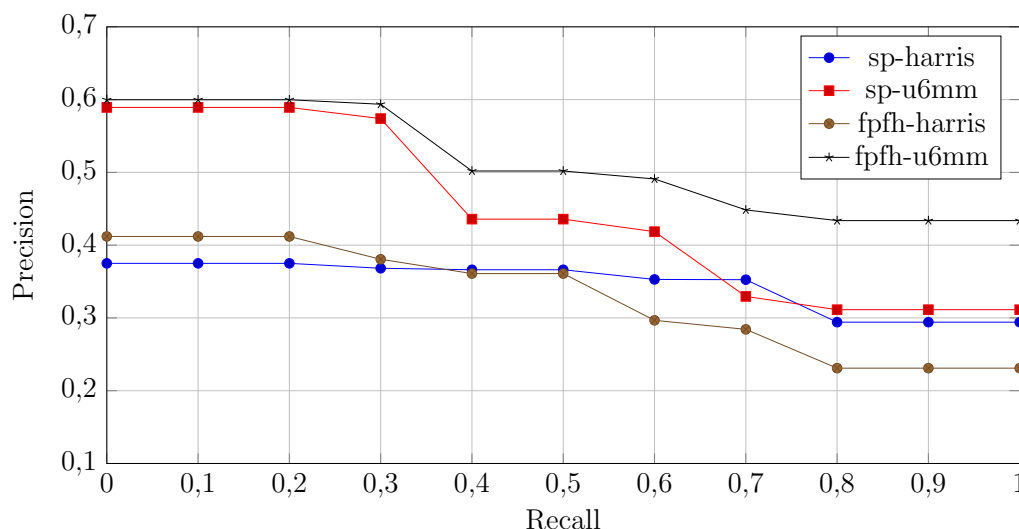


Figura 5.5: Comparación de los distintos métodos.

puntos tiene mejor desempeño que el selector de puntos de interés Harris 3D. No obstante, es importante destacar que el tamaño de esta colección es relativamente mediano, y sus resultados pueden que no sean del todo representativos. Por otra parte, el método de selección densa de puntos de interés es bastante más lento, ya que requiere el cálculo de cerca del doble de descriptores que si se utilizara Harris 3D. Con respecto a los indicadores *Nearest neighbor* (NN), *First tier* (FT) y *Second tier* (ST), la metodología de selección densa de puntos de interés también muestra mejor desempeño. Los resultados del gráfico de *precision-recall* mostrados en la figura 5.5 son consistentes con los resultados encontrados con el indicador MAP. Mostrando mejores desempeños en los experimentos que utilizaron selección densa de puntos de interés.

De la figura 5.5 podemos apreciar que para un *recall* de 0.1 (al recuperar el 10% de los objetos de la clase) se obtiene un nivel de *precision* cercano a 0.6 (el 60% de los objetos recuperados son relevantes) con el mejor método, mientras que para un *recall* de 1.0 (al recuperar el 100% de los objetos de la clase), se obtiene un nivel de *precision* de 0.43 (el 43% de los objetos recuperados son relevantes) con el mejor método.

Por otra parte, se encontró que la selección densa de puntos de interés depende del espaciado en la selección de puntos, obteniendo mejores resultados cuando el

5.5. Resultados

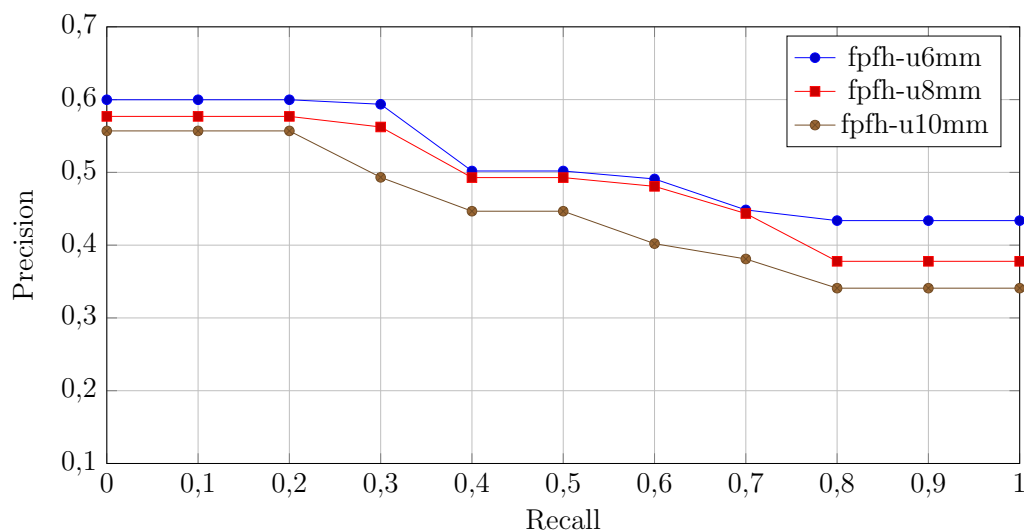


Figura 5.6: Comparación del método de selección densa de puntos de interés y descriptor local *Fast Point Feature Histogram* con niveles de selección de puntos de interés cada 6mm, 8mm y 10mm.

espaciado es menor. Para este experimento se encontró que los mejores resultados se encuentran con un espaciado de 6mm. La figura 5.6 muestra la comparación del método de selección densa de puntos de interés y descriptor local *Fast Point Feature Histogram* con niveles de selección de puntos de interés cada 6mm, 8mm y 10mm.

Capítulo 6

Conclusiones

6.1. Conclusiones

En este trabajo se creó una colección de referencia 3D para evaluar algoritmos de matching parcial utilizando un escáner 3D constituido por un tablet Android y un proyector. La objetos 3D presentes en esta colección permitieron evaluar los algoritmos usando datos reales. Junto con la colección RGB-D de Lai et al. [16], este es uno de los primeros intentos en crear una colección 3D de acceso público que permita evaluar algoritmos de matching parcial usando información real. La información real es información imperfecta, que contiene ruido y que significa una mayor complejidad a la hora de realizar consultas de similitud.

Se encontró que la técnica de descriptores locales, utilizando *clustering* local y distancia *SQFD* [4] es una metodología que permite determinar similitud entre objetos 3D cuando la información es parcial y proveniente de datos reales. Los resultados de los experimentos realizados en este trabajo muestran que el problema es de búsqueda por similitud parcial es un problema desafiante. Si bien los resultados fueron moderados en término de eficacia, y fue posible encontrar algunas mejoras, los resultados indican que este problema está lejos de ser resuelto. El área de búsqueda por similitud en objetos 3D es un área en constante avance y el problema del matching parcial, en particular, sigue siendo un problema abierto debido a su alta complejidad, debido a que la información de las consultas es incompleta.

Si bien se lograron conseguir mejores resultados, en términos de efectividad, al usar selección densa de puntos de interés, tal metodología es poco práctica debido a su baja eficiencia. Lo que se busca de un algoritmo de búsqueda parcial es que

6.2. Trabajo futuro

sea efectivo, es decir, que logre su objetivo de encontrar objetos similares y que sea eficiente, o sea, que cumpla su objetivo realizando la menor cantidad de operaciones posibles. En este caso se logró mejorar la efectividad a costa de una disminución de la eficiencia, lo cual impide que tal metodología sea usada en aplicaciones a tiempo real debido a la lentitud del del cálculo de descriptores (dependiendo de la cantidad de descriptores calculados, el proceso puede tardar desde unos segundos, hasta varios minutos por consulta, en un computador con procesador *Intel Core i7 930*, con 8M de cache, 2.80 GHz y 8Gb de RAM).

Un objetivo esencial de cualquier metodología de matching parcial es que sea robusta y que funcione sin importar la calidad de las mallas geométricas ingresadas. En este caso, las mallas geométricas tienen hoyos, información faltante y ruido. Aunque no todas las metodologías existentes actualmente son capaces de funcionar ante estos escenarios, las metodologías evaluadas lograron obtener métricas de similitud a pesar de todos estos inconvenientes.

Por otra parte, la creación de un escáner 3D desde cero, fue un trabajo de ingeniería aplicada lleno de problemáticas y aprendizajes, que permitió conocer a fondo el cómo se adquiere la información 3D de un objeto de la vida real y las limitantes de los procesos actuales. Este proceso permitió entender cómo, al variar las condiciones de iluminación o las propiedades reflectantes de un objeto, se interfiere de manera directa en la información 3D que se obtendrá o como una calibración deficiente del sistema provoca que los objetos digitalizados tengan deformaciones indeseadas.

En conclusión, el problema de recuperación parcial 3D es un problema que está lejos de ser resuelto. Se necesita continuar la investigación, probar métodos más sofisticados, probar con nuevas maneras de describir la forma de los objetos 3D y las mallas parciales. Siempre con el foco en encontrar metodologías que no solo sean efectivas sino que eficientes.

6.2. Trabajo futuro

Uno de los principales problemas que persiste en el área de recuperación parcial de objetos 3D es la eficiencia. Los métodos actuales aún requieren bastante tiempo de cómputo (el preprocesamiento de la colección demora varias horas), debido a que el cálculo de descriptores es costoso, lo cual significa un problema de escalabilidad al momento de utilizar los algoritmos frente a miles de objetos 3D. Si bien la metodo-

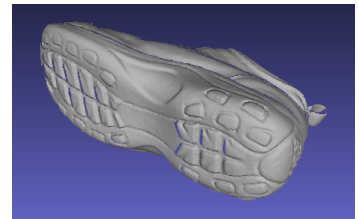
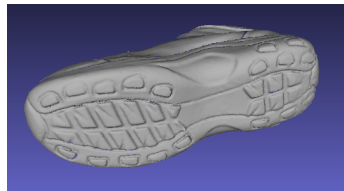
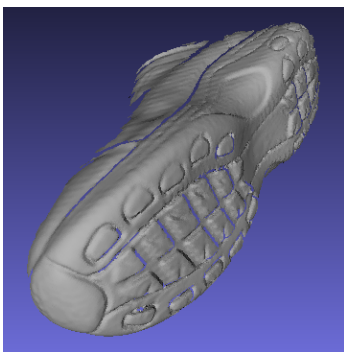
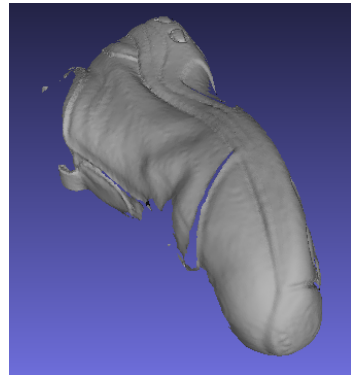
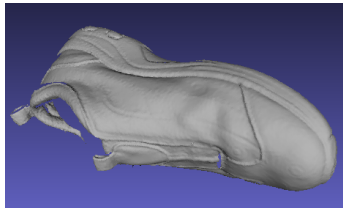
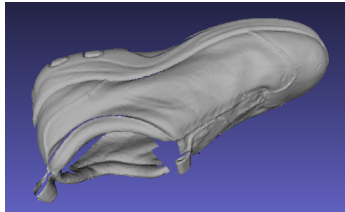
logía de descriptores locales, *clustering* local y distancia *SQFD* es una técnica que permite determinar similitud entre objetos 3D cuando la información es parcial y proveniente de objetos reales, es necesario encontrar metodologías más eficientes que hagan uso, por ejemplo, de índices métricos.

Otro problema que necesita ser mejorado es la robustez de los algoritmos frente a la ausencia de información (información parcial). Muy pocos algoritmos son robustos ante el ruido y la ausencia de información, y la eficiencia (tanto en términos de espacio como de tiempo) de los algoritmos que logran superar tal desafío continúa siendo baja. Se necesita idear nuevas metodologías que permitan determinar la similitud entre objetos cuando la información es parcial.

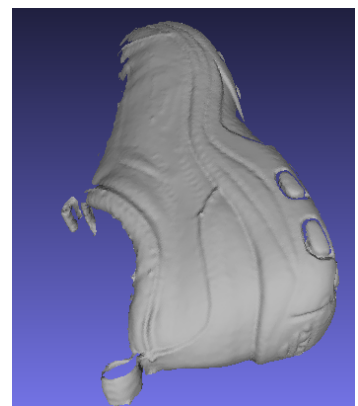
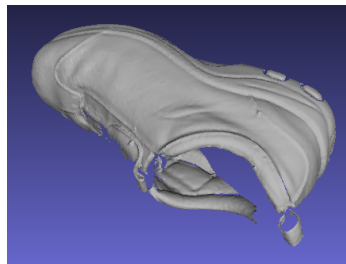
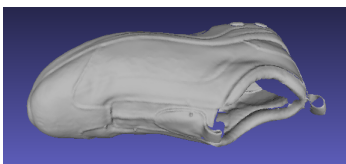
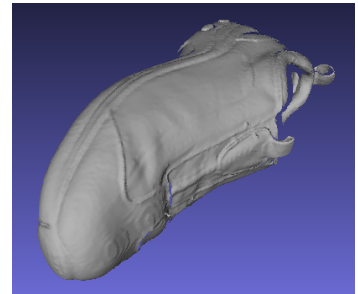
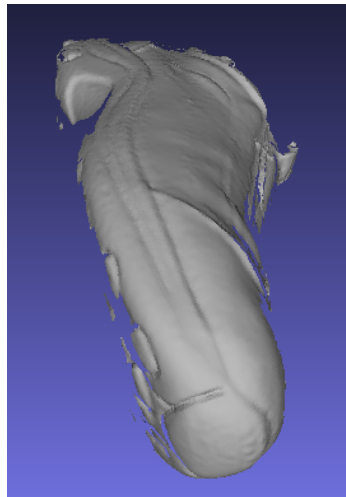
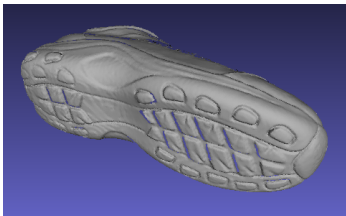
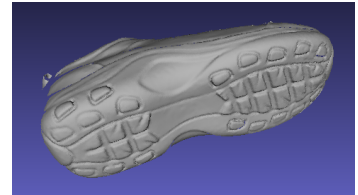
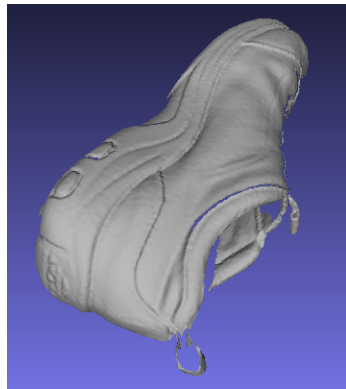
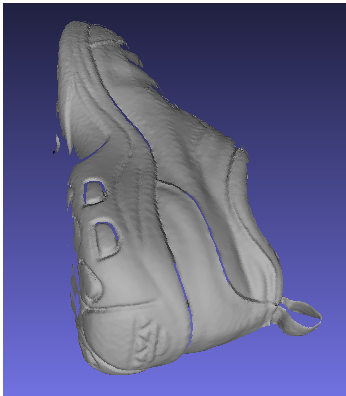
Apéndice A

Vistas parciales y colección de objetos

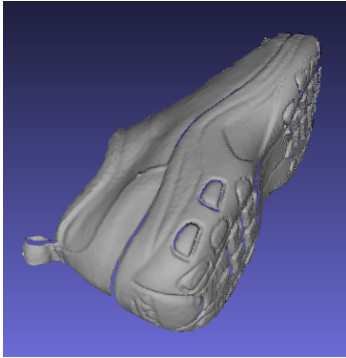
A.1. Vistas parciales de un objeto de la colección



A.1. Vistas parciales de un objeto de la colección



A.2. Objetos 3D de la colección creada



A.2. Objetos 3D de la colección creada

machinegun's group (PRISMA dataset)



Figura A.1: Grupo *machinegun*

A.2. Objetos 3D de la colección creada

suv's group (PRISMA dataset)



Object D00005



Object D00006



Object D00008



Object D00007

Figura A.2: Grupo *suv*

A.2. Objetos 3D de la colección creada

car's group (PRISMA dataset)



Figura A.3: Grupo *car*

A.2. Objetos 3D de la colección creada

doll's group (PRISMA dataset)



Figura A.4: Grupo *doll*

A.2. Objetos 3D de la colección creada

shoe's group (PRISMA dataset)



Figura A.5: Grupo *shoe*

A.2. Objetos 3D de la colección creada

bipedaldinosaur's group (PRISMA dataset)



Figura A.6: Grupo *bipedal dinosaur*

A.2. Objetos 3D de la colección creada

quadrupedal dinosaur's group (PRISMA dataset)



Figura A.7: Grupo *quadrupedal dinosaur*

Bibliografía

- [1] B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranić. An experimental effectiveness comparison of methods for 3d similarity search. *International Journal on Digital Libraries*, 6(1):39–54, 2006.
- [2] Ivan Sipiran, Rafael Meruane, Benjamin Bustos, Tobias Schreck, Bo Li, Yijuan Lu, and Henry Johan. A benchmark of simulated range images for partial shape retrieval. *The Visual Computer*, pages 1–16, 2014.
- [3] B. Bustos, D. Keim, D. Saupe, and T. Schreck. Content-based 3d object retrieval. *Computer Graphics and Applications, IEEE*, 27(4):22–27, 2007.
- [4] Christian Beecks, Merih Seran Uysal, and Thomas Seidl. Signature quadratic form distances for content-based similarity. In *Proceedings of the 17th ACM international conference on Multimedia*, pages 697–700. ACM, 2009.
- [5] B. Curless. Overview of active vision techniques. *SIGGRAPH 99 Course on 3D Photography*, 1999.
- [6] Mitutoyo america corporation. <http://www.mitutoyo.com>.
- [7] Faro technologies inc. <http://www.faro.com>.
- [8] R.C. Gonzalez and E. Richard. Woods, digital image processing, 2002.
- [9] E. Prados and O. Faugeras. Shape from shading: a well-posed problem? In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 870–877. IEEE, 2005.
- [10] S.K. Nayar. Shape from focus system. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pages 302–308. IEEE, 1992.

- [11] Y. Matsumoto, H. Terasaki, K. Sugimoto, and T. Arakawa. A portable three-dimensional digitizer. In *3-D Digital Imaging and Modeling, 1997. Proceedings., International Conference on Recent Advances in*, pages 197–204. IEEE, 1997.
- [12] K.N. Kutulakos and S.M. Seitz. A theory of shape by space carving. *International Journal of Computer Vision*, 38(3):199–218, 2000.
- [13] A.S. Ogale and Y. Aloimonos. Shape and the stereo correspondence problem. *International Journal of Computer Vision*, 65(3):147–162, 2005.
- [14] Ivan Sipiran, Rafael Meruane, Benjamin Bustos, Tobias Schreck, Henry Johan, Bo Li, and Yijuan Lu. Shrec’13 track: large-scale partial shape retrieval using simulated range images. In *Proceedings of the Sixth Eurographics Workshop on 3D Object Retrieval*, pages 81–88. Eurographics Association, 2013.
- [15] Christian Beecks, Merih Seran Uysal, and Thomas Seidl. Signature quadratic form distance. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, pages 438–445. ACM, 2010.
- [16] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 1817–1824. IEEE, 2011.
- [17] Andrew E. Johnson and Martial Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(5):433–449, 1999.
- [18] M. Heczko, D. Keim, D. Saupe, and D. Vranic. Methods for similarity search on 3d databases. *Datenbank-Spektrum*, 2(2):54–63, 2002.
- [19] D.V. Vranic and D. Saupe. Description of 3d-shape using a complex function on the sphere. In *Multimedia and Expo, 2002. ICME’02. Proceedings. 2002 IEEE International Conference on*, volume 1, pages 177–180. IEEE, 2002.
- [20] D. Saupe and D. Vranić. 3d model retrieval with spherical harmonics and moments. *Pattern Recognition*, pages 392–397, 2001.
- [21] D.V. Vranić and D. Saupe. 3d shape descriptor based on 3d fourier transform. In *Proceedings of the EURASIP Conference on Digital Signal Processing for Multimedia Communications and Services (ECMCS 2001), Budapest, Hungary, 2001*.

- [22] D.V. Vranić and D. Saupe. 3d model retrieval. *Proc. SCCG 2000*, pages 3–6, 2004.
- [23] M.T. Suzuki, T. Kato, and N. Otsu. A similarity retrieval of 3d polygonal models using rotation invariant shape descriptors. In *Systems, Man, and Cybernetics, 2000 IEEE International Conference on*, volume 4, pages 2946–2952. IEEE, 2000.
- [24] T. Funkhouser, P. Min, M. Kazhdan, J. Chen, A. Halderman, D. Dobkin, and D. Jacobs. A search engine for 3d models. *ACM Transactions on Graphics (TOG)*, 22(1):83–105, 2003.
- [25] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin. Shape distributions. *ACM Transactions on Graphics (TOG)*, 21(4):807–832, 2002.
- [26] E. Paquet, M. Rioux, A. Murching, T. Naveen, and A. Tabatabai. Description of shape information for 2-d and 3-d objects. *Signal Processing: Image Communication*, 16(1):103–122, 2000.
- [27] T. Zaharia and F. Prêteux. 3d shape-based retrieval within the mpeg-7 framework. In *Proc. SPIE Conf. on Nonlinear Image Processing and Pattern Analysis XII*, volume 4304, pages 133–145, 2001.
- [28] B. Bustos, D.A. Keim, D. Saupe, T. Schreck, and D.V. Vranić. Feature-based similarity search in 3d object databases. *ACM Computing Surveys (CSUR)*, 37(4):345–387, 2005.
- [29] B. Bustos, D. Keim, D. Saupe, T. Schreck, and D. Vranic. An experimental comparison of feature-based 3d retrieval methods. In *3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004. Proceedings. 2nd International Symposium on*, pages 215–222. IEEE, 2004.
- [30] R. Gal and D. Cohen-Or. Salient geometric features for partial shape matching and similarity. *ACM Transactions on Graphics (TOG)*, 25(1):130–150, 2006.
- [31] J. Hu and J. Hua. Salient spectral geometric features for shape matching and retrieval. *The Visual Computer*, 25(5):667–675, 2009.
- [32] R. Toldo, U. Castellani, and A. Fusiello. Visual vocabulary signature for 3d object retrieval and partial matching. In *Eurographics 2009 Workshop on 3D Object Retrieval*, pages 21–28. The Eurographics Association, 2009.

- [33] T. Furuya and R. Ohbuchi. Dense sampling and fast encoding for 3d model retrieval using bag-of-visual features. In *Proceeding of the ACM International Conference on Image and Video Retrieval*, page 26. ACM, 2009.
- [34] M. Ovsjanikov, A.M. Bronstein, M.M. Bronstein, and L.J. Guibas. Shape google: a computer vision approach to isometry invariant shape retrieval. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 320–327. IEEE, 2009.
- [35] H. Dutagaci, A. Godil, A. Axenopoulos, P. Daras, T. Furuya, and R. Ohbuchi. Shrec’09 track: Querying with partial models. In *Eurographics 2009 Workshop on 3D Object Retrieval*, pages 69–76. The Eurographics Association, 2009.
- [36] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser. The princeton shape benchmark. In *Shape Modeling Applications, 2004. Proceedings*, pages 167–178. Ieee, 2004.
- [37] A.M. Bronstein, M.M. Bronstein, M. Bronstein, and R. Kimmel. *Numerical geometry of non-rigid shapes*. Springer-Verlag New York Inc, 2008.
- [38] J. Batlle, E. Mouaddib, and J. Salvi. Recent progress in coded structured light as a technique to solve the correspondence problem: a survey. *Pattern recognition*, 31(7):977, 1998.
- [39] J. Salvi, J. Pages, and J. Batlle. Pattern codification strategies in structured light systems. *Pattern Recognition*, 37(4):827–849, 2004.
- [40] J. Salvi, S. Fernandez, T. Pribanic, and X. Llado. A state of the art in structured light patterns for surface profilometry. *Pattern Recognition*, 43(8):2666–2680, 2010.
- [41] WJ Walecki, F. Szondy, and MM Hilali. Fast in-line surface topography metrology enabling stress calculation for solar cell manufacturing for throughput in excess of 2000 wafers per hour. *Measurement Science and Technology*, 19:025302, 2008.
- [42] Leica Geosystems. Leica scanstation c10. <http://www.leica-geosystems.us/forensic/>.
- [43] P. Cignoni and R. Scopigno. Sampled 3d models for ch applications: A viable and enabling new medium or just a technological exercise? *Journal on Computing and Cultural Heritage (JOCCH)*, 1(1):2, 2008.

- [44] M. Levoy. The digital michelangelo project. In *3-D Digital Imaging and Modeling, 1999. Proceedings. Second International Conference on*, pages 2–11. IEEE, 1999.
- [45] S. Kumar, D. Snyder, D. Duncan, J. Cohen, and J. Cooper. Digital preservation of ancient cuneiform tablets using 3d-scanning. In *3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings. Fourth International Conference on*, pages 326–333. IEEE, 2003.
- [46] F. Berryman, P. Pynsent, J. Fairbank, and S. Disney. A new system for measuring three-dimensional back shape in scoliosis. *European Spine Journal*, 17(5):663–672, 2008.
- [47] P.J. Cobelli, A. Maurel, V. Pagneux, and P. Petitjeans. Global measurement of water waves by fourier transform profilometry. *Experiments in fluids*, 46(6):1037–1047, 2009.
- [48] B. Bustos and I. Sipiran. 3d shape matching for retrieval and recognition. *3D Imaging, Analysis and Applications*, pages 265–308, 2012.
- [49] Greg Flitton, Toby P Breckon, and Najla Megherbi. A 3d extension to cortex like mechanisms for 3d object class recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3634–3641. IEEE, 2012.
- [50] K.W. Bowyer, K. Chang, and P. Flynn. A survey of approaches and challenges in 3d and multi-modal 3d+ 2d face recognition. *Computer Vision and Image Understanding*, 101(1):1–15, 2006.
- [51] M. Novotni and R. Klein. Geometric 3d comparison-an application. *ECDL WS Generalized Documents*, 2001.
- [52] D.A. Keim. *Efficient geometry-based similarity search of 3D spatial databases*, volume 28. ACM, 1999.
- [53] O. Ronneberger, H. Burkhardt, and E. Schultz. General-purpose object recognition in 3d volume data sets using gray-scale invariants-classification of airborne pollen-grains recorded with a confocal laser scanning microscope. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 2, pages 290–295. IEEE, 2002.

- [54] S. Goodall, P. Lewis, K. Martinez, P. Sinclair, F. Giorgini, M. Addis, M. Boniface, C. Lahanier, and J. Stevenson. Sculpteur: multimedia retrieval for museums. *Image and Video Retrieval*, pages 1962–1963, 2004.
- [55] H. Chen and B. Bhanu. Human ear recognition in 3d. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):718–737, 2007.
- [56] I. Atmosukarto, K. Wilamowska, C. Heike, and L.G. Shapiro. 3d object classification using salient point patterns with application to craniofacial research. *Pattern Recognition*, 43(4):1502–1517, 2010.
- [57] J.S. Yeh, D.Y. Chen, B.Y. Chen, and M. Ouhyoung. A web-based three-dimensional protein retrieval system by matching visual similarity. *Bioinformatics*, 21(13):3056–3057, 2005.
- [58] E. Paquet, H.L. Viktor, et al. Exploring protein architecture using 3d shape-based signatures. 2007.
- [59] E. Paquet and H.L. Viktor. Capri/mr: exploring protein databases from a structural and physicochemical point of view. *Proceedings of the VLDB Endowment*, 1(2):1504–1507, 2008.
- [60] C.F. You and Y.L. Tsai. 3d solid model retrieval for engineering reuse based on local feature correspondence. *The International Journal of Advanced Manufacturing Technology*, 46(5):649–661, 2010.
- [61] Q.X. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann. Reassembling fractured objects by geometric matching. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 569–578. ACM, 2006.
- [62] P. Huang, A. Hilton, and J. Starck. Shape similarity for 3d video sequences of people. *International Journal of Computer Vision*, 89(2):362–381, 2010.
- [63] C. Böhm, S. Berchtold, and D.A. Keim. Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. *ACM Computing Surveys (CSUR)*, 33(3):322–373, 2001.
- [64] E. Chávez, G. Navarro, R. Baeza-Yates, and J.L. Marroquín. Searching in metric spaces. *ACM Computing Surveys (CSUR)*, 33(3):273–321, 2001.
- [65] A. Flisch, J. Wirth, R. Zanini, M. Breitenstein, A. Rudin, F. Wendt, F. Mnich, and R. Golz. Industrial computed tomography in reverse engineering applications. *DGZ-fP-Proceedings BB*, 1999.

- [66] U.R. Dhond and J.K. Aggarwal. Structure from stereo—a review. *Systems, Man and Cybernetics, IEEE Transactions on*, 19(6):1489–1510, 1989.
- [67] R. Zhang, P.S. Tsai, J.E. Cryer, and M. Shah. Shape-from-shading: a survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(8):690–706, 1999.
- [68] M. Subbarao, T.S. Choi, and A. Nikzad. Focusing techniques (journal paper). *Optical Engineering*, 32(11):2824–2836, 1993.
- [69] M. Subbarao and G. Surya. Application of spatial-domain convolution/deconvolution transform for determining distance from image defocus. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 1822, pages 159–167, 1993.
- [70] A. Laurentini. The visual hull concept for silhouette-based image understanding. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(2):150–162, 1994.
- [71] B. Horn and M.J. Brooks. *Shape from shading*. MIT press, 1989.
- [72] E. Krotkov. Focusing. *International Journal of Computer Vision*, 1(3):223–237, 1988.
- [73] M. Watanabe and S.K. Nayar. Rational filters for passive depth from defocus. *International Journal of Computer Vision*, 27(3):203–225, 1998.
- [74] J. Carranza, C. Theobalt, M.A. Magnor, and H.P. Seidel. Free-viewpoint video of human actors. *ACM Transactions on Graphics (TOG)*, 22(3):569–577, 2003.
- [75] W. Matusik, C. Buehler, R. Raskar, S.J. Gortler, and L. McMillan. Image-based visual hulls. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 369–374. ACM Press/Addison-Wesley Publishing Co., 2000.
- [76] J. Starck and A. Hilton. Model-based multiple view reconstruction of people. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 915–922. IEEE, 2003.
- [77] G. Hobrough and T. Hobrough. Stereopsis for robots by iterative stereo image matching. 1984.

- [78] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 519–528. Ieee, 2006.
- [79] N. Ayache and F. Lustman. Fast and reliable passive trinocular stereovision. In *Proceedings ICCV*, volume 87, pages 422–427, 1987.
- [80] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*, volume 2. Cambridge Univ Press, 2000.
- [81] A. Fusiello, V. Roberto, and E. Trucco. Efficient stereo with multiple windowing. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 858–863. IEEE, 1997.
- [82] C. Sun. A fast stereo matching method. In *Digital Image Computing: Techniques and Applications*, pages 95–100. Citeseer, 1997.
- [83] J. Weng, N. Ahuja, and T.S. Huang. Matching two perspective views. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14(8):806–825, 1992.
- [84] R.J. Woodham. Photometric method for determining surface orientation. *Optical engineering*, 1(7):139–144, 1980.
- [85] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *International Journal of Computer Vision*, 72(3):239–257, 2007.
- [86] A. Treuille, A. Hertzmann, and S. Seitz. Example-based stereo with general brdfs. *Computer Vision-ECCV 2004*, pages 457–469, 2004.
- [87] T. Higo, Y. Matsushita, N. Joshi, and K. Ikeuchi. A hand-held photometric stereo camera for 3-d modeling. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1234–1241. IEEE, 2009.
- [88] C. Hernández, G. Vogiatzis, G.J. Brostow, B. Stenger, and R. Cipolla. Non-rigid photometric stereo with colored lights. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.
- [89] Z. Ji and MC Leu. Design of optical triangulation devices. *Optics & Laser Technology*, 21(5):339–341, 1989.

- [90] P.J. Besl. Active, optical range imaging sensors. *Machine vision and applications*, 1(2):127–152, 1988.
- [91] CREAFORM. Handyscan 3d. <http://www.creaform3d.com>.
- [92] Polhemus. Polhemus fastscan. <http://www.fastscan3d.com/>.
- [93] Inc. NextEngine. Nextengine 3d laser scanner. <http://www.nextengine.com>.
- [94] H. Hiroshi Kawasaki, R. Furukawa, R. Sagawa, and Y. Yagi. Dynamic scene shape reconstruction using a single structured light pattern. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.
- [95] J. Salvi, J. Batlle, and E. Mouaddib. A robust-coded pattern projection for dynamic 3d scene measurement. *Pattern Recognition Letters*, 19(11):1055–1065, 1998.
- [96] KL Boyer and AC Kak. Color-encoded structured light for rapid active ranging. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (1):14–28, 1987.
- [97] T. Etzion. Constructions for perfect maps and pseudorandom arrays. *Information Theory, IEEE Transactions on*, 34(5):1308–1316, 1988.
- [98] Zhengyou Zhang. Microsoft kinect sensor and its effect. *MultiMedia, IEEE*, 19(2):4–10, 2012.
- [99] M. Gupta, A. Agrawal, A. Veeraraghavan, and S.G. Narasimhan. Structured light 3d scanning in the presence of global illumination. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 713–720. IEEE, 2011.
- [100] D.C. Brown. Decentering distortion of lenses. *Photometric Engineering*, 32(3):444–462, 1966.
- [101] J.Y. Bouguet. Camera calibration toolbox for matlab. 2004.
- [102] Z. Zhang. Flexible camera calibration by viewing a plane from unknown orientations. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 666–673. Ieee, 1999.

- [103] J. Heikkila and O. Silven. A four-step camera calibration procedure with implicit image correction. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 1106–1112. IEEE, 1997.
- [104] Z. Zhang. A flexible new technique for camera calibration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(11):1330–1334, 2000.
- [105] R.I. Hartley and P. Sturm. Triangulation. *Computer vision and image understanding*, 68(2):146–157, 1997.
- [106] R. Hartley. Estimation of relative camera positions for uncalibrated cameras. In *Computer Vision—ECCV’92*, pages 579–587. Springer, 1992.
- [107] HC Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms*, MA Fischler and O. Firschein, eds, pages 61–62, 1987.
- [108] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. Numerical recipes in c: the art of scientific computing, 1992. *Cité en*, page 92, 1992.
- [109] JL Posdamer and MD Altschuler. Surface measurement by space-encoded projected beam systems. *Computer graphics and image processing*, 18(1):1–17, 1982.
- [110] S. Inokuchi, K. Sato, and F. Matsuda. Range imaging system for 3-d object recognition. In *Proceedings of the International Conference on Pattern Recognition*, pages 806–808, 1984.
- [111] Kosuke Sato, Hiroyuki Yamamoto, and Seiji Inokuchi. Tuned range finder for high precision 3d data. In *International Conference on Pattern Recognition*, pages 1168–1171, 1986.
- [112] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on pattern analysis and machine intelligence*, 14(2):239–256, 1992.
- [113] Zhengyou Zhang. Iterative point matching for registration of free-form curves. 1992.
- [114] Marc Alexa, Johannes Behr, Daniel Cohen-Or, Shachar Fleishman, David Levin, and Claudio T Silva. Computing and rendering point set surfaces. *Visualization and Computer Graphics, IEEE Transactions on*, 9(1):3–15, 2003.

- [115] R. Baeza-Yates, B. Ribeiro-Neto, et al. *Modern information retrieval*, volume 82. Addison-Wesley New York, 1999.
- [116] Bo Li and Henry Johan. Sketch-based 3d model retrieval by incorporating 2d-3d alignment. *Multimedia tools and applications*, 65(3):363–385, 2013.
- [117] Bo Li, Tobias Schreck, Afzal Godil, Marc Alexa, Tamy Boubekeur, Benjamin Bustos, Jipeng Chen, Mathias Eitz, Takahiko Furuya, Kristian Hildebrand, et al. Shrec’12 track: Sketch-based 3d shape retrieval. In *3DOR*, pages 109–118, 2012.
- [118] Bo Li, Yijuan Lu, and Henry Johan. Sketch-based 3d model retrieval by viewpoint entropy-based adaptive view clustering. In *Proceedings of the Sixth Eurographics Workshop on 3D Object Retrieval*, pages 49–56. Eurographics Association, 2013.
- [119] Ivan Sipiran, Benjamin Bustos, and Tobias Schreck. Data-aware 3d partitioning for generic shape retrieval. *Computers & Graphics*, 37(5):460–472, 2013.
- [120] Bo Li and Henry Johan. View context: a 3d model feature for retrieval. In *Advances in Multimedia Modeling*, pages 185–195. Springer, 2010.
- [121] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *NIPS*, volume 2, page 3, 2000.
- [122] Shigeo Takahashi, Issei Fujishiro, Yuriko Takeshima, and Tomoyuki Nishita. A feature-driven approach to locating optimal viewpoints for volume visualization. In *Visualization, 2005. VIS 05. IEEE*, pages 495–502. IEEE, 2005.
- [123] Serge Belongie, Jitendra Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, 2002.
- [124] I. Sipiran and B. Bustos. Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *The Visual Computer*, 27(11):963–976, 2011.
- [125] Dejan V Vranic. Desire: a composite 3d-shape descriptor. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 4–pp. IEEE, 2005.

- [126] Alexander M Bronstein, Michael M Bronstein, Leonidas J Guibas, and Maks Ovsjanikov. Shape google: Geometric words and expressions for invariant shape retrieval. *ACM Transactions on Graphics (TOG)*, 30(1):1, 2011.
- [127] Yi Liu, Hongbin Zha, and Hong Qin. Shape topics: A compact representation and new algorithms for 3d partial shape retrieval. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2025–2032. IEEE, 2006.
- [128] Silvia Biasotti, X Bai, Benjamin Bustos, Andrea Cerri, Daniela Giorgi, L Li, Michela Mortara, Ivan Sipiran, S Zhang, and Michela Spagnuolo. Shrec'12 track: Stability on abstract shapes. In *3DOR*, pages 101–107, 2012.
- [129] Wee Kheng Leow and Rui Li. The analysis and applications of adaptive-binning color histograms. *Computer Vision and Image Understanding*, 94(1):67–91, 2004.
- [130] Andrew Edie Johnson. *Spin-images: a representation for 3-D surface matching*. PhD thesis, Citeseer, 1997.
- [131] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 3212–3217. IEEE, 2009.
- [132] Radu Bogdan Rusu, Nico Blodow, Zoltan Csaba Marton, and Michael Beetz. Aligning point cloud views using persistent feature histograms. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3384–3391. IEEE, 2008.
- [133] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, and Michael Beetz. Learning informative point classes for the acquisition of object model maps. In *Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on*, pages 643–650. IEEE, 2008.