



INSTITUTO POLITÉCNICO NACIONAL  
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN



RECUPERACIÓN Y COMPARACIÓN DE FIGURAS EN 3D

TESIS  
QUE PARA OBTENER EL GRADO DE  
MAESTRO EN CIENCIAS DE LA COMPUTACIÓN

PRESENTA  
ABEL VALLE CHÁVEZ

DIRECTOR  
DR. JESÚS G. FIGUEROA NAZUNO

México D.F.

Junio 2012





# INSTITUTO POLITÉCNICO NACIONAL SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

## ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D.F. siendo las 12:00 horas del día 04 del mes de Junio de 2012 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación del:

**Centro de Investigación en Computación**

para examinar la tesis titulada:

**“Recuperación y comparación de figuras en 3D”**

Presentada por el alumno:

**VALLE**

Apellido paterno

**CHÁVEZ**

Apellido materno

**ABEL**

Nombre(s)

Con registro:

A	1	0	0	3	3	2
---	---	---	---	---	---	---

aspirante de: **MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN**

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

### LA COMISIÓN REVISORA

Director de Tesis

Dr. Jesús Guillermo Figueroa Nazuno

Dr. Edgardo Manuel Felipe Riverón

Dr. Sergio Suárez Guerra

Dr. Miguel Jesús Torres Ruiz

Dr. Amadeo José Argüelles Cruz

PRESIDENTE DEL COLEGIO DE PROFESORES

Dr. Luis Alfonso Villa Vargas







**INSTITUTO POLITECNICO NACIONAL**  
**SECRETARÍA DE INVESTIGACIÓN Y POSGRADO**

*CARTA CESION DE DERECHOS*

En la Ciudad de **México** el día **12** del mes de **Junio** del año **2012**, el (la) que suscribe **C. Abel Valle Chávez** alumno del Programa de la **Maestría en Ciencias de la Computación** con número de registro **A100332**, adscrito al **Centro de Investigación en Computación del Instituto Politécnico Nacional**, manifiesta que es autor intelectual del presente trabajo de Tesis bajo la dirección del **Dr. Jesús G. Figueroa Nazuno** y cede los derechos del trabajo intitulado **“Recuperación y comparación de figuras en 3D”**, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben de reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección **a.valle.chavez@gmail.com**. Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

---

Abel Valle Chávez



## RESUMEN

---

Distintos campos de estudio como por ejemplo, la arqueología, biología molecular, medicina, gráficos por computadora y la industria, están conformando conjuntos de figuras en 3D para diversos propósitos de análisis. La cantidad de modelos se incrementa y surge la necesidad de tener sistemas de análisis de similitud entre figuras 3D para realizar búsquedas entre conjuntos grandes de objetos.

El primer problema que se estudia en este trabajo es el de representación. La aproximación que se proporciona consiste en la propuesta y prueba de tres técnicas nuevas de representación, que tienen como base la transformación de dominio 3D a 1D.

También se propone y se prueba una metodología que permite la comparación y recuperación de figuras 3D. La metodología se compone de las etapas de selección de conjuntos de datos, control de invariantes, extracción de secuencias representativas, comparación elástica y construcción de matrices de distancias y finalmente la evaluación de desempeño de recuperación.

Los resultados muestran una mejora sustancial en comparación a técnicas del estado del arte. La medida de similitud propuesta puede aplicarse como medida de distancia para clasificación y reconocimiento de patrones. Las técnicas de representación y la técnica de comparación se pueden utilizar para la implementación de sistemas de administración de figuras 3D de sólidos específicos.



## ABSTRACT

---

Different fields of study such as archeology, molecular biology, medicine, computer graphics and industry are constructing sets of 3D shapes for different purposes of analysis. The amount of models is increasing and the need for analysis systems of similarity between 3D shapes raises to improve the search through large sets of objects.

The first problem studied in this work, is that of representation. The approach provided consists on the proposal and testing of three new techniques of representation, which are based on the transformation from 3D to 1D domain.

A methodology that allows comparison and retrieval of 3D shapes is also proposed and tested. The methodology comprises the steps of selecting data sets, rotation, scaling and translation transformations control, representative sequences extraction, comparison and construction of distance matrices, and, finally, evaluation of the precision and recall ratio.

The implementation of the proposed methodology gets, measuring average performance, better results than the eight relevant techniques surveyed in a study involving state of the art representative techniques.

The proposed similarity measure can be applied as a distance measure for classification and pattern recognition. The representation and comparison techniques can be used to implement management systems of specific solid 3D shapes.



## Dedicatoria

A mis padres Susana y Abel, mis hermanos Eridani y Edgar, ya que a ellos les debo mi desarrollo, mi formación, parte de mi educación y en gran medida lo que soy.



*We have seen that computer programming is an art,  
because it applies accumulated knowledge to the world,  
because it requires skill and ingenuity, and especially  
because it produces objects of beauty.*

— Donald E. Knuth [24]

## AGRADECIMIENTOS

---

Al Dr. Jesús F. Nazuno por su guía en mi formación y por el apoyo incondicional en esta etapa de mi desarrollo.

A E. Araceli Santos Camacho por su paciencia, comentarios y opiniones que contribuyeron al desarrollo de este trabajo.

A los amigos y compañeros que estuvieron ahí, donde las pláticas y convivencia aportaron de manera constructiva.

Al comité tutorial, que contribuyeron con comentarios y observaciones en el desarrollo de esta tesis:

Dr. Eduardo Gómez Ramírez, Dr. Sergio Suárez Guerra, Dr. Edgardo Manuel Felipe Riverón, Dr. Miguel Jesús Torres Ruiz y el Dr. Amadeo José Argüelles Cruz.

Se agradece al Centro de Investigación en Computación (CIC), al Instituto Politécnico Nacional (IPN) y al Consejo Nacional de Ciencia y Tecnología (CONACyT) por su imprescindible apoyo en la realización de este trabajo.



## LISTA DE SÍMBOLOS

---

$C = \{c_1, c_2, \dots, c_n\}$	Secuencia $C$ con $n$ elementos
$Q = \{q_1, q_2, \dots, q_n\}$	Secuencia $Q$ con $n$ elementos
$S = \{s_1, s_2, \dots, s_n\}$	Conjunto de figuras 3D con $n$ elementos
$E = \{e_1, e_2, \dots, e_n\}$	Secuencia de valores de distancia con $n$ elementos
$T = \{T_1, T_2, \dots, T_n\}$	Conjunto de triángulos de $n$ elementos
$P = \{P_1, P_2, \dots, P_n\}$	Conjunto de puntos de $n$ elementos
$S_i$	Área del triángulo $T_i$
$st_i$	Secuencia numérica de datos para $i \in \{0, 1, 2, \dots, N\}$
$M$	Arreglo rectangular de datos de $m$ renglones y $n$ columnas
$M_C$	Matriz de covarianza
$R_x$	Matriz de rotación en un ángulo $\alpha$ alrededor del eje $x$
$R_y$	Matriz de rotación en un ángulo $\beta$ alrededor del eje $y$
$R_z$	Matriz de rotación en un ángulo $\gamma$ alrededor del eje $z$
$S_v$	Matriz de escala
$m_c$	Centro geométrico de una figura en 3D
$V_d = \{v_1, v_2, \dots, v_{n-1}\}$	Secuencia de diferencias entre parejas de distancias de $n - 1$ elementos



## SIGLAS Y ACRÓNIMOS

---

**CAD** (*Computer-Aided Design*): Diseño asistido por computadora.

**CAM** (*Computer-Aided Manufacturing*): Fabricación asistida por computadora.

**DFT** (*Discrete Fourier Transform*): Transformada discreta de Fourier.

**FDTW**: *Fast Dynamic Time Warping*.

**IR** (*Information Retrieval*): Recuperación de información.

**NN** (*Nearest Neighbor*): Vecino más cercano.

**SH** (*Spherical Harmonics*): Armónicos esféricos (AE).

**VR** (*Virtual Reality*): Realidad virtual.

**VRML** (*Virtual Reality Modeling Language*): Lenguaje de modelado de realidad virtual.

**1D**: Una dimensión o unidimensional.

**2D**: Dos dimensiones o bidimensional.

**3D**: Tres dimensiones o tridimensional.

**HT** (*Hough Transform*): Transformada de Hough.



## GLOSARIO

---

**Alineamiento elástico:** El término empleado para describir el proceso de medir el esfuerzo de alineación entre los elementos de las secuencias representativas.

**Comparación:** Análisis de entidades para establecer sus semejanzas y/o diferencias. En las ciencias usualmente implica el cálculo de algún indicador numérico o métrica.

**Distancia:** Espacio o período de tiempo que transcurre entre dos eventos. Para este trabajo se puede concebir como una magnitud complementaria a la similitud que consecuentemente nos indica diferencia.

**DTW** (*Dynamic Time Warping*): Es una técnica para cuantificar y obtener la alineación que representa la distancia mínima entre dos secuencias de datos bajo restricciones de monotonicidad.

**FV** (*Feature Vector*). Es un vector n-dimensional de características numéricas que representan a un objeto.

**Matriz:** Es un arreglo rectangular de datos numéricos, los cuales se pueden sumar y multiplicar. Se emplea ampliamente para la representación de diversos conceptos en todas las ciencias.

**PSB** (*Princeton Shape Benchmark*): Repositorio de modelos 3D para la evaluación de algoritmos de recuperación y análisis de figuras 3D.

**Recuperación de información:** Se refiere a la capacidad de un sistema para aproximar respuestas a una solicitud en particular. En general se denomina así al proceso realizado para responder a consultas en aplicaciones tales como las bases de datos y los motores de búsqueda.

**Similitud:** También conocida como semejanza. Se puede definir como la intensidad de la relación entre dos entidades cuya medida se basa en las características que comparten

y en las que difieren. En el presente trabajo se plantea otra aproximación donde la similitud se establece como el efecto de sustituir a las entidades originales y combinarlas (conjugarlas) con las características de otras. La similitud se puede definir también en términos de las diferencias o como una función de distancia.

## ÍNDICE GENERAL

---

Acta de revisión de tesis . . . . .	iii
Carta de cesión de derechos . . . . .	iv
Resumen . . . . .	vii
Abstract . . . . .	ix
Dedicatoria . . . . .	xi
Agradecimientos . . . . .	xiii
Lista de símbolos . . . . .	xv
Siglas y acrónimos . . . . .	xvii
Glosario . . . . .	xix
1 INTRODUCCIÓN . . . . .	1
1.1 Planteamiento del problema . . . . .	1
1.2 Objetivo general . . . . .	2
1.3 Objetivos particulares . . . . .	2
1.4 Motivación . . . . .	2
1.5 Alcances . . . . .	3
1.6 Panorama de los sistemas de análisis de similitud entre figuras 3D . . . . .	3
1.7 La necesidad del análisis de figuras en 3D . . . . .	5
1.8 Análisis de similitud de figuras 3D en distintos campos de aplicación . . . . .	6
1.9 Problemas asociados con el análisis de entidades en 3D . . . . .	9
1.10 Organización del documento . . . . .	11
2 ESTADO DEL ARTE . . . . .	13
2.1 Antecedentes . . . . .	13
2.2 Descripción de la propuesta . . . . .	17
3 MARCO CONCEPTUAL . . . . .	21
3.1 Representación de figuras 3D . . . . .	21
3.1.1 Requerimientos de la representación . . . . .	22
3.2 Conceptos de comparación . . . . .	22
3.2.1 El problema de la similitud y la comparación . . . . .	23
3.2.2 Métricas de similitud . . . . .	26
3.3 Técnicas de comparación . . . . .	29

3.3.1	Comparación elástica desde el enfoque de esfuerzo de alineación	29
3.3.2	Distancia <i>Dynamic Time Warping</i>	32
3.3.3	Distancia <i>Minkowski</i>	35
3.3.4	Comparación de dos figuras en 3D	36
3.4	Conceptos de recuperación	37
4	METODOLOGÍA	41
4.1	Descripción de la metodología	41
4.1.1	Limitaciones	42
4.2	Datos tridimensionales	43
4.3	Control de invariantes de una figura 3D	45
4.3.1	Invariante de rotación	45
4.3.2	Invariante de escala	48
4.3.3	Invariante de traslación	50
4.4	Construcción de secuencias representativas	51
4.4.1	Representación de medidas sobre un cubo ( $R1$ )	53
4.4.2	Representación de medidas sobre un cilindro ( $R2$ )	54
4.4.3	Representación de medidas sobre una esfera ( $R3$ )	56
4.4.4	Reconstrucción parcial de figuras 3D	57
4.4.5	Robustez de las técnicas de representación	58
4.4.6	Propiedades de las secuencias representativas	59
4.5	Comparación de las secuencias representativas	60
4.5.1	Combinación de valores de distancia	60
4.6	Complejidad computacional de las técnicas	61
4.7	Indicador de similitud entre figuras 3D	63
4.7.1	Propiedades del indicador de similitud	63
4.7.2	Relación <i>distorsión-distancia</i>	63
4.7.3	Relación <i>diferencia de distancias</i> entre <i>distorsiones</i> consecutivas	64
4.8	Análisis de la recuperación	64
4.9	Los conjuntos de datos	64
4.9.1	Figuras 3D del <i>PSB</i>	64
4.9.2	Figuras geométricas primitivas con deformaciones cuantificadas	66
4.10	El archivo de clasificación	67
4.11	La matriz de distancias	67
4.12	Condiciones experimentales	68
4.13	Procedimiento para la visualización de la alineación de secuencias	69
4.14	Comentarios de implementación	71
5	RESULTADOS EXPERIMENTALES	73
5.1	Comentarios generales	73

5.2	Experimento 1 . . . . .	74
5.2.1	Tiempo de ejecución de control de invariantes y construcción de secuencias . . . . .	74
5.3	Experimento 2 . . . . .	74
5.3.1	Tiempo de ejecución de comparación de secuencias representativas . . . . .	74
5.4	Experimento 3 . . . . .	75
5.4.1	Robustez ante distintas resoluciones . . . . .	75
5.5	Experimento 4 . . . . .	77
5.5.1	Descripción del experimento de recuperación de las tres técnicas de representación . . . . .	77
5.5.2	Resultados de ejemplo de recuperación de la técnica de representación <i>R1</i> . . . . .	78
5.5.3	Resultados de ejemplo de recuperación de la técnica de representación <i>R2</i> . . . . .	78
5.5.4	Resultados de ejemplo de recuperación de la técnica de representación <i>R3</i> . . . . .	79
5.5.5	Comparación de las técnicas <i>R1</i> , <i>R2</i> y <i>R3</i> . . . . .	80
5.6	Experimento 5 . . . . .	82
5.6.1	Resultados de ejemplo de recuperación de la técnica de combinación de valores de distancias . . . . .	82
5.6.2	Comparación de la combinación de valores de distancia . . . . .	83
5.7	Comparación de resultados con técnicas del estado del arte . . . . .	84
5.8	Experimento 7 . . . . .	85
5.8.1	Relación <i>distorsión-distancia</i> . . . . .	85
5.9	Experimento 8 . . . . .	88
5.9.1	Relación <i>diferencia de distancias</i> entre <i>distorsiones</i> consecutivas . . . . .	88
5.10	Experimento 9 . . . . .	90
5.10.1	Otras medidas de desempeño . . . . .	90
5.10.2	Análisis del <i>vecino más cercano</i> . . . . .	90
5.10.3	Análisis de <i>medida-E</i> . . . . .	90
5.11	Experimento 10 . . . . .	93
5.11.1	Visualización de alineación de secuencias y reconstrucción parcial de 1D a 3D . . . . .	93
6	CONCLUSIONES . . . . .	97
6.1	Conclusiones . . . . .	97
6.2	Contribuciones . . . . .	99
6.3	Trabajo futuro . . . . .	100

A	APÉNDICE A . . . . .	103
A.1	Transformaciones de datos en 3D . . . . .	103
A.1.1	Conceptos y propiedades de matrices numéricas . . . . .	103
A.1.2	Transformaciones de matrices de tres dimensiones . . . . .	105
A.2	Intersección rayo-polígono . . . . .	107
A.2.1	Intersección rayo-triángulo . . . . .	107
A.2.2	Cálculo de intersección y distancia rayo-polígono en 3D . . . . .	108
A.3	Formatos de archivos . . . . .	109
A.3.1	Formato de archivo de objeto 3D (.off) . . . . .	109
A.3.2	Formato de archivo de clasificación . . . . .	110
A.4	Algoritmos de registro de medidas . . . . .	111
A.4.1	Algoritmo completo de registros de medidas sobre un cubo . . . . .	111
A.4.2	Algoritmo completo de registro de medidas sobre cilindros . . . . .	113
B	APÉNDICE B . . . . .	115
B.1	La aplicación SIM3D . . . . .	115
B.1.1	Sobre <i>QT Creator</i> y <i>C++</i> . . . . .	115
B.1.2	Sobre la aplicación SIM3D . . . . .	116
B.1.3	Archivos utilizados por la aplicación . . . . .	116
B.1.4	Módulos de la aplicación . . . . .	117
B.2	Manual de la aplicación SIM3D . . . . .	119
B.2.1	El módulo de análisis y comparación . . . . .	119
B.2.2	El módulo de medidas de desempeño . . . . .	120
B.2.3	El módulo de recuperación . . . . .	122
	REFERENCIAS . . . . .	125

## ÍNDICE DE FIGURAS

---

Figura 1	Ejemplo de búsqueda de similitud en una base de datos de objetos 3D, mostrando un objeto de consulta ( $q$ ) y un conjunto de respuestas recuperadas posiblemente relevantes (a). . . . .	4
Figura 2	Ilustración de interacción humano-objeto en un entorno 3D virtual. . . . .	5
Figura 3	a) Representación 2D ( $x, y$ ) de una figura 3D. b) Representación en 3D ( $x, y, z$ ) donde en (1) se muestran los múltiples valores de $z$ que no se pueden registrar en 2D. . . . .	6
Figura 4	Reconocimiento de figuras 3D a partir de primitivas [17]. . . . .	13
Figura 5	Modelo del proceso de extracción del descriptor. . . . .	15
Figura 6	Categorías de técnicas de recuperación de objetos 3D. . . . .	18
Figura 7	Un objeto 3D con orientación y escala diferentes (izquierda) y también representado en varios niveles de detalle (derecha). . . . .	22
Figura 8	Paradigma clásico para el problema de similitud entre objetos. La función de similitud se ilustra como un proceso de toma de decisión binario. . . . .	23
Figura 9	Estructura conceptual del problema de similitud. Las entidades a comparar deben de ser cuantificables. . . . .	24
Figura 10	Similitud por atributos y por valores numéricos. . . . .	25
Figura 11	Cubos de voxeles que ejemplifican la cantidad de combinaciones de valores de ceros y unos en la complejidad de su comparación. . . . .	26
Figura 12	Descripción de la implicación del valor de una función de distancia. . . . .	27
Figura 13	Problemas en la comparación de secuencias numéricas. (a) La secuencia $R$ presenta ruido en la posición 3 de la secuencia $S$ . (b) $R$ tiene escala y posición diferente, (c) $R$ se encuentra desfasada y (d) $R$ se encuentra con diferente escala con respecto a la serie $S$ . . . . .	30
Figura 14	La comparación directa no es suficiente para reconocer la letra que se ilustra. . . . .	30

Figura 15	Alineación usando distancia <i>Euclidiana</i> y alineación no lineal ( <i>FDTW</i> ). . . . .	31
Figura 16	Ejemplo de comparación elástica. . . . .	31
Figura 17	Proceso de alineación de secuencias usando <i>FDTW</i> : (a) secuencias originales, (b) proceso de alineación, (c) alineación encontrada. . . . .	34
Figura 18	Ejemplo de la idea para la alineación entre una función seno con ruido y una coseno en $[0, 2\pi]$ , con líneas guías trazadas cada $\frac{\pi}{4}$ . . . . .	35
Figura 19	Patrones de la distancia <i>Minkowski</i> en dos dimensiones que satisfacen $\ x\ _p$ . Se considera la distancia desde el punto $(0,0)$ a un punto $(x,y)$ . . . . .	36
Figura 20	Diagrama general de la comparación de representaciones de figuras 3D. . . . .	37
Figura 21	Metodología propuesta. . . . .	42
Figura 22	Ejemplo de figura con dificultades para una recuperación. . . . .	43
Figura 23	Una figura en 3D de un cerebro, representada mediante malla de polígonos triangulares. . . . .	44
Figura 24	a) Al desplazar los puntos $p1$ y $p2$ cualquier punto en el área del triángulo se mantiene en el mismo plano. b) Al desplazar los puntos $p1$ y $p2$ los puntos en el área del rectángulo se encuentran en distintos planos. . . . .	45
Figura 25	Proceso de control de la invariante de rotación. . . . .	48
Figura 26	Proceso de control de invariante de escala. . . . .	49
Figura 27	Proceso de control de invariante de traslación. . . . .	51
Figura 28	Intersección de un vector de dirección y triángulo en 3D. . . . .	52
Figura 29	(a) Representación de medidas usando como marco de referencia un cubo. (b) Representación usando un cilindro. (c) Representación usando una esfera. . . . .	53
Figura 30	Puntos de observación sobre un cubo como marco de referencia. . . . .	54
Figura 31	Puntos de observación sobre cilindros como marco de referencia. . . . .	55
Figura 32	Puntos de observación sobre una esfera como marco de referencia. . . . .	56
Figura 33	Ejemplo de figura 3D y su reconstrucción parcial a partir de una secuencia numérica en 1D. . . . .	58
Figura 34	Metodología de robustez para distintas resoluciones. . . . .	59

Figura 35	Comparación de dos figuras 3D usando <i>FDTW</i> . Las secuencias se obtienen a partir de: (a) Representación de medidas sobre un cubo. (b) Representación de medidas sobre cilindros. (c) Representación de medidas sobre una esfera. El resultado son valores de similitud en (1), (2) y (3). . . . .	60
Figura 36	Suma de matrices de distancias. . . . .	61
Figura 37	Ejemplos de figuras 3D del <i>PSB</i> . . . . .	66
Figura 38	Figura 3D de un “tubo” cuantificando sus deformaciones. . . . .	66
Figura 39	Ejemplos de figuras 3D primitivas con deformaciones graduales. . . . .	67
Figura 40	Captura de la herramienta <i>Blender</i> editando un vértice de un cilindro en 3D. . . . .	68
Figura 41	Opciones de condiciones experimentales para el análisis de figuras 3D. . . . .	69
Figura 42	Procedimiento de un experimento extenso para la técnica de representación <i>R2</i> . . . . .	70
Figura 43	Diagrama de metodología de visualización de la alineación de secuencias y reconstrucción parcial en 3D. . . . .	71
Figura 44	Tiempo promedio de control de invariantes y construcción de secuencias para una figura 3D. . . . .	74
Figura 45	Tiempo promedio de comparación de dos secuencias representativas. . . . .	75
Figura 46	Figura 3D con simplificaciones de polígonos. . . . .	76
Figura 47	Recuperación de la clase <i>aircraft</i> $\supset$ <i>balloon_vehicle</i> $\supset$ <i>hot_air_balloon</i> . . . . .	78
Figura 48	Recuperación de la clase <i>hourglass</i> . . . . .	78
Figura 49	Recuperación de la clase <i>vehicle</i> $\supset$ <i>car</i> $\supset$ <i>sedan</i> . . . . .	78
Figura 50	Recuperación de la clase <i>aircraft</i> $\supset$ <i>airplane</i> $\supset$ <i>stealth_bomber</i> . . . . .	79
Figura 51	Recuperación de la clase <i>animal</i> $\supset$ <i>quadruped</i> $\supset$ <i>dog</i> . . . . .	79
Figura 52	Recuperación de la clase <i>blade</i> $\supset$ <i>sword</i> . . . . .	79
Figura 53	Recuperación de la clase <i>aircraft</i> $\supset$ <i>airplane</i> $\supset$ <i>fighter_jet</i> . . . . .	80
Figura 54	Recuperación de la clase <i>body_part</i> $\supset$ <i>face</i> . . . . .	80
Figura 55	Recuperación de la clase <i>plant</i> $\supset$ <i>potted_plant</i> . . . . .	80
Figura 56	Gráfica de comparación entre las técnicas propuestas. . . . .	81
Figura 57	Recuperación de la clase <i>furniture</i> $\supset$ <i>seat</i> $\supset$ <i>chair</i> $\supset$ <i>dining_chair</i> . . . . .	82
Figura 58	Recuperación de la clase <i>liquid_container</i> $\supset$ <i>glass_with_stem</i> . . . . .	82
Figura 59	Recuperación de la clase <i>plant</i> $\supset$ <i>bush</i> . . . . .	83

Figura 60	Gráfica de comparación de la combinación de las técnicas de representación propuestas con respecto a las técnicas de manera individual. . . . .	83
Figura 61	Gráfica de comparación de las técnicas de representación propuestas, con respecto a las técnicas estudiadas en Bustos et al. [7].	85
Figura 62	Relación <i>grado de distorsión - distancia</i> . . . . .	86
Figura 63	Relación <i>grado de distorsión - distancia</i> . . . . .	87
Figura 64	Relación <i>grado de distorsión - distancia</i> . . . . .	87
Figura 65	Relación entre <i>distorsiones consecutivas</i> contra <i>diferencia entre distancias</i> . . . . .	88
Figura 66	Relación entre <i>distorsiones consecutivas</i> contra <i>diferencia entre distancias</i> . . . . .	89
Figura 67	Relación entre <i>distorsiones consecutivas</i> contra <i>diferencia entre distancias</i> . . . . .	89
Figura 68	Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D. . . . .	93
Figura 69	Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D. . . . .	94
Figura 70	Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D. . . . .	94
Figura 71	Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D. . . . .	95
Figura 72	Izquierda: Malla de triángulos de la superficie de una parte mecánica. Derecha: Construcción de un árbol <i>AABB</i> . . . . .	108
Figura 73	Contenido de carpeta del archivo ejecutable. . . . .	116
Figura 74	Pestañas con los módulos de la aplicación (1), (2) y (3). . . . .	118
Figura 75	Diagrama de módulos de la aplicación que muestra sus dependencias. . . . .	119
Figura 76	Módulo de análisis y comparación. . . . .	120
Figura 77	Módulo de medidas de desempeño. . . . .	121
Figura 78	Módulo de recuperación. . . . .	123

## ÍNDICE DE TABLAS

---

Tabla 1	Métodos de obtención de descriptores de objetos 3D . . . . .	19
Tabla 2	Tiempo promedio de extracción de medidas por técnica de representación y número de muestras para 2000 modelos . .	74
Tabla 3	Tiempo promedio de comparación con <i>FDTW</i> por técnica de representación y número de muestras para 2000 parejas de secuencias. . . . .	75
Tabla 4	Valores de distancias obtenidos para distintas resoluciones usando la técnica R1 (registro de medidas sobre un cubo). . . . .	76
Tabla 5	Valores de distancias obtenidos para distintas resoluciones usando la técnica R2 (registro de medidas sobre cilindros). . . . .	76
Tabla 6	Valores de distancias obtenidos para distintas resoluciones usando la técnica R3 (registro de medidas sobre una esfera). . . . .	77
Tabla 7	Opciones seleccionadas de experimento para cada técnica de representación. . . . .	77
Tabla 8	Precisión obtenida para distintas proporciones de recuperación	82
Tabla 9	Precisión obtenida para distintos proporciones de recuperación	84
Tabla 10	Precisión promedio obtenida para distintas proporciones de recuperación . . . . .	86
Tabla 11	Medida del <i>vecino más cercano</i> para las técnicas de representación propuestas . . . . .	91
Tabla 12	Medida de <i>medida-E</i> para las técnicas de representación propuestas . . . . .	92
Tabla 13	Ejemplos de listados de archivos. . . . .	117



## INTRODUCCIÓN

---

En este capítulo se plantea el problema y se describen aspectos importantes sobre la tesis, tales como: los objetivos, la motivación, una síntesis de la propuesta y finalmente se describe la estructura del resto del documento.

### 1.1.Planteamiento del problema

Existen grandes problemas en el campo de investigación asociado con el análisis de datos en 3D, también se han planteado problemas específicos de especial interés en relación con la recuperación de información en 3D.

Este trabajo plantea una aproximación a los siguientes problemas específicos:

1. Diseño de una representación que permita derivar una estructura unificada, capaz de mantener componentes descriptivos globales de la figura 3D.
2. Establecer un criterio de comparación con base en el análisis de similitud no lineal.
3. Diseño y prueba experimental de una metodología de comparación, que permita organizar representaciones de figuras 3D que tengan aspectos en común, con el propósito de que contribuyan a la recuperación de dichas figuras.
4. Desarrollo de procesos de análisis espacial.

Dado un conjunto de varias figuras 3D, la aproximación al primer problema es crear representaciones simples de las figuras. Las representaciones deben permitir realizar computacionalmente, comparaciones de similitud bajo un criterio adecuado. Para el segundo problema se propone un criterio de comparación no lineal.

El tercer problema se aborda utilizando el enfoque de medir el esfuerzo de alineación de representaciones de figuras 3D.

Para el cuarto problema se plantea tener un marco de referencia de análisis y un punto de origen de análisis fijo para cada figura 3D.

## 1.2. Objetivo general

Desarrollar una metodología que permita comparar y recuperar figuras tridimensionales, utilizando como medida de comparación el esfuerzo de alineación entre representaciones de figuras 3D.

## 1.3. Objetivos particulares

- Proponer y probar representaciones que se puedan obtener mediante una transformación de 3D a 1D, donde la representación describa globalmente a la figura 3D, permitiendo aplicar técnicas de comparación de secuencias de datos.
- Desarrollar y probar experimentalmente una metodología que permita comparar y recuperar figuras 3D, con la capacidad de incorporar distintas opciones de representación del problema, medidas de distancia y cantidad de datos.
- Generar un conjunto de datos compacto de figuras geométricas primitivas en 3D, con deformaciones asociadas a un valor, en las que se permita observar el comportamiento de comparación a partir de las representaciones propuestas.
- Realizar la validación experimental de desempeño y comportamiento de las técnicas de representación propuestas, utilizando el conjunto de figuras de pruebas del *Princeton Shape Benchmark (PSB)* y sobre el conjunto de figuras con deformaciones graduales, comparando resultados contra los de varias técnicas del estado del arte.

## 1.4. Motivación

La motivación principal de este trabajo surge a partir de una reunión de distintos elementos y factores. Uno de los factores es que se especifica la comparación figuras en 3D para recuperación como un problema abierto de la computación [19, 38]. También se toma la continuidad de trabajos previos [2, 1, 29] y la experiencia en desarrollo de realidad virtual usando herramientas de geometría computacional [29].

La importancia de nuestro trabajo radica en que se proporciona una aproximación al problema de comparación en 3D, que puede utilizarse como base de cualquier sistema de recuperación por contenido. El análisis de datos 3D por contenido es de gran

relevancia, dado que a partir de sus componentes, se pueden extender aplicaciones útiles de comparación para la medicina e industria. La comparación de figuras 3D es un concepto importante en varias aplicaciones que realizan recuperación, reconocimiento o registro de entidades 3D. Este trabajo proporciona una aproximación alternativa al problema de similitud.

### 1.5. Alcances

Los aspectos de semántica asociada al problema de similitud, no son considerados. En cambio, se establece un enfoque cuantitativo y computacional que parte de la observación de la superficie de las figuras.

Los objetos de estudio son figuras construidas a partir de polígonos triangulares. **No tienen asociados color ni textura**; a lo largo del documento se ilustran figuras teñidas con color con el propósito de resaltarlas.

Las figuras en 3D pueden ser geométricas o irregulares, inclusive figuras que no encierran un espacio.

Este trabajo, no es un desarrollo de técnicas para generar descriptores de características, ya que se plantea una representación que considera toda la información del objeto.

No existe un criterio de similitud universalmente aceptado. Existen clasificaciones bajo criterios de forma y funcionalidad de las figuras, pero nótese que dichas clasificaciones tienen problemas de ambigüedad por conceptos semánticos asociados a las figuras.

El enfoque de esta tesis no incorpora la intervención del criterio humano en la comparación. El análisis de similitud es **no supervisado**, por lo tanto tampoco se necesita una muestra de supervisión de entrenamiento.

### 1.6. Panorama de los sistemas de análisis de similitud entre figuras 3D

El desarrollo de bases de datos multimedia, sistemas y componentes de recuperación es cada vez más importante debido a la disponibilidad de una gran cantidad de datos multimedia. Debido a los avances en los campos de adquisición, almacenamiento y disposición de varios formatos multimedia, es muy deseable la aplicación de sistemas de administración de bases de datos que ayuden al control y manejo de esa información. La necesidad es real para contenido de imagen y video. Para el caso de figuras 3D, se espera un desarrollo parecido en un futuro próximo. Las mejoras en tecnología de captura 3D y la disponibilidad de modelos 3D ampliamente distribuidos en Internet,

están contribuyendo rápidamente a la creación de grandes bases de datos de este tipo de información multimedia. El continuo desarrollo en hardware de gráficos ayuda a acelerar el procesamiento de datos 3D lo más posible. Dicha tecnología, se encuentra al alcance de un amplio rango de usuarios potenciales a un bajo costo, en comparación con la situación hace 10 años.

Una de las tareas importantes en un sistema de recuperación de información, es la implementación de algoritmos de análisis de similitud efectivos y eficientes. Los objetos multimedia no pueden consultarse sencillamente en el sentido clásico (búsqueda exacta), ya que la probabilidad de que dos objetos multimedia sean idénticos es muy poca, al menos que sean copias del mismo origen. En lugar de ello, una consulta requiere como respuesta los objetos más similares o bien, se hace una especificación detallada manualmente.

Un enfoque para implementar búsqueda de similitud en una base de datos multimedia, consiste en usar etiquetas de información que describen el contenido del objeto multimedia. Desafortunadamente, este enfoque no es muy factible en repositorios multimedia grandes porque en muchos de los casos, las descripciones textuales tienen que ser generadas manualmente y son difíciles de extraer automáticamente. También son sujeto de estándares adoptados por las personas que las desarrollaron y no se puede codificar toda la información disponible en los objetos multimedia.

Un enfoque más prometedor para la implementación de sistemas de análisis de similitud, es mediante el uso de la información multimedia por sí misma, la cual es llamada análisis con base en el contenido (*content-based analysis*). En este enfoque, los datos multimedia son usados para una consulta de similitud. En la figura 1, se ilustra el concepto de análisis de similitud con base en el contenido 3D. El objeto de análisis es un modelo 3D de una silla, en el cual se espera que el sistema recupere objetos 3D similares.



Figura 1: Ejemplo de búsqueda de similitud en una base de datos de objetos 3D, mostrando un objeto de consulta (q) y un conjunto de respuestas recuperadas posiblemente relevantes (a).

## 1.7.La necesidad del análisis de figuras en 3D

Un aspecto importante en el que se fundamenta el desarrollo del trabajo, parte de la siguiente afirmación.

El mundo en que vivimos y en que nos movemos en lo cotidiano es tridimensional, agregando tiempo (en 3 y 1/2 dimensiones). Nuestro mundo no es una pantalla de computadora con mecanismos complicados para simular un entorno tridimensional en un plano.



Figura 2: Ilustración de interacción humano-objeto en un entorno 3D virtual.

Científicos e ingenieros utilizan expresiones gráficas de su entendimiento de la realidad mediante dibujos o varias escenas. Las descripciones gráficas de la realidad en 3D no es un tema nuevo. Los dibujos en los que se visualiza una perspectiva, datan del periodo del Renacimiento [11]. Las descripciones 3D de la realidad en perspectiva, cambian con respecto a la posición del observador, así que la elaboración de la descripción es tediosa. En 2D se presenta el problema, por ejemplo en los mapas tradicionales se utilizan proyecciones ortogonales del área geográfica, donde las proyecciones muestran una representación 3D muy limitada (ver figura 3).

Con el uso de la tecnología de cómputo, el conocimiento de la realidad puede ser transferido directamente en una figura digital 3D, mediante un proceso conocido como modelado en 3D. Una descripción de la realidad en 3D es independiente de la posición.

$CAD^1$  es una herramienta típica de gráficos por computadora para modelado en 3D, usada en diseño de autos, maquinaria, aeronaves y naves espaciales, en la industria de

1 Del inglés *Computer-Aided Design*.

la construcción y arquitectura. *CAD* se enfoca en el aspecto geométrico de la figura y su visualización en 3D. *CAD* es cuestionable, si la tarea consiste en comparar las entidades en 3D y encontrar similitud entre los objetos.

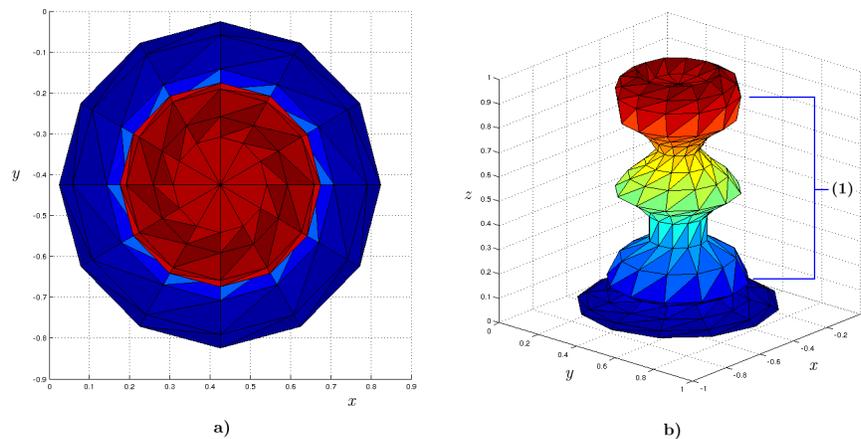


Figura 3: a) Representación 2D  $(x, y)$  de una figura 3D. b) Representación en 3D  $(x, y, z)$  donde en (1) se muestran los múltiples valores de  $z$  que no se pueden registrar en 2D.

*CAD* puede proporcionar una vista desde una perspectiva deseada con omisión de superficies ocultas, también calcula el volumen, área de la superficie, centro de masa y una agrupación temática ya diseñada. El problema de la similitud consiste en encontrar una correspondencia entre las entidades a estudiar, para identificar una relación (o parecido) entre ellas mediante un juicio. *CAD* no incorpora alguna función con las características mencionadas.

### 1.8. Análisis de similitud de figuras 3D en distintos campos de aplicación

El problema de análisis de similitud en figuras 3D se origina en ciertos campos.

Ejemplos de dominios de problemas incluyen:

Diseño Asistido por Computadora y Fabricación Asistida por Computadora (*CAD/CAM<sup>a</sup>*), Realidad Virtual (*VR<sup>b</sup>*), Medicina, Biología Molecular, Investigación Forense, aplicaciones militares, industria del vestido y calzado e industria del entretenimiento.

<sup>a</sup> Del inglés *Computer-Aided Manufacturing*.

<sup>b</sup> Del inglés *Virtual Reality*.

A continuación se hace una descripción más específica.

- En medicina, la detección de deformaciones de órganos similares, se puede utilizar con fines de diagnóstico. Por ejemplo, en la teoría médica actual de la epilepsia infantil, se supone que un desarrollo irregular de una parte específica del cerebro, llamada hipocampo, es la razón para la epilepsia. Varios estudios muestran que el tamaño y la forma de la deformación del hipocampo pueden indicar el defecto, esto se utiliza para decidir si se debe o no eliminar el hipocampo por medio de cirugía del cerebro. El análisis de similitud en una base de datos de modelos 3D de hipocampos, puede apoyar en el proceso de decisión y ayudar a evitar cirugías innecesarias [21].
- La clasificación estructural es una tarea básica en biología molecular. Esta clasificación puede ser exitosamente abordada por el análisis de similitud, donde las proteínas y las moléculas son modeladas como objetos 3D. Las imprecisiones en el modelo de la molécula en 3D debidas a la medición, el muestreo, el redondeo numérico y los pequeños errores de desplazamiento, deben ser manejadas con congruencia [3].
- Desde hace varios años, muchos centros de pronóstico del tiempo incluyen las predicciones de polen en sus informes, con el fin de advertir y ayudar a las personas alérgicas a los diferentes tipos de polen. Ronneberger et al. [31], desarrollaron un sistema de reconocimiento de patrones que clasifica el polen de los datos volumétricos en 3D, adquiridos mediante un microscopio láser de foco común. La estructura 3D del polen se puede extraer. Invariantes de la escala de grises proporcionan componentes de los vectores de características para su clasificación.
- Institutos forenses en todo el mundo deben hacer frente a la compleja tarea de identificación de tabletas y píldoras ilícitas de drogas. En conjunción con el análisis químico, las características físicas de la píldora (por ejemplo, la forma e impresión) se utilizan en el proceso de identificación. La forma y los métodos de reconocimiento de pie de imprenta incluyen la caja de delimitación del objeto (*bounding box*) y la forma con base en el contorno, elementos que se pueden utilizar para definir un modelo 3D de la píldora. Se puede utilizar un sistema de búsqueda de similitud para informar similitudes entre la píldora estudiada y los modelos de comprimidos conocidos como ilícitos [14].
- Para apoyar a las herramientas *CAD*, se propone utilizar una base de datos de objetos 3D. Dado que un objeto puede modelar con exactitud la geometría de un objeto, toda la información necesaria al respecto se obtiene a partir del modelo 3D, por ejemplo, cualquier vista 2D del objeto. Las herramientas de *CAD* tienen muchas aplicaciones en el diseño industrial. Por ejemplo, las piezas estándar en

una empresa de fabricación pueden ser modeladas como objetos 3D. Cuando se diseña un nuevo producto, puede estar compuesto por muchas piezas pequeñas, que encajan entre sí para formar el producto. Si alguna de estas partes es similar a una de las piezas estándar, la sustitución de la pieza original con la parte estándar, puede reducir los costos de producción.

- Otra aplicación industrial es el problema de los zapatos que no se adecúan al pie de un cliente [27]. Se propone generar un modelo del pie del cliente utilizando una herramienta de digitalización en 3D. A continuación, se realiza un análisis de similitud para descartar los modelos de ajuste más improbables de acuerdo con los pies del cliente. Los candidatos restantes son inspeccionados para determinar la mejor coincidencia. Lo mismo puede aplicarse en la industria del vestido.
- Se supone un sistema de detección amigo/enemigo para determinar si un objeto (por ejemplo, un avión o un tanque) se considera amistoso u hostil, con base en su clasificación geométrica. Este tipo de sistema, tiene aplicaciones obvias en la defensa militar. Una forma de implementar un sistema de detección consiste en almacenar los modelos en 3D de los objetos amistosos u hostiles, el sistema determina la clasificación de un objeto usando como base la definición de la similitud y la base de datos de objetos de referencia. Estas decisiones son críticas y deben alcanzarse en tiempo real, los requisitos de alta eficiencia y eficacia del sistema de recuperación, son dominantes para esta aplicación.
- Los productores de cine y videojuegos hacen un uso intensivo de modelos 3D para aumentar el realismo en las aplicaciones de entretenimiento. La reutilización y adaptación de los objetos en 3D mediante el análisis de similitud en bases de datos existentes, es un enfoque prometedor para reducir los costos de producción.

Dado que los objetos 3D se utilizan en diversos dominios de aplicación, se han desarrollado diferentes formas de representación de objetos, de manipulación y de presentación. En el dominio de *CAD*, los objetos se construyen a menudo mediante la unión de pequeñas superficies paramétricas que son editadas por personal técnico. Además, se emplean frecuentemente técnicas de geometría sólida constructiva, donde los objetos complejos se modelan mediante la composición de primitivas. Los dispositivos de adquisición de 3D, suelen producir aproximaciones de objetos mediante voxelización (por ejemplo, escáneres de tomografía computarizada) o nubes de puntos 3D (por ejemplo, escáneres de detección de fase de luz estructurada). Existen otras formas de representación, como volúmenes de barrido o gramáticas en 3D.

Probablemente, la representación más utilizada para objetos 3D, es la de malla de polígonos, generalmente triángulos. En [8] se presenta un estudio sobre las formas de

representación más importantes. Para la recuperación de 3D, todos estos formatos sirven como entrada a una consulta de similitud. Si se encuentra disponible, se explota la información de los datos de la geometría pura, por ejemplo, los datos estructurales que se incluyen en una representación *VRML*.

Hasta el momento, muchos métodos de análisis de similitud que se presentan en la literatura, se basan en polígonos triangulares con la opción de conversión a otras formas de representación.

Se ha hecho amplia investigación durante varias décadas en la descripción de figuras y el establecimiento de relaciones de similitud entre la figura geométrica y visual en los campos de visión por computadora, análisis de figuras y geometría computacional. En visión por computadora, el método usual es tratar de reconocer objetos en una escena mediante la segmentación de una imagen en 2D y luego combinar estos segmentos en un conjunto de objetos de referencia conocido a priori. Los problemas específicos implican el cumplimiento de invariantes con respecto a condiciones de iluminación, vista en perspectiva, desorden y oclusión. Desde la perspectiva de base de datos, se supone que los objetos ya se han descrito y pueden ser utilizados directamente. Los problemas surgen en las formas de representaciones de objetos heterogéneos (no se puede asegurar una cantidad de diez propiedades de los objetos 3D), y el problema de decisión de por sí es difícil. Para la implementación de un sistema de análisis de figuras 3D, se tienen que resolver las siguientes preguntas: ¿Cuál es la noción de similitud?, ¿Cuánta tolerancia es sostenible en un contexto de aplicación determinado? y ¿qué tamaño se requiere del conjunto respuesta?. Además, la perspectiva de base de datos se refiere a un número grande de objetos, por lo tanto la atención se centra no sólo en los métodos precisos, sino también en métodos rápidos, que ofrezcan un tiempo de procesamiento adecuado, incluso en repositorios de objetos de gran tamaño.

### 1.9. Problemas asociados con el análisis de entidades en 3D

Este trabajo se centra en desarrollar y probar experimentalmente, una propuesta de comparación y recuperación de figuras 3D, pero existen otros temas que están en el comienzo de su desarrollo. Es posible indicar algunos de los problemas abiertos actuales.

Para el desarrollo de un sistema de información 3D computacional, se deben integrar los componentes necesarios. Dependiendo de las entidades 3D que se manejan, se requiere una solución o aproximación de los siguientes problemas:

1. Diseño de la entidad tridimensional.

- Proporcionar un modelo de datos compacto o esquema que permita derivar una estructura unificada, capaz de mantener todos los componentes de la representación geométrica de entidades del mundo real.
2. Construcción de la entidad tridimensional.
    - Desarrollo de medidas y métodos apropiados para adquisición de datos en 3D.
    - Desarrollo de un método de estructuración de datos que unifique datos de varias entradas de múltiples orígenes, capaz de ser administrado por un único sistema.
    - Diseño de metodologías de comparación que permitan organizar representaciones de entidades del mundo real que tengan aspectos en común.
    - Resolver el problema de datos falsos originados por las discrepancias entre conjuntos de datos durante un proceso de integración, convirtiendo los datos falsos en datos de calidad, útiles para un usuario final.
  3. Uso de la entidad tridimensional.
    - Uso de componentes existentes como los son lo que se usan para manipular datos en 2D y la preparación de los componentes para una futura incorporación de datos de mayor dimensión con el propósito de disminuir costos en la adquisición de datos.
    - Desarrollo de operadores espaciales y de funciones de análisis espacial.
    - Desarrollo de visualización gráfica con capacidad de manipulación, permitiendo la selección de puntos de observación.
    - Diseño de información de aspectos no espaciales como lo son nombre, color, temperatura, generalización, etc., así como su presentación.
    - Diseño de interfaces de usuario y un lenguaje de consulta permitiendo a los usuarios acceder a las representaciones de un conjunto de datos.
    - Diseño de una estructura de indexación espacial que acelere las operaciones de almacenamiento y recuperación de datos.
    - Desarrollo de herramientas para navegar entre distintas representaciones almacenadas en diferentes bases de datos, en diferentes lugares físicos y diferentes plataformas de cómputo.
  4. Mantenimiento de la entidad tridimensional.
    - Diseño de procedimientos de actualización que incluyen reglas de consistencia para asegurar la integridad de la base de datos, especialmente durante un proceso de actualización.

## 1.10. Organización del documento

En el capítulo 2 se resume el estado del arte de la comparación y recuperación de figuras 3D y se proporciona una breve descripción de la propuesta de esta tesis.

En el capítulo 3 se describe el marco conceptual del problema de representación de figuras 3D, conceptos de similitud y comparación, definiciones de técnicas de comparación y conceptos de medidas de recuperación de información aplicados a figuras en 3D.

En el capítulo 4 se desarrollan todos los aspectos metodológicos, comenzando con la definición de los objetos con los que se trabaja y una serie de procedimientos para obtener representaciones adecuadas del problema. Los aspectos de comparación tienen su origen en los conceptos del capítulo 3, los cuales se adaptan a una técnica computacional de comparación. Se menciona la utilización de la medida de desempeño, condiciones experimentales y aspectos de la implementación.

En el capítulo 5 se exponen los resultados experimentales, donde se describe la interpretación y observaciones de cada experimento. Los resultados permiten mostrar el potencial de las técnicas de análisis de similitud, desde el punto de vista del enfoque así como desde la perspectiva de implementación.

En el capítulo 6 se discuten las conclusiones, las contribuciones y el trabajo futuro.



## ESTADO DEL ARTE

El capítulo comprende una síntesis del estado del arte que sirve para colocar en contexto a la propuesta de este trabajo.

### 2.1. Antecedentes

El problema de similitud entre figuras 3D (también denominadas objetos 3D o modelos 3D), ha sido desarrollado por varios investigadores. Sin embargo las ideas principales de los primeros trabajos han permanecido.

Los primeros trabajos que se aproximaron al problema de similitud entre figuras geométricas fueron Roberts [30] y Guzmán [17], ver figura 4.

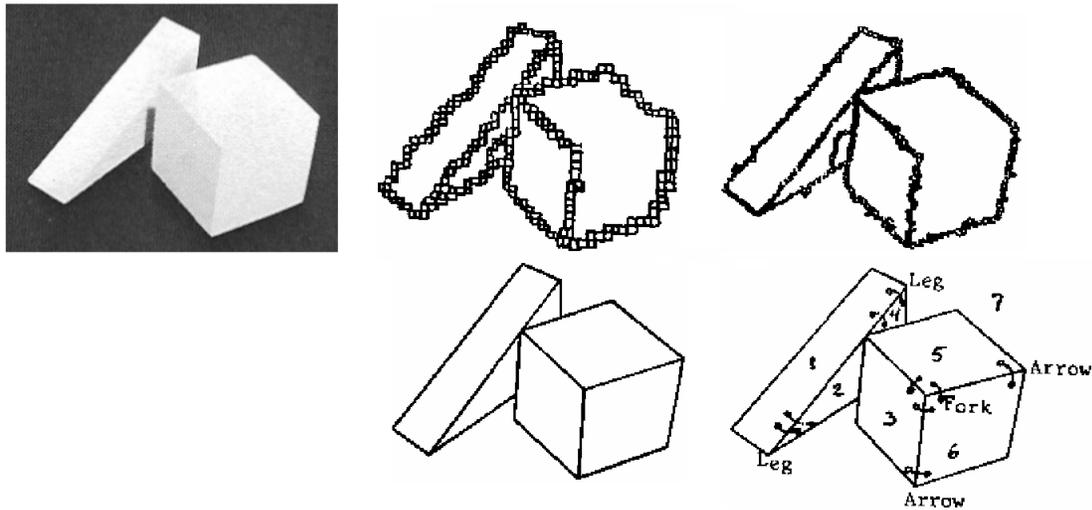


Figura 4: Reconocimiento de figuras 3D a partir de primitivas [17].

Guzmán introdujo la siguiente idea; existen elementos base, *atòmum* (átomo), cosas indivisibles, que constituyen un patrón, suficiente para describir a una figura. Una vez que se tiene dicha caracterización de estas figuras, es posible determinar su clase.

A partir de estos trabajos se refuerza la idea de dividir y vencer en el área de reconocimiento de patrones, así como la idea de definir ciertas premisas o características relevantes de un objeto dado.

Una cantidad considerable del estudio actual, tiene como principio el encontrar aspectos relevantes de la entidad de estudio, mediante la selección, extracción y construcción de características que en conjunto conforman un descriptor.

La clasificación de métodos para la descripción de datos 3D, puede determinarse utilizando diferentes criterios. Una categorización popular de descriptores, en el campo de funciones de análisis de figuras, según [25], corresponde con los siguientes puntos.

- Los descriptores puede ser construidos tomando como base la superficie de un objeto o con base en las propiedades del interior. La curvatura del contorno es un ejemplo del primer tipo de descriptor, mientras que las medidas para la distribución de la masa objeto son del segundo tipo de descripción.
- Dependiendo del tipo de descriptor de objeto, es posible producir un vector de valores escalares mediante métodos numéricos que representan el objeto, mientras que los métodos espaciales utilizan otros medios, por ejemplo, una secuencia de formas primitivas que se aproximan a la forma original o un gráfico que representa la estructura del objeto.
- Los descriptores de preservación conservan la información completa de objetos que permite la reconstrucción sin pérdida del objeto original. Los descriptores que no son de preservación, descartan cierta cantidad de información de los objetos, por lo general conserva sólo la parte de la información que se considera el más importante.

Una categorización más específica entre descriptores con respecto a los modelos 3D, se puede hacer en función del tipo de información contenida en el modelo, por ejemplo, la geometría, color, propiedades de textura, la distribución de masa y material. Por lo general, la geometría es considerada como más importante para los objetos 3D, por lo tanto todos los descriptores presentados en lo siguiente, usan la geometría como única entrada. Esto se debe a que la geometría siempre se especifica en los modelos, mientras que otras características son más dependientes de la aplicación y no se pueden suponer que están presentes arbitrariamente en conjuntos de datos de objetos 3D.

Por otra parte, se podrían diferenciar descriptores con respecto a las restricciones de integridad asumidas para los modelos, por ejemplo, la propiedad de sólido, la orientación consistente de imagen o el tipo de entrada que se asume (malla de polígonos, voxelización, conjunto de imágenes, etc.) La mayoría de los métodos existentes son flexibles, ya que permiten inconsistencias del modelo y asumen triangulaciones. Por supuesto, la flexibilidad de la descripción depende de los supuestos del modelo, la información adicional es útil para obtener más opciones para el diseño de los descriptores.

Una forma de clasificar los métodos de recuperación de modelos 3D se puede encontrar en [7]. En esta clasificación, la extracción de descriptores de forma se considera como un proceso de varias etapas (ver figura 5). En el proceso, un objeto 3D dado, generalmente representado por una malla de polígonos, en primera instancia, se preprocesa para aproximar la invariante requerida y las propiedades de robustez. Entonces, se genera una abstracción del objeto de manera que sea de tipo superficie, volumétrica o capturada por una o varias imágenes 2D. A continuación, se puede realizar un análisis numérico de forma, a partir del resultado se extraen los vectores de características.

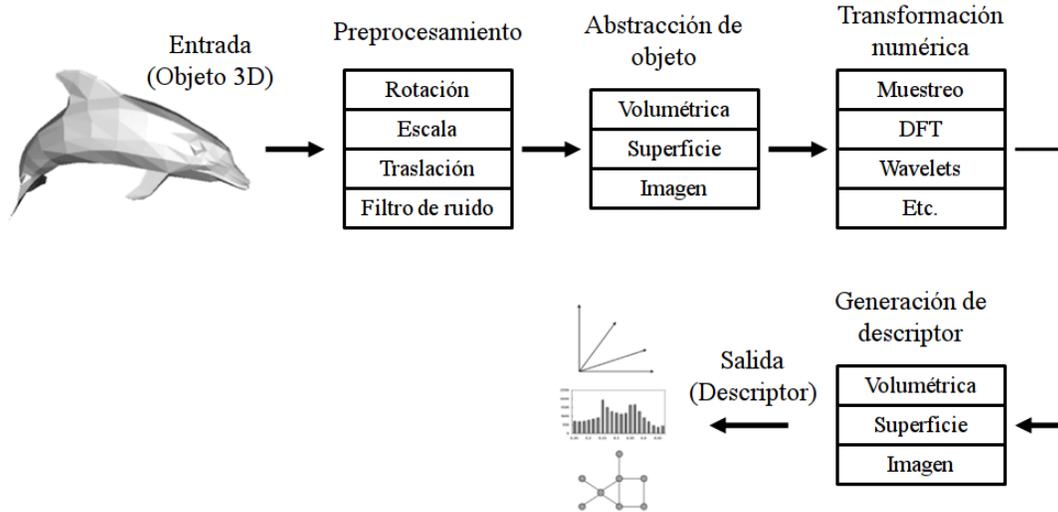


Figura 5: Modelo del proceso de extracción del descriptor.

De manera breve se muestran los pasos básicos de la figura 5. Sin pérdida de generalización se asume que los objetos 3D se representan como una malla de polígonos.

1. Preprocesamiento. Si el descriptor lo requiere, el modelo 3D se preprocesa para el control de invariantes de rotación ( $R$ ), traslación ( $T$ ) y escala ( $E$ ).

2. Tipo de abstracción del objeto. Hay tres tipos diferentes de abstracción del objeto: superficie, volumétrica e imagen. Un ejemplo de descriptor de superficie es la estadística de la curvatura del objeto. Las medidas de distribución de masa en 3D, usan como base descriptores de momento. Dichas medidas, pertenecen al tipo de abstracción volumétrica. Otra forma de capturar características de una malla es proyectarla en una imagen o varias imágenes, produciendo imágenes que corresponden con mapas de profundidad, siluetas y demás, de donde se puede obtener descriptores. A esta forma se le denomina abstracción de objeto con base en imagen.
3. Transformación numérica. Las características principales de una malla de polígonos se pueden capturar numéricamente utilizando distintos métodos. Por ejemplo, arreglos de voxeles y arreglos de imágenes se transforman utilizando wavelets o bien es posible muestrear la superficie. Otras transformaciones numéricas incluyen armónicas esféricas (*SH*), ajuste de curva y transformada discreta de Fourier (*DFT*). Dichas transformaciones utilizan representaciones numéricas del objeto.
4. Descriptores generados. En esta etapa se genera un descriptor final que puede pertenecer a una de tres clases:
  - a) Vectores de características (*FV*), que consiste en elementos en un espacio vectorial equipado con una métrica adecuada. Usualmente el espacio vectorial Euclidiano se toma con dimensiones que fácilmente puede llegar a varios cientos.
  - b) En el enfoque estadístico, los objetos 3D son inspeccionados buscando características específicas, las cuales pueden ser acumuladas y representadas en un histograma. Por ejemplo, en los casos más simples, las cantidades sumadas de área en regiones volumétricas específicas o, más complejo, se puede recolectar estadísticas de las distancias entre pares de puntos seleccionados aleatoriamente a partir del objeto.
  - c) La tercera categoría se adapta a la descripción estructural de objetos 3D, que puede ser representada mediante un grafo [32, 18]. Un grafo puede representar más fácilmente la estructura de un objeto y los elementos que la componen. Puede descomponerse en partes significativas, por ejemplo, el torso y extremidades de los modelos de animales.

La tabla 1 muestra algunos algoritmos importantes del estado del arte, pasos de preprocesamiento, tipo de abstracción de objeto, transformación numérica aplicada y tipo de descriptor obtenido. Con la información anterior, los descriptores se organizan de la siguiente manera.

- **Estadísticos.** Los descriptores estadísticos reflejan propiedades básicas de los objetos, como el número de vértices y polígonos, el área de la superficie, volumen y momentos estadísticos. En algunos dominios de aplicación, medidas simples como de extensión espacial o medidas volumétricas pueden ser suficientes para recuperar objetos de interés.
- **Descriptores con base en la extensión.** Estos métodos construyen el descriptor del objeto a partir de características muestreadas a lo largo de una dirección espacial desde el centro del objeto.
- **Descriptores con base en el volumen.** Determinan características del objeto mediante representaciones volumétricas obtenidas por discretización de la superficie del objeto en arreglos de voxels, o bien, usando los modelos que ya se encuentran voxelizados.
- **Geometría de la superficie.** Estos descriptores se basan en obtener las características de la superficie del modelo.
- **Descriptores con base en las imágenes.** El problema de similitud en 3D se reduce a un problema de similitud de imágenes, comparando proyecciones en 2D de los modelos 3D.

Existe un trabajo considerable de visión por computadora y análisis de figuras, áreas que desarrollan descriptores de modelos 3D avanzados y se fundamentan en el análisis estructural de la figura, complementado con el uso de estructuras de datos y funciones de distancia. En principio, dichos descriptores se pueden usar para la implementación de algoritmos de análisis de similitud entre objetos 3D.

La figura 6 sintetiza la organización de los métodos de la tabla 1.

## 2.2.Descripción de la propuesta

La idea general para realizar la comparación y recuperación de figuras en 3D se constituye de una etapa de preprocesamiento, que consiste en el control de invariantes de un modelo para su análisis. Luego del preprocesamiento, el modelo es analizado para generar una nueva representación útil para la comparación; dicha representación es una secuencia numérica.

Lo siguiente es realizar la comparación calculando la distancia de *Fast Dynamic Time Warping* entre dos secuencias, donde cada secuencia representa a un modelo 3D. La distancia medida es un valor, considerado como indicador numérico del esfuerzo de alineación entre los modelos 3D originales.

Al terminar todas las comparaciones se tiene una matriz de distancias por cada técnica de representación.

A partir de las matrices de distancias, es posible realizar la recuperación de cualquier figura y obtener una lista ordenada de menor a mayor con respecto al valor de la distancia.

Mediante una serie de recuperaciones es posible observar características particulares de cada técnica en la comparación. También es posible obtener indicadores de desempeño de la recuperación.

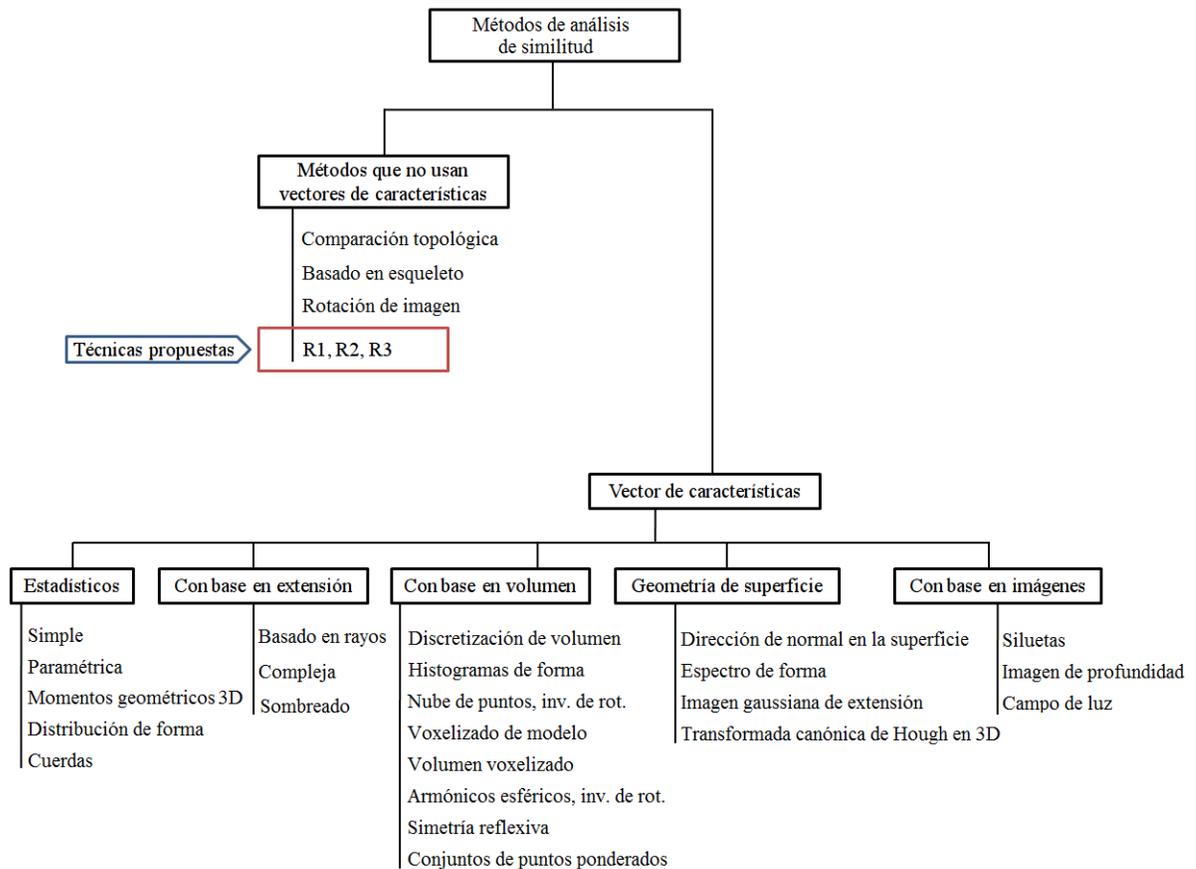


Figura 6: Categorías de técnicas de recuperación de objetos 3D.

Tabla 1: Métodos de obtención de descriptores de objetos 3D

Nombre de descriptor	Prep.	Extracción	Trans. Num.	Tipo
Estadística simple	RTE	Volum.	Ninguna	VC
Estadística paramétrica	RTE	Superficie	Muestreo	VC
Momentos geom. en 3D	RTE	Superficie	Muestreo	VC
Momentos de rayo	RTE	Superficie	Muestreo	VC
Distribución de forma (D2)	Ninguna	Superficie	Muestreo	Hist.
Basado en cuerdas	RT	Superficie	Muestreo	Hist.
Basado en rayos con AE	RTE	Imagen	Muest.+AE	VC
Sombreado con AE	RTE	Imagen	Muest.+AE	VC
Complejo con AE	RTE	Imagen	Muest.+AE	VC
Basado en extensión de rayo	RTE	Imagen	Muest.+AE	VC
Histograma de forma	RTE	Volum.	Muestreo	Hist.
Nube de puntos, inv. de rot.	RTE	Volum.	Muestreo	Hist.
Voxel	RTE	Volum.	Ninguna	Hist.
3D <i>DFT</i>	RTE	Volum.	3D <i>DFT</i>	VC
Volumen voxelizado	RTE	Volum.	Ondeletas	VC
Volumen	RTE	Volum.	Ninguna	VC
Cobertura de secuencias	RTE	Volum.	Ninguna	VC
Armónicos esféricos, inv. de rot.	TE	Volum.	Muest.+AE	VC
Simetría reflexiva	TE	Volum.	Muestreo	VC
Conjunto de puntos ponderados	RTE	Volum.	Ninguna	Hist.
Dir. de normal en la superficie	Ninguno	Superficie	Ninguna	Hist.
Espectro de forma	Ninguno	Superficie	Ajuste de curva	Hist.
Imagen gaussiana de extensión	R	Superficie	Ninguna	Hist.
3D <i>HT</i> basada en forma	Ninguno	Superficie	Muestreo	VC
Silueta	RTE	Imagen	Muest.+AE	VC
Imagen de profundidad	RTE	Imagen	2D <i>DFT</i>	VC
Campo de luz	TE	Imagen	<i>DFT</i> , Zernike	VC
Comparación topológica	Ninguno	Superficie	Muestreo	Grafo
Esqueletización	Ninguno	Volum.	Trans. dist., <i>clustering</i>	Grafo
Rotación de imagen	Ninguno	Superficie	Segmentación	Hist. en 2D

Notas: RTE - (R) rotación, (T) traslación, (E) escala, AE - armónicos esféricos, HT - transformada de *Hough*, VC - vector de características.



Se definen varios conceptos importantes y métodos de utilidad para el seguimiento del resto de los capítulos.

### 3.1. Representación de figuras 3D

En principio, los objetos 3D pueden representar información compleja. Las dificultades que se originan en la definición de similitud entre objetos espaciales son comparables con las que corresponden para la misma tarea aplicada a imágenes en 2D. Las propiedades geométricas de los objetos 3D, pueden ser proporcionadas por cierta cantidad de formatos de representación. Dependiendo del formato, es posible que se especifiquen propiedades como superficie y material. La resolución de los objetos puede establecerse de manera arbitraria. Dado que no hay una teoría que fundamente universalmente la descripción de figuras 3D o una forma de compararlas directamente, los datos 3D son transformados de alguna manera para obtener representaciones numéricas para indexación y recuperación.

También nos referimos a estas representaciones como secuencias representativas. La idea básica, es extraer datos numéricos que describan objetos bajo algún aspecto geométrico e inferir la similitud de los modelos a partir de la distancia de las representaciones numéricas en algún espacio métrico.

La noción de similitud se deriva del contexto de la aplicación, que define los aspectos de similitud relevantes. Las relaciones de similitud obtenidas de esta manera, son de interés para un paradigma específico de similitud y puede no reflejar similitud en un contexto diferente de aplicación.

La propuesta de representación por secuencia de medidas tiene ventajas en comparación con otras técnicas en términos de implementación. Por ejemplo, se puede calcular la distancia *Minkowski* de manera eficiente. También, se pueden utilizar métodos de acceso espacial [6] o métodos de acceso a partir de métricas [10] para indexar los vectores de

medidas obtenidos. Estas ventajas hacen de una representación, un buen candidato para la implementación de un módulo de análisis de similitud entre objetos 3D.

### 3.1.1. Requerimientos de la representación

Se pueden definir varios requerimientos para el diseño de una representación. Una buena representación debe de considerar condiciones invariantes de rotación, escala y traslación. Esto es, un sistema de análisis de similitud debe permitir recuperar geoméricamente objetos 3D similares con diferentes posiciones y orientación. Una representación también debe ser robusta a pequeños cambios geométricos y del nivel de detalle. Si tales transformaciones son aplicadas a los objetos, el indicador de similitud no debe de cambiar abruptamente.

Las condiciones invariantes se pueden obtener por medio de un procesamiento de control, en el cual se transforma a un objeto de tal manera que sea colocado en un marco de referencia de análisis Bustos et al. [7]. Otras representaciones obtienen condiciones invariantes de manera implícita.

En la figura 7 (izquierda), se ilustra las diferentes posiciones de un objeto de un auto en 3D, así que la representación debe ser la misma en todos los casos. A la derecha de la figura 7, se muestra el mismo objeto en cuatro niveles de resolución.

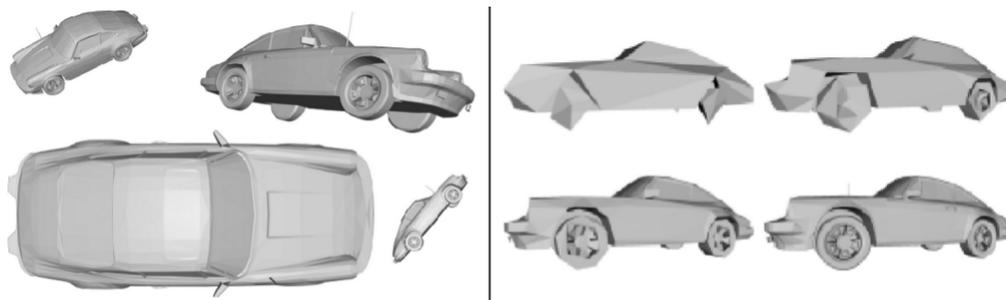


Figura 7: Un objeto 3D con orientación y escala diferentes (izquierda) y también representado en varios niveles de detalle (derecha).

### 3.2. Conceptos de comparación

Las mejores técnicas son aquellas que nos permiten emplear la mayor cantidad de información disponible sobre las entidades involucradas. Para el problema específico de comparación de datos en 3D, se preferirán aquellos métodos que trabajen con toda la información contenida en cada uno de los elementos de los conjuntos de vértices y polígonos tridimensionales.

### 3.2.1. El problema de la similitud y la comparación

El problema de la similitud es complejo de abordar en las ciencias de la computación. Las aproximaciones al problema conceptualizan relaciones de semejanza entre objetos; las relaciones pueden ser observadas en un espacio de representación mediante una métrica. Aunque la aproximación es metódica, de acuerdo con Ortega-González [28], el ser humano emplea múltiples criterios y mecanismos para observar dichas relaciones, inclusive sin rigor alguno, que resultan prácticamente desconocidos para la ciencia.

Relacionar entidades es un aspecto fundamental de la inteligencia humana. Los resultados de un criterio de relación suelen ser diferentes para distintas personas. Es por ello que es importante mencionar, que no existen criterios universales para determinar relaciones de semejanza.

En [13] se presenta la estructura conceptual del problema de similitud. En el paradigma clásico (ver figura 8), se incluyen etapas que abarcan los procesos de preprocesamiento, selección extracción de características, normalización, entre otros. En todos, se requiere conocer a priori, las características descriptivas de las entidades a comparar, dejando de lado el resto de la información.

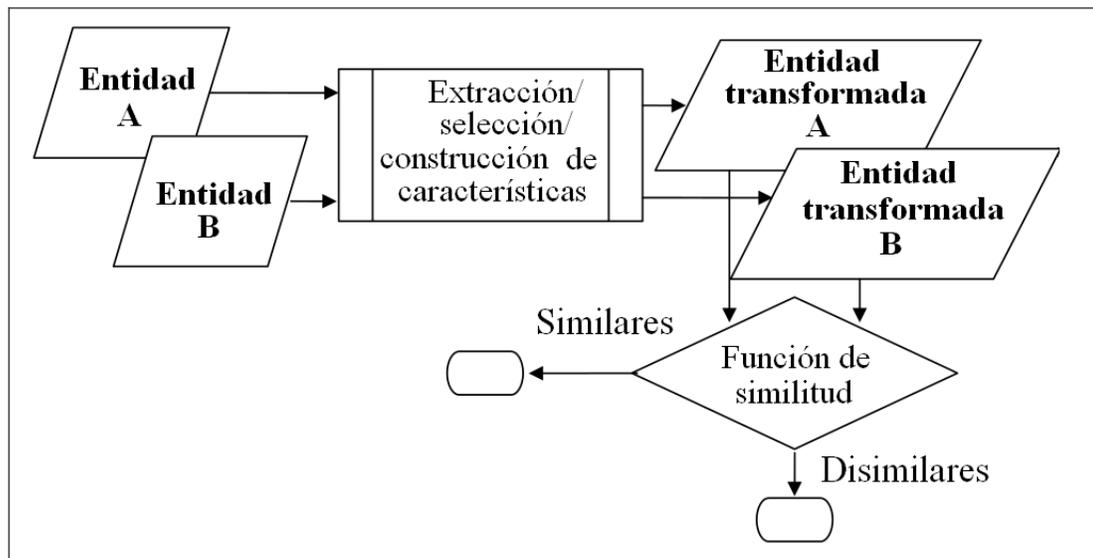


Figura 8: Paradigma clásico para el problema de similitud entre objetos. La función de similitud se ilustra como un proceso de toma de decisión binario.

El paradigma moderno (ver figura 9), establece los procesos previos a la función de semejanza del paradigma clásico como opcionales, permitiendo comparar los objetos como

un todo, donde el criterio de comparación es un proceso para asignar números en base a alguna regla.

El problema de similitud puede ser abordada desde dos grandes espacios dentro del campo de la computación (ver figura 10); en la cual se han realizado grandes esfuerzos para encontrar criterios de comparación entre un par de secuencias de valores, por ejemplo, en el caso discreto, se han propuesto diferentes índices de similitud como la distancia de *Jaccar*, *Rusell* y *Rao*, *Hamming* [12], entre otros. En el caso continuo se tiene la distancia *Euclidiana*, *City Block* y *Minkowski*. Este trabajo se enfoca en el segundo caso.

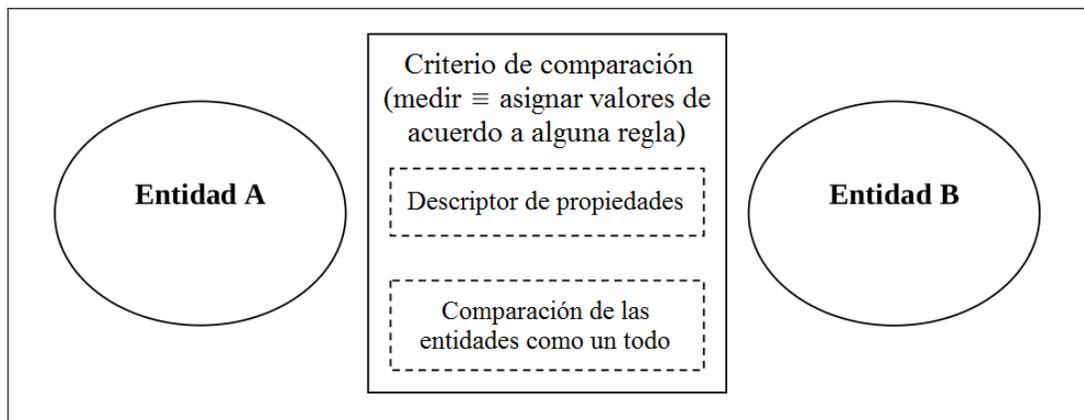


Figura 9: Estructura conceptual del problema de similitud. Las entidades a comparar deben de ser cuantificables.

Considerando el enfoque de utilizar representaciones numéricas, se puede definir el criterio de comparación para este trabajo.

El principio de comparación parte de la idea de realizar la comparación geométrica directa o bajo una representación equivalente, donde la medida de similitud está dada por la cantidad de esfuerzo requerido para que un objeto pueda ser transformado en otro.

### 3.2.1.1.El problema de similitud entre figuras 3D

A diferencia de la comparación y análisis de similitud en imágenes, en modelos 3D, no existe un concepto matemático de similitud. Es posible construir una aproximación mediante el concepto de *cubo de datos*, muy utilizado en el campo de minería de datos. Medir la similitud entre cubos de datos puede requerir un gasto computacional enorme y complejo, debido a que la combinación de valores que puede contener un cubo es enorme.

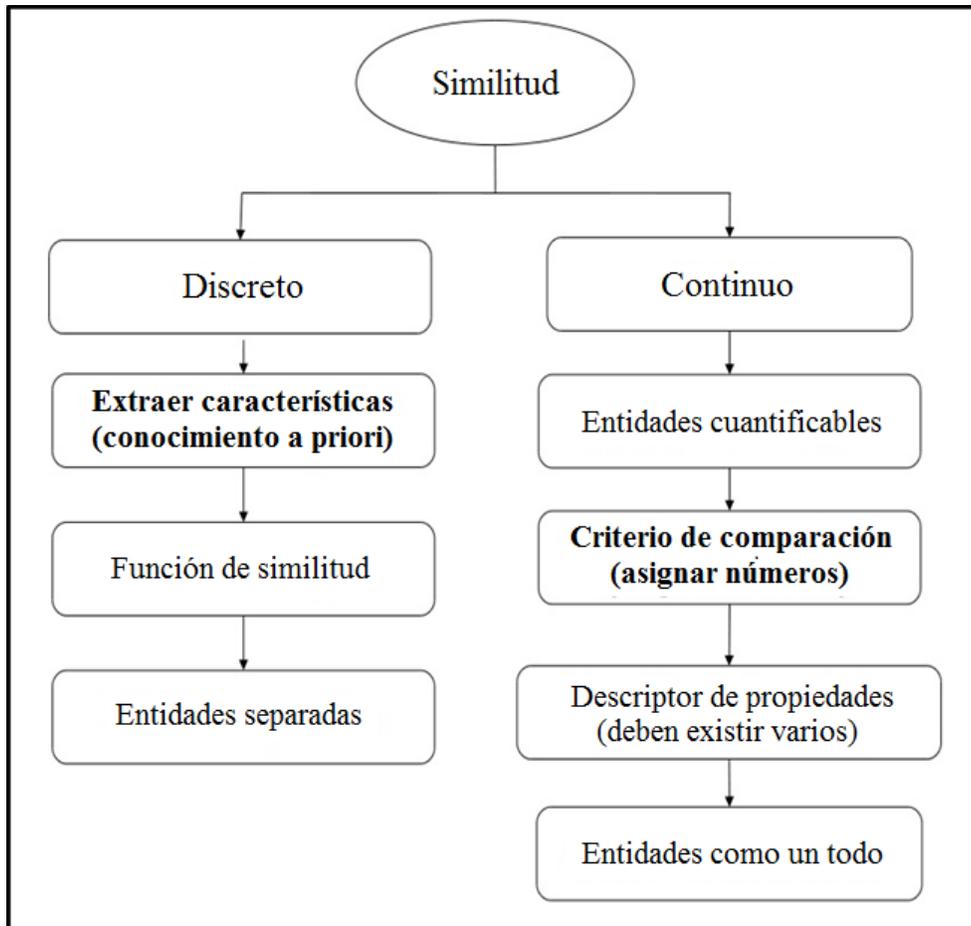


Figura 10: Similitud por atributos y por valores numéricos.

Por ejemplo, para un modelo resultado de una voxelización, de muy baja resolución, con un tamaño de  $10 \times 10 \times 10$  (ver figura 11-(a)), la combinación de valores de presencia y ausencia de volumen (1 y 0 respectivamente) en cada celda resulta ser de  $((2^{10})^{10})^{10} = 2^{1000}$ . Comparando directamente el cubo de voxes de una figura con el cubo de voxes de otra figura (ver figura 11), se cuantifica el esfuerzo de encontrar la distancia mínima entre el contenido de ambos cubos.

Si se considera medir la alineación elástica entre voxeles de cubos para encontrar la distancia mínima, el problema se vuelve una tarea realmente compleja. Como referencia se tiene que el esfuerzo medido como resultado de una alineación elástica en 2D es un problema NP-Completo [23]; es de esperarse que el problema en 3D, sea mucho más complejo.

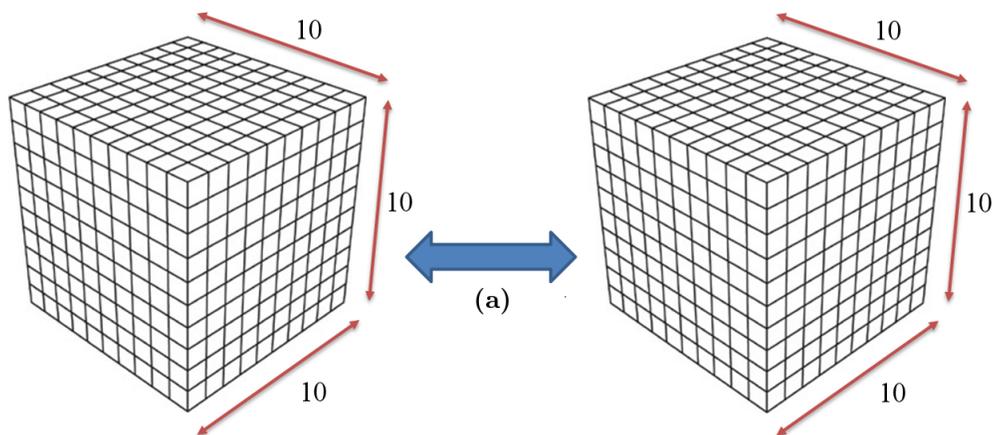


Figura 11: Cubos de voxeles que ejemplifican la cantidad de combinaciones de valores de ceros y unos en la complejidad de su comparación.

### 3.2.2. Métricas de similitud

La similitud se puede concebir como un valor numérico que refleja la fuerza o intensidad de la relación entre dos entidades a estudiar. Varios de los conceptos de esta sección fueron obtenidos de [33].

La distancia es la medida de la disimilitud entre los objetos. La disimilitud se refiere a la diferencia entre dos objetos con base en varias características comparadas. Usualmente, las características de los objetos son representadas como coordenadas dentro de un espacio geométrico. Existen varios tipos de métricas e indicadores para distancias y similitudes, cada una con sus características propias. Una métrica de similitud es una función que asocia un valor numérico entre un par de descriptores (usualmente vectores o secuencias, ambos numéricos). Es posible utilizar una métrica de disimilitud para expresar la similitud, esto es, si la disimilitud es una función de la distancia que entrega un valor con magnitud pequeña, representa una relación de similitud más grande (ver figura 12).

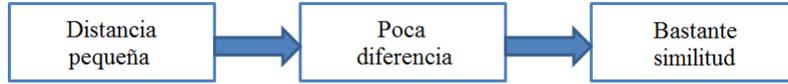


Figura 12: Descripción de la implicación del valor de una función de distancia.

La similitud o la distancia entre dos objetos, puede ser medida en base a varias variables de características. Las métricas dependen estrictamente de los valores de dichas variables. Una vez que la distancia o la similitud entre dos objetos ha sido calculada, tal información puede ser utilizada para la indexación o del conjunto de objetos. En general las métricas básicas pueden dividirse en tres categorías. A continuación se describen las métricas y la categoría a la que corresponden, según el trabajo de E.A. Santos-Camacho [12].

1. Distancias para variables binarias. Coeficiente de Concordancia Simple (*Simple Matching Coefficient*), Coeficiente de *Jaccard*, Distancia de *Hamming*.
2. Distancias para variables nominales. Transformación de Rango Normalizado (*Normalized Rank Transformation*), Distancia de *Spearman*, Distancia *Footrule*, Distancia de *Kendall*, Distancia de *Cayley*, Distancia de *Ulam*, Distancia de *Hamming*.
3. Distancias para variables ordinales. Distancia *Euclidiana*, Distancia *City Block (Manhattan)*, Distancia *Chebyshev*, Distancia de *Minkowski*, Distancia de *Canberra*, Separación Angular (Coseno), Coeficiente de Correlación.

A continuación se mencionan las métricas usadas comúnmente en la literatura para comparación en 3D. Veltkamp [37] proporciona una introducción de comparación de figuras utilizando teoría de geometría computacional, que incluye medidas de similitud usadas para comparar polígonos y curvas.

- Norma  $L_p$  (Distancia *Minkowski*)

Esta medida es usada bajo el principio de que se tienen representaciones numéricas que consideran globalmente a una figura. Las representaciones deben de ser vectores de un tamaño determinado (pueden ser vectores de características), así que las comparaciones de similitud se realizan sobre los vectores. La norma  $L_p$  no es una medida de similitud invariante a transformaciones.

- Distancia *Hausdorff*

La distancia *Hausdorff* es una métrica. No es invariante a transformaciones y no es robusta ante ruido. La ventaja de esta métrica es que es posible la comparación parcial.

- Distancia del movimiento terrestre (*Earth Mover's Distance*), también conocida como distancia de transporte.

Dados dos caminos ponderados de puntos  $A$  y  $B$ , se determina la cantidad mínima de trabajo que lleva transformar  $A$  en  $B$ .

Las métricas mencionadas en esta sección, sirven como medidas de comparación de una metodología convencional de análisis de similitud entre datos multimedia en general. Cada una de ellas, se utiliza dependiendo de la manera en que los modelos son representados para su comparación, pero por la naturaleza de las métricas de comparar características, no son efectivas con las representaciones que se proponen en este trabajo.

### 3.2.2.1. Propiedades de una función de distancia

La comparación de figuras es un concepto importante, ya que es el proceso de determinar a partir de dos figuras, qué tan similares son. Generalmente, esto se hace calculando la distancia en términos de una medida de disimilitud, donde una distancia pequeña indica poca disimilitud indicando bastante similitud [37].

**Definición 1.** Sea un conjunto de figuras  $S = \{s_1, s_2, \dots, s_N\}$ , la distancia de similitud es definida como  $d(s_i, s_j): S \times S \rightarrow R^+0$  donde  $s_i, s_j \in S$ .

1. Identidad:

$$\forall s_i \in S, d(s_i, s_i) = 0 \quad (1)$$

2. Positividad:

$$\forall s_i, s_j \in S, s_i \neq s_j, d(s_i, s_j) > 0 \quad (2)$$

3. Simetría:

$$\forall s_i, s_j \in S, d(s_i, s_j) = d(s_j, s_i) \quad (3)$$

4. Desigualdad triangular:

$$\forall s_i, s_j, s_k \in S, d(s_i, s_k) \leq d(s_i, s_j) + d(s_j, s_k) \quad (4)$$

5. Invariante de transformación: Sea un grupo de transformaciones  $G$ ,

$$\forall s_i, s_j \in S, g \in G, d(s_i, g(s_j)) = d(s_i, s_j) \quad (5)$$

La función  $d$  debe de cumplir por lo menos las primeras tres propiedades.

**Definición 2.** Una función de distancia que tiene las propiedades de identidad, positividad, simetría y desigualdad del triángulo se le denomina *métrica*.

**Definición 3.** Una función de distancia que tiene las propiedades de identidad, simetría y desigualdad triangular se le denomina *seudo métrica*.

**Definición 4.** Una función de distancia que tiene las propiedades de identidad, positividad y simetría se le denomina *semi métrica*.

### 3.3. Técnicas de comparación

El criterio de comparación permite tener una función de distancia, componente fundamental para una función de disimilitud. Dicha función es útil para agrupar objetos analizados en un espacio de medida y mediante algún método de clasificación, determinar la pertenencia a alguna clase.

La metodología de esta tesis, no realiza clasificación. En cambio proporciona una técnica de comparación donde el resultado es un valor de distancia.

Con un criterio de agrupación, el valor de distancia entre entidades comparadas, es útil para clasificación.

#### 3.3.1. Comparación elástica desde el enfoque de esfuerzo de alineación

Uno de los problemas más comunes en computación es la forma de representar la información extraída de diversos fenómenos; por ejemplo, la representación de una imagen en una computadora, primero tiene que ser mapeada a una matriz de números para poder ser procesada. Otros ejemplos: las mediciones de diferentes variables como las de una bolsa de valores, se representa con un arreglo de datos o el diagnóstico médico, que puede ser representado por un árbol. Como es de esperarse, la búsqueda de similitud entre representaciones y estructuras de datos se vuelve compleja.

Los seres humanos por otro lado, tienen la capacidad de encontrar similitud de forma natural, pero hasta el momento no se sabe cómo el cerebro realiza esta función. En la actualidad conocer el fenómeno de similitud en diferentes áreas del conocimiento requiere encontrar en primera instancia, una correspondencia que no sea directa entre dos entidades (en ese caso secuencias numéricas), es decir, saber si un patrón es similar a otro en cierto grado. Este es, un problema central en el área de Reconocimiento de Patrones. En la figura 13, se presentan los problemas al comparar dos secuencias numéricas y en la figura 14 se muestra que para la tarea de reconocimiento de una letra, la comparación directa no es suficiente.

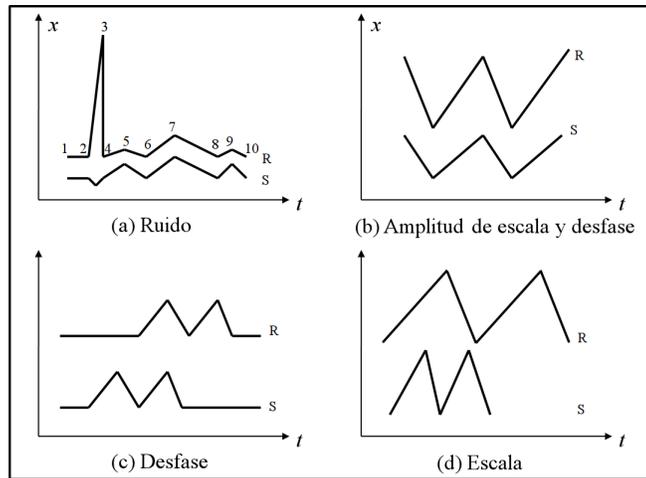


Figura 13: Problemas en la comparación de secuencias numéricas. (a) La secuencia  $R$  presenta ruido en la posición 3 de la secuencia  $S$ . (b)  $R$  tiene escala y posición diferente, (c)  $R$  se encuentra desfasada y (d)  $R$  se encuentra con diferente escala con respecto a la serie  $S$ .



Figura 14: La comparación directa no es suficiente para reconocer la letra que se ilustra.

Por otro lado, dados estos problemas, la alineación producida con la distancia *Euclidiana* (comparación del punto  $i$ -ésimo de la primera secuencia con el  $i$ -ésimo de la segunda secuencia), no aminora los problemas de la figura 13, aun cuando estos son muy pequeños. En la figura 15, se presenta la alineación producida con la distancia *Euclidiana* la serie  $Q$  está desfasada con respecto a la secuencia  $C$ , es por esta razón que esta medida es poco útil bajo los problemas de alineamiento mencionados.

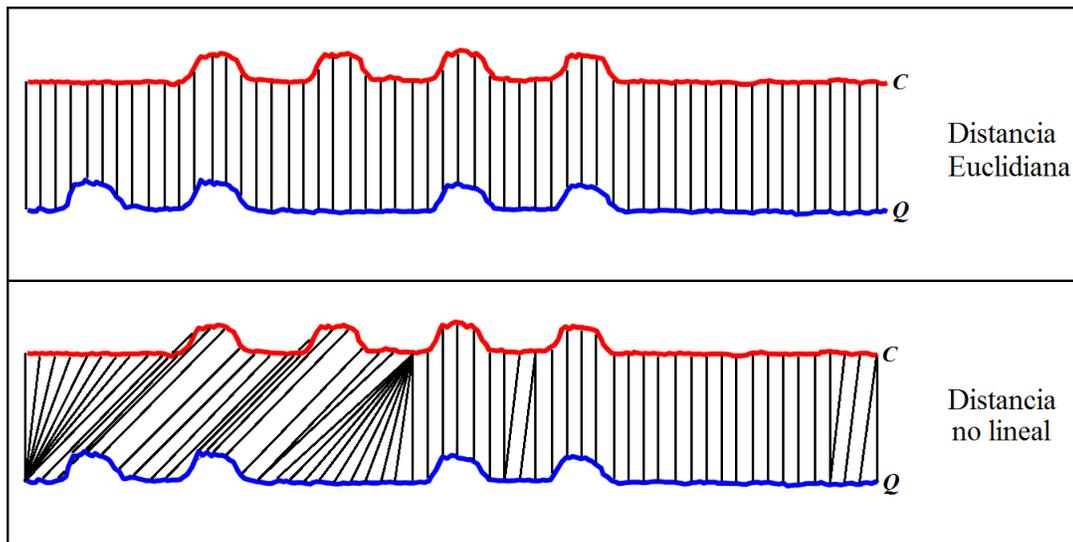


Figura 15: Alineación usando distancia *Euclidiana* y alineación no lineal (*FDTW*).

Entonces, la *comparación elástica* es la relación que existe entre la información contenida de dos secuencias numéricas. En la figura 16, se ilustra un ejemplo de alineación elástica, donde la comparación es no lineal.

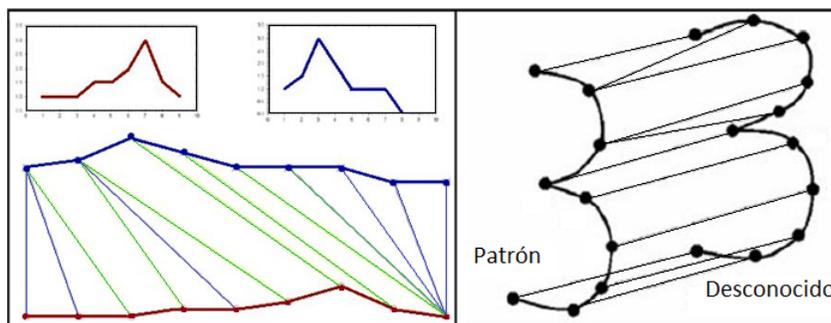


Figura 16: Ejemplo de comparación elástica.

### 3.3.2. Distancia *Dynamic Time Warping*

*Dynamic Time Warping* es una técnica que ha sido usada ampliamente en reconocimiento de voz. Lo que caracteriza a la técnica es que permite un mapeo no lineal de una secuencia con respecto a otra minimizando la distancia base entre las dos, que comúnmente corresponde con la distancia euclidiana. *DTW* es propia para calcular la similitud entre secuencias que se encuentran desfasadas una con respecto a la otra, o cuando una ellas presenta u omite segmentos.

Dadas dos secuencias  $Q$  y  $C$  especificadas en (6), *DTW* es definido como (9).

$$Q = \langle q_1, q_2, \dots, q_{|Q|} \rangle, \langle C = c_1, c_2, \dots, c_{|C|} \rangle \quad (6)$$

Donde se cumple:

$$DTW(\langle \rangle, \langle \rangle) = 0 \quad (7)$$

$$DTW(Q, \langle \rangle) = DTW(\langle \rangle, C) = \infty \quad (8)$$

$$DTW(Q, C) = D_{base}(h(Q), h(C)) + \min \begin{cases} DTW(Q, r(C)) \\ DTW(r(Q), C) \\ DTW(r(Q), r(C)) \end{cases} \quad (9)$$

Donde:

$$D_{base} = (q_i, c_i) = \sqrt{(q_i - c_i)^2} \quad (10)$$

Lo anterior corresponde con la forma tradicional de definir *DTW*, una técnica de complejidad temporal dada por una expresión  $O(3^n)$ , siendo intratable para datos de tamaño considerable.

#### 3.3.2.1. Distancia *Fast Dynamic Time Warping (FDTW)*

*FDTW* es la versión optimizada no recursiva a *DTW*. La solución de la comparación de dos secuencias con *FDTW* es la distancia mínima total y la ruta de ajuste (*warpath*), ver figura 17. La ruta de ajuste nos indica cuales son las relaciones entre los elementos de ambas secuencias dadas por la alineación encontrada. Esta distancia se encuentra en un rango de 0 (cero) a infinito; donde la distancia  $d$  hacia 0 indica similitud y hacia infinito disimilitud. Es importante mencionar que *FDTW* tiene algunas restricciones porque se reduce el espacio de soluciones que se pueden encontrar [1, 22].

En el algoritmo 3.1 se describe la solución a la alineación de secuencias con *FDTW*.

Se puede observar que el algoritmo tiene una complejidad dada por un término polinomial cuadrático.

En cuanto a la distancia base, en observaciones experimentales se identificó que es suficiente con la distancia *Euclidiana*. Es así que se define la misma distancia base de la ecuación (10).

Experimentalmente, al cambiar la distancia base por la distancia *Minkowski* ( $L_p$ ,  $p = 3$ ), se obtienen valores proporcionales de distancia que no afectan el resultado. Lo anterior se debe a que las secuencias numéricas se mantienen en una cantidad de registros constante así como un intervalo de valores acotado. Las características mencionadas se derivan del control de invariantes de las figuras 3D. Por otro lado, la inclusión de una distancia base diferente, muestra en *FDTW* un efecto de mantener la relación entre la comparación local de los datos.

---

**Algoritmo 3.1** Algoritmo de *FDTW*

---

Inicializar  $r_{i,j} \leftarrow 0$  para  $i = 2, \dots, n$  y  $j = 2, \dots, m$

Asignar  $r_{1,1} \leftarrow d_{base}(q_1, c_1)$

Repetir para  $i = 2, \dots, n$

    Calcular  $r_{i,1} \leftarrow d_{base}(q_i, c_1) + r_{(i-1),1}$

Fin repetir

Repetir para  $j = 2, \dots, m$

    Calcular  $r_{1,j} \leftarrow d_{base}(q_1, c_j) + r_{1,(j-1)}$

Fin repetir

Repetir para  $i = 2, \dots, n$

    Repetir para  $j = 2, \dots, m$

        Calcular  $u \leftarrow d_{base}(q_i, c_i) + r_{(i-1),(j-1)}$

        Calcular  $v \leftarrow d_{base}(q_i, c_i) + r_{i,(j-1)}$

        Calcular  $w \leftarrow d_{base}(q_i, c_i) + r_{(i-1),j}$

        Asignar  $r_{i,j} \leftarrow \min\{u, v, w\}$

    Fin repetir

Fin repetir

Asignar  $d \leftarrow r_{n,m}$

---

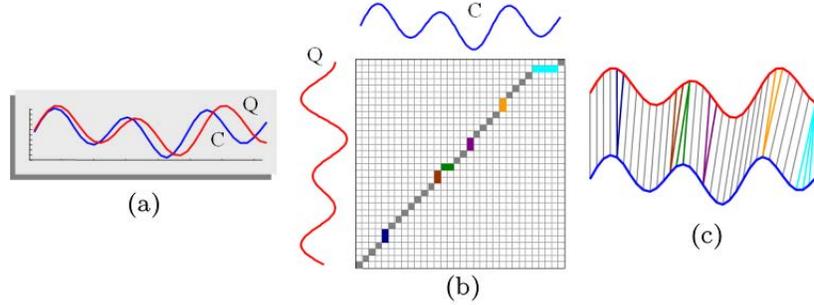


Figura 17: Proceso de alineación de secuencias usando *FDTW*: (a) secuencias originales, (b) proceso de alineación, (c) alineación encontrada.

Para la indexación de series de datos, se sugiere ocupar aproximaciones donde se realiza un submuestreo de los datos o bien se representan por una combinación de los máximos y los mínimos. Consulte más información sobre esta alternativa en [28].

### 3.3.2.2. La ruta de ajuste mínima (*warpath*)

En el apartado 3.3.2.1 el algoritmo de *FDTW* trabaja con la matriz  $r_{N,M}$  denominada matriz de costo acumulado. Para determinar la ruta de ajuste mínima se aplica el siguiente procedimiento.

Siendo la ruta de ajuste (*warpath*)  $p = (p_1, \dots, p_L)$  se calcula en orden inverso tomando como referencia los índices con comienzo en  $p_L = (N, M)$ . Supóngase que  $p_l = (n, m)$  será calculada. En caso de que  $(n, n) = (1, 1)$ , se tiene que  $l = 1$  y el procedimiento ha terminado. De otro modo se tiene (11).

$$p_{l-1} \begin{cases} (1, m-1), & \text{si } n = 1 \\ (n-1, 1), & \text{si } m = 1 \\ \min\{r_{n-1, m-1}, r_{n-1, m}, r_{n, m-1}\}, & \text{en otro caso} \end{cases} \quad (11)$$

Para analizar pares de secuencias considerando sus alineaciones, se considera lo siguiente. Se toma la secuencia de referencia y se coloca horizontalmente en la parte inferior de la matriz de costo acumulado, mientras que la otra secuencia se coloca verticalmente a la izquierda de la matriz. Así que la matriz de costo acumulado contendrá la ruta de ajuste mínima (ver figura 18). De esta manera se pueden tener puntos de comparación que correspondan con los índices de la secuencia comparada, recorriendo la ruta de ajuste de manera inversa y proyectando en la secuencia de la izquierda para determinar el índice de la secuencia de referencia.

La ventaja de este método es que la ruta de ajuste se puede determinar directamente, por lo tanto es sencillo visualizar el impacto de propiedades locales y globales de una secuencia con respecto a la otra.

**3.3.3. Distancia *Minkowski***

La distancia *Minkowski* resulta ser muy útil como indicador de disimilitud, como se mostró en los resultados obtenidos en el trabajo [12]. Donde se plantea, que la comparación con medidas clásicas, como la distancia *Euclidiana*, presenta diferentes problemas en la comparación de secuencias numéricas. Se observó que de un estudio de 21 medidas, con la distancia *Minkowski* se logra aminorar dichos problemas.

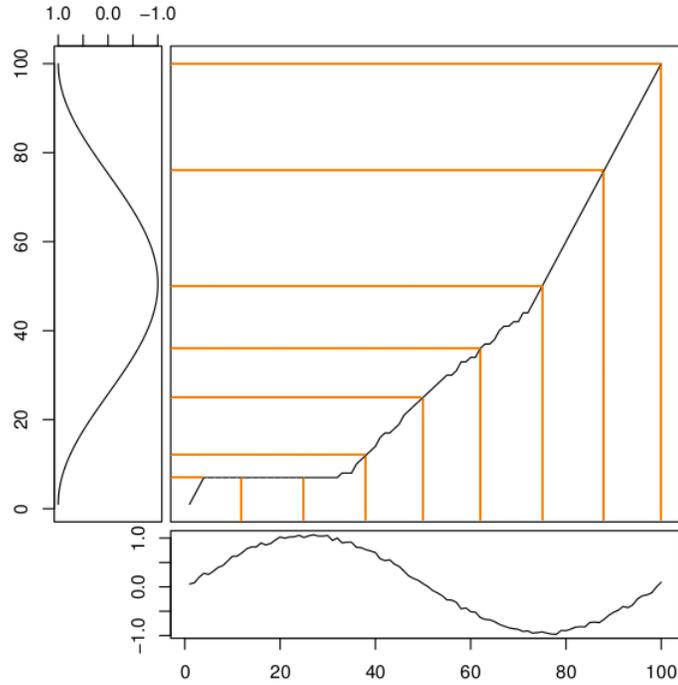


Figura 18: Ejemplo de la idea para la alineación entre una función seno con ruido y una coseno en  $[0, 2\pi]$ , con líneas guías trazadas cada  $\frac{\pi}{4}$ .

Esta medida es usada bajo el principio de que se tienen representaciones numéricas que consideran globalmente a una figura. Las representaciones deben de ser vectores de un tamaño determinado (pueden ser vectores de características), así que las comparaciones de similitud se realizan sobre los vectores.

**Definición 5.** Dados dos puntos  $x, y \in R^k$ , la distancia *Minkowski* ( $L_p$ ) se define como:

$$L_p = \left( \sum_{i=1}^k |x_i - y_i|^p \right)^{1/p} \quad (12)$$

Para  $p \geq 1$ ,  $L_p$  es una métrica.

Si  $p = 1$ , se le denomina norma  $L_1$ , distancia *Manhattan* o distancia *City Block*.

Si  $p = 2$ , se le denomina norma  $L_2$  o distancia *Euclidiana*.

La norma  $L_p$  no es una medida de similitud invariante a transformaciones.

La distancia *Minkowski* ( $L_3$ ) se utiliza como medida de distancia de referencia en los experimentos. El algoritmo 3.2 describe el procedimiento para calcular dicha distancia.

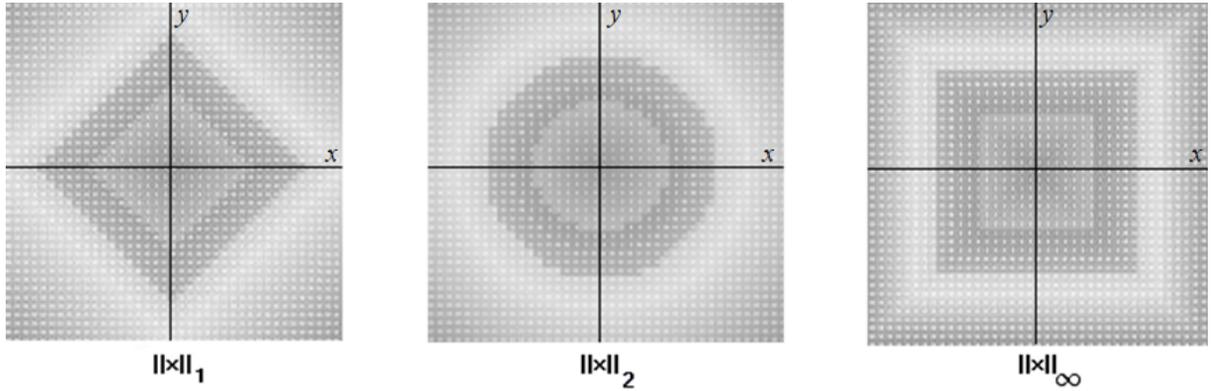


Figura 19: Patrones de la distancia *Minkowski* en dos dimensiones que satisfacen  $\|x\|_p$ . Se considera la distancia desde el punto  $(0,0)$  a un punto  $(x,y)$ .

---

**Algoritmo 3.2** Algoritmo de distancia *Minkowski*

---

Inicializar  $d \leftarrow 0$

Inicializar  $p \leftarrow 3$

Repetir para  $i = 1, \dots, n$

Calcular  $d \leftarrow d + |x_i - y_i|^p$

Fin repetir

Asignar  $d \leftarrow (d)^{1/p}$

---

### 3.3.4. Comparación de dos figuras en 3D

A continuación se describe la comparación entre dos figuras en 3D a través de la transformación de dominio de  $\mathbb{R}^3$  a  $\mathbb{R}^1$  (3D a 1D). Se propone una alternativa para la medición del esfuerzo de la alineación elástica entre dos figuras 3D.

La comparación se compone de una etapa de preprocesamiento (ver figura 20), que consiste en el control de invariantes de una figura para su análisis. Después del preprocesamiento, la figura se analiza para generar una nueva representación útil para la comparación, dicha representación es una secuencia numérica. Lo siguiente es realizar la comparación calculando la distancia entre dos secuencias, donde cada secuencia representa a una figura 3D. La distancia medida es un valor, considerado como indicador numérico del esfuerzo de alineación entre las figuras 3D originales.

En la figura 20 se muestra el diagrama de comparación descrita con dos figuras de ejemplo.

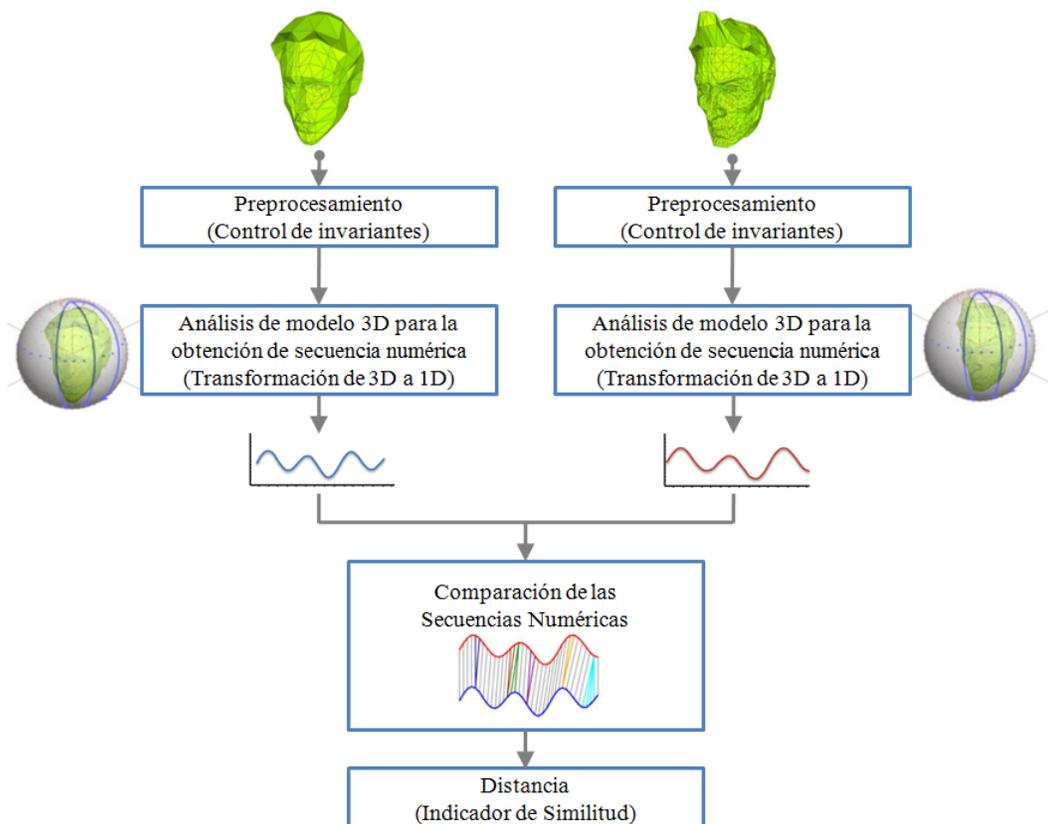


Figura 20: Diagrama general de la comparación de representaciones de figuras 3D.

### 3.4. Conceptos de recuperación

Si una técnica de comparación calcula la distancia numérica entre dos figuras, las condiciones están dadas para calcular cierto número de medidas, que sirven para

comparar a la técnica con otras, bajo el principio de que todas trabajan con el mismo conjunto de datos.

Para una técnica de comparación dada y un conjunto de datos de modelos 3D ( $S = \{s_1, s_2, \dots, s_N\}$ ), se calcula una matriz de distancias entre pares de modelos. Para cualquier modelo  $q \in S$ , se puede recuperar la lista de los  $k$  modelos más similares a partir de la matriz de distancias.

Se enumeran las medidas cualitativas y cuantitativas proporcionadas por *Princeton Shape Benchmark* [5], que indican el desempeño de una técnica para la recuperación de figuras 3D.

1. Listado de imágenes de las figuras ordenadas en orden de similitud descendiente
2. Gráfica de *precisión-recuperación* (*precision-recall*)
3. Imagen de matriz de distancias
4. Imagen de nivel (*tier image*)
5. Vecino más cercano (*Nearest Neighbor, NN*)
6. Medida-E (*E-Measure*)

Tanto 1, 3 y 4 son medidas que implican la apreciación de los diseñadores del conjunto de datos, mientras que la gráfica de *precisión-recuperación* es una medida cuantificable que parte de la siguiente descripción.

De acuerdo con Baeza-Yates y Ribeiro-Neto [4], *precisión* es la fracción de los elementos recuperados que son relevantes de acuerdo con un elemento a comparar y recuperar, la *recuperación* es la fracción de elementos relevantes que han sido recuperados del conjunto de datos.

Sea:

$R = \{\text{conjunto de elementos relevantes}\}$ ,  $|R|$ :Número de elementos relevantes.

$A = \{\text{conjunto de elementos respuesta}\}$ ,  $|A|$ :Número de elementos relevantes.

$R_a = \{\text{conjunto de elementos relevantes recuperados}\}$ ,  $|R_a|$ :Número de elementos relevantes recuperados.

$q$  =Modelo a recuperar por el algoritmo de comparación y recuperación.

$$\text{recuperación} = \frac{|R_a|}{|R|} \quad (13)$$

$$\text{precisión} = \frac{|R_a|}{|A|} \quad (14)$$

La gráfica de la relación *precisión-recuperación* es una medida común que se calcula en la mayoría de los trabajos del estado del arte.

Sea  $Rq = \{\text{conjunto de elementos relevantes para una consulta } q\}$ ,  $|Rq|$ : Número de elementos relevantes para un modelo a recuperar  $q$ .

Para evaluar el comportamiento de un sistema de recuperación, se ejecutan varias consultas.

Sea:

$Nq$  = Número de modelos recuperados.

$P_i(r)$  = *Precisión* que corresponde con la *recuperación*  $r$  para el modelo a recuperar  $i$ .

$\bar{P}(r)$  = *Precisión* promedio que corresponde con la *recuperación*  $r$ .

Finalmente:

$$\bar{P}(r) = \sum_{i=1}^{Nq} \frac{P_i(r)}{Nq} \quad (15)$$

En el análisis de las curvas de *precisión-recuperación* es posible obtener comportamientos que consisten en los puntos siguientes.

- A. La recuperación de elementos no es muy buena pero ordena bien los elementos relevantes en las primeras posiciones de los resultados.
- B. La recuperación de elementos relevantes es buena, pero no ordena los elementos relevantes en las primeras posiciones de los resultados.

La propuesta en este trabajo es usando el enfoque (A). Lo que se obtiene son resultados donde lo relevante es la precisión de los elementos recuperados.

La medida del **vecino más cercano** (5) indica el porcentaje de las  $k$  mejores comparaciones que pertenecen a la misma clase de la figura que se recuperó. Con dicha medida se proporciona un indicador de qué tan bien una técnica recupera, considerando las primeras respuestas.

El algoritmo 3.3 muestra el cálculo de la medida para  $k = 1$ .

---

**Algoritmo 3.3** Cálculo de la medida del vecino más cercano

---

Inicializar  $nn \leftarrow 0$ Repetir para cada figura del conjunto  $S$ 

Si la primera figura respuesta pertenece a la misma clase de la figura a recuperar

        Asignar  $nn \leftarrow nn + 1$ 

Fin si

Fin repetir

Regresar  $|S|/nn$ 

---

La *medida-E* (6) está compuesta de los valores de *precisión* y *recuperación* para un número predeterminado de resultados recuperados. La idea, es que un usuario de un sistema de recuperación, está más interesado en los primeros resultados que en el resto. Es entonces, que la *medida-E* considera únicamente cierta cantidad de modelos recuperados para cada consulta y calcula la *precisión* y la *recuperación* sobre esos resultados.

La *medida-E* se define como:

$$E = \frac{2}{\left(\frac{1}{precision} + \frac{1}{recall}\right)} \quad (16)$$

Existen otros conjuntos de datos de pruebas más especializados, por ejemplo:

- Escultor (*Sculpteur, SCU*) [16]
- Concurso de recuperación de figuras (*Shape Retrieval Contest Watertight, SHREC-W*) [15]
- Conjunto de figuras de ingeniería de Purdue (*Purdue Engineering Shape Benchmark, ESB*) [20]

Es frecuente que los trabajos de análisis de similitud entre figuras 3D, utilicen su propio conjunto de datos de acuerdo al campo de aplicación; pero es complicado comparar el desempeño de los métodos sin un conjunto de datos unificado.

## METODOLOGÍA

---

El capítulo describe los aspectos metodológicos que conforman la propuesta. Se describen los conjuntos de datos con los que se experimentaron. También se mencionan las herramientas que apoyan la implementación indicando sus características importantes.

### 4.1.Descripción de la metodología

La metodología que se propone se resume brevemente en los siguientes puntos.

#### **Metodología propuesta**

1. Selección del conjunto de datos.
2. Preprocesamiento de cada figura 3D para el control de invariantes de rotación, escala y traslación, a partir de los puntos en 3D que conforman la descripción geométrica del modelo.
3. Transformación de dominio de 3D a 1D mediante tres técnicas diferentes, obteniendo tres secuencias representativas para cada figura 3D.
4. Comparación de las secuencias representativas usando *FDTW* y construcción de la matriz de distancias. La comparación se hace entre secuencias generadas por la misma técnica de transformación de dominio, obteniendo tres matrices de distancias, una por cada técnica de representación.
5. Consultas sobre la matriz de distancias que determinan el comportamiento de la metodología mediante indicadores de recuperación.

La metodología se inicia con la selección del conjunto de datos. Tanto la definición como la descripción de los datos tridimensionales se exponen la siguiente sección.

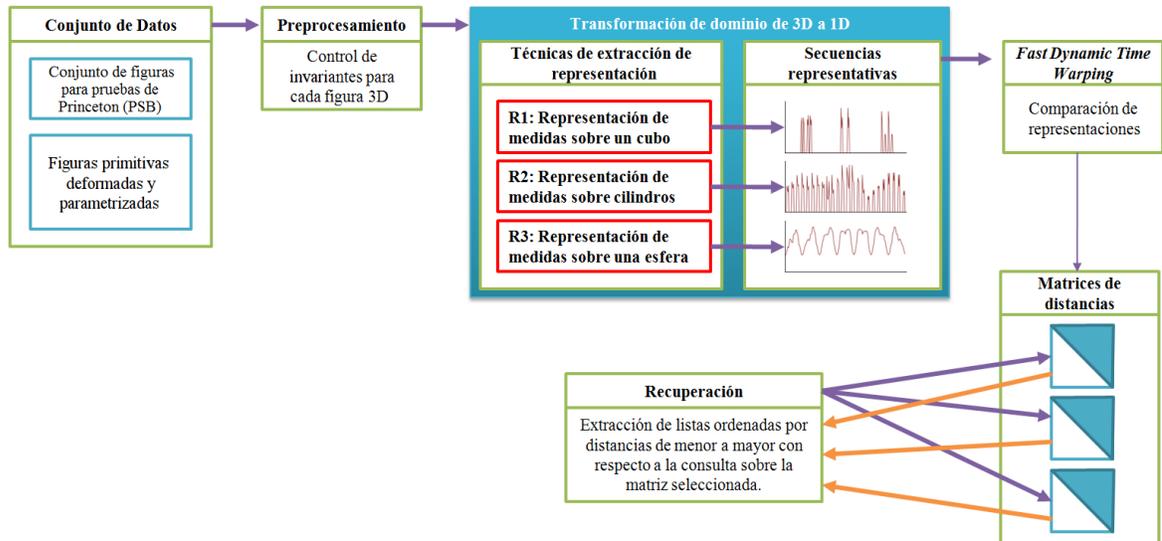


Figura 21: Metodología propuesta.

#### 4.1.1.Limitaciones

En la introducción se mencionan los alcances de la propuesta, principalmente son los siguientes.

- Los objetos de estudio son figuras tridimensionales que **no tienen asociados color ni textura** y pueden ser geométricas o irregulares, inclusive figuras que no encierran un espacio.

Algunas condiciones de color determinan el contexto de la figura, por ejemplo, si en una imagen se tiene en una mitad color verde y en otra mitad color azul, es posible que la imagen sea un paisaje de un prado en un día despejado. Este trabajo no determina ni considera aspectos semánticos de las figuras.

- En general las técnicas de representación propuestas tienen resultados de similitud visual. Las figuras 3D con partes desconectadas o bien estructuras 3D que delimitan una forma pero que no son sólidos cerrados presentan problemas en la búsqueda de similitud visual.

En la figura 22 se muestra una "mariposa" donde las alas están "huecas" y tiene sólidos en 3D desconectados de la figura. Si un vector de dirección no tiene intersección con un plano, no se obtiene un registro diferente de 0 (cero).



Figura 22: Ejemplo de figura con dificultades para una recuperación.

#### 4.2.Datos tridimensionales

Una figura 3D puede representarse en distintas formas, por ejemplo: nubes de puntos, imágenes de profundidad, “*polygon soups*” y malla de polígonos. Dependiendo de las herramientas de análisis es como se define la representación que se va a utilizar. En el caso de este trabajo existen 2 razones que conducen a utilizar la representación de malla de polígonos que a continuación se especifican.

1. Se utilizaron herramientas matemáticas del álgebra lineal.
2. Los resultados serán comparados con el *Princeton 3D Shape Benchmark (PSB)*, donde todas las figuras se encuentran bajo la representación de malla de polígonos triangulares.

La representación de datos espaciales de una figura 3D está dada por la definición de malla de polígonos, la cual es muy popular para modelos tridimensionales debido a su simplicidad. Una figura tridimensional entonces es definida por un par de listas ordenadas:

$$M = \langle P, V \rangle \tag{17}$$

Donde  $V = \{v_1, v_2, \dots, v_N\}$  es la lista de vértices y  $v_n = \{x_n, y_n, z_n\}$ .  $P = \{p_1, p_2, \dots, p_R\}$  es la lista de polígonos planares y  $p_r = \{v_{n,1}, v_{n,2}, \dots, v_{n,k_r}\}$  en donde  $k_r$  es el número de vértices en un polígono  $p_r$ . Si  $k = 3$  para todo  $p_r$  entonces lo que se obtiene se le denomina malla de triángulos.

En el anexo A.3.1 se especifican los archivos de formato de objeto 3D (.off), donde cada archivo es la información necesaria que compone a un modelo 3D.

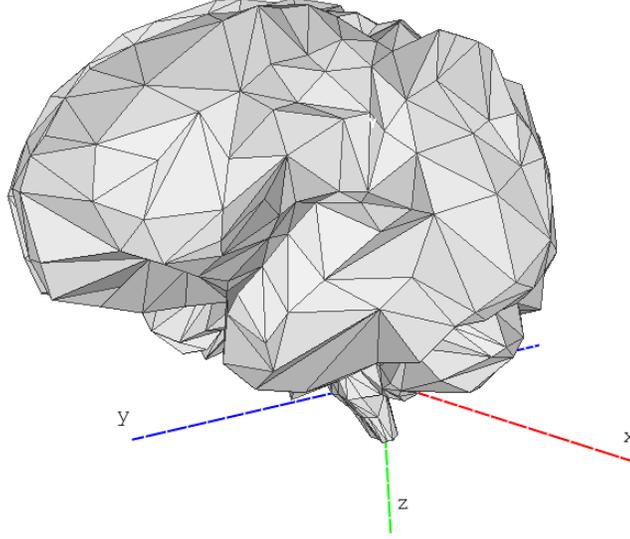


Figura 23: Una figura en 3D de un cerebro, representada mediante malla de polígonos triangulares.

La malla de triángulos es útil en el sentido que si se trabaja con la superficie de un triángulo delimitado por 3 puntos espaciales, al mover cualquiera de los puntos siempre se encontrarán en el mismo plano (coplanares), situación que no es así para un polígono de cuatro vértices, ver figura 24. El realizar el análisis a partir de polígonos triangulares, se traduce en menor complejidad y reducción de cálculos.

Una malla consiste en un conjunto de triángulos denotada como:

$$T = \{T_1, T_2, \dots, T_m\}, T_i \in \mathbb{R}^3 \quad (18)$$

Donde  $T_i$  se considera como un conjunto de tres vértices:

$$T_i = \{P_A, P_B, P_C\} \quad (19)$$

La metodología se compone de una etapa de preprocesamiento que consiste en controlar las invariantes de rotación, escala y traslación de las figuras 3D del conjunto de datos que a continuación se describe.

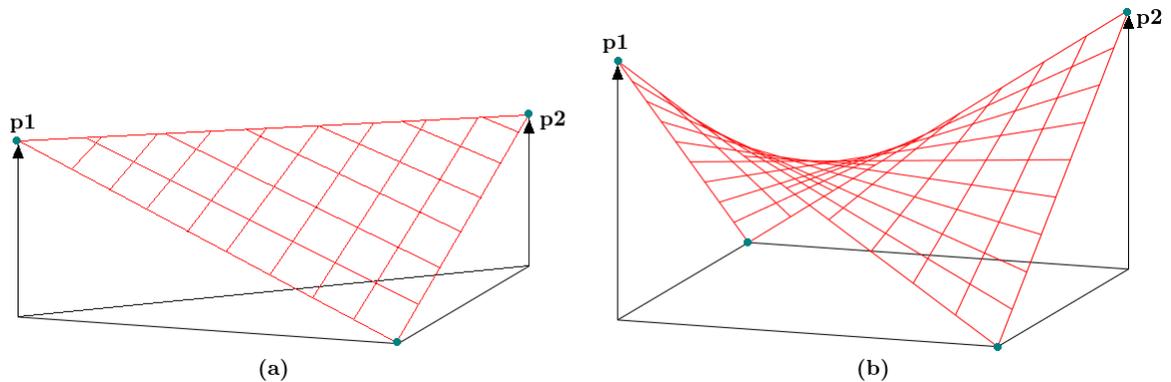


Figura 24: a) Al desplazar los puntos  $p1$  y  $p2$  cualquier punto en el área del triángulo se mantiene en el mismo plano. b) Al desplazar los puntos  $p1$  y  $p2$  los puntos en el área del rectángulo se encuentran en distintos planos.

### 4.3. Control de invariantes de una figura 3D

En general, las figuras 3D de una base de datos están en unidades, posición y orientación de forma arbitraria. Por ello se aplican procesos de control de invariantes de rotación, escala y traslación, con el objetivo de obtener propiedades adecuadas controlando cada invariante.

Como observación importante, al final del control de invariantes, el resultado conserva la topología y la estructura de la figura 3D original.

#### 4.3.1. Invariante de rotación

La condición de rotación de una figura 3D es un problema complicado en comparación a las invariantes de escala y traslación. La herramienta más importante para obtener una alineación en términos de rotación, es la técnica general de Análisis de Componente Principal (*PCA*), también llamada transformada de *Karhunen-Loeve* o transformada *Hotelling* en los campos de procesamiento de señales, estadística, compresión y cómputo neuronal. Se usa *PCA* para calcular los ejes principales de la figura, de tal manera que la figura es rotada para que sus ejes principales se asocien con un sistema coordenado canónico.

##### 4.3.1.1. Principio básico de *PCA* en datos 3D

El *PCA* se basa en una representación estadística de variables aleatorias como sigue:

Sea

$$P = P_1, P_2, \dots, P_n \quad (20)$$

Donde

$$P_i = (x_i, y_i, z_i) \in \mathbb{R}^3 \quad (21)$$

Así que [20](#) es el conjunto de vértices en la superficie de una figura 3D. Para operar con el conjunto de vértices utilizamos el concepto de matriz [A.1.1](#), donde  $P$  puede definirse como una matriz de dimensiones  $n \times 3$  (ver ecuación [22](#)).

$$P = \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ x_n & y_n & z_n \end{bmatrix} \quad (22)$$

El objetivo del *PCA* es encontrar un mapeo  $\tau = \mathbb{R}^3 \rightarrow \mathbb{R}^3$  de tal forma que una figura 3D en cualquier estado de rotación, sea puesta en un estado fijo. El procedimiento se especifica a continuación.

1. Se encuentra el baricentro de la figura, dado como sigue:

$$x_c = \frac{1}{N} \sum_{i=1}^N x_i, \quad y_c = \frac{1}{N} \sum_{i=1}^N y_i, \quad z_c = \frac{1}{N} \sum_{i=1}^N z_i \quad (23)$$

2. Se determina un nuevo punto que corresponde con el centro de la figura:

$$P_c = (x_c, y_c, z_c) \quad (24)$$

3. Por cada punto  $P_i = (x_i, y_i, z_i) \in P$  se aplica la transformación de traslación del centro de la figura al origen:

$$P'_i = (x_i - x_c, y_i - y_c, z_i - z_c) \quad (25)$$

4. Se determina un nuevo conjunto de puntos:

$$P' = (P'_1, P'_2, \dots, P'_n) \quad (26)$$

5. A partir del nuevo conjunto  $P' = (P'_1, P'_2, \dots, P'_n)$  donde ahora  $P'_i = (x_i, y_i, z_i) \in P'$ , se calcula la matriz de covarianza:

$$M_C = \frac{1}{n} \sum_{i=1}^n P_i'^T P_i' = \frac{1}{n} \begin{bmatrix} \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i y_i & \sum_{i=1}^n x_i z_i \\ \sum_{i=1}^n x_i y_i & \sum_{i=1}^n y_i^2 & \sum_{i=1}^n y_i z_i \\ \sum_{i=1}^n x_i z_i & \sum_{i=1}^n y_i z_i & \sum_{i=1}^n z_i^2 \end{bmatrix} \quad (27)$$

6. La matriz  $M_C$  tiene la propiedad de ser simétrica de números reales. Luego, a  $C$  se le calculan sus *eigenvalores*, se ordenan en forma decreciente, encontrando sus correspondientes *eigenvectores*. Los *eigenvectores* son escalados a la unidad y se forma una matriz de rotación  $R$ . Se rotan todos los puntos en  $P'$  formando un nuevo conjunto de puntos  $P''$ .

$$P'' = \{P'' \mid P'' = P'_i R, P'_i \in P', i = 1, \dots, n\} \quad (28)$$

Al terminar, la figura es rotada de tal manera que el eje  $x$  mapea al *eigenvector* con el eigenvalor más grande, el eje  $y$  se mapea con *eigenvector* con el segundo eigenvalor más grande y el eje  $z$  mapea al *eigenvector* con el eigenvalor más pequeño.

En el trabajo de Ortega-González [28] se explican conceptos generales, definiciones y métodos para el cálculo de los *eigenvalores* y *eigenvectores*.

Hasta este punto se tiene que  $P''_i = (x_i, y_i, z_i) \in P''$ , al graficar los puntos se obtiene una ambigüedad que consiste en que *PCA* puede mostrar las distribuciones de las figuras ordenadas de acuerdo a los ejes  $x, y$  y  $z$  pero no considera si la figura se orientará en dirección de los números positivos o negativos.

Se propone una aproximación al problema que consiste en calcular el área de la superficie de la figura a partir de las áreas de los polígonos que la componen. Se denota  $S_i$  como el área del triángulo  $T_i$  definido en (19).

$$S_i = | (P_{C_i} - P_{A_i}) \times (P_{B_i} - P_{C_i}) | / 2 \quad (29)$$

El área total de la superficie de la figura corresponde con  $S$ .

$$S = \sum_{i=1}^m S_i \quad (30)$$

Cada polígono puede encontrarse del lado de los números positivos o negativos, se verifican si las componentes son negativas  $(x_i, y_i, z_i)$ , en el caso de serlo se acumula el área  $S_i$  en  $S_{negativosx}, S_{negativosy}$  y  $S_{negativosz}$  según corresponda.

$$S_i : (x_{A_i}, x_{B_i}, x_{C_i} < 0) \rightarrow S_{negativosx} = \sum S_i \quad (31)$$

$$S_i : (y_{A_i}, y_{B_i}, y_{C_i} < 0) \rightarrow S_{negativosy} = \sum S_i \quad (32)$$

$$S_i : (z_{A_i}, z_{B_i}, z_{C_i} < 0) \rightarrow S_{negativosz} = \sum S_i \quad (33)$$

La ambigüedad entre los ejes positivos y negativos se resuelve seleccionando la dirección de los ejes positivos para colocar el área mayor de la figura en las componentes  $x, y$  y  $z$ .

$$Si : (S_{negativosx} > (S - S_{negativosx})) \rightarrow P'' = P'' R_x \quad (34)$$

$$Si : (S_{negativosy} > (S - S_{negativosy})) \rightarrow P'' = P'' R_y \quad (35)$$

$$Si : (S_{negativosz} > (S - S_{negativosz})) \rightarrow P'' = P'' R_z \quad (36)$$

En las ecuaciones 34,35 y 36, los términos  $P'' R_x, P'' R_y$  y  $P'' R_z$ , son la multiplicación del estado actual de los vértices del modelo por una matriz de rotación de 180 grados en los ejes correspondientes.

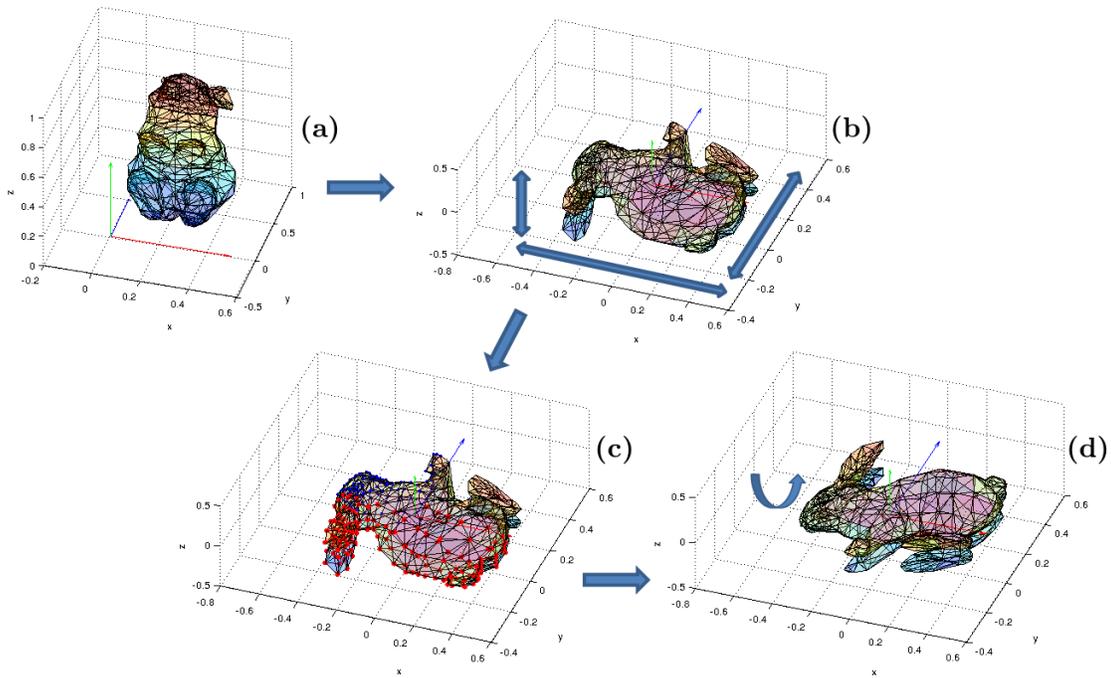


Figura 25: Proceso de control de la invariante de rotación.

#### 4.3.2. Invariante de escala

El tamaño de la figura se determina modificando sus dimensiones, donde ningún punto quede fuera del intervalo  $(0,1)$  de valores reales.

Tomando como referencia el centro geométrico de la figura, ningún vértice queda fuera de los límites de una esfera de radio  $1/2$ . Definiendo  $r_{max}$  como la distancia máxima de una figura a partir de su centro geométrico,  $r_{max}$  se calcula como sigue:

$$r_{max} = \sqrt{\max(x_i'')^2 + \max(y_i'')^2 + \max(z_i'')^2} \quad (37)$$

Obteniendo finalmente el factor de escala:

$$s = \frac{1}{2r_{max}} \quad (38)$$

En lo siguiente, para cada punto en  $P''$ , se realiza la transformación y se obtiene el conjunto de puntos escalados:

$$P''' = \{(sx_i'', sy_i'', sz_i'') | (x_i'', y_i'', z_i'') = P_i'' \in P'', i = 1, \dots, n\} \quad (39)$$

De manera análoga, utilizamos una matriz  $M_s$  para realizar la operación sobre el conjunto de puntos  $P''$ .

$$M_s = \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix} \quad (40)$$

$$P''' = P'' M_s \quad (41)$$

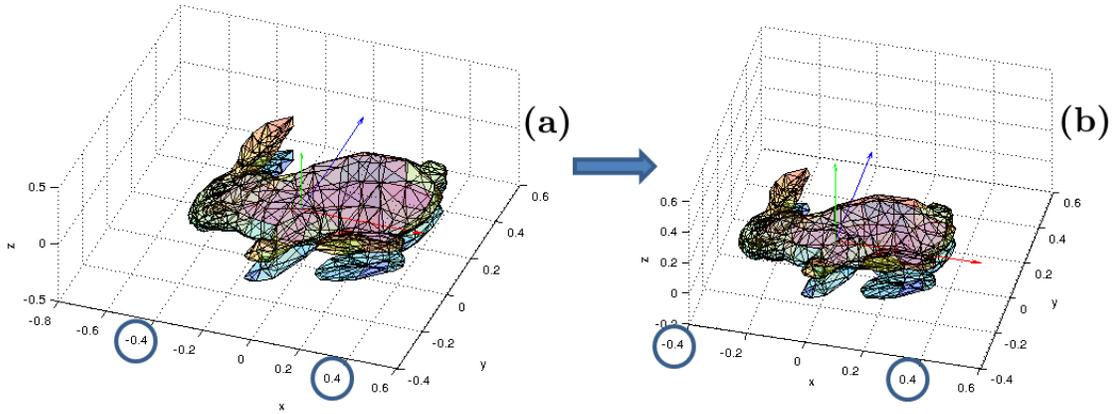


Figura 26: Proceso de control de invariante de escala.

### 4.3.3. Invariante de traslación

Considerando que se han controlado las invariantes de rotación y escala, el modelo conserva la misma estructura topológica que en el modelo original. Es entonces que todos los datos de la figura son trasladados a un marco de referencia con valores dentro del intervalo (0,1), con su centro fijo en las coordenadas:

$$m_c = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right) \quad (42)$$

Donde el factor de traslación se suma para cada componente de cada punto [A.1.2.3](#).

$$P'''' = \left\{ \left( x_i''' + \frac{1}{2}, y_i''' + \frac{1}{2}, z_i''' + \frac{1}{2} \right) \mid (x_i''', y_i''', z_i''') = P_i''' \in P''', i = 1, \dots, n \right\} \quad (43)$$

De manera análoga se define una matriz de traslación  $M_T$ .

$$M_T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix} \quad (44)$$

Se agrega una dimensión a los puntos  $P'''$  según (45).

$$P''''_{aux} = \begin{bmatrix} x_1''' & y_1''' & z_1''' & 1 \\ x_2''' & y_2''' & z_2''' & 1 \\ \vdots & \vdots & \vdots & 1 \\ x_n''' & y_n''' & z_n''' & 1 \end{bmatrix} \quad (45)$$

Así  $P''''_{aux}$  y  $M_T$  permite calcular la operación de traslación.

$$P''''_{aux} = P''''_{aux} M_T \quad (46)$$

Finalmente obtenemos los puntos trasladados a un punto fijo en las primeras 3 dimensiones de  $P''''_{aux}$ , por lo que se ignora la dimensión agregada.

$$P'''' = \{ P''''_{aux}(x_i'''' , y_i'''' , z_i'''' ), i = 1, \dots, n \} \quad (47)$$

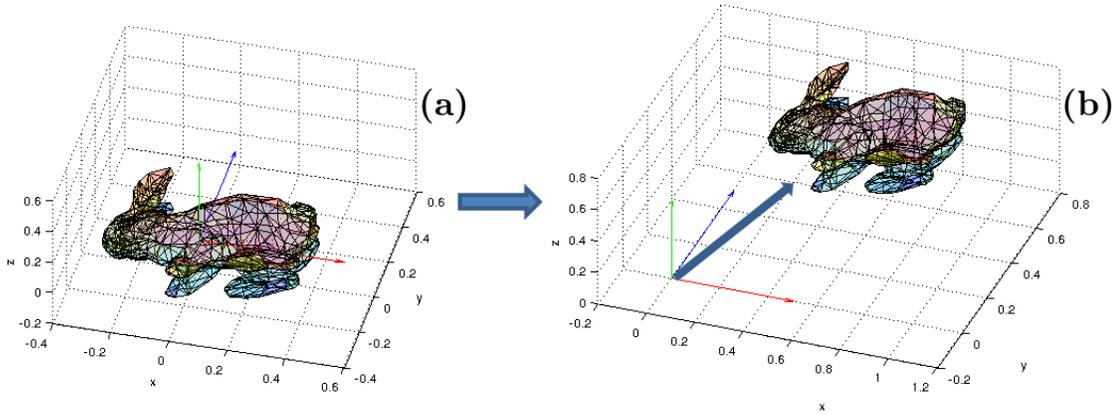


Figura 27: Proceso de control de invariante de traslación.

#### 4.4. Construcción de secuencias representativas

La siguiente etapa, utiliza el resultado de la etapa de preprocesamiento para realizar la transformación de dominio de 3D a 1D ( $\mathbb{R}^3 \rightarrow \mathbb{R}^1$ ), dicha transformación se realiza mediante tres procedimientos distintos. El procedimiento general de transformación de 3D a 1D consiste en realizar un registro de medidas sobre la superficie del modelo en un orden específico para conformar una secuencia numérica, también denominada secuencia representativa. La secuencia numérica se almacena en un archivo plano que únicamente consiste en una columna de datos, donde cada renglón es una medida registrada. Como la transformación de 3D a 1D se realiza mediante tres procedimientos distintos, se tienen tres archivos donde en cada uno hay una secuencia representativa. Cada archivo se nombra usando el identificador de figura y la distinción del procedimiento de extracción utilizado. La secuencia numérica de mediciones es una representación más sencilla de analizar, administrar y almacenar.

A lo largo del apartado se proponen diferentes representaciones que permitan derivar una estructura unificada, capaz de mantener componentes descriptivos globales de la figura 3D. Para lograrlo se parte de la observación de la naturaleza del problema, consideración que los métodos de comparación en 3D clásicos, dejan como menos importante. Por ejemplo, en la disciplina de reconocimiento de patrones se considera como primera aproximación la creación de vectores de características, útiles para procesar computacionalmente y generar clases a partir de su espacio de representación. Es común ver que al elegir las características de un vector, tal vez no contenga componentes descriptivos de una figura 3D. Un valor que se considera regularmente es el volumen;

es claro que dos figuras 3D con condiciones geométricas totalmente diferentes pueden tener el mismo volumen.

Las secuencias representativas tienen como objetivo registrar un mapeo de las figuras 3D en 1D, donde el resultado sirva para el criterio de comparación.

Para generar una secuencia representativa, se parte de la idea de tomar muestras de un objeto 3D con invariantes controladas de rotación, escala y posición. Las muestras se toman de manera regular sobre la estructura poligonal, utilizando vectores con dirección como se visualiza en la figura 29. Las muestras se obtienen a partir de medir la distancia desde puntos de una función paramétrica  $o(x_o, y_o, z_o)$  en una dirección  $b(x_b, y_b, z_b)$ .

Cada valor medido se registra en una secuencia numérica, donde cada valor mide la extensión ( $xt$ ) en la dirección  $b$  hasta encontrar el primer punto de intersección entre algún triángulo de la malla y el vector, con el origen en  $o$  en la dirección  $b$ , un ejemplo se observa en la figura 28. Si el vector no tiene intersección con algún polígono, la extensión medida se considera con un valor de 0. El número de muestras está dado por la función paramétrica, que en un registro uniforme, capta suficiente información del objeto.

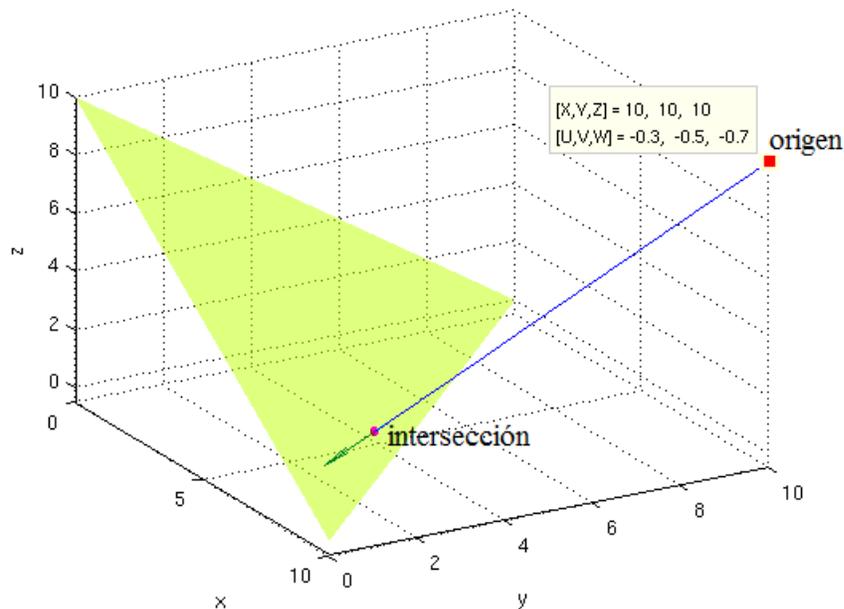


Figura 28: Intersección de un vector de dirección y triángulo en 3D.

La secuencia representativa se construye mediante el seguimiento de una trayectoria espacial, dada por el orden en que se toman las distancias al objeto desde un origen, las cuales se generan por los algoritmos de los siguientes apartados. Cada uno de los algoritmos tiene una consideración importante, que consiste en que se pueda registrar la mayor cantidad de superficie de la figura en 3D. En la figura 29 se muestra la intuición del cómo se concibió la idea para el análisis.

Cada uno de los algoritmos cuenta con un parámetro  $t$ , que permite ajustar la cantidad de mediciones. Dichas mediciones se distribuyen regularmente según el marco de referencia, así que la cantidad de mediciones está dada en función de  $t$ .

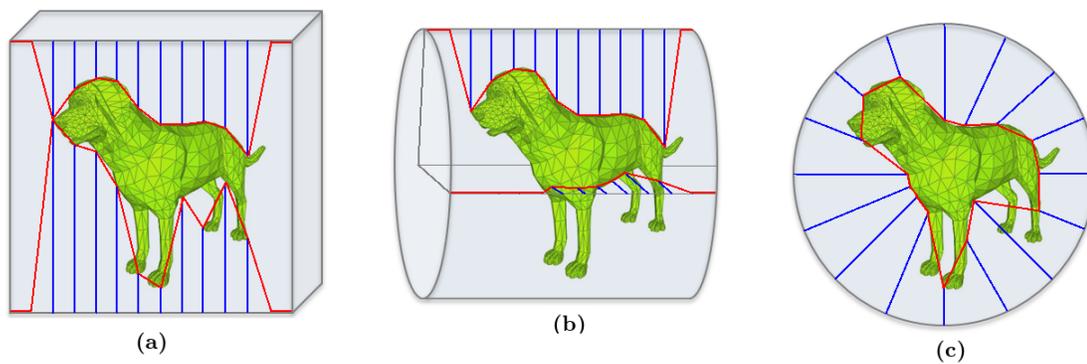


Figura 29: (a) Representación de medidas usando como marco de referencia un cubo. (b) Representación usando un cilindro. (c) Representación usando una esfera.

#### 4.4.1. Representación de medidas sobre un cubo ( $R1$ )

La idea de los algoritmos tienen como base la observación de la figura 3D desde un marco de referencia fijo. El marco de referencia se concibió como una estructura de condiciones geométricas. Sobre el marco de referencia se especifican distintos puntos de observación que sirven como referencia para realizar una medición de distancia desde el punto de observación al modelo en 3D.

El primer algoritmo parte de la idea de tener 6 caras de un cubo, e ir colocando los puntos de observación en un orden predeterminado (ver figura 30). Las mediciones desde los puntos de observación sobre el cubo, están dadas por un orden definido.

El algoritmo 4.1 muestra la idea general para caras laterales del cubo, en el anexo A.4.1 se muestra el algoritmo completo para generar todos los puntos de observación.

---

**Algoritmo 4.1** Algoritmo base para la representación de medidas sobre un cubo.

---

Inicializar  $x_b \leftarrow 1/2$

Inicializar  $y_b \leftarrow 1/2$

Repetir para  $y_o = 0, \dots, 1$  en intervalos de  $1/t$

    Inicializar  $z_o = 0, x_{pos} = 0$

    Repetir para  $i = 0, \dots, (2t + 1)$  en intervalos de 1

        Asignar  $x_o \leftarrow x_{pos}$

        Asignar  $x_b \leftarrow x_{pos}$

        Si  $i \geq t - 1$

            Asignar  $x_{pos} \leftarrow x_{pos} - (1/t)$

            Asignar  $z_o \leftarrow 1$

        Sino

            Asignar  $x_{pos} \leftarrow x_{pos} + (1/t)$

        Fin Si

    Obtener intersección del rayo con origen  $o(x_o, y_o, z_o)$  y dirección  $b(x_b, y_b, z_b)$

    Fin Repetir

Fin Repetir

---

La cantidad de elementos en la secuencia de medidas en función de  $t$  se define en la ecuación (48).

$$N = 6(t^2 + 2t + 1) \quad (48)$$

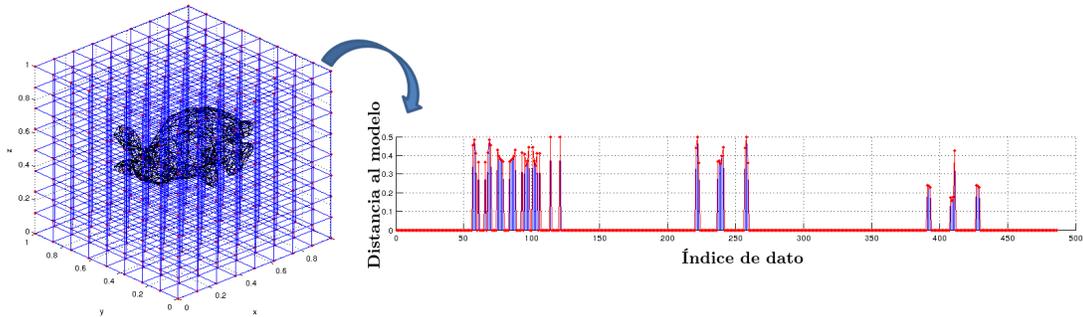


Figura 30: Puntos de observación sobre un cubo como marco de referencia.

#### 4.4.2. Representación de medidas sobre un cilindro ( $R^2$ )

Virtualmente se generan puntos de observación sobre un cilindro en cada eje  $(x, y, z)$ . Finalmente se genera el orden en que se realizan las medidas sobre los cilindros. El

resultado es una representación que contiene secciones de siluetas consecutivas, parecidas a “rebanadas” de la figura.

---

**Algoritmo 4.2** Algoritmo base para el registro de medidas sobre un cilindro.

---

Inicializar  $x_b \leftarrow 1/2$

Inicializar  $y_b \leftarrow 1/2$

Inicializar  $z_b \leftarrow 1/2$

Inicializar  $r_c \leftarrow 1/2$

Repetir para  $\theta = 0, \dots, (2 - 1/t)$  en intervalos de  $2(1/t)$

    Repetir para  $h = 0, \dots, 1$  en intervalos de  $1/t$

        Asignar  $x_o \leftarrow h$

        Asignar  $y_o \leftarrow r_c \cos(\theta\pi) + y_c$

        Asignar  $z_o \leftarrow r_c \sen(\theta\pi) + z_c$

        Asignar  $x_b \leftarrow h$

        Obtener intersección del rayo con origen  $o(x_o, y_o, z_o)$  y dirección  $b(x_b, y_b, z_b)$

    Fin Repetir

Fin Repetir

---

El algoritmo 4.2 muestra la idea general para un sólo cilindro a lo largo de un eje, en el anexo A.4.2 se muestra el algoritmo completo para generar todos los puntos de observación, que es equivalente a colocar un cilindro a lo largo de cada eje  $(x, y, z)$ .

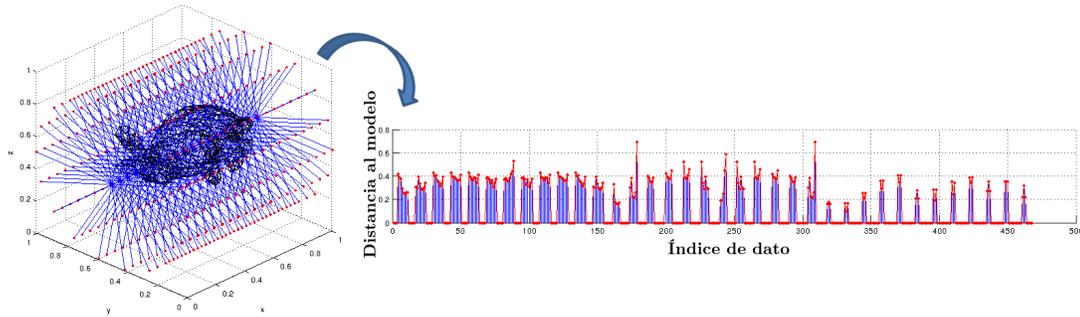


Figura 31: Puntos de observación sobre cilindros como marco de referencia.

La cantidad de elementos en la secuencia de medidas en función de  $t$  se define en la ecuación (49).

$$N = 3(t^2 + t) \quad (49)$$

#### 4.4.3. Representación de medidas sobre una esfera ( $R^3$ )

Para este algoritmo (4.3) se considera el poder analizar al modelo 3D en una forma bastante regular, con la propiedad de eliminar algunos datos de mediciones sin intersección con la malla de polígonos. Las observaciones se realizan desde puntos de observación en un marco de referencia con forma de esfera.

---

**Algoritmo 4.3** Algoritmo de trayectoria de puntos de observación describiendo una esfera.

---

Inicializar  $x_b \leftarrow 1/2$

Inicializar  $y_b \leftarrow 1/2$

Inicializar  $z_b \leftarrow 1/2$

Inicializar  $r_c \leftarrow 1/2$

Repetir para  $\theta = 0, \dots, (1 - (1/t))$  en intervalos de  $2(1/t)$

    Repetir para  $\phi = 0, \dots, (2 - (1/t))$  en intervalos de  $1/t$

        Asignar  $x_o \leftarrow r_c \cos(\phi\pi) + x_c$

        Asignar  $y_o \leftarrow r_c \cos(\theta\pi) \text{sen}(\phi\pi) + y_c$

        Asignar  $z_o \leftarrow r_c \text{sen}(\theta\pi) \text{sen}(\phi\pi) + z_c$

        Obtener intersección del rayo con origen  $o(x_o, y_o, z_o)$  y dirección  $b(x_b, y_b, z_b)$

    Fin Repetir

Fin Repetir

---

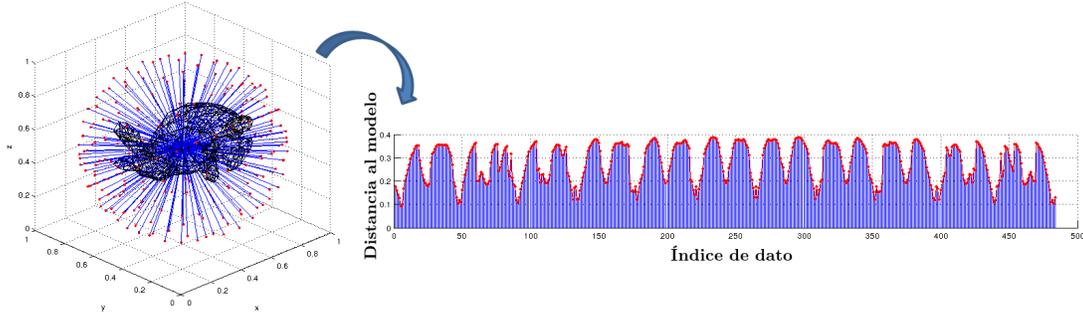


Figura 32: Puntos de observación sobre una esfera como marco de referencia.

La cantidad de elementos en la secuencia de medidas en función de  $t$  se define en la ecuación (50).

$$N = (t^2 + t) \quad (50)$$

#### 4.4.4.Reconstrucción parcial de figuras 3D

Distintas técnicas de análisis de similitud en 3D existentes, no cuentan con la capacidad de reconstruir la figura 3D a partir de la extracción de descripciones. Por ejemplo, las técnicas que tienen como base el análisis estadístico de distribución de masa, que al dividir el espacio y contabilizar la presencia o ausencia de volumen, pierden características que constituyen a la figura, siendo imposible la reconstrucción del modelo analizado.

Las técnicas propuestas en este trabajo, a diferencia de las del estado del arte, se cuenta con la capacidad de obtener una reconstrucción parcial de conjunto de vértices  $V$  en 3D, que corresponden con la superficie del modelo, lo anterior se origina a partir de la secuencia de medidas en 1D de cualquiera de las 3 técnicas.

Para realizar la reconstrucción de una figura se considera la representación en 1D como una secuencia de valores. Los valores consisten en medidas de extensión, tomadas desde los puntos de referencia origen  $o(x_0, y_0, z_0)$  con dirección  $b(x_b, y_b, z_b)$  almacenadas en una secuencia numérica:

$$st_i \tag{51}$$

Donde  $i \in \{0, 1, 2, \dots, N\}$  y  $N$  es el tamaño predeterminado de la secuencia. Los siguiente, es procesar la secuencia de medidas de la misma forma en que fue obtenida, tomando la extensión a la que se encontraba el punto de origen en el marco de referencia, cambiándola por la extensión registrada en la secuencia. El algoritmo 4.4 toma la idea de registro sobre una esfera (4.3) y describe lo anterior.

---

**Algoritmo 4.4** Algoritmo de reconstrucción parcial de 1D a 3D.

---

Inicializar  $r_c \leftarrow 1/2$

Inicializar  $i \leftarrow 0$

Repetir para  $\theta = 0, \dots, (1 - (1/t))$  en intervalos de  $2(1/t)$

    Repetir para  $\phi = 0, \dots, (2 - (1/t))$  en intervalos de  $1/t$

        Si  $st_i \neq 0$

            Asignar  $x \leftarrow (r_c - st_i)\cos(\phi\pi) + x_c$

            Asignar  $y \leftarrow (r_c - st_i)\cos(\theta\pi)\sen(\phi\pi) + y_c$

            Asignar  $z \leftarrow (r_c - st_i)\sen(\theta\pi)\sen(\phi\pi) + z_c$

            Asignar  $i \leftarrow i + 1$

            Regresar punto en 3D  $pr(x, y, z)$

        Fin Si

    Fin Repetir

Fin Repetir

---

Usando el pseudocódigo de reconstrucción y graficando los puntos en 3D en el orden en que fueron tomados, es posible representar gráficamente al modelo, por medio de líneas uniendo los puntos, siguiendo la trayectoria que se siguió en el registro de la representación.

Como ejemplo, en la figura 33 se puede observar la figura 3D original y su reconstrucción parcial a partir de la representación en 1D. En el apartado 5.11.1 se muestran más ejemplos de reconstrucción de 1D a 3D.

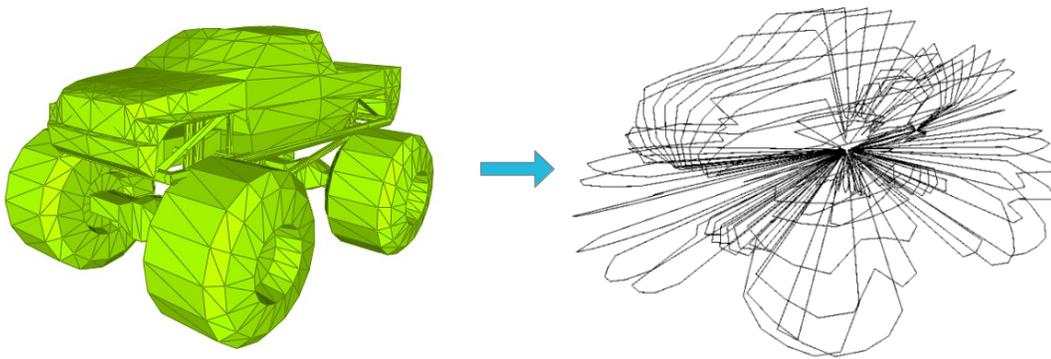


Figura 33: Ejemplo de figura 3D y su reconstrucción parcial a partir de una secuencia numérica en 1D.

El mismo procedimiento puede aplicarse haciendo las adecuaciones para los algoritmos de registro de medidas 4.1 y 4.2, únicamente considerando que si  $st_i$  es igual a cero, no considerar el dato como valor que pertenece a la figura.

#### 4.4.5. Robustez de las técnicas de representación

Una de las propiedades de las técnicas propuestas, consiste en que son robustas ante diferentes resoluciones de modelos en 3D.

En la figura 34 se muestra la metodología que consiste en la selección de una figura 3D, a la cual se le aplican simplificaciones de cantidad de polígonos en 3 etapas. En cada etapa la cantidad de polígonos es aproximadamente de la mitad de la etapa anterior. Posteriormente, se compara la figura original con respecto a las otras 3 figuras. El procedimiento se realiza para las 3 técnicas de representación propuestas.

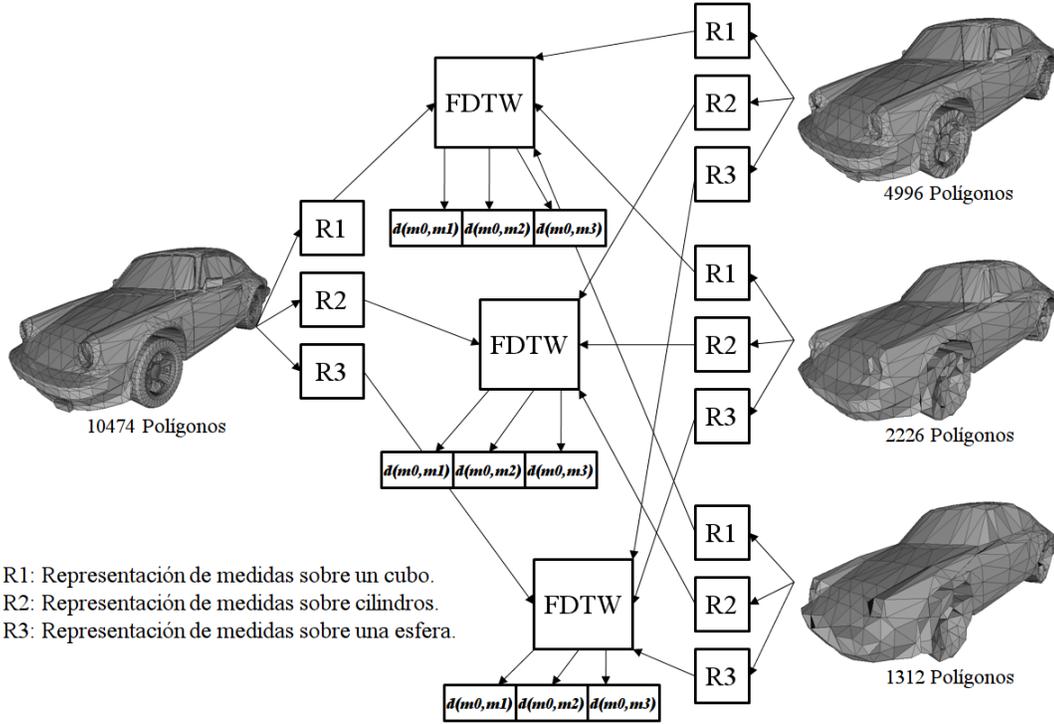


Figura 34: Metodología de robustez para distintas resoluciones.

#### 4.4.6. Propiedades de las secuencias representativas

Debido al control de invariantes se obtienen otras propiedades para las secuencias representativas. Se enumeran las propiedades generales que se obtienen mediante el proceso de establecer un marco de referencia de análisis fijo.

1. Los valores de las secuencias numéricas son siempre positivos.
2. Cada dato de la secuencia tiene un valor dentro del intervalo  $[0, 1) \in \mathbb{R}$ .
3. El preprocesamiento de datos 3D permite que los valores se mantengan dentro de la misma escala para un mismo tipo de representación, sin importar la forma o distribución espacial del modelo.

Las propiedades mencionadas permiten que las medidas de distancia como *FDTW* y *Minkowski* ( $L_3$ ) tengan una estabilidad importante. Por ejemplo si modificamos la distancia base de *FDTW* no tiene efecto en la relación de similitud entre las representaciones, únicamente cambia la proporción de las distancias calculadas.

#### 4.5.Comparación de las secuencias representativas

La etapa de comparación consiste en utilizar la técnica de *Fast Dynamic Time Warping* como medida de similitud entre secuencias representativas. Para calcular la medida de similitud se toma cada secuencia representativa y se determinan los valores de distancia con respecto a las demás secuencias. En la figura 35 nótese que la comparación se determina entre secuencias con origen en la misma técnica de representación.

Al terminar todas las comparaciones se tiene una matriz de distancias por cada técnica de representación. La matriz se especifica con más detalle en el apartado 4.11.

A partir de la matriz, se realizan recuperaciones de cualquier figura. El principio consiste en obtener una lista ordenada de menor a mayor con respecto al valor de distancia. Mediante la recuperación de cada figura se observa la respuesta, permitiendo de manera automática, registrar los indicadores de precisión y recuperación bajo un criterio de categorización dado.

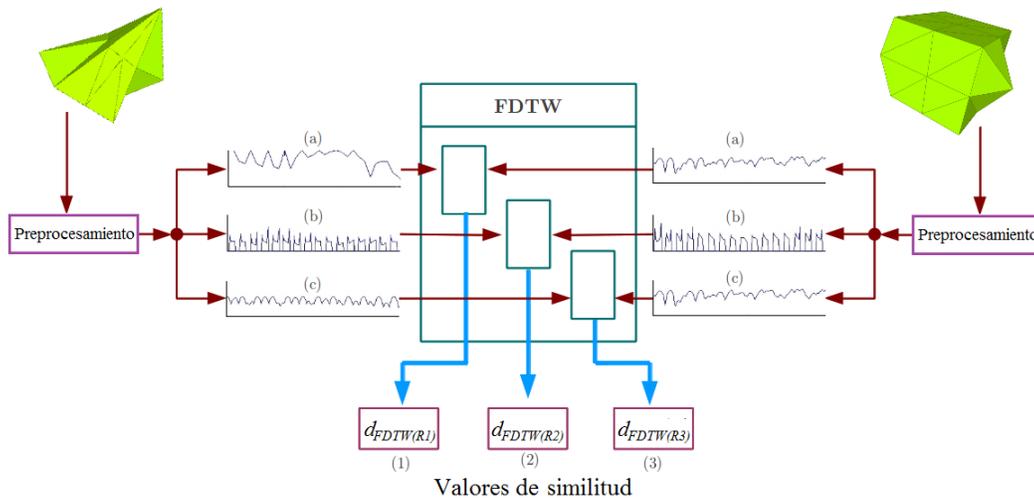


Figura 35: Comparación de dos figuras 3D usando *FDTW*. Las secuencias se obtienen a partir de: (a) Representación de medidas sobre un cubo. (b) Representación de medidas sobre cilindros. (c) Representación de medidas sobre una esfera. El resultado son valores de similitud en (1), (2) y (3).

##### 4.5.1. Combinación de valores de distancia

Para beneficiar los resultados se propone realizar una conjugación de aportaciones de distancias, tomando como principio hacer la combinación de matrices de distancias.

Mediante un análisis de correlación entre secuencias representativas de una misma figura 3D, existe poca relación y por otro lado, cada técnica obtiene mejores resultados de recuperación para clases de figuras distintas. Se determina realizar la conjugación de valores de distancia para mejorar el desempeño, partiendo de que las representaciones consideran diferentes aspectos de la figura. El tener a cada técnica de representación con los mejores resultados en clases distintas es evidencia de que no existe técnica que sea eficaz en todos los casos y esto por el teorema de *No-Free Lunch* [39].

Se parte de la idea de que cada técnica considera diferentes aspectos de las figuras 3D. Existen formas sofisticadas para combinar técnicas, una alternativa es usar un esquema básico de suma de distancias para obtener un nuevo valor de distancia ( $d_T$ ). Donde  $d_T = d_{FDTW(R1)} + d_{FDTW(R2)} + d_{FDTW(R3)}$ , donde  $d_{FDTW(R1)}$ ,  $d_{FDTW(R2)}$  y  $d_{FDTW(R3)}$  son las distancias obtenidas en la comparación de representaciones para cada técnica  $R1$ ,  $R2$  y  $R3$ .

El procedimiento consiste en utilizar las matrices de distancias  $D_{FDTW(R1)}$ ,  $D_{FDTW(R2)}$  y  $D_{FDTW(R3)}$ , las cuales fueron obtenidas para cada técnica de representación, de donde se obtiene una nueva matriz  $D_T$  de distancias, con la aportación de cada técnica en la distancia final de una comparación de secuencias representativas como se indica en la ecuación (52).

$$D_T = D_{FDTW(R1)} + D_{FDTW(R2)} + D_{FDTW(R3)} \quad (52)$$

La figura 36 ilustra el cálculo de la nueva matriz de distancias a partir de las matrices de distancias  $D_{FDTW(R1)}$ ,  $D_{FDTW(R2)}$  y  $D_{FDTW(R3)}$ .

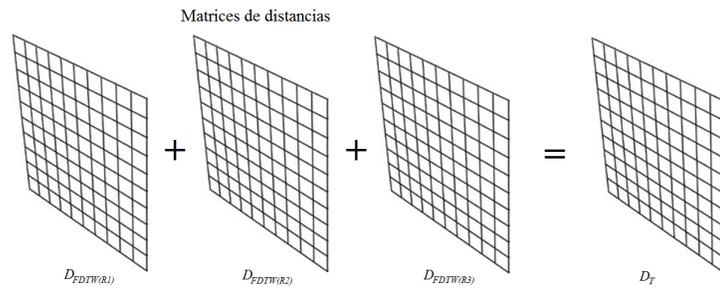


Figura 36: Suma de matrices de distancias.

#### 4.6. Complejidad computacional de las técnicas

En el análisis de algoritmos existen dos aspectos muy importantes a considerar los cuales son útiles en la comparación entre técnicas, métodos o algoritmos que dan una solución a

algún problema. Dichos aspectos son la complejidad temporal y la complejidad espacial. La complejidad temporal de un problema es el número de pasos que toma resolver una instancia de un problema, a partir del tamaño de la entrada utilizando el algoritmo más eficiente a disposición. Intuitivamente, si se toma una instancia con entrada de longitud  $n$  que puede resolverse en  $n^2$  pasos, se dice que ese problema tiene una complejidad en tiempo de  $n^2$ .

A continuación se describe la complejidad calculada para cada una de las etapas que componen la metodología.

La consideración de  $n$  está dada de la siguiente manera: para la etapa 1 se considera a  $n_v$  como la cantidad de vértices de una figura y  $m_v$  para la segunda figura 3D. Para la etapa 2 se considera a  $n_p$  como la cantidad de polígonos de un figura y  $m_p$  para la segunda figura 3D.

Se considera al parámetro  $n$  como el tamaño de las secuencias representativas. En todos los casos las expresiones representan a las cotas máximas.

1. Complejidad del preprocesamiento de control de invariantes:
  - a) Rotación:  $(2n_v + k + (n_v \times 3 \times 4)) + (2m_v + k + (m_v \times 3 \times 4)) = 14n_v + 14m_v + 2k$
  - b) Escala:  $(n_v \times 3 \times 3) + (m_v \times 3 \times 3) = 9n_v + 9m_v$
  - c) Traslación:  $(n_v \times 4 \times 4) + (m_v \times 4 \times 4) = 16n_v + 16m_v$
2. Complejidad de inserción de polígonos en una estructura espacial para el análisis:  $n_p^2 + m_p^2$  para el peor caso y  $(n_p \log n_p) + (m_p \log m_p)$  cuando los polígonos están distribuidos de manera uniforme en la caja que delimita la figura [36].
3. Complejidad del análisis de intersección rayo-polígono en 3D para la construcción de secuencias representativas:  $(\log n)$  [36].
4. Complejidad del alineamiento de secuencias representativas usando *FDTW*:  $2 \times (4n^2 + 2n)$
5. Complejidad teórica del proceso completo:  $8n^2 + n_p^2 + m_p^2 + 39n_v + 39m_v + 4n + (\log n)$

En 1(a)  $k$  es una constante resultado de determinar los *eigenvalores* y *eigenvectores* de la matriz de covarianza de tamaño constante de  $3 \times 3$ . La complejidad espacial esta dada en función de la cantidad de figuras  $|S|$  y en función de la cantidad de elementos de las secuencias. El espacio que ocupan todas las secuencias es de  $N * |S|$  y el de una matriz de distancias es de  $|S| \times |S|$ .

## 4.7. Indicador de similitud entre figuras 3D

La naturaleza de las secuencias representativas, incorporando una técnica de comparación adecuada para la representación, determina a  $FDTW$  como una medida de distancia entre secuencias numéricas comparables.

**Definición 6.** Dados dos modelos en 3D  $A$  y  $B$ , el esfuerzo mínimo de alineación entre los registros de superficie está dado por:

$$similitud(A, B) = FDTW(g(A), g(B)) \quad (53)$$

Donde  $g$  son las transformaciones que consisten en el preprocesamiento de datos de los modelos 3D y la transformación de dominio de 3D a 1D, donde se obtiene como resultado, una secuencia numérica.

### 4.7.1. Propiedades del indicador de similitud

El indicador de similitud o función de distancia propuesta, cumple con las siguientes propiedades definidas en el apartado 3.2.2.1. Considere que  $a$  y  $b$  son figuras 3D de un conjunto  $S$ .

1. No negatividad:  $similitud(a, b) \geq 0$  para todo  $a, b \in S$
2. Identidad:  $similitud(a, a) = 0$  para todo  $A \in S$
3. Unicidad:  $similitud(a, b) = 0$  implica necesariamente que  $A = B$
4. Simetría:  $similitud(a, b) = similitud(b, a)$  para todo  $a, b \in S$

### 4.7.2. Relación *distorsión-distancia*

Se propone establecer una relación *grado de distorsión* contra *distancia*. En dicha relación, se tiene como principio que en medida de que el grado de distorsión aumente con respecto a una figura base, la distancia entre la figura base y la figura modificada también deberá aumentar.

La comparación considera un conjunto de figuras 3D  $S = \{s_1, s_2, \dots, s_n\}$  de  $n$  elementos, la secuencia del aumento de la distorsión sobre la figura base se especifica como:

$$E = \{e_1, e_2, \dots, e_n\} \quad (54)$$

Donde  $E$  es la secuencia de valores de distancias,  $e_i = similitud(s_1, s_{i+1})$  para  $i = 0, 1, 2, \dots, n$ .

### 4.7.3. Relación *diferencia de distancias* entre *distorsiones consecutivas*

Se propone establecer una relación de *distorsiones consecutivas* contra *diferencia entre distancias*. La relación propuesta parte de la secuencia en  $E = \{e_1, e_2, \dots, e_n\}$  (54) de  $n$  comparaciones, la secuencia que caracteriza el cambio en la distancia, de acuerdo al aumento de la distorsión sobre la figura base se especifica como:

$$V_d = \{v_1, v_2, \dots, v_{n-1}\} \quad (55)$$

Donde  $V_d$  es la secuencia de diferencias entre parejas de distancias,  $v_i = |e_i - e_{i+1}|$  para  $i = 1, 2, \dots, n - 1$ .

La relación muestra el cambio proporcional entre grados de distorsión consecutivos. La interpretación es la siguiente; cuando el cambio es estable o se mantiene en un rango acotado, la medida de similitud es constante respecto al grado de distorsión. En el caso de que el cambio sea desproporcional, es indicativo de que la similitud entre grados de distorsión consecutivos cambia su estabilidad. Es posible utilizar la interpretación para determinar un umbral de similitud para cada técnica de representación.

## 4.8. Análisis de la recuperación

Utilizando la especificación del *PSB*, las gráficas de *precisión-recuperación* se generan a partir de un proceso que corresponde con (15) que tiene como entradas un archivo de clasificación y una matriz de distancias. La secuencia de pasos del proceso es la siguiente.

1. Leer un archivo de clasificación con extensión \*.cla, que especifica el orden de los modelos con su correspondencia a cada clase.
2. Crear un mapeo de la posición del modelo a una clase.
3. A partir de la matriz de distancias calcular el desempeño por modelo.
4. Promediar el valor de *precisión* considerando cada modelo para los valores de *recuperación* especificados.

## 4.9. Los conjuntos de datos

### 4.9.1. Figuras 3D del *PSB*

El conjunto de datos se compone de un subconjunto de entrenamiento (907 figuras en 90 clases) y un subconjunto de pruebas (907 figuras en 92 clases). Para el caso del subconjunto de entrenamiento, se sugiere utilizarlo para sistemas o algoritmos que

requieren conjuntos de supervisión. Se aclara que la clasificación base con la que cuenta el *PSB* es inducida por funcionalidad y forma de los modelos.

En general las figuras del *PSB* tienen resolución variada, son figuras irregulares, de formas toscas y contienen degeneraciones como triángulos no conectados, de tamaños variables y posiciones arbitrarias, en la figura 37 se muestran algunos ejemplos.

La primera consideración fue el estudio de los datos con los que se contaba, agregándolos a un sistema de base de datos convencional. Mediante observaciones se pudo determinar lo siguiente.

Las figuras representan clases de objetos que son familiares con el mundo real, por ejemplo, animales, plantas, vehículos, herramientas o accesorios (ver figura 37). No incluye clases de figuras de dominios especializados de aplicación, por ejemplo, ingeniería CAD o biología molecular.

Los esquemas definidos de clases en el *PSB* son muy selectivos para la clasificación, agrupando objetos estrictamente por función (concepto semántico) así como agrupación de figuras por geometría.

Con el enfoque especificado de este trabajo, no se considera la relación semántica entre la información geométrica codificada por la nueva representación y la funcionalidad aparente de la figura 3D. Por lo tanto existe el planteamiento interesante de observar el comportamiento de nuestra propuesta ante el hecho de que una tercera parte de las clases presentan figuras con relación a su funcionalidad.

El conjunto de figuras de Princeton, consiste en un archivo de compresión (ya sea *zip*, *tar.gz*), que contiene las figuras poligonales recolectadas de Internet. Para cada figura 3D, existe un formato de archivo de objeto (\*.off) con la geometría poligonal de la figura, información del modelo en un archivo de texto y una imagen en formato *JPEG* con una vista. En total se compone de 1814 figuras 3D.

La información de cada modelo consiste en los datos de identificador de modelo, dirección URL de descarga, URL de referencia, formato original, número de polígonos, componentes conectados, ángulo promedio entre dos planos, caja que delimita a la figura, profundidad promedio, centro geométrico, escala, ejes principales y valores principales.

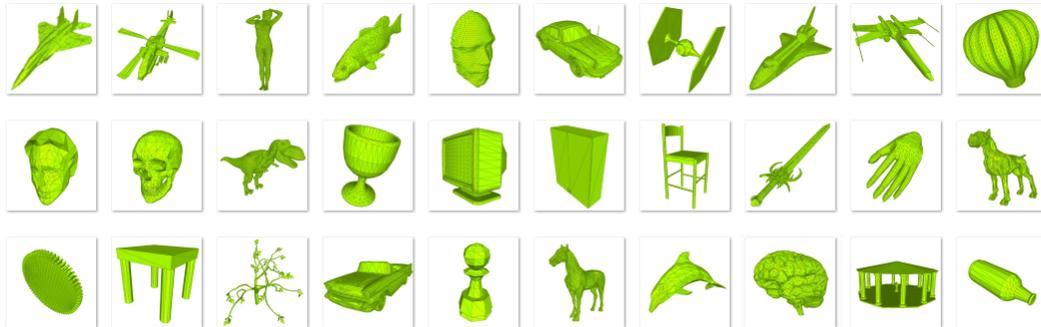


Figura 37: Ejemplos de figuras 3D del *PSB*.

#### 4.9.2. Figuras geométricas primitivas con deformaciones cuantificadas

El conjunto está compuesto de 120 figuras 3D modeladas con malla de polígonos triangulares. Para la construcción del conjunto se tomaron 6 figuras primitivas (cono, cubo, cilindro, toroide, tubo) a las cuales se les aplicaron modificaciones; las cuales consisten en el incremento del grado de deformación de manera sistemática e incremental sobre un vértice de la figura. En medida del incremento en la deformación, se asoció un valor, cuantificando de manera general el grado de deformación. Cada deformación hecha, constituye una figura nueva del conjunto de datos para un total de 20 modelos por figura primitiva.

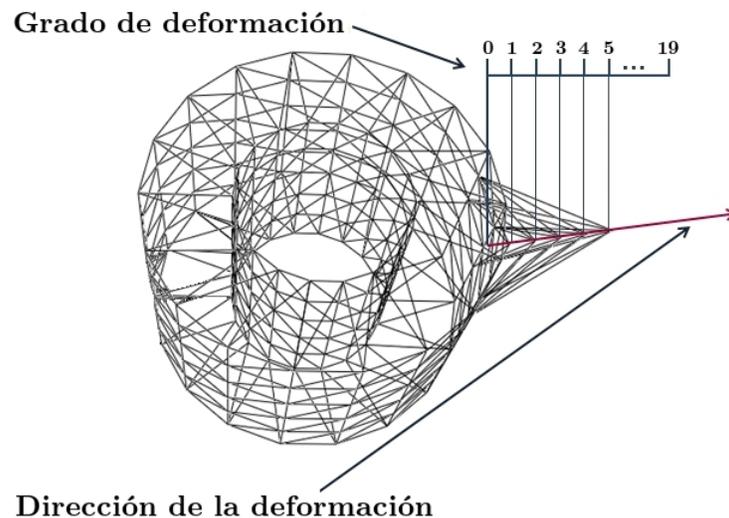


Figura 38: Figura 3D de un “tubo” cuantificando sus deformaciones.

Específicamente se seleccionó un vértice al costado de las figuras primitivas y únicamente se incrementó la componente en  $x$  del vértice. En la figura 39 se muestran ejemplos de las figuras primitivas y cómo se visualiza en 3D la deformación en forma gradual.

Para aplicar el grado de deformación, se utilizó la herramienta *Blender*, la cual es gratuita y de código abierto. *Blender* permite realizar la creación de modelos 3D a partir de figuras primitivas. También permite la modificación de los modelos a nivel de vértices (ver figura 40), permitiendo guardar las modificaciones en el formato *.off* .

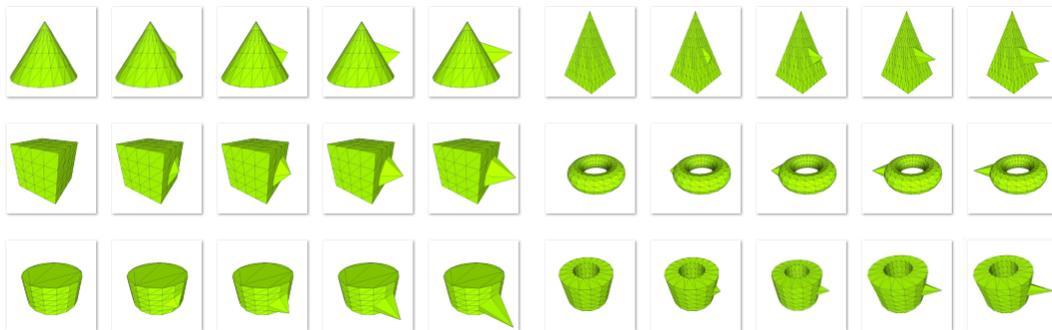


Figura 39: Ejemplos de figuras 3D primitivas con deformaciones graduales.

#### 4.10.El archivo de clasificación

El archivo de clasificación especifica dos fuentes de información. La primera, indica el orden en que los modelos deben de aparecer en la matriz de distancias, dicho orden debe corresponder con los renglones y columnas de la matriz. La segunda consiste en que el archivo proporciona la agrupación de los modelos, permitiendo determinar si un modelo corresponde o no con la agrupación.

El detalle de la descripción del archivo se dispone en (A.3.2).

#### 4.11.La matriz de distancias

La matriz de distancias representa la similitud entre todos los pares de modelos del conjunto a evaluar. Para  $N$  modelos, la matriz es una secuencia de  $N \times N$  valores flotantes en formato binario, donde el valor en la posición  $i * N + j$  representa la distancia entre los modelos  $i$  y  $j$ . Un valor de 0 (cero) indica que las representaciones son idénticas y valores mayores indican menos similitud entre los modelos 3D. Todos los valores de distancia son positivos.

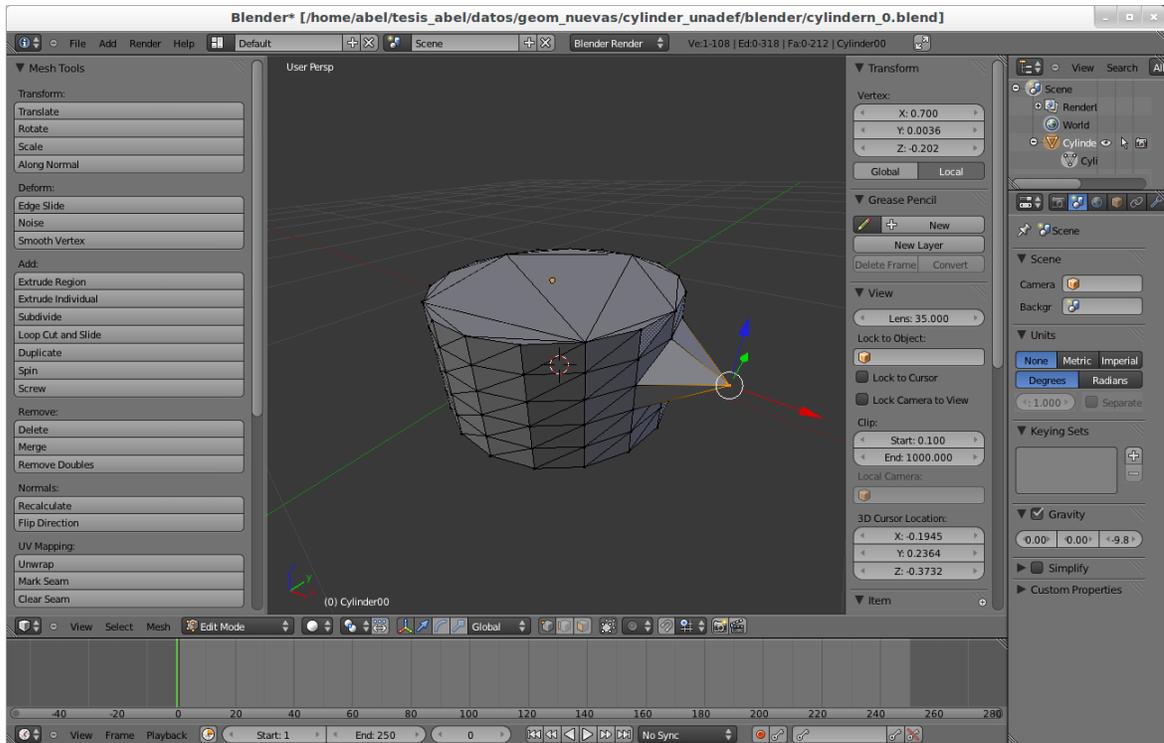


Figura 40: Captura de la herramienta *Blender* editando un vértice de un cilindro en 3D.

#### 4.12. Condiciones experimentales

El diseño de cada experimento requiere de establecer las condiciones requeridas. Las condiciones de un experimento consisten en las opciones con las que se cuentan para realizar un experimento, en la figura 41 se muestran dichas opciones. En primer lugar se selecciona con qué técnica de representación se hará el experimento, luego se determina la cantidad de datos y finalmente se determina la medida de comparación, ya sea utilizar *FDTW* o distancia *Minkowski*.

En la figura 42, se muestra un procedimiento ejemplificando un experimento extenso, únicamente para la técnica de representación *R2*. Se utilizan 4 tamaños distintos se secuencias representativas comparadas con las 2 medidas de distancia. Al final se tienen valores de distancia obtenidos por una combinación de opciones, los valores de distancia sirven para registrar observaciones.

**1) Opciones de técnicas de obtención de representación:**

R1: Representación de medidas usando un cubo como marco de referencia

R2: Representación de medidas usando cilindros como marco de referencia

R3: Representación de medidas usando una esfera como marco de referencia

**2) Opciones de tamaños de secuencias representativas \*:**

N  $\approx$  200    N  $\approx$  500    N  $\approx$  1000    N  $\approx$  1500

\* Son tamaños sugeridos que muestran el comportamiento global de las técnicas.

**3) Opciones de medidas de distancia:**

*Fast Dynamic Time Warping*    Distancia *Minkowski*

Figura 41: Opciones de condiciones experimentales para el análisis de figuras 3D.

#### 4.13. Procedimiento para la visualización de la alineación de secuencias

Como se ha estudiado a lo largo de este trabajo, se parte de la metodología de comparación y recuperación de figuras en 3D y se incorpora la técnica de visualización de alineación en 1D y la reconstrucción parcial en 3D.

En resumen, se tiene la etapa de preprocesamiento, que consiste en el control de invariantes de las figuras 3D. Mediante un registro de medidas de extensión desde un marco de referencia fijo al modelo, se generan secuencias numéricas representativas para cada figura (apartado 4.4). Las secuencias son almacenadas en archivos asociándolos con los modelos.

Lo siguiente es realizar la comparación calculando *FDTW*, lo cual consiste en tomar cada secuencia y se determina su valor de distancia con respecto a todas las demás. Al mismo tiempo se calcula la ruta de ajuste mínima (apartado 3.3.2.2).

Al terminar todas las comparaciones, se tienen los registros de las rutas de ajuste por cada comparación. Ya almacenados los registros, es posible visualizar cualitativamente el impacto de las propiedades locales y globales de una representación con respecto a otra. Para la visualización se hace un análisis de similitud parcial asociando mediante colores la correspondencia de semejanza entre dos representaciones.

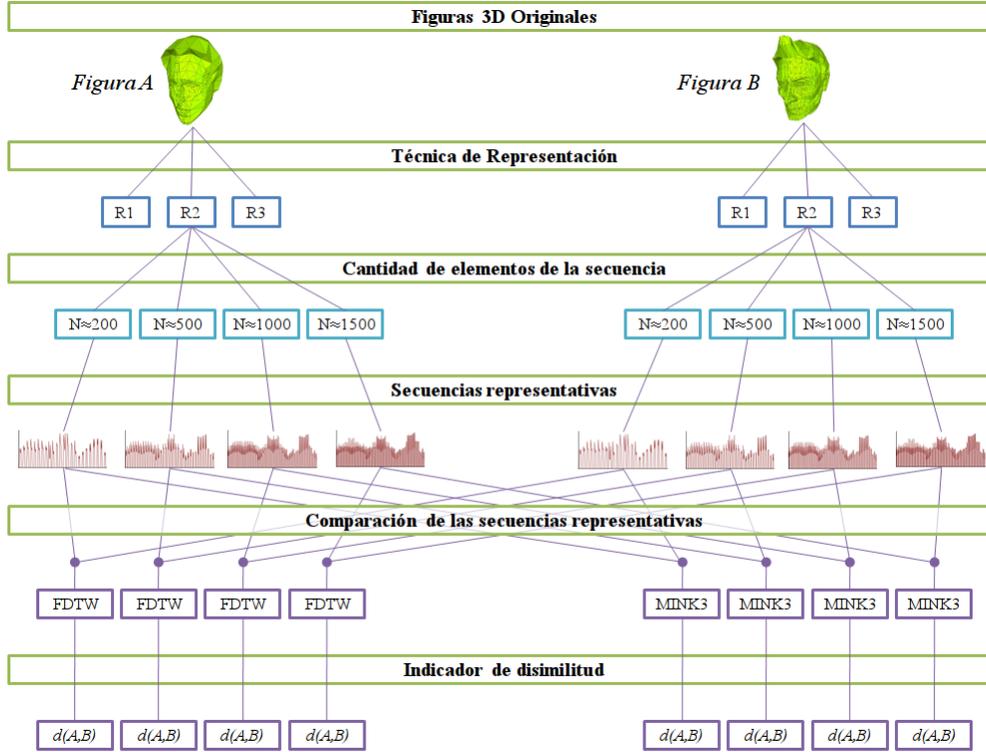


Figura 42: Procedimiento de un experimento extenso para la técnica de representación  $R2$ .

Siendo la ruta de ajuste (*warpath*)  $p = (p_1, \dots, p_L)$ , se tiene que  $p_l = (i, j)$  donde el último par es  $p_L = (I, J)$ . Como se observa  $p_l = (i, j)$  denota una lista de índices que marca la correspondencia de alineación de un punto  $i$  en la secuencia  $Q_i$  con respecto a otro punto  $j$  en la secuencia  $C_j$ . De  $p_l$  podemos tomar los datos  $p_l(i)$  y  $p_l(j)$ . La asignación de colores para una secuencia de tamaño  $N$  se indexa como tomando valores de  $\{1, 2, \dots, N\}$ , un valor por color. Finalmente la asignación de colores para una secuencia  $Q$  está dada por  $SC_Q$ , para una secuencia  $C$  está dada por  $SC_C$ .

La asignación de índices de colores, que corresponde con la alineación entre secuencias está dada por:

Sea  $l \in \{1, 2, \dots, L\}$ ,  $i, j \in \{1, 2, \dots, N\}$ ,

$$SC_Q(p_l(i)) = p_l(i) \quad (56)$$

$$SC_C(p_l(j)) = p_l(j) \quad (57)$$

En la figura 43 se muestra el esquema de la metodología de reconstrucción y visualización de la alineación de secuencias representativas.

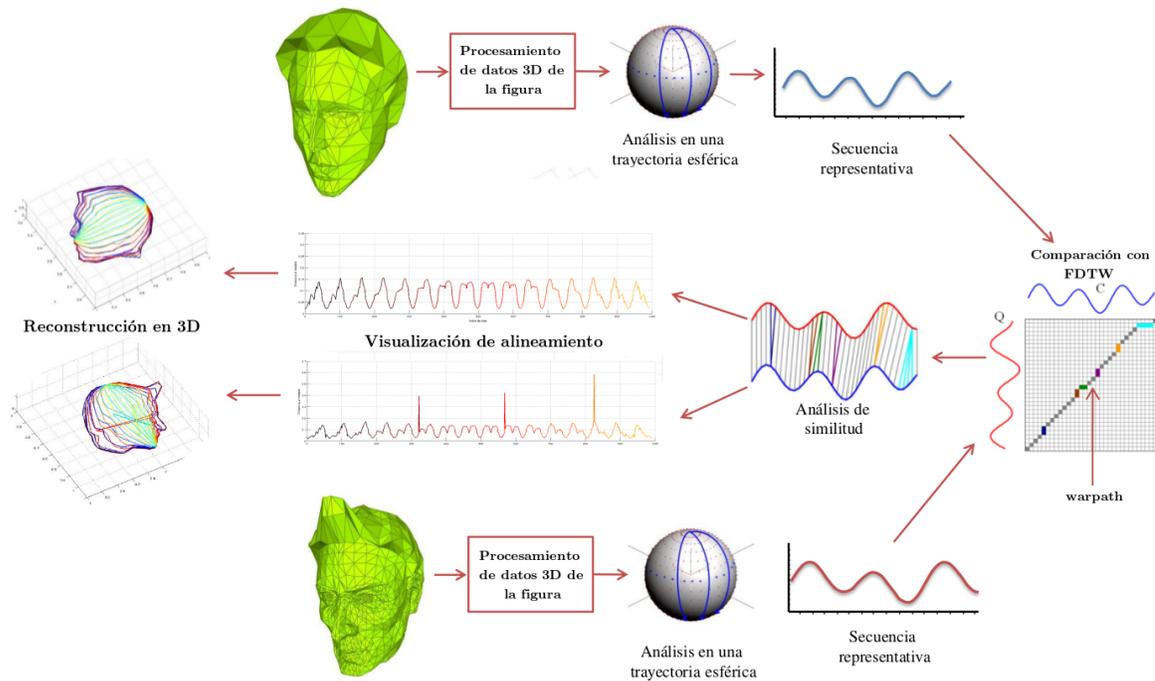


Figura 43: Diagrama de metodología de visualización de la alineación de secuencias y reconstrucción parcial en 3D.

#### 4.14. Comentarios de implementación

Para la implementación de la metodología propuesta en el apartado 4.1, se utilizó como herramienta de desarrollo *MATLAB* en su versión R2010a. La herramienta ayudó a obtener rápidamente resultados indicativos.

El primer paso de implementación consiste en hacer la lectura de las representaciones geométricas a partir de los archivos *\*.off* y realizar el análisis de los modelos para obtener los registros de medidas.

En primera instancia, se utilizó el algoritmo propuesto por Müller and Trumbore [26], que permite la utilización mínima de memoria y no necesita preprocesamiento para calcular la intersección entre un vector dirigido (rayo) y un plano triangular en 3D. El listado completo del algoritmo se encuentra en el anexo A.1.

El algoritmo resulta muy útil cuando se busca la intersección entre rayos y únicamente un triángulo, para el caso de un modelo 3D, se tienen miles o millones de polígonos. La implementación en *MATLAB* presentaba tiempos de análisis estaba en función de  $N * |F|$ , donde  $N$  es la cantidad de medidas que se requieren hacer y  $|F|$  es la cantidad de polígonos a analizar. Los tiempos de experimentación dependieron directamente de la resolución de los modelos.

En una nueva versión de la implementación se utilizó la biblioteca *CGAL* [9] (*Computational Geometry Algorithms Library*, versión 3.8) para el lenguaje de desarrollo *C++*. La biblioteca proporciona un acceso sencillo a algoritmos eficientes y viables comúnmente utilizados en geometría computacional. A diferencia de *MATLAB*, la biblioteca es *GPL* (*General Public License*).

De *CGAL* se utilizó el algoritmo de cálculo de intersección rayo-polígono en 3D, permitiendo realizar 38471 consultas de intersecciones sobre una figura de 14400 polígonos triangulares en un tiempo de 1 segundo.

Para trabajar con matrices y realizar operaciones en 3D se utilizó la biblioteca *Eigen* [34] en su versión 3.0.1. La biblioteca para *C++*, es denominada como *template library for linear algebra*, la cual sirve para realizar operaciones con matrices y vectores. La biblioteca también incorpora algoritmos de soluciones numéricas así como algoritmos relacionados. De la biblioteca se utilizan las estructuras de datos para almacenar los conjuntos de puntos en 3D y realizar operaciones matriciales, así como para el cálculo de *eigenvalores* y *eigenvectores* de matrices cuadradas.

Los experimentos se llevaron a cabo en una PC con procesador Pentium 4 (CPU de 2.4 GHz., 3 GB de RAM).

## RESULTADOS EXPERIMENTALES

---

El este capítulo se presentan los resultados experimentales sobre la prueba de las diferentes técnicas, para el problema de recuperación y análisis de la similitud entre figuras 3D, se utilizan conjuntos de datos homogéneos y heterogéneos. También se muestran pruebas de desempeño y resultados de experimentos de visualización y reconstrucción.

### 5.1.Comentarios generales

Las técnicas de medida de distancia mediante la alineación elástica, establece una relación de orden entre representaciones de figuras 3D. Lo que se calcula es una distancia de todos los elementos en relación con cada uno de ellos para después ser ordenados. Una observación importante es que el método que se utiliza no se considera como un clasificador binario, pero si se agregan mecanismos de discriminación, podría funcionar como un clasificador.

En las siguientes secciones se muestran, interpretan y discuten varios experimentos relevantes.

Considerando que existen distintos conjuntos de datos y diversos campos de aplicación, es complejo definir una evaluación definitiva de una técnica de recuperación de figuras 3D. Se menciona el caso del esfuerzo del *PSB* para proponer herramientas que sirvan para comparar técnicas, pero aún así existen casos que las categorías que se diseñan, presentan problemas utilizando ciertos criterios. Lo anterior, se deriva en escenarios relevantes y representativos, en los que se observa el principio del funcionamiento y viabilidad de la propuesta.

## 5.2. Experimento 1

### 5.2.1. Tiempo de ejecución de control de invariantes y construcción de secuencias

En la tabla 2 se muestran los tiempos promedio para 2000 modelos, considerando el análisis y control de invariantes más la extracción de medidas de las tres técnicas propuestas, también se toma en cuenta distintas cantidades de muestras (tamaño de secuencia) representativas.

Tabla 2: Tiempo promedio de extracción de medidas por técnica de representación y número de muestras para 2000 modelos

Tec. Rep.	No.Muestras	T. Prom.(ms)						
R1	216	9.18	486	9.36	1014	9.87	1536	10.71
R2	216	10.04	468	11.07	1026	13.73	1518	16.08
R3	210	12.65	506	17.97	1056	28.00	1482	35.25

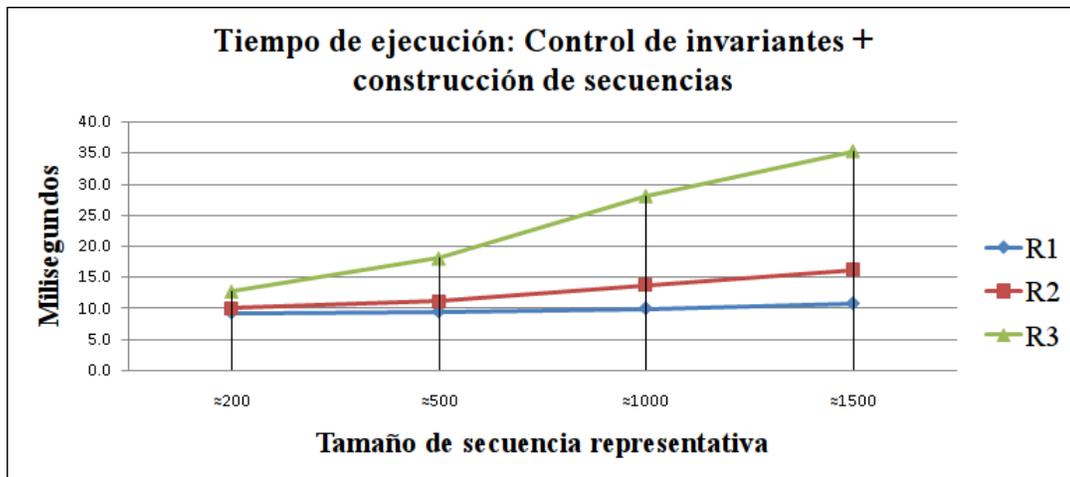


Figura 44: Tiempo promedio de control de invariantes y construcción de secuencias para una figura 3D.

## 5.3. Experimento 2

### 5.3.1. Tiempo de ejecución de comparación de secuencias representativas

Un aspecto a considerar para la implementación en una aplicación real, es la relación directa de los tiempos de ejecución de la implementación con la complejidad temporal teórica, la cual se puede observar en la tabla 3.

Tabla 3: Tiempo promedio de comparación con *FDTW* por técnica de representación y número de muestras para 2000 parejas de secuencias.

Tec. Rep.	No.Muestras	T. Prom.(ms)						
R1	216	0.26	486	1.42	1014	11.75	1536	58.34
R2	216	0.23	468	1.47	1026	12.00	1518	30.71
R3	210	0.30	506	1.89	1056	19.40	1482	27.91

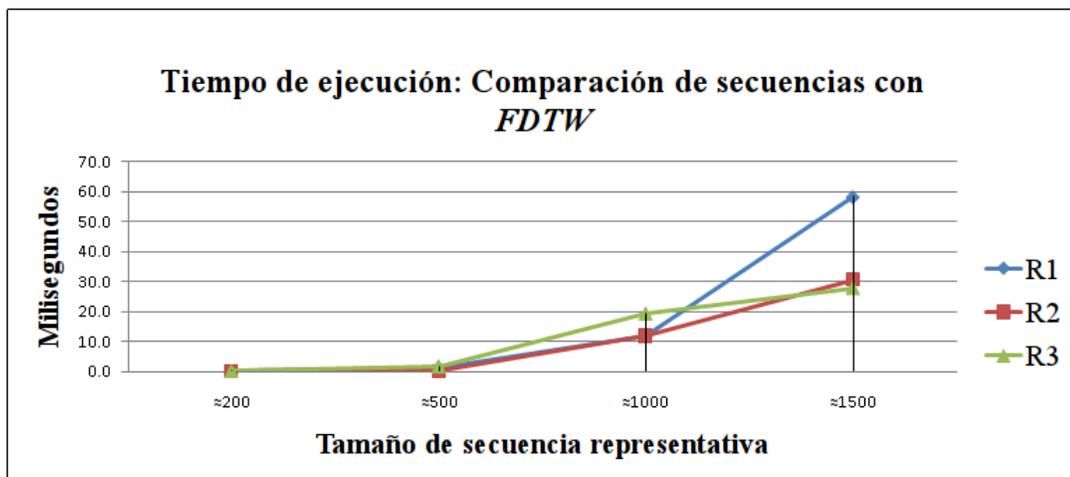


Figura 45: Tiempo promedio de comparación de dos secuencias representativas.

## 5.4.Experimento 3

### 5.4.1.Robustez ante distintas resoluciones

El experimento consiste en considerar una figura 3D, a la cual se le hicieron simplificaciones de cantidad de polígonos en 3 etapas como se muestra en la figura 46. En cada etapa la cantidad de polígonos era de casi la mitad de la etapa anterior. Posteriormente se comparó la figura original con respecto a las otras 3 figuras, el procedimiento se realizó para las 3 técnicas de representación propuestas con secuencias representativas de tamaño  $N \approx 1000$ .

Como resultado se obtuvieron las tablas 4, 5 y 6, donde las distancias obtenidas con las técnicas propuestas, son menores al promedio de la distancia más pequeña por cada técnica. El que los valores de distancia sean menores en relación a la magnitud promedio que se obtiene mediante la medida de distancia, es una condición necesaria para que una técnica de representación sea robusta ante distintas resoluciones.

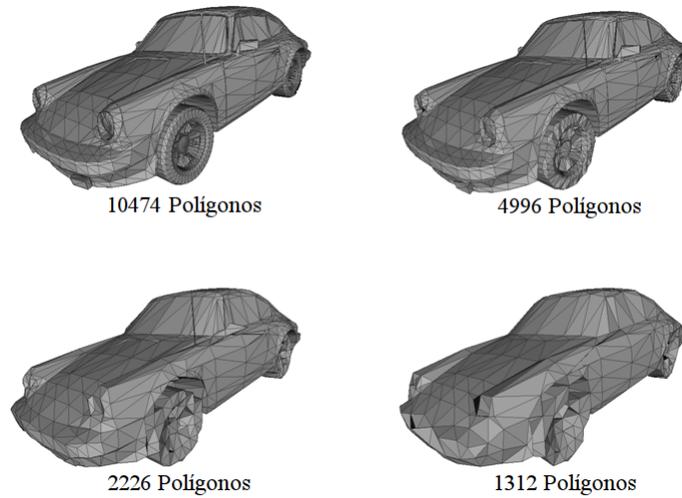


Figura 46: Figura 3D con simplificaciones de polígonos.

Tabla 4: Valores de distancias obtenidos para distintas resoluciones usando la técnica R1 (registro de medidas sobre un cubo).

Modelo A	Modelo B	Distancia	Resultado	Dist. Prom.
Porsche (10474 poligonos)	Porsche (4996 poligonos)	$FDTW(R1(A),R1(B))$	0.6622	3.3870
Porsche (10474 poligonos)	Porsche (2226 poligonos)	$FDTW(R1(A),R1(B))$	1.2982	
Porsche (10474 poligonos)	Porsche (1312 poligonos)	$FDTW(R1(A),R1(B))$	0.9530	

Tabla 5: Valores de distancias obtenidos para distintas resoluciones usando la técnica R2 (registro de medidas sobre cilindros).

Modelo A	Modelo B	Distancia	Resultado	Dist. Prom.
Porsche (10474 poligonos)	Porsche (4996 poligonos)	$FDTW(R2(A),R2(B))$	3.2346	14.1815
Porsche (10474 poligonos)	Porsche (2226 poligonos)	$FDTW(R2(A),R2(B))$	9.1570	
Porsche (10474 poligonos)	Porsche (1312 poligonos)	$FDTW(R2(A),R2(B))$	7.0028	

Tabla 6: Valores de distancias obtenidos para distintas resoluciones usando la técnica R3 (registro de medidas sobre una esfera).

Modelo A	Modelo B	Distancia	Resultado	Dist. Prom.
Porsche (10474 poligonos)	Porsche (4996 poligonos)	$FDTW(R3(A),R3(B))$	5.7359	20.8918
Porsche (10474 poligonos)	Porsche (2226 poligonos)	$FDTW(R3(A),R3(B))$	11.4336	
Porsche (10474 poligonos)	Porsche (1312 poligonos)	$FDTW(R3(A),R3(B))$	9.9877	

## 5.5.Experimento 4

### 5.5.1.Descripción del experimento de recuperación de las tres técnicas de representación

El experimento consiste en mostrar la respuesta de la metodología de análisis de similitud, utilizando cada una de las técnicas de registro de medidas ( $R1$ ,  $R2$  y  $R3$ ) en dos formas. La primera consiste en mostrar algunos ejemplos de recuperaciones y la segunda es mediante una comparación de curvas de *precisión-recuperación* promedio para cada técnica.

La configuración de opciones de los experimentos se observan en la tabla 7, cada configuración corresponde con los mejores resultados observados de la técnica.

Tabla 7: Opciones seleccionadas de experimento para cada técnica de representación.

Técnica	Tamaño de secuencia	Medida de distancia	Indicador de similitud
$R1$	$N \approx 1500$	$FDTW$	$d_{FDTW(R1)}$
$R2$	$N \approx 1000$	$FDTW$	$d_{FDTW(R2)}$
$R3$	$N \approx 1500$	$FDTW$	$d_{FDTW(R3)}$

El conjunto de datos para el experimento, es el que corresponde con el de pruebas del *PSB*. El conjunto consiste en 907 figuras que pertenecen a 92 clases. Es un conjunto de datos heterogéneo y con cantidad no balanceada de elementos por clase.

En cada apartado de resultados de recuperación, se indica en color verde la figura a recuperar, en color azul lo que *PSB* especifica como respuesta correcta y en color rojo la respuesta incorrecta.

### 5.5.2. Resultados de ejemplo de recuperación de la técnica de representación $R1$

En las figuras 47, 48 y 49, se presentan tres ejemplos de resultados para la técnica de representación  $R1$ .

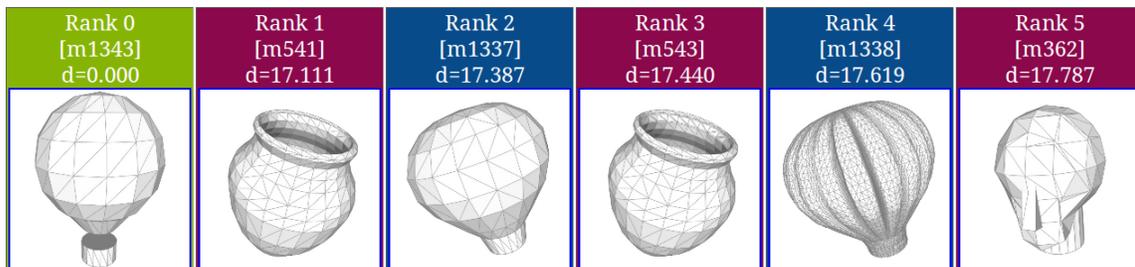


Figura 47: Recuperación de la clase  $aircraft \supset balloon\_vehicle \supset hot\_air\_balloon$ .

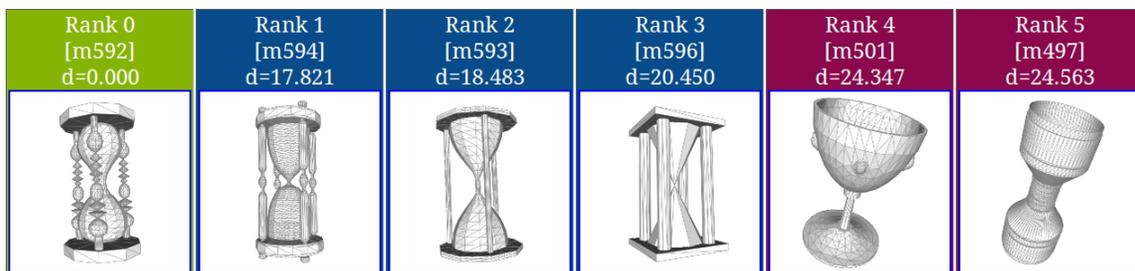


Figura 48: Recuperación de la clase  $hourglass$ .

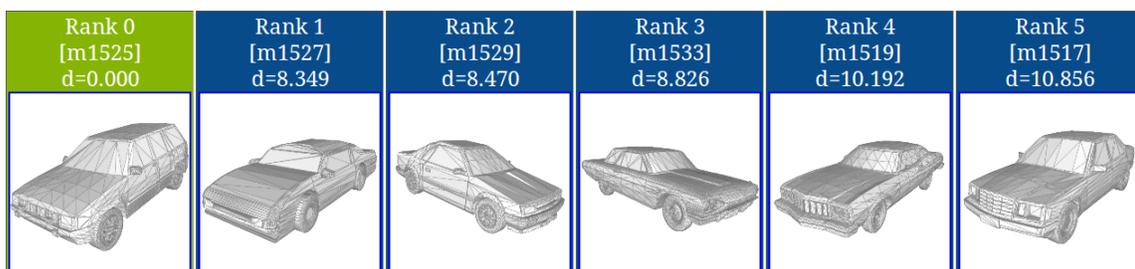


Figura 49: Recuperación de la clase  $vehicle \supset car \supset sedan$ .

### 5.5.3. Resultados de ejemplo de recuperación de la técnica de representación $R2$

A continuación, en las figuras 50, 51 y 52 se muestran tres ejemplos resultados para la técnica de representación  $R2$ .

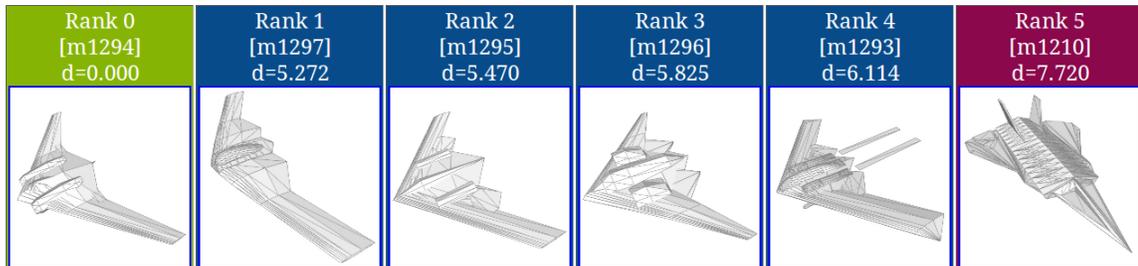


Figura 50: Recuperación de la clase *aircraft*  $\supset$  *airplane*  $\supset$  *stealth\_bomber*.

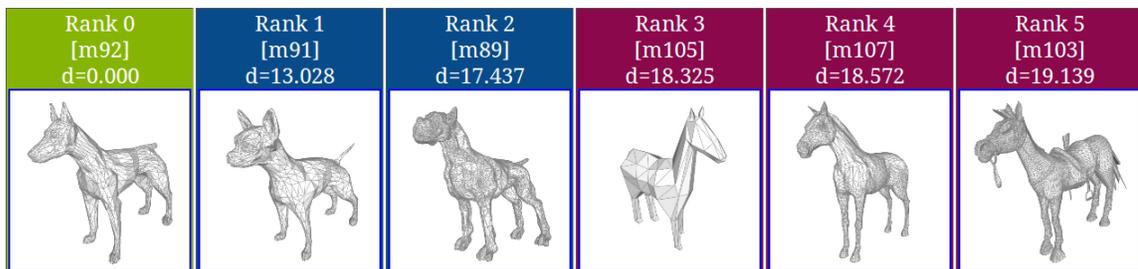


Figura 51: Recuperación de la clase *animal*  $\supset$  *quadruped*  $\supset$  *dog*.

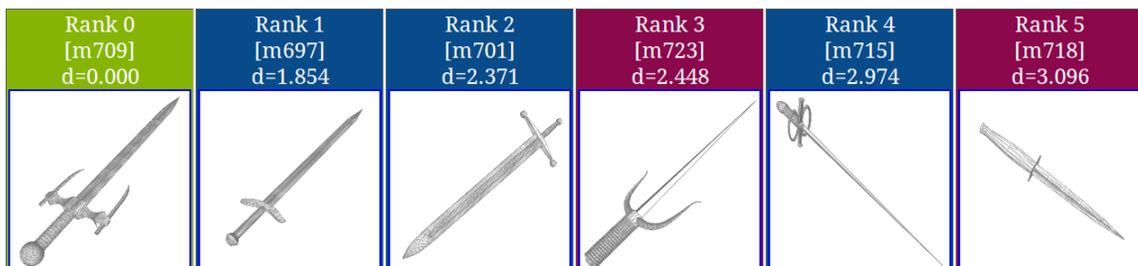
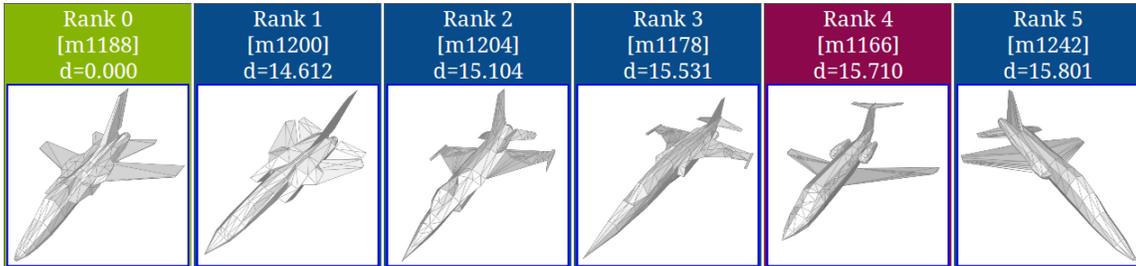
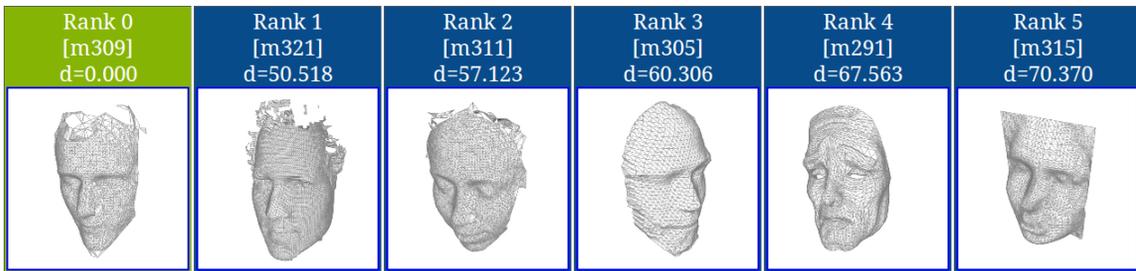
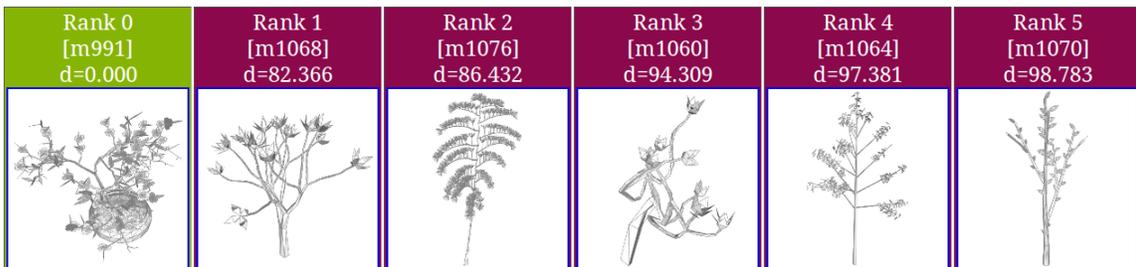


Figura 52: Recuperación de la clase *blade*  $\supset$  *sword*.

#### 5.5.4. Resultados de ejemplo de recuperación de la técnica de representación *R3*

A continuación, en las figuras 53, 54 y 55, se muestran tres ejemplos de resultados para la técnica de representación *R3*.

Figura 53: Recuperación de la clase *aircraft*  $\supset$  *airplane*  $\supset$  *fighter\_jet*.Figura 54: Recuperación de la clase *body\_part*  $\supset$  *face*.Figura 55: Recuperación de la clase *plant*  $\supset$  *potted\_plant*.

Una observación importante para las distintas figuras recuperadas consiste, por ejemplo, en la figura 55 la respuesta es de plantas visiblemente similares. La clasificación indica que todas las respuestas son incorrectas a pesar de que son de la clase *plant*. Es un claro ejemplo de las inconsistencias y ambigüedades en los métodos de evaluación.

### 5.5.5. Comparación de las técnicas *R1*, *R2* y *R3*

Para observar el comportamiento de las técnicas en una comparación, se calculan las curvas de *precisión-recuperación*.

A partir del procedimiento en la sección 3.4 para el cálculo de curvas de *precisión-recuperación*, se comparan las técnicas de representación propuestas.

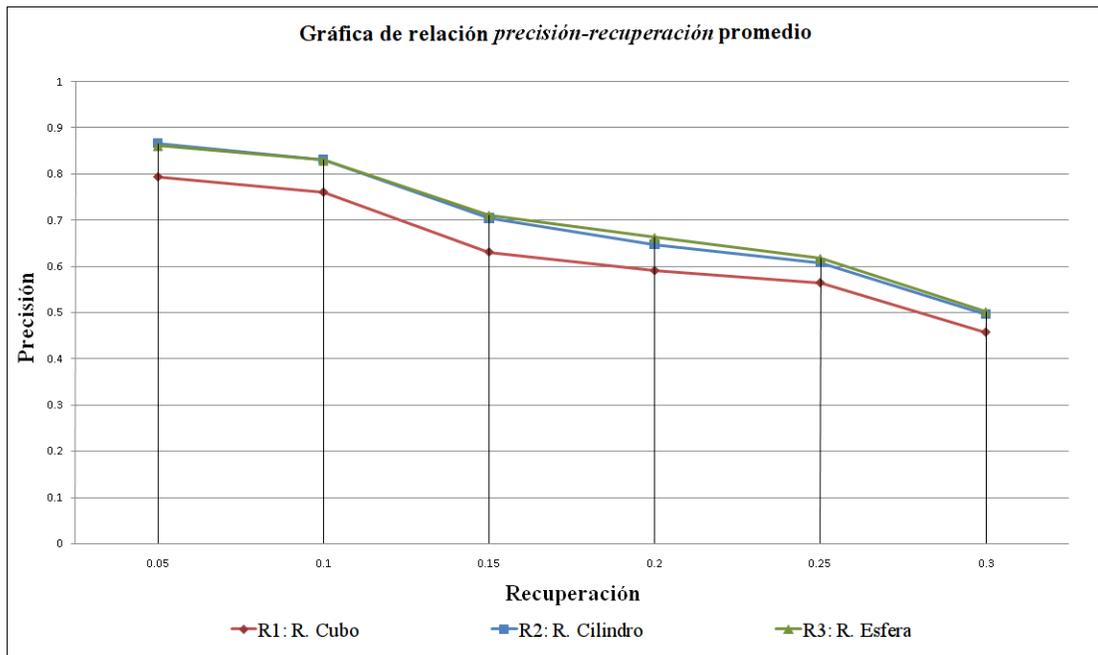


Figura 56: Gráfica de comparación entre las técnicas propuestas.

Considérese lo siguiente, para una figura a recuperar  $q$ , la cual es un miembro de la clase  $C$  de tamaño  $|C|$  se tiene: *precisión* (eje vertical) es la razón de elementos relevantes  $k_q$  (los elementos que pertenecen a la misma clase que la figura a recuperar) con respecto al número de figuras recuperadas  $k_{rec}$ . La *recuperación* (eje horizontal) es la razón de elementos relevantes  $k_q$  con respecto al tamaño de la clase de la figura a recuperar  $|C|$ . Idealmente esta curva debe ser una línea horizontal con precisión de valor uno.

Lo que se observa en la figura 56 es que las 3 representaciones propuestas muestran valores mayores a 0.5 (50 %) de precisión para una recuperación del 0.25 (25 %).

En la tabla 8 se observa el comportamiento de cada técnica en términos de precisión para el porcentaje de recuperación indicado. Donde se están considerando los mejores resultados con la configuración para cada técnica de la tabla 7. Se observa que  $R2$  tiene mejor comportamiento de precisión hasta un 0.10 de la recuperación, para posteriores proporciones de recuperación, la técnica de representación  $R3$  obtuvo mejores resultados.

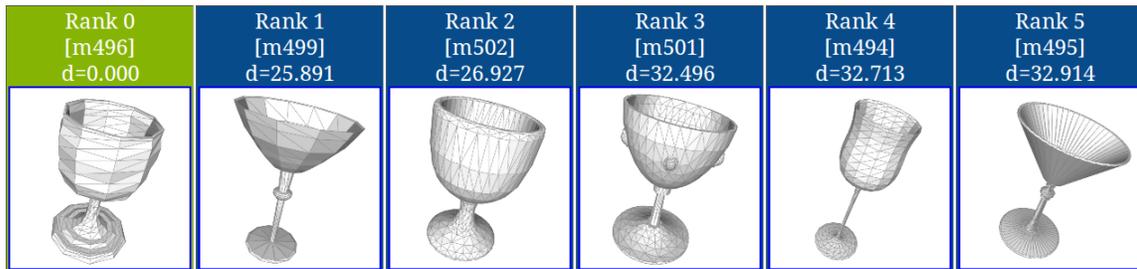
Tabla 8: Precisión obtenida para distintas proporciones de recuperación

Técnica	Recuperación					
	0.05	0.10	0.15	0.20	0.25	0.30
$d_{FDTW(R2)}$	0.866	0.831	0.704	0.647	0.607	0.496
$d_{FDTW(R3)}$	0.860	0.829	0.710	0.662	0.616	0.501
$d_{FDTW(R1)}$	0.793	0.759	0.630	0.590	0.563	0.456

## 5.6. Experimento 5

### 5.6.1. Resultados de ejemplo de recuperación de la técnica de combinación de valores de distancias

A continuación, en las figuras 57, 58 y 59 se muestran tres ejemplos de resultados para la combinación de distancias  $d_T = d_{FDTW(R1)} + d_{FDTW(R2)} + d_{FDTW(R3)}$ .

Figura 57: Recuperación de la clase  $furniture \supset seat \supset chair \supset dining\_chair$ .Figura 58: Recuperación de la clase  $liquid\_container \supset glass\_with\_stem$ .

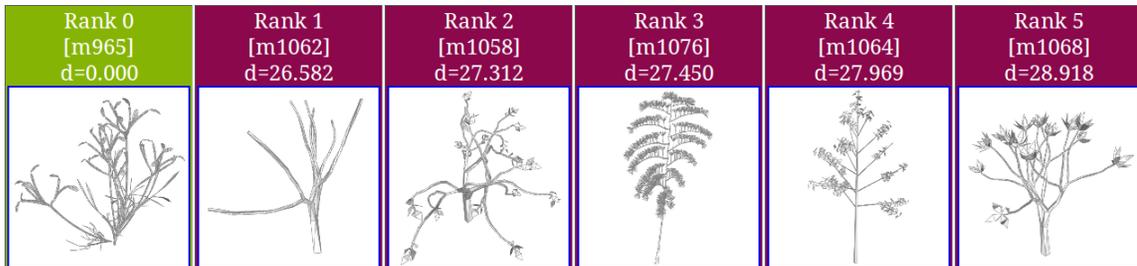


Figura 59: Recuperación de la clase  $plant \supset bush$ .

### 5.6.2. Comparación de la combinación de valores de distancia

Para observar el comportamiento de las técnicas en una comparación, se calculan las curvas de *precisión-recuperación*.

A partir del procedimiento de la sección 3.4 para el cálculo de curvas de *precisión-recuperación*, se comparan las técnicas de representación propuestas y la combinación de resultados en  $d_T = d_{FDTW(R1)} + d_{FDTW(R2)} + d_{FDTW(R3)}$ .

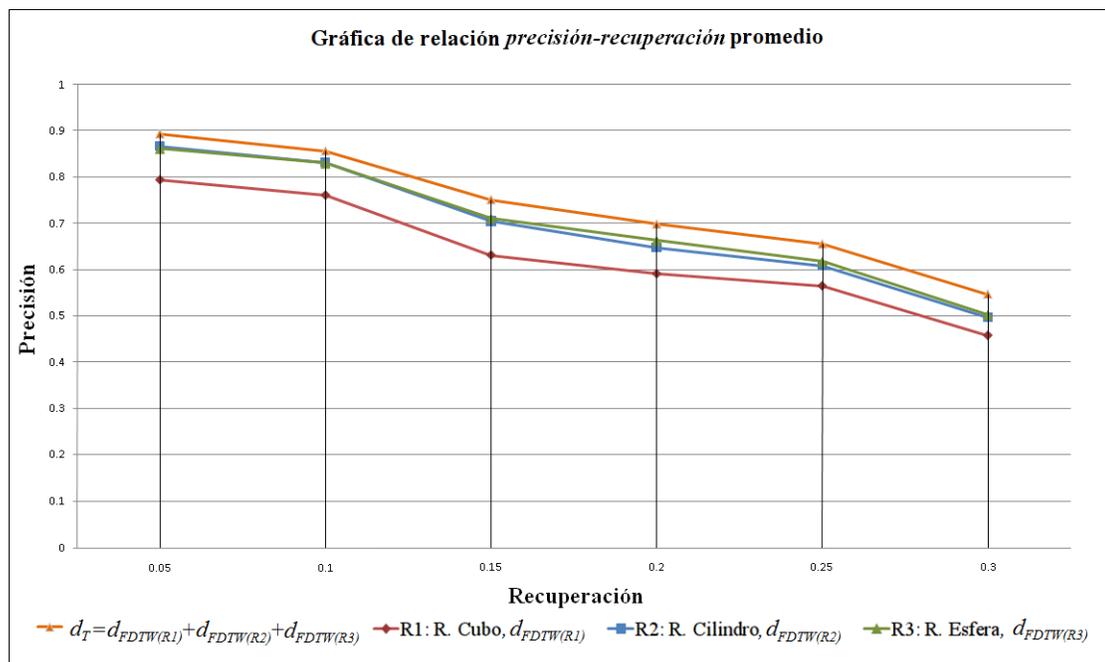


Figura 60: Gráfica de comparación de la combinación de las técnicas de representación propuestas con respecto a las técnicas de manera individual.

En la tabla 9 se observa el comportamiento de cada técnica en términos de precisión para el porcentaje de recuperación indicado.

Tabla 9: Precisión obtenida para distintos proporciones de recuperación

Técnica	Recuperación					
	0.05	0.10	0.15	0.20	0.25	0.30
$d_T$	0.892	0.855	0.750	0.697	0.654	0.545
$d_{FDTW(R2)}$	0.866	0.831	0.704	0.647	0.607	0.496
$d_{FDTW(R3)}$	0.860	0.829	0.710	0.662	0.616	0.501
$d_{FDTW(R1)}$	0.793	0.759	0.630	0.590	0.563	0.456

Se observa que los resultados de precisión mejoran con  $d_T$  en comparación con cada técnica de representación. La suma de las distancias funciona mejor que la medida de distancia de una sola técnica de representación. Lo anterior se le atribuye a que los valores de distancia medidos por cada técnica, mantienen un valor de escala diferente y coincide con la aportación que realiza en la combinación.  $R2$  aporta en mayor medida con los mejores resultados globales, seguida de la técnica  $R3$  y finalmente  $R1$ .

### 5.7. Comparación de resultados con técnicas del estado del arte

Para observar el comportamiento de las técnicas en una comparación, se calculan las curvas de *precisión-recuperación*.

A partir del procedimiento de la sección 3.4 para el cálculo de curvas de *precisión-recuperación*, se comparan las técnicas de representación propuestas, la combinación de resultados del indicador de disimilitud  $d_T$  y ocho técnicas del estado del arte. Los datos de resultados se obtuvieron de Bustos et al. [7]. En el trabajo mencionado comparan 16 técnicas de análisis de similitud entre figuras 3D de las cuales sólo se tomaron las ocho con mejores resultados.

En la tabla 10 se observa el comportamiento de cada técnica en términos de precisión para el porcentaje de recuperación indicado.

Como se observa el principio de la propuesta, de manera global es mejor en términos de precisión para una recuperación de hasta 0.30 en comparación a varias técnicas.

Como se explica en el estado del arte todas parten del principio de clasificación, creando descriptores que muchas veces durante el proceso de selección, extracción y construcción de características, tienen pérdidas de información valiosa.

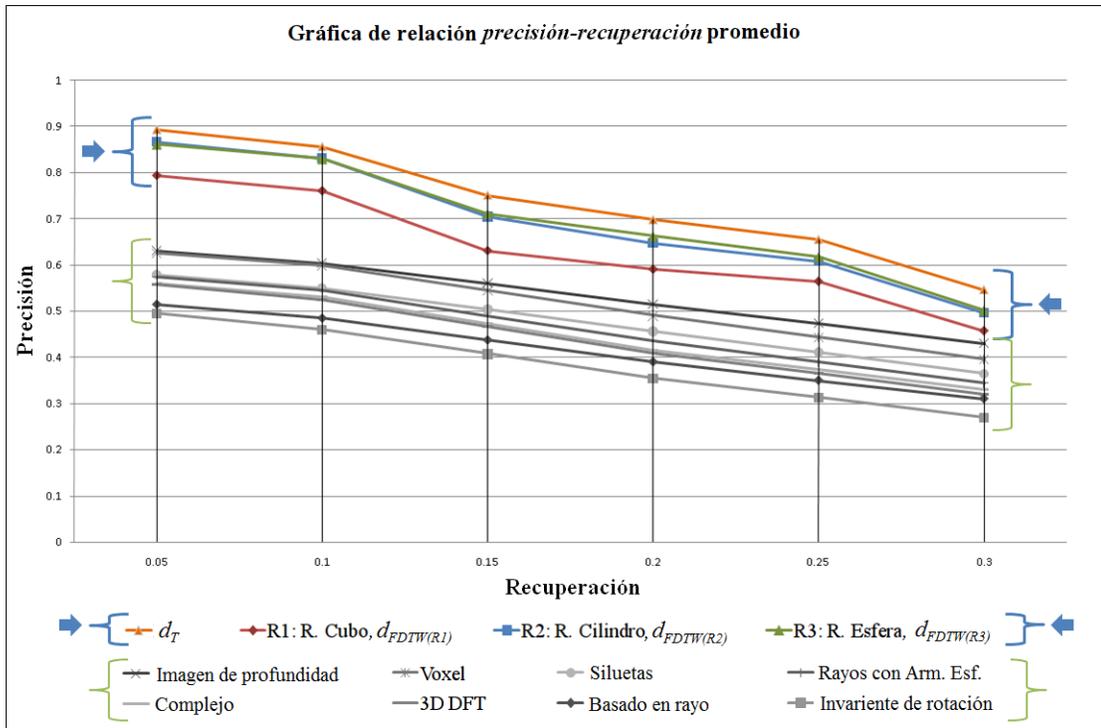


Figura 61: Gráfica de comparación de las técnicas de representación propuestas, con respecto a las técnicas estudiadas en Bustos et al. [7].

## 5.8. Experimento 7

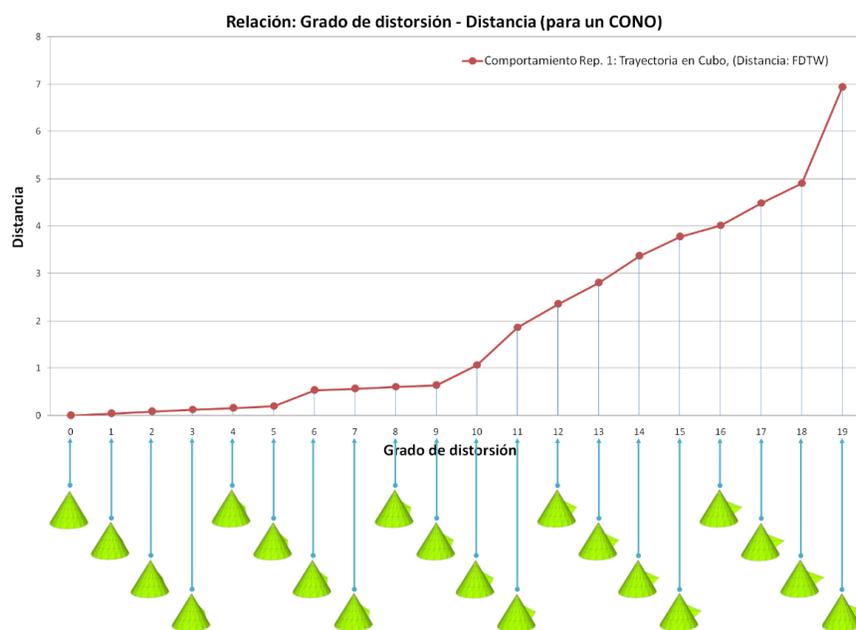
### 5.8.1. Relación *distorsión-distancia*

El objetivo del experimento, es mostrar el comportamiento de las técnicas propuestas, con un conjunto de datos de figuras geométricas primitivas, especificado en el apartado 4.9.2, el conjunto de datos tiene la característica de ser homogéneo. Las figuras se encuentran modificadas gradualmente.

A continuación en las figuras 62, 63 y 64 se presenta la relación propuesta de *grado de distorsión* contra *distancia*.

Tabla 10: Precisión promedio obtenida para distintas proporciones de recuperación

Técnica	Recuperación					
	0.05	0.10	0.15	0.20	0.25	0.30
$d_T$	0.892	0.855	0.750	0.697	0.654	0.545
$d_{FDTW(R2)}$	0.866	0.831	0.704	0.647	0.607	0.496
$d_{FDTW(R3)}$	0.860	0.829	0.710	0.662	0.616	0.501
$d_{FDTW(R1)}$	0.793	0.759	0.630	0.590	0.563	0.456
Imagen de profundidad	0.630	0.604	0.559	0.515	0.472	0.430
Voxel	0.625	0.598	0.544	0.490	0.443	0.396
Siluetas	0.578	0.550	0.503	0.456	0.410	0.365
Rayos con Arm. Esf.	0.575	0.545	0.490	0.435	0.389	0.344
Complejo	0.560	0.530	0.472	0.415	0.372	0.330
3D DFT	0.558	0.525	0.467	0.410	0.365	0.320
Basado en rayo	0.515	0.485	0.437	0.390	0.350	0.310
Invariante de rotación	0.495	0.460	0.407	0.355	0.312	0.270

Figura 62: Relación *grado de distorsión* - *distancia*.

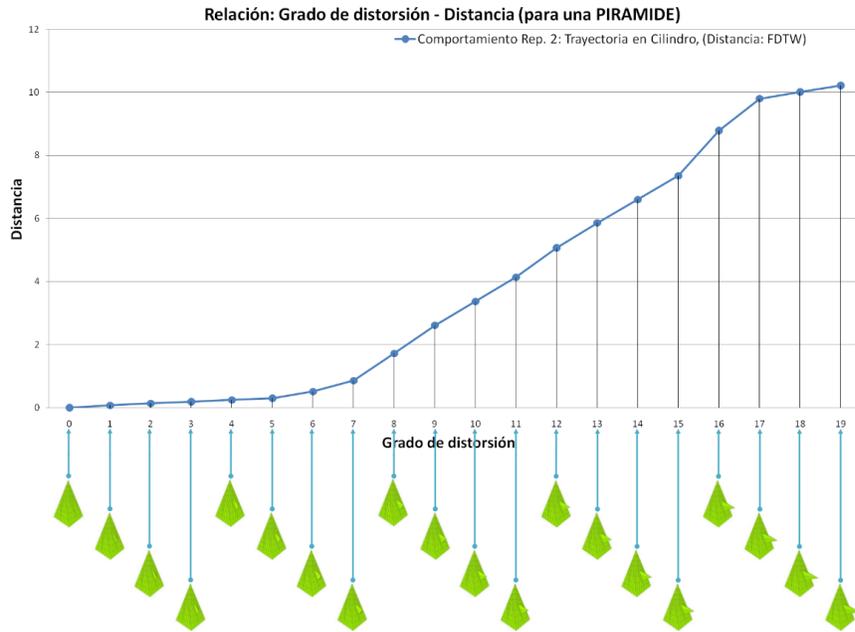


Figura 63: Relación *grado de distorsión - distancia*.

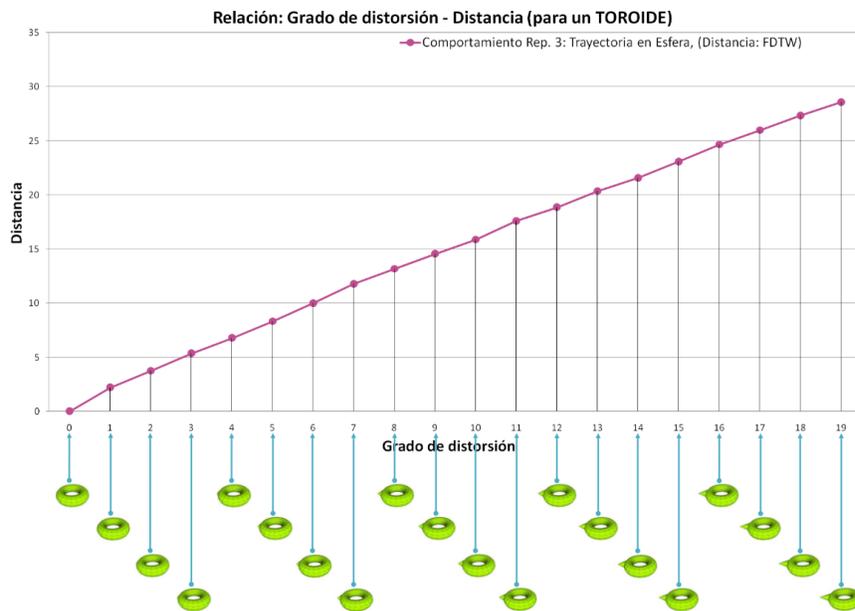


Figura 64: Relación *grado de distorsión - distancia*.

### 5.9.Experimento 8

#### 5.9.1.Relación *diferencia de distancias* entre *distorsiones* consecutivas

El objetivo consiste en mostrar el comportamiento de las técnicas propuestas, con un conjunto de datos de figuras geométricas. Las figuras se encuentran modificadas gradualmente.

La función de *distancia* corresponde con la medida de distancia especificada en la gráfica.

A continuación en las figuras 65, 66 y 67 se presenta la relación propuesta de *grado de distorsión* contra *diferencia entre distancias* de parejas.

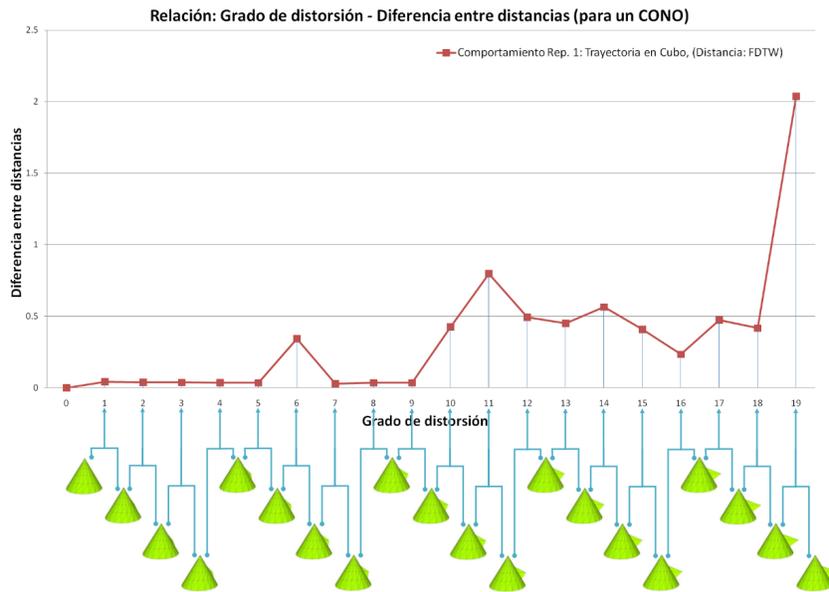


Figura 65: Relación entre *distorsiones consecutivas* contra *diferencia entre distancias*.

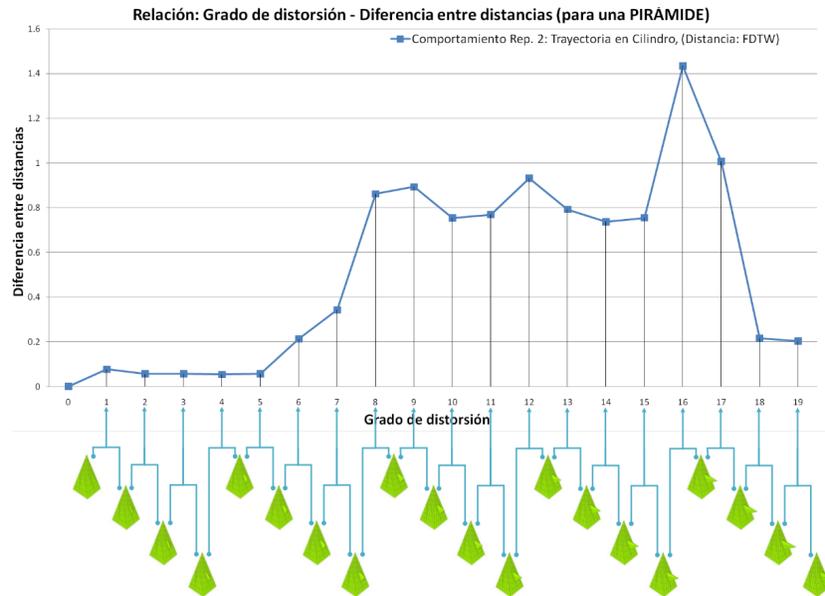


Figura 66: Relación entre *distorsiones consecutivas* contra *diferencia entre distancias*.

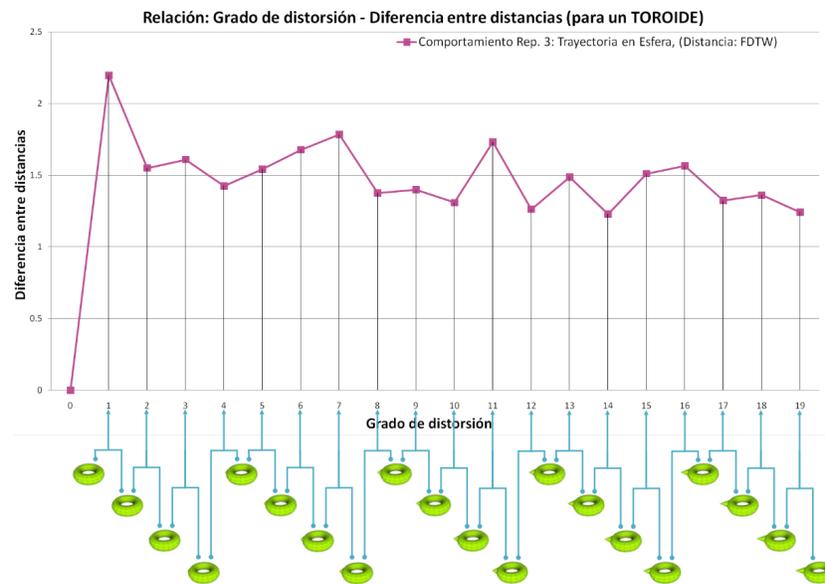


Figura 67: Relación entre *distorsiones consecutivas* contra *diferencia entre distancias*.

## 5.10. Experimento 9

### 5.10.1. Otras medidas de desempeño

El objetivo del experimento consiste en calcular la medida del *vecino más cercano* y *medida-E* para las técnicas propuestas, alternando opciones representativas de experimentación.

Las siguientes medidas son útiles para caracterizar a las técnicas de comparación y recuperación. El trabajo con el que se comparan resultados, no reportan estas medidas y frecuentemente no se reportan en la literatura.

### 5.10.2. Análisis del *vecino más cercano*

La medida del vecino más cercano indica el porcentaje de las  $k$  mejores comparaciones que pertenecen a las misma clase de la figura que se consultó, para este análisis  $k = 1$ . Con dicha medida se proporciona un indicador de qué tan bien una técnica está recuperando en términos de similitud, con respecto al primer elemento recuperado.

La tabla 11 enlista los resultados ordenados desde el mejor resultado y en orden descendiente los demás.

Se observa, en las primeras 9 posiciones a *FDTW* como la mejor medida de distancia, por encima de la distancia clásica *Minkowski*.

Para la medida de distancia *FDTW*, con la técnica de representación *R2* considerando  $\approx 1000$  datos, se obtiene un 34.8% de modelos correctamente recuperados, los cuales se consideran como la primera respuesta.

### 5.10.3. Análisis de *medida-E*

La *medida-E* se compone de los valores de *precisión* y *recuperación*, para los primeros 32 modelos. Es entonces que la *medida-E*, considera únicamente cierta cantidad de modelos recuperados para cada figura a recuperar y calcula la *precisión* y la *recuperación* sobre esos resultados.

La tabla 12, enlista los resultados ordenados desde el mejor resultado y en orden descendiente los demás.

Se observa en las primeras 6 posiciones a *FDTW*, como la mejor medida de distancia, por encima de la distancia clásica *Minkowski*.

Para la medida de distancia *FDTW* con la técnica de representación *R2* considerando  $\approx 1000$  datos, se obtiene un 11.3%, que corresponde con la capacidad de mantener una relación de precisión y recuperación para los primeros 32 modelos recuperados.

Tabla 11: Medida del *vecino más cercano* para las técnicas de representación propuestas

Representación	Cantidad de datos	Distancia	Valor de la medida <i>vecino más cercano</i>
<i>R2</i>	1000	<i>FDTW</i>	0.348
<i>R2</i>	1500	<i>FDTW</i>	0.344
<i>R3</i>	1500	<i>FDTW</i>	0.329
<i>R3</i>	1000	<i>FDTW</i>	0.317
<i>R3</i>	500	<i>FDTW</i>	0.303
<i>R2</i>	500	<i>FDTW</i>	0.301
<i>R3</i>	200	<i>FDTW</i>	0.285
<i>R2</i>	200	<i>FDTW</i>	0.281
<i>R1</i>	1500	<i>FDTW</i>	0.260
<i>R1</i>	1500	<i>Minkowski</i>	0.245
<i>R1</i>	500	<i>FDTW</i>	0.232
<i>R3</i>	1000	<i>Minkowski</i>	0.228
<i>R1</i>	1000	<i>FDTW</i>	0.223
<i>R3</i>	500	<i>Minkowski</i>	0.218
<i>R3</i>	1500	<i>Minkowski</i>	0.215
<i>R1</i>	1000	<i>Minkowski</i>	0.212
<i>R1</i>	500	<i>Minkowski</i>	0.204
<i>R3</i>	200	<i>Minkowski</i>	0.196
<i>R2</i>	1000	<i>Minkowski</i>	0.195
<i>R2</i>	500	<i>Minkowski</i>	0.186
<i>R2</i>	200	<i>Minkowski</i>	0.178
<i>R2</i>	1500	<i>Minkowski</i>	0.176
<i>R1</i>	200	<i>FDTW</i>	0.146
<i>R1</i>	200	<i>Minkowski</i>	0.139

Nota: El orden de la distancia *Minkowski* es  $p = 3$

Tabla 12: Medida de *medida-E* para las técnicas de representación propuestas

Representación	Cantidad de datos	Distancia	<i>Medida-E</i>
<i>R2</i>	1000	<i>FDTW</i>	0.113
<i>R3</i>	1000	<i>FDTW</i>	0.112
<i>R2</i>	1500	<i>FDTW</i>	0.111
<i>R3</i>	1500	<i>FDTW</i>	0.111
<i>R3</i>	500	<i>FDTW</i>	0.110
<i>R3</i>	200	<i>FDTW</i>	0.107
<i>R1</i>	1500	<i>Minkowski</i>	0.104
<i>R2</i>	500	<i>FDTW</i>	0.103
<i>R1</i>	1000	<i>Minkowski</i>	0.097
<i>R2</i>	200	<i>FDTW</i>	0.096
<i>R1</i>	1500	<i>FDTW</i>	0.093
<i>R2</i>	1000	<i>Minkowski</i>	0.086
<i>R2</i>	1500	<i>Minkowski</i>	0.084
<i>R1</i>	1000	<i>FDTW</i>	0.083
<i>R1</i>	500	<i>Minkowski</i>	0.082
<i>R2</i>	500	<i>Minkowski</i>	0.079
<i>R1</i>	500	<i>FDTW</i>	0.076
<i>R3</i>	1500	<i>Minkowski</i>	0.071
<i>R3</i>	500	<i>Minkowski</i>	0.071
<i>R3</i>	1000	<i>Minkowski</i>	0.070
<i>R3</i>	200	<i>Minkowski</i>	0.068
<i>R2</i>	200	<i>Minkowski</i>	0.068
<i>R1</i>	200	<i>Minkowski</i>	0.059
<i>R1</i>	200	<i>FDTW</i>	0.053

Nota: El orden de la distancia *Minkowski* es  $p = 3$

## 5.11. Experimento 10

### 5.11.1. Visualización de alineación de secuencias y reconstrucción parcial de 1D a 3D

Se muestran algunos resultados representativos, utilizando el procedimiento de visualización de alineación de secuencias. El procedimiento muestra la alineación de representaciones en 1D y la reconstrucción de 1D a 3D.

En primer lugar se muestran las figuras comparadas originales, luego se muestran las representaciones en 1D indicando en color la correspondencia entre las dos secuencias, finalmente también se muestra la correspondencia en color en las reconstrucciones parciales de las figuras en 3D.

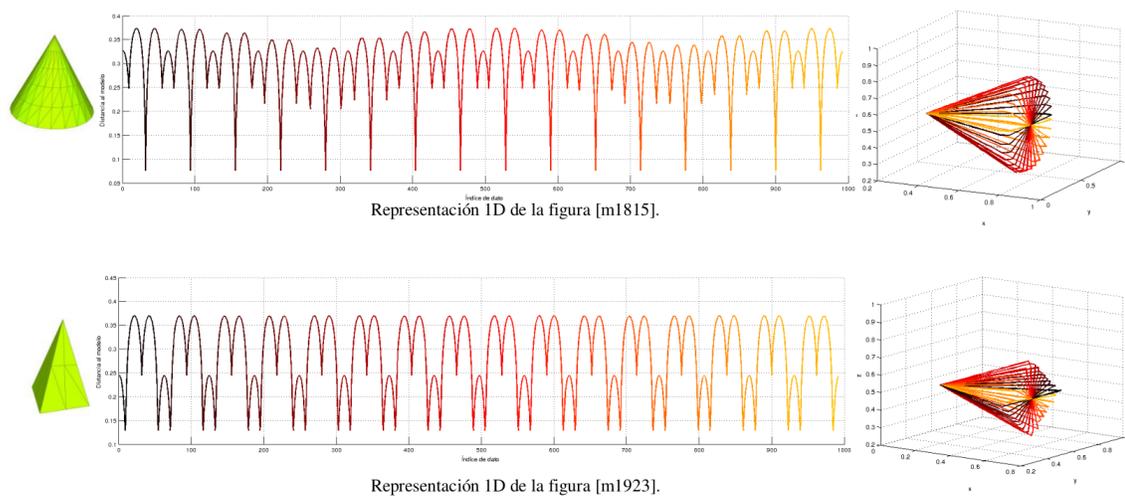


Figura 68: Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D.

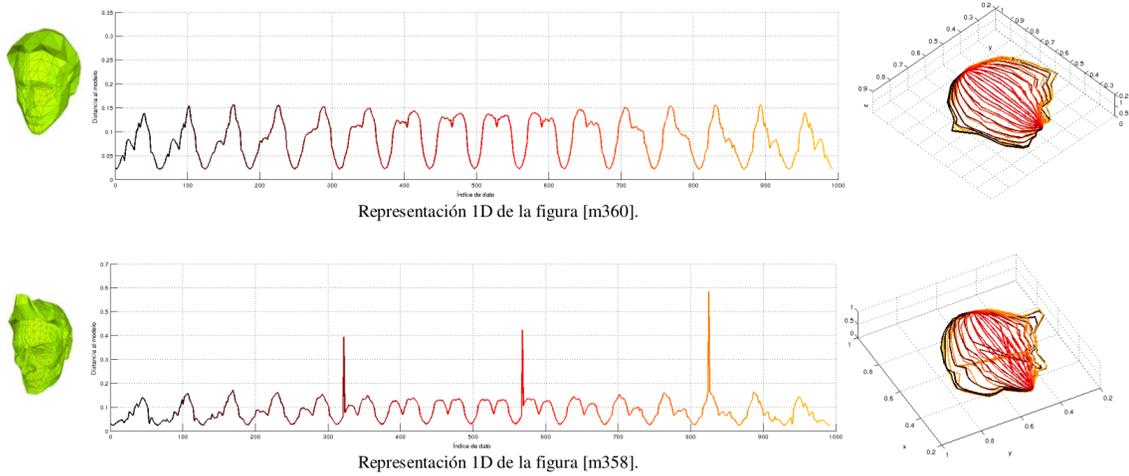


Figura 69: Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D.

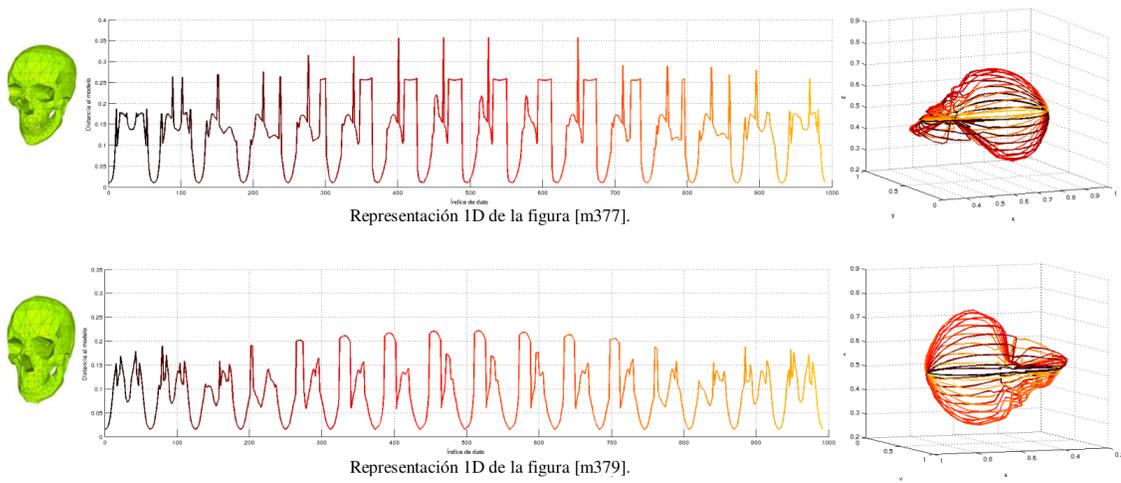


Figura 70: Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D.

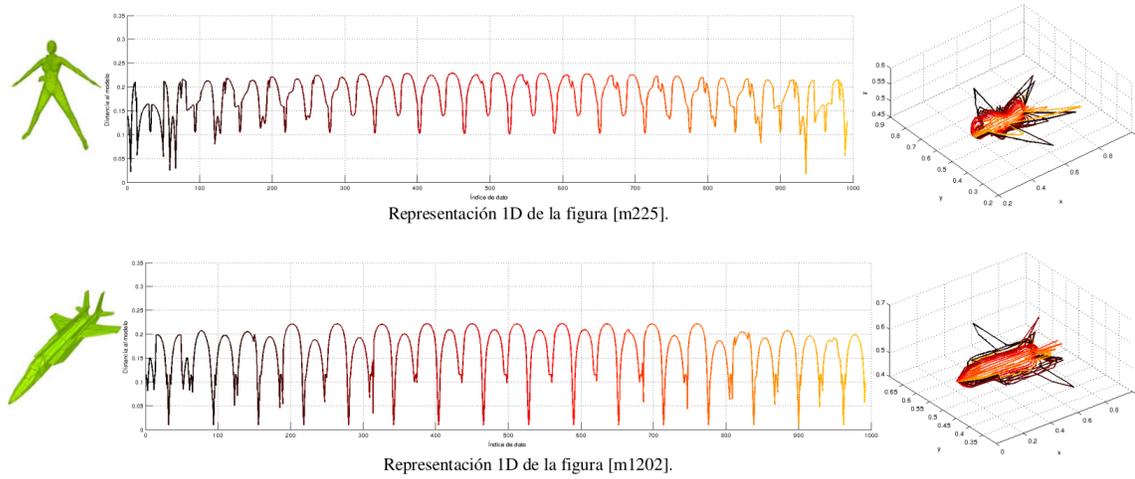


Figura 71: Visualización de alineación en la comparación y reconstrucción parcial de figuras 3D.



## CONCLUSIONES

---

En este capítulo se presentan las conclusiones, de diferentes aspectos observados en el desarrollo de esta tesis, además de describir las contribuciones y líneas de trabajo futuro.

### 6.1. Conclusiones

- La implementación de la metodología propuesta obtiene mejores resultados que las ocho mejores técnicas de un estudio presentado en Bustos et al. [7], el cual considera a 16 técnicas.

La mejora de los resultados en el análisis de similitud entre figuras 3D, radica en la observación de la naturaleza del problema y el enfoque de comparación que se adopta. En primer lugar, se parte de la idea de registrar las variaciones de la superficie de los objetos, desde marcos de referencia fijos y luego la comparación de dichos registros mediante comparación elástica, acoplando las dos ideas, se proporciona un mejor resultado que medidas de distancia clásicas de comparación.

- Se proponen tres distintas técnicas de representación de figuras, las técnicas realizan una transformación de dominio de 3D a 1D y tienen la propiedad de transformación inversa, que consiste en reconstruir parcialmente la figura 3D.

Partiendo de la intuición de que se puede determinar distintas formas de observación de un objeto, la observación es mejor si se registra la mayor cantidad de superficie de manera continua, captando las variaciones de superficie, lo cual se aproxima a registrar una silueta seccionada. Las representaciones captan la información necesaria para operar con figuras de topologías diferentes, planos en 3D, sólidos cerrados, sólidos no cerrados y polígonos desconectados que son figuras mal construidas. La reconstrucción parcial es una propiedad que no la tiene ningún otro descriptor de características y se le atribuye a las técnicas que registran medidas con vectores de dirección.

- El desarrollo de la tesis no parte del enfoque de clasificación, que en muchos casos consiste en la selección, extracción y construcción de características para determinar un descriptor. Mas bien, se proponen representaciones que registran las propiedades geométricas de la superficie en forma global de una figura 3D que resultan más fáciles de analizar.
- Las secuencias representativas obtenidas usando las técnicas  $R1, R2$  y  $R3$  son robustas ante distintas resoluciones de las figuras 3D.

Es una necesidad para las representaciones de figuras, tengan propiedades de invariantes y robustez, las técnicas propuestas consiguen dichas propiedades mediante análisis de componentes principales agregando un análisis de área para eliminar la ambigüedad de orientación en cada componente  $(x, y, z)$ . La robustez ante distintos niveles de resolución se encuentra implícita en las técnicas. La resolución está dada por la cantidad de polígonos y las técnicas propuestas no se basan en la información directa de los polígonos que componen a los modelos.

- Los resultados de análisis, de curvas de *precisión-recuperación* promedio y *medida-E*, determina en todos los casos, mejores resultados globales comparando con la distancia *Fast Dynamic Time Warping (FDTW)* que con la métrica *Minkowski*. La técnica de representación  $R2$  considerando  $\approx 1000$  valores y comparando con *FDTW*, obtuvo mejores resultados en las distintas medidas de desempeño. El análisis de *vecinos más cercanos* permite ubicar a las técnicas propuestas acorde a un criterio de similitud entre figuras 3D. El análisis de la *medida-E*, determina el desempeño global de las técnicas considerando la relación entre precisión y recuperación. Las medidas mencionadas se recomiendan, como consideraciones importantes para la implementación de un sistema de recuperación de modelos 3D.
- No existe técnica definitiva para el análisis de similitud entre figuras 3D, una observación en este trabajo, es que cada técnica de representación obtuvo mejores resultados en consultas distintas, lo cual determinó una prueba de conjugación de valores de distancia. Los resultados de la combinación de distancias mostraron una mejora clara con respecto a las técnicas de representación de manera individual.
- Aunque *Princeton Shape Benchmark* cuenta con un conjunto de datos y herramientas de evaluación de desempeño, utiliza como referencia una clasificación hecha a partir de un criterio humano y al comparar los resultados obtenidos hay inconsistencias. Esto, debido a la construcción degenerada de la figura o bien falta de correspondencia semántica.

- El análisis de figuras geométricas, con deformaciones graduales muestra un criterio cuantitativo del comportamiento de una técnica de análisis de similitud.

## 6.2. Contribuciones

El amplio estudio de diseño de descriptores para clasificación, tiene como resultado que el desarrollo de las técnicas de descripción se dirija en esa dirección. En cambio, este trabajo plantea la conservación de los componentes globales en las representaciones, con el propósito de que sean útiles para su comparación, determinando la adecuada combinación entre representación y criterio de comparación.

A lo largo del trabajo, se incluyen observaciones de la importancia de representar la naturaleza del problema. También se incluye una medida de distancia clásica, que ayuda a referenciar a la propuesta de la medida de esfuerzo de alineación entre dos figuras 3D.

Se puntualizan las aportaciones de este trabajo.

- Se propone una metodología de comparación y recuperación, que puede trabajar con tres técnicas distintas de representación y dos medidas distintas de distancia.
- Se evalúan los resultados con las medidas de desempeño del *vecino más cercano* y *medida-E*, que comúnmente no se reportan en el estado del arte.
- Capacidad de las técnicas de capturar información de figuras geométricas, figuras que no encierran un espacio, planos en 3D y figuras irregulares.
- Se refuerza el enfoque de análisis de similitud a partir de una representación global y la comparación de representaciones midiendo el esfuerzo de alineación.
- Se proporcionan representaciones con propiedades, como positividad, longitud de datos fija y valores de datos dentro de un intervalo fijo.
- Transformación del dominio 3D a 1D con la propiedad de transformación inversa de 1D a 3D. Esta propiedad no la tiene ningún descriptor de características.
- Se proporciona una forma visual de indicar la comparación parcial entre figuras 3D. La comparación parcial es una propiedad que no se considera en los métodos que usan descriptores de características.
- Se proporciona una combinación de medidas de distancia para mejorar los resultados de precisión en la recuperación.
- Se propone una técnica de visualización de la alineación de las figuras en 1D y en 3D.

### 6.3. Trabajo futuro

A continuación, se describen algunas líneas de trabajo futuro que se derivan de la metodología propuesta.

- Aplicación directa del indicador de similitud propuesto para la clasificación de modelos 3D.
- Incorporación de información semántica relativa al modelo 3D para mejorar la recuperación.
- Utilizar un método compuesto de distintas técnicas (mixto) para análisis de similitud de figuras 3D.
- Aplicación de la metodología de comparación de modelos 3D en cráneos para el diagnóstico de plagiocefalia, mediante la comparación de la secuencia representativa de un modelo de un cráneo sano, contra la del modelo de un cráneo a diagnosticar.
- Utilizar las secuencias representativas como elementos de estudio, para determinar descriptores de superficie de modelos 3D para clasificación, donde se utilizarán redes neuronales o máquinas de soporte vectorial. En este punto se incorpora el conjunto de datos de entrenamiento del *PSB*, como muestra de supervisión.
- Utilizar las secuencias representativas como elementos de estudio, para determinar descriptores de modelos 3D para clasificación, se plantea usar ondeletas <sup>1</sup>.
- Reconstrucción de puntos en 3D que componen la superficie de la figura. A partir de los puntos en 3D, realizar un análisis por cada componente  $x$ ,  $y$  y  $z$ , con el propósito de encontrar relaciones entre componentes de otras figuras 3D.
- Comparación elástica parcial, seccionando las secuencias representativas.
- Creación de un archivo nuevo de clasificación, que considere todas las figuras del *PSB*.
- Desarrollar una técnica de evaluación de comparación cuantitativa, con figuras primitivas geométricas con deformaciones graduales.

---

<sup>1</sup> Idea planteada con la colaboración del Dr. Sergio Suárez

## PUBLICACIONES

---

[1] A. Valle-Chávez and E. A. Valle-Chávez. Visualización de comparación parcial en estructuras de figuras 3D. *3rd International Supercomputing Conference in México 2012*, 2012.

[2] A. Valle-Chávez. Comparison of new techniques of representation for 3d shape data base retrieval. *International Journal of Combinatorial Optimization Problems and Informatics*, 3(3):2007-1558, Sep-Dec 2012.

[3] A. Valle-Chávez and J. Figueroa-Nazuno. Comparación de figuras 3d basada en secuencias representativas. *17th International Congress on Computer Science Research CIICC'11*, 2011.

[4] A. Valle-Chávez and J. Figueroa-Nazuno. Metodología para la búsqueda de similitud entre figuras 3d. *IX Congreso Internacional sobre Innovación y Desarrollo Tecnológico CIINDET 2011*, 2011.

[5] A. Valle-Chávez, C. Bustillo-Hernández, and J. Figueroa-Nazuno. Coherencia en fluctuaciones de precios de acciones de la bolsa mexicana de valores. *Vigésimaprimer Reunión de Otoño de Comunicaciones, Computación, Electrónica y Exposición Industrial ROC&C 2010*, 2010.

[6] A. Valle-Chávez. Implementación de galga extensiométrica de bajo costo en un guante de datos. *Vigésimaprimer Reunión de Otoño de Comunicaciones, Computación, Electrónica y Exposición Industrial ROC&C 2010*, 2010.

[7] V. Ortega-González, A. Valle-Chávez, A. Avilés-Gamboa, and D. Araujo-Díaz J. Ferrer-Tenorio E. Morales-Hernández, F. Vázquez-Velásquez. Ervin, Espacios Didácticos de Realidad Virtual para Niños. *II Congreso Nacional Estudiantil en Investigación, Querétaro, México*, 2006.



## A.1. Transformaciones de datos en 3D

### A.1.1. Conceptos y propiedades de matrices numéricas

Un aspecto importante que se persigue en los métodos de semejanza entre figuras geométricas, es la capacidad del sistema de reconocer un par de entidades distinguidas por una o más transformaciones.

En este apartado se resume algunas definiciones y propiedades elementales de las matrices. Primero daremos la definición de “matriz”.

Una *matriz*

$$M = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \quad (58)$$

es un arreglo rectangular de datos.

Si  $M$  tiene  $m$  renglones y  $n$  columnas, se dice que el tamaño de  $M$  es de  $m$  por  $n$  (escrito  $m \times n$ ).

Con frecuencia se abrevia como  $M = (a_{ij})$ . En esta ecuación  $a_{ij}$  denota el elemento de  $M$  que aparece en el renglón  $i$  y la columna  $j$ .

Una matriz  $M : \mathbb{R}^3 \rightarrow \mathbb{R}^3$  es denominada transformación lineal donde se mapea vectores a vectores por  $Y = MX$ . La linealidad se refiere a la propiedad de que  $M(cU + V) = cMU + MV$  para cualquier escalar  $c$  y cualquier vector  $U$  y  $V$ .

La *matriz de ceros* es una matriz con todos sus valores en cero.

Ejemplo.

$$0_{3,3} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad (59)$$

La *matriz identidad* es la matriz denotada como  $I$ , con valores de 1 en la diagonal y valores de 0 en las demás posiciones.

Ejemplo.

$$I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (60)$$

Una *matriz* se dice que es *inversa* si existe una matriz denotada como  $M^{-1}$ , tal que  $MM^{-1} = M^{-1}M = I$ .

Ejemplo. Sea  $A = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 5 & 1 \\ 4 & 10 & 1 \end{pmatrix}$ , entonces  $A^{-1} = \begin{pmatrix} 5 & 2 & -2 \\ -2 & -1 & 1 \\ 0 & 2 & -1 \end{pmatrix}$ .

$$AA^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = I_3 \quad (61)$$

La *transpuesta* de una *matriz*  $M = [a_{ij}]$  es la matriz  $M^T = [a_{ji}]$ , esto es, que los renglones de  $M$  se convierten en las columnas de  $M^T$ .

Ejemplo. Sea  $A = \begin{pmatrix} 1 & 2 & 0 \\ 2 & 5 & 1 \\ 4 & 10 & 1 \end{pmatrix}$

$$A^T = \begin{pmatrix} 1 & 2 & 4 \\ 2 & 5 & 10 \\ 0 & 1 & 1 \end{pmatrix} \quad (62)$$

Una *matriz* es *simétrica* si  $M^T = M$ .

Ejemplo. Sea  $A = \begin{pmatrix} 1 & 2 & 4 \\ 2 & 5 & 1 \\ 4 & 1 & 3 \end{pmatrix}$

$$A^T = \begin{pmatrix} 1 & 2 & 4 \\ 2 & 5 & 1 \\ 4 & 1 & 3 \end{pmatrix} = A \tag{63}$$

La *matriz diagonal*  $D = [d_{ij}]$  tiene la propiedad  $d_{ij} = 0$  para  $i \neq j$ , comúnmente denotada como  $D = \text{diag}\{a, b, c\}$ .

Ejemplo. Sea  $A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 3 \end{pmatrix}$

$$A = \text{diag}\{1, 5, 3\} \tag{64}$$

**A.1.2.Transformaciones de matrices de tres dimensiones**

Algunas matrices de  $3 \times 3$  se consideran especiales, debido a que se utilizan comúnmente en gráficas por computadora, ideales para transformaciones de datos en 3D.

**A.1.2.1.Rotación**

Sean  $R_x(\alpha)$ ,  $R_y(\beta)$  y  $R_z(\gamma)$  las matrices de rotación para un ángulo  $\alpha$  alrededor del eje  $x$ , un ángulo  $\beta$  alrededor del eje  $y$ , y  $\gamma$  alrededor del eje  $z$ , se define:

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\text{sen}(\alpha) & 0 \\ 0 & \text{sen}(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{65}$$

$$R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \text{sen}(\beta) & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen}(\beta) & 0 & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{66}$$

$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & -\text{sen}(\gamma) & 0 & 0 \\ \text{sen}(\gamma) & \cos(\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{67}$$

La rotación espacial de un punto  $p(x, y, z)$  está dada en los siguientes casos.

Rotación de  $\alpha$  en el eje  $x$ :

$$R_x(\alpha)p = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha) & -\text{sen}(\alpha) & 0 \\ 0 & \text{sen}(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ 0 \end{bmatrix} \quad (68)$$

Rotación de  $\beta$  en el eje  $y$ :

$$R_y(\beta)p = \begin{bmatrix} \cos(\beta) & 0 & \text{sen}(\beta) & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen}(\beta) & 0 & \cos(\beta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ 0 \end{bmatrix} \quad (69)$$

Rotación de  $\gamma$  en el eje  $z$ :

$$R_z(\gamma)p = \begin{bmatrix} \cos(\gamma) & -\text{sen}(\gamma) & 0 & 0 \\ \text{sen}(\gamma) & \cos(\gamma) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ 0 \end{bmatrix} \quad (70)$$

#### A.1.2.2. Escala

Sea  $S_v$  la matriz de escala, donde cada componente del vector  $v$  es el factor por el que se multiplica la componente de un punto  $p(x, y, z)$ . La transformación de escala en 3D del punto  $p$  está dada por:

$$S_v p = \begin{bmatrix} v_x & 0 & 0 \\ 0 & v_y & 0 \\ 0 & 0 & v_z \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} v_x p_x \\ v_y p_y \\ v_z p_z \end{bmatrix} \quad (71)$$

#### A.1.2.3. Traslación

Sea  $T_v$  la matriz de translación, donde cada componente del vector  $v$  es la magnitud de desplazamiento (puede ser negativa) en el eje correspondiente y sea  $p$  un punto con componentes en  $(x, y, z)$ . La transformación de translación en 3D del punto  $p$  está dada por:

$$T_v p = \begin{bmatrix} 1 & 0 & 0 & v_x \\ 0 & 1 & 0 & v_y \\ 0 & 0 & 1 & v_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \begin{bmatrix} p_x + v_x \\ p_y + v_y \\ p_z + v_z \\ 1 \end{bmatrix} = p + v \quad (72)$$

## A.2. Intersección rayo-polígono

### A.2.1. Intersección rayo-triángulo

La primera aproximación fue utilizar el algoritmo de T. Möller y B. Trumbore [26], que permite la utilización mínima de memoria y no necesita preprocesamiento. El algoritmo A.1 construye una transformación y la aplica al origen del rayo. El resultado de la transformación consiste en un vector que contiene la distancia  $t$  del origen a la intersección y las coordenadas.

---

**Algoritmo A.1** Algoritmo de intersección entre rayo y triángulo.

---

Entrada:

$o$ : punto de origen

$d$ : dirección

$p_0, p_1, p_2$ : vértices del triángulo

Asignar  $\varepsilon \leftarrow 0.0001$

Asignar  $e_1 \leftarrow p_1 - p_0$

Asignar  $e_2 \leftarrow p_2 - p_0$

Asignar  $q \leftarrow d \times e_2$

Asignar  $a \leftarrow e_1 \cdot q$

Si  $a > -\varepsilon$  y  $a < \varepsilon$

// El vector es paralelo al plano (la intersección es infinito).

Regresar

Fin Si

Asignar  $c \leftarrow \frac{1}{a}$

Asignar  $s \leftarrow o - p_0$

Asignar  $u \leftarrow c(s \cdot q)$

Si  $u < 0.0$

// La intersección está fuera del triángulo.

Regresar

Fin Si

---

---

```

Asignar  $r \leftarrow s \times e$ 
Asignar  $v \leftarrow c(d \cdot r)$ 
Si  $v < 0.0$  o  $(u + v) > 1.0$ 
    // La intersección está fuera del triángulo.
    Regresar
Fin Si
// Se calcula la distancia desde el origen.
Asignar  $t \leftarrow c(e2 \cdot r)$ 
Regresar  $t$ 

```

---

### A.2.2. Cálculo de intersección y distancia rayo-polígono en 3D

La manera de ofrecer una estructura de datos y algoritmos para llevar a cabo mediciones de intersección y distancia de manera eficiente sobre conjuntos de objetos finitos en 3D, es mediante un árbol *AABB*. El conjunto de objetos geométricos se almacena en la estructura de datos, donde es posible recuperarlos y realizar mediciones de intersección y registro de distancia. La idea consiste en disparar un rayo en contra de un conjunto de planos delimitados por triángulos.

El árbol de datos *AABB* toma como entrada un iterador sobre un intervalo de datos geométricos, posteriormente se convierten en primitivas. A partir de las primitivas se construye una jerarquía de espacios delimitados (*bounding boxes*) alineados de acuerdo a los ejes coordenados. La jerarquía de *bounding boxes* sirve para acelerar el cálculo de intersecciones y medición de distancias. Cada primitiva proporciona acceso a un identificador y los datos geométricos de un objeto. La primitiva consiste en un triángulo.

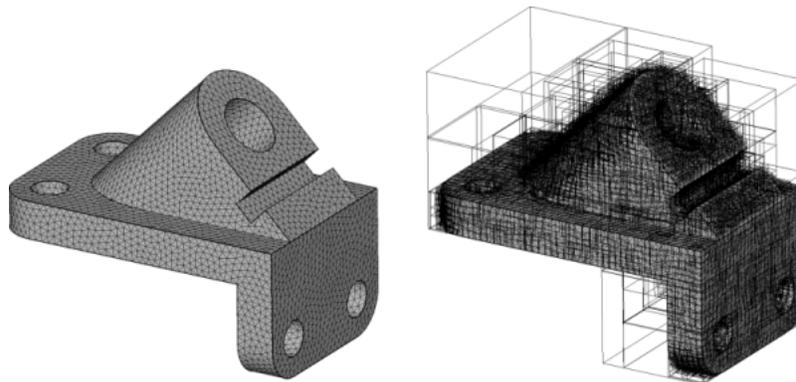


Figura 72: Izquierda: Malla de triángulos de la superficie de una parte mecánica. Derecha: Construcción de un árbol *AABB*.

## A.3.Formatos de archivos

### A.3.1.Formato de archivo de objeto 3D (.off)

Los archivos de formato de objeto (.off<sup>1</sup>) se utilizan para representar la geometría de un modelo, especificando la superficie del modelo mediante polígonos. Los polígonos se pueden componer de cualquier cantidad de vértices.

Los archivos .off que conforman el *Princeton Shape Benchmark*, corresponden con el siguiente estándar. Los archivos OFF son de tipo de codificación ASCII comenzando con el texto OFF. La siguiente línea especifica el número de vértices, el número de caras y el número de aristas. El número de aristas puede ser ignorado.

Los vértices se encuentran listados en columnas correspondientes con las coordenadas x, y ,z, donde cada renglón es un vértice en 3D. Después de la lista de vértices, se enlistan las caras, indicando una cara por renglón. Para cada cara, se especifica el número de vértices, seguido de los índices correspondientes en la lista de vértices. A continuación se muestra un ejemplo.

Nótese que los vértices se encuentran numerados comenzando desde el 0 (no comienzan en 1) y que el número de aristas se ignora colocando un cero.

Un ejemplo simple de un cubo.

```
OFF
8 6 0
-0.500000 -0.500000 0.500000
0.500000 -0.500000 0.500000
-0.500000 0.500000 0.500000
0.500000 0.500000 0.500000
-0.500000 0.500000 -0.500000
0.500000 0.500000 -0.500000
-0.500000 -0.500000 -0.500000
0.500000 -0.500000 -0.500000
4 0 1 3 2
4 2 3 5 4
4 4 5 7 6
4 6 7 1 0
4 1 7 5 3
4 6 0 2 4
```

---

<sup>1</sup> Del inglés *Object File Format*

**A.3.2.Formato de archivo de clasificación**

Un archivo de clasificación enlista las clases y los modelos que corresponden para cada clase. La clasificación contiene subconjuntos de modelos de la base de datos. Hay variedad en la forma de agrupar los modelos y el PSB proporciona clasificaciones para tomarse como referencia.

Descripción básica para el formato de clasificación con versión 1. Clasificaciones futuras tendrán un valor mayor de versión.

```
PSB NUM_VERSION
NUM_CLASES NUM_MODELOS

NOM_CLASE NOM_CLASEPADRE NUM_MODELOS_EN_CLASE
Un identificador de modelo por línea.
```

Un nombre de clase padre con un valor de 0 (cero) indica el nivel raíz de la jerarquía. Todas las clases padre deben definirse antes que las subclases. Después de la línea que especifica la clase incluyendo la clase padre y el número de modelos, se especifican los identificadores de los modelos.

Los nombres de las clases deben ser únicos.

El archivo no incluye comentarios.

A continuación un ejemplo de un archivo .cla.

```
PSB 1
5 10
aircraft 0 4
1119
1121
1123

animal 0 0

quadruped animal 0

dog quadruped 7
86
87
88
89

horse quadruped 6
103
104
105
```

## A.4. Algoritmos de registro de medidas

### A.4.1. Algoritmo completo de registros de medidas sobre un cubo

---

**Algoritmo A.2** Algoritmo de registro de medidas sobre un cubo.

---

Repetir para  $axs = 0, \dots, 2$  en intervalos de 1

    Repetir para  $cn = 0, \dots, 1$  en intervalos de  $1/t$

        Verificar  $axs$

            caso 0:

                Asignar  $z_o \leftarrow 0$

                Asignar  $x_{pos} \leftarrow 0$

                Asignar  $y_o \leftarrow cn$

            caso 1:

                Asignar  $y_o \leftarrow 0$

                Asignar  $x_{pos} \leftarrow 0$

                Asignar  $z_o \leftarrow cn$

            caso 2:

                Asignar  $x_o \leftarrow 0$

                Asignar  $y_{pos} \leftarrow 0$

                Asignar  $z_o \leftarrow cn$

        Fin Verificar

    Repetir  $i = 0, \dots, (2t + 1)$  en intervalos de 1

        Verificar  $axs$

            caso 0:

                Asignar  $x_o \leftarrow x_{pos}$

                Asignar  $x_b \leftarrow x_{pos}$

                Asignar  $y_b \leftarrow cn$

            caso 1:

                Asignar  $x_o \leftarrow x_{pos}$

                Asignar  $x_b \leftarrow x_{pos}$

                Asignar  $z_b \leftarrow z_o$

            caso 2:

                Asignar  $y_o \leftarrow y_{pos}$

                Asignar  $y_b \leftarrow y_{pos}$

                Asignar  $z_b \leftarrow z_o$

        Fin Verificar

        Obtener intersección del rayo con origen  $o(x_o, y_o, z_o)$  y dirección  $b(x_b, y_b, z_b)$

---

---

---

```
Si  $i \geq N$ 
  Si  $i > N$ 
    Verificar  $axs$ 
      caso 0:
         $x_{pos} \leftarrow x_{pos} - (1/t)$ 
      caso 1:
         $x_{pos} \leftarrow x_{pos} - (1/t)$ 
      caso 2:
         $y_{pos} \leftarrow y_{pos} - (1/t)$ 
    Fin Verificar
  Fin Si
  Verificar  $axs$ 
    caso 0:
       $z_o \leftarrow 1$ 
    caso 1:
       $y_o \leftarrow 1$ 
    caso 2:
       $x_o \leftarrow 1$ 
  Fin Verificar
Si No
  Verificar  $axs$ 
    caso 0:
       $x_{pos} \leftarrow x_{pos} + (1/t)$ 
    caso 1:
       $x_{pos} \leftarrow x_{pos} + (1/t)$ 
    caso 2:
       $y_{pos} \leftarrow y_{pos} + (1/t)$ 
  Fin Verificar
Fin Si
Fin repetir
Fin Repetir
```

---

**A.4.2. Algoritmo completo de registro de medidas sobre cilindros**


---

**Algoritmo A.3** Algoritmo para el registro de medidas sobre cilindros.
 

---

 Repetir  $axs = 0, \dots, 2$  en intervalos de 1

 Asignar  $x_b \leftarrow 1/2$ 

 Asignar  $y_b \leftarrow 1/2$ 

 Asignar  $z_b \leftarrow 1/2$ 

 Repetir  $\theta = 0, \dots, (2 - (1/t))$  en intervalos de  $(2(1/t))$ 

 Repetir  $h = 0, \dots, 1$  en intervalos de  $(1/t)$ 

 Validar  $axs$ 

caso 0:

 Asignar  $x_o \leftarrow h$ 

 Asignar  $y_o \leftarrow r_c \cos(\theta\pi) + 1/2$ 

 Asignar  $z_o \leftarrow r_c \sen(\theta\pi) + 1/2$ 

 Asignar  $x_b \leftarrow h$ 

caso 1:

 Asignar  $y_o \leftarrow h$ 

 Asignar  $x_o \leftarrow r_c \cos(\theta\pi) + 1/2$ 

 Asignar  $z_o \leftarrow r_c \sen(\theta\pi) + 1/2$ 

 Asignar  $y_b \leftarrow h$ 

caso 2:

 Asignar  $z_o \leftarrow h$ 

 Asignar  $x_o \leftarrow r_c \cos(\theta\pi) + 1/2$ 

 Asignar  $y_o \leftarrow r_c \sen(\theta\pi) + 1/2$ 

 Asignar  $z_b \leftarrow h$ 

Fin Validar

 Obtener intersección del rayo con origen  $o(x_o, y_o, z_o)$  y dirección  $b(x_b, y_b, z_b)$ 

Fin Repetir

Fin Repetir

 Fin Repetir
 

---



### B.1. La aplicación SIM3D

En esta sección se describen algunos aspectos técnicos sobre la plataforma donde se desarrolló y se ejecuta la aplicación.

#### B.1.1. Sobre *QT Creator* y *C++*

En el comienzo del desarrollo de la herramienta se utilizó *MATLAB (R2010a)*, como se menciona en comentarios de la implementación del capítulo 4, *MATLAB* ayudó a obtener rápidamente resultados indicativos.

Los tiempos de análisis en *MATLAB* no eran favorables para el análisis de una figura 3D de 300,000 polígonos. Se hicieron pruebas en lenguaje *C++* y los tiempos de respuesta fueron mucho mejor. En la nueva implementación se utilizó la interface de desarrollo *Qt Creator* (Qt 4.7.3, 32 bit), que permitió la inclusión de bibliotecas necesarias para el desarrollo. El sistema operativo de desarrollo fue con la distribución libre *LinuxMint 12*, con base en *Ubuntu Oneiric Ocelot 11.10*.

A parte de *QT Creator* y bibliotecas estándar de *C++* se utilizaron las siguientes bibliotecas.

1. *CGAL* [9] (*Computational Geometry Algorithms Library*, versión 3.8)
2. *Eigen* [34] (*Template library for linear algebra*, versión 3.0.1)
3. *VTK* [35] (*The Visualization ToolKit*, versión 5.6)

Tanto el entorno de desarrollo como las bibliotecas son *GPL (General Public License)*, esto es, de licencia general pública. Cada biblioteca se encuentra referenciada donde se puede obtener el sitio de donde fueron descargadas. Cuentan con las respectivas explicaciones de instalación y documentación de uso.

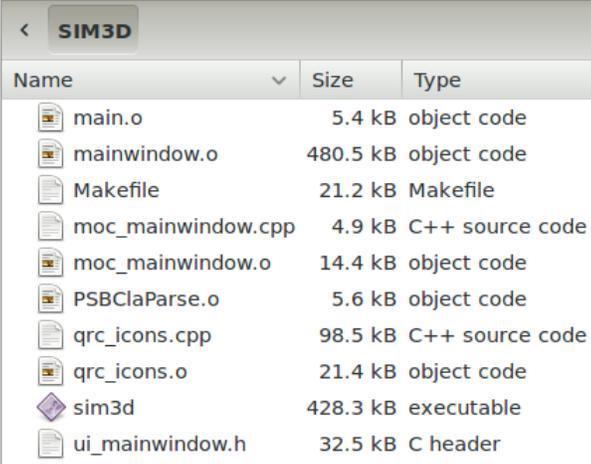
### B.1.2. Sobre la aplicación SIM3D

El ejecutable de la aplicación desarrollada y compilada en *QT Creator* se encuentra en el archivo “sim3d.zip” incluido en la carpeta “SIM3D” en el disco compacto (CD) que acompaña la tesis.

Para que la aplicación funcione, se requiere el sistema operativo *Linux*, específicamente que tenga base en *Ubuntu Oneiric Ocelot 11.10* (32 bit). También se requiere que las bibliotecas estén instaladas. Los demás archivos necesarios (por ejemplo archivos \*.o), se encuentran junto con el ejecutable “sim3d”. Al descomprimirse tanto el ejecutable y los archivos necesarios, deberán estar en la misma carpeta (ver figura 73).

La ejecución de la aplicación puede ser de dos formas distintas.

1. Hacer doble clic sobre el archivo “sim3d”.
2. Desde la línea de comandos, posicionarse en la carpeta donde está el ejecutable y teclear el comando “./sim3d”.



Name	Size	Type
main.o	5.4 kB	object code
mainwindow.o	480.5 kB	object code
Makefile	21.2 kB	Makefile
moc_mainwindow.cpp	4.9 kB	C++ source code
moc_mainwindow.o	14.4 kB	object code
PSBCLaParse.o	5.6 kB	object code
qrc_icons.cpp	98.5 kB	C++ source code
qrc_icons.o	21.4 kB	object code
sim3d	428.3 kB	executable
ui_mainwindow.h	32.5 kB	C header

Figura 73: Contenido de carpeta del archivo ejecutable.

### B.1.3. Archivos utilizados por la aplicación

Se consideran los siguientes tipos de archivos.

1. \*.off ([A.3.1](#))
2. \*.cla ([A.3.2](#))

3. \*.matrix (4.11)
4. \*.lst
5. \*.txt
6. \*.plot

El archivo \*.lst consiste en un listado simple de identificadores de figuras 3D, el orden debe de corresponder con los identificadores del archivo \*.cla. La primera línea denota una cadena que etiqueta el conjunto de figuras. La segunda línea especifica la cantidad de elementos que compone el conjunto de datos.

El archivo \*.txt se genera a partir de realizar el análisis de la figuras. Lo que contiene es una secuencia de números en una columna, donde cada renglón contiene un dato flotante.

El archivo \*.plot contiene valores de tipo flotante en dos columnas, la primera columna son los valores de recuperación, la segunda columna corresponde con los valores de precisión.

En la tabla 13 se ejemplifican los contenidos de los archivos descritos.

Tabla 13: Ejemplos de listados de archivos.

*.lst	*.txt	*.plot	
CONJUNTO	0.14978	0.050000	0.892693
3	0.169002	0.100000	0.855747
	0.223277	0.150000	0.750357
22	0.280055	0.200000	0.697632
23	0.318875	0.250000	0.654784
24	0.336879	0.300000	0.545833

#### B.1.4. Módulos de la aplicación

El sistema se compone de tres módulos:

1. Análisis y comparación
2. Medidas de desempeño
3. Recuperación de figuras 3D

El módulo (1) analiza las figuras y genera las secuencias representativas utilizando tres procesos diferentes. Después de la selección de la medida de distancia, se calcula la comparación entre secuencias representativas. A partir de la comparación se genera la matriz de distancias en el archivo \*.matrix.

Para utilizar el módulo (2), se tiene que tener el archivo de matriz de distancias y el archivo de clasificación. Se pueden realizar tres análisis distintos con dos variantes. Los análisis son: resultados de curva de precisión-recuperación, medida de vecinos más cercanos y medida-E. Cada análisis puede ser global o por cada clase. El análisis global calcula el promedio de resultados de todas las figuras y el análisis por cada clase, calcula los resultados en detalle para cada agrupación de figuras.

El módulo (3) requiere de los archivos de las figuras \*.off, la matriz de distancias \*.matrix y el archivo de clasificación \*.cla. en el módulo se puede realizar recuperaciones de figuras 3D, visualizando información asociada, por ejemplo, en la recuperación de una figura se despliegan en 3D las primeras tres respuestas, mostrando el valor de distancia calculado. También es posible exportar en formato html, todos los listados de respuestas para cada figura del conjunto de datos.

Cada uno de los módulos es accesible en las pestañas, en la parte superior de la ventana de la aplicación (ver figura 74).

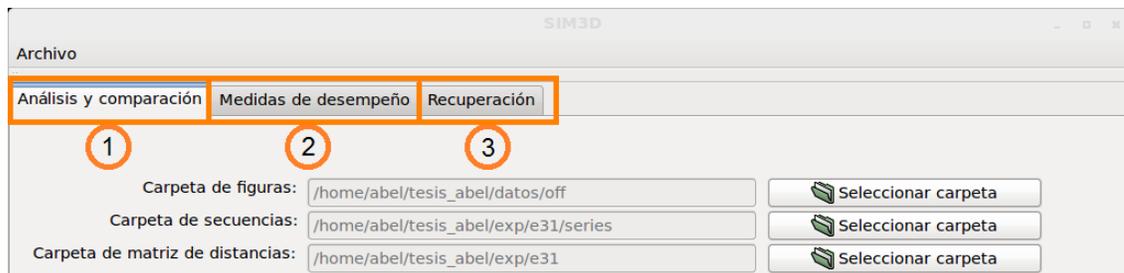


Figura 74: Pestañas con los módulos de la aplicación (1), (2) y (3).

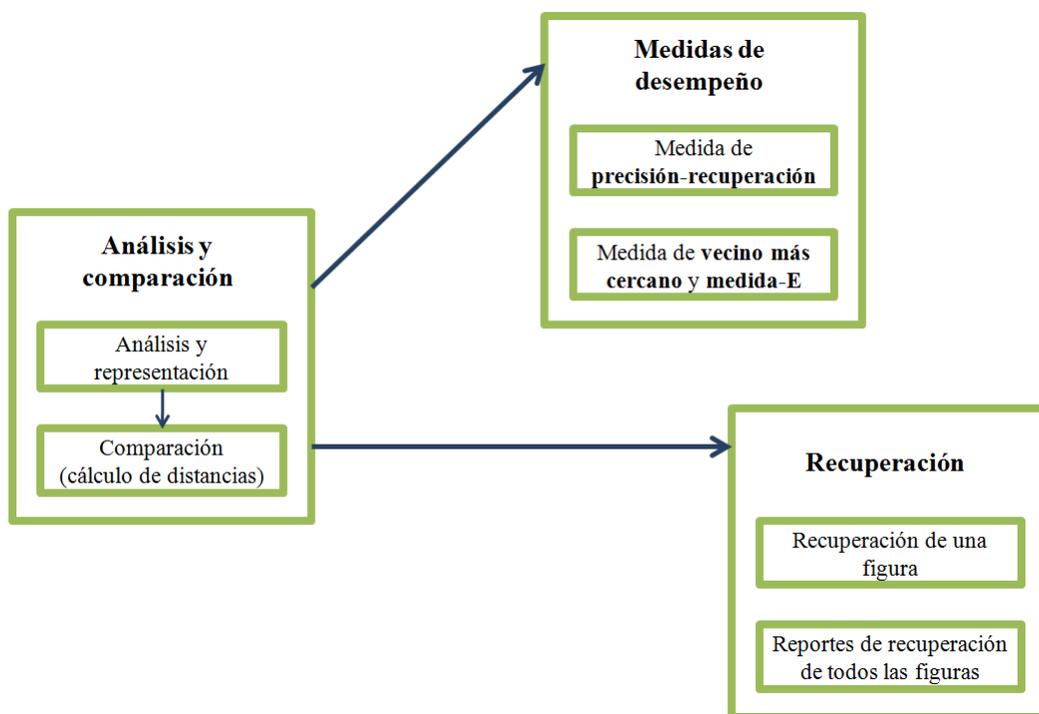


Figura 75: Diagrama de módulos de la aplicación que muestra sus dependencias.

## B.2.Manual de la aplicación SIM3D

### B.2.1.El módulo de análisis y comparación

Para visualizar el módulo, hacer clic en la pestaña de “Análisis y comparación”.

Para ejecutar el proceso de análisis de un conjunto de figuras 3D, construcción de secuencias y la comparación se siguen los siguientes pasos.

1. Seleccionar la carpeta que contiene a las figuras en formato \*.off.
2. Seleccionar la carpeta donde se escribirán los archivos de las secuencias numéricas (archivos planos de texto \*.txt).
3. Seleccionar la carpeta donde se escribirá el archivo de la matriz de distancias \*.matrix.
4. Seleccionar el archivo del conjunto de figuras a analizar \*.lst.
5. Seleccionar el o los marcos de referencia de análisis.

6. Seleccionar la cantidad de datos para la secuencia numérica por marco de referencia.
7. Seleccionar la medida de distancia.
8. Presionar el botón de comenzar el análisis.

Después de comenzar el análisis, en la parte inferior se muestra el progreso del análisis y construcción de secuencias representativas. En otro indicador de progreso se muestra el avance de la comparación de todas las figuras contra todas las figuras.

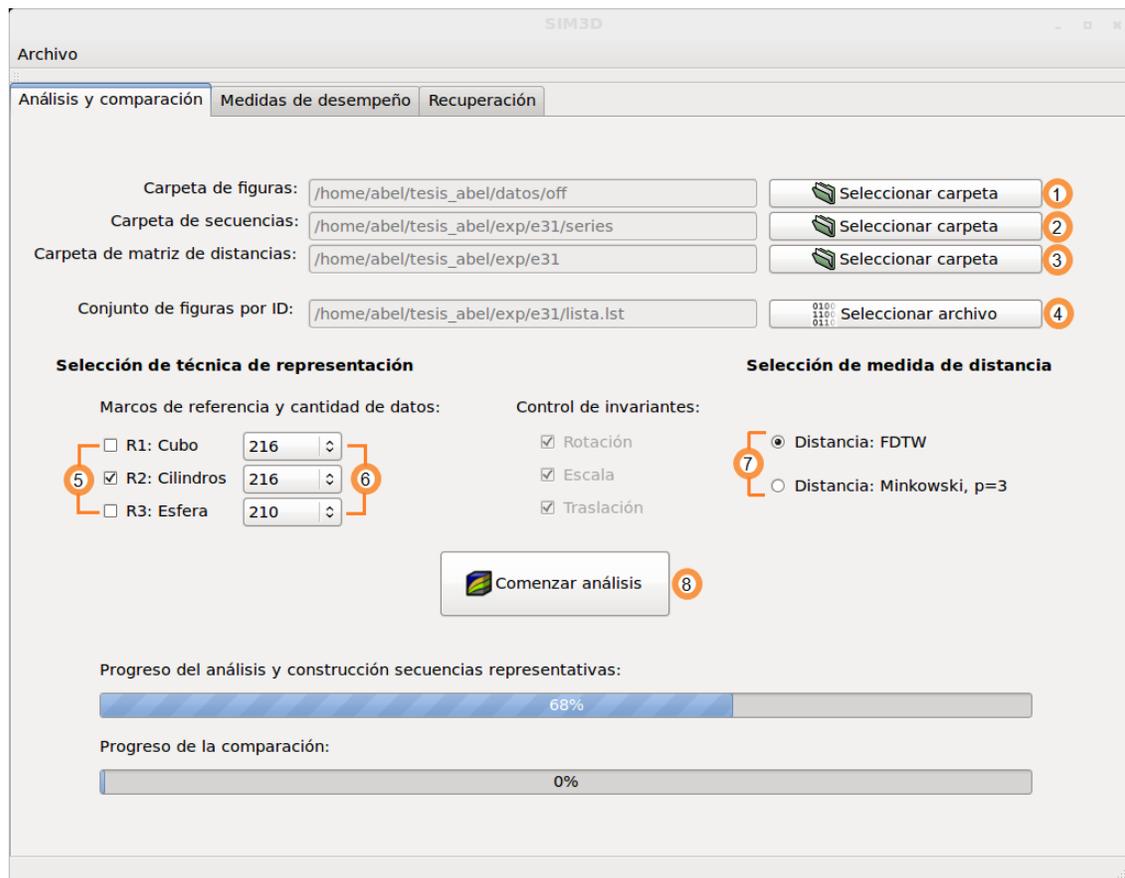


Figura 76: Módulo de análisis y comparación.

### B.2.2. El módulo de medidas de desempeño

Para visualizar el módulo, hacer clic en la pestaña de “Medidas de desempeño”.

- a) Para obtener los resultados de evaluación de *precisión-recuperación* seguir los siguientes pasos.

1. a) Seleccionar el archivo de matriz de distancias \*.matrix.
2. a) Seleccionar el archivo de clasificación \*.cla.
3. a) Seleccionar la carpeta donde se escribirá el archivo \*.plot con los valores correspondientes con los resultados.
4. a) Seleccionar el modo del análisis ya sea un análisis promedio global o por cada clase.
5. a) Presionar el botón de “exportar resultados de precisión-recuperación”.

Al finalizar el proceso se obtiene un archivo \*.plot, cada renglón es la relación precisión-recuperación.

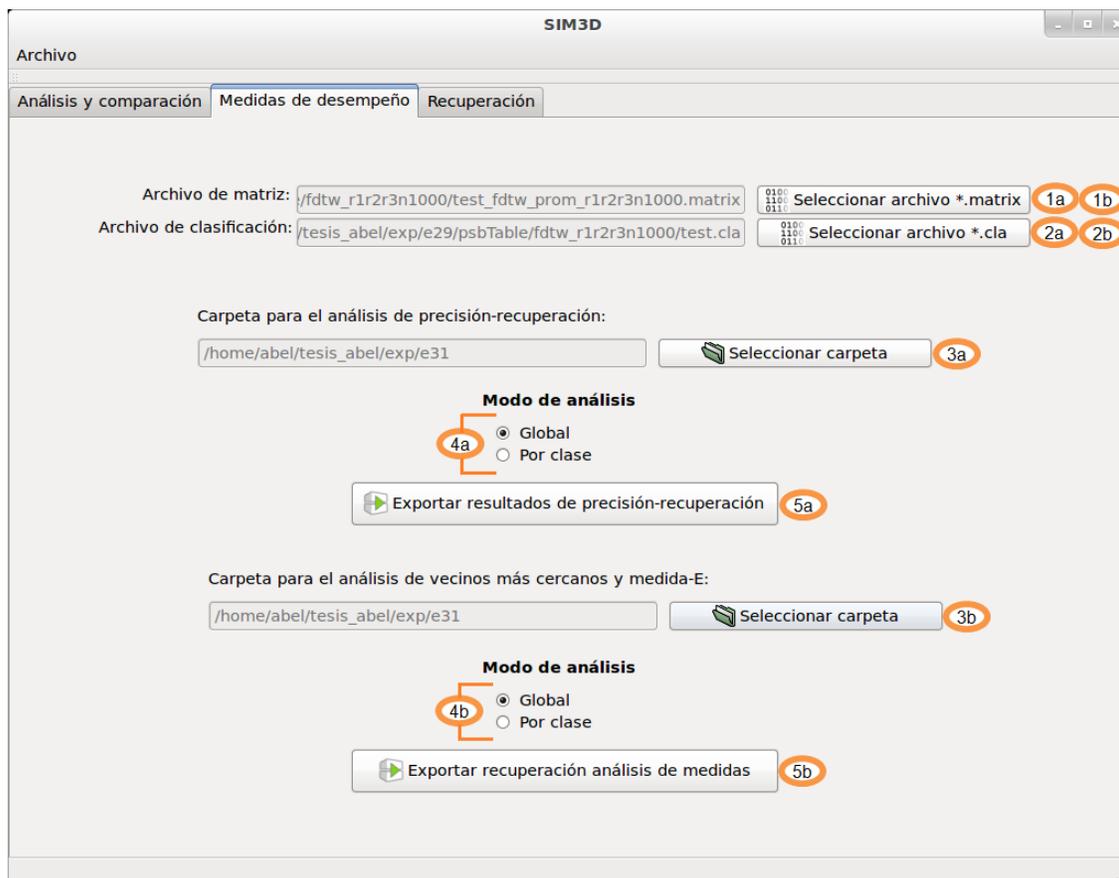


Figura 77: Módulo de medidas de desempeño.

b) Para obtener los resultados de evaluación de *vecinos más cercanos* y la *medida-E* se utilizan los archivos seleccionados en los pasos anteriores.

1. b) Verificar que se haya seleccionado el archivo de matriz de distancias \*.matrix.
2. b) Verificar que se haya seleccionado el archivo de clasificación \*.cla.
3. b) Seleccionar la carpeta donde se escribirá los archivos \*.txt con el listado de los resultados.
4. b) Seleccionar el modo del análisis ya sea un análisis promedio global o por cada clase.
5. b) Presionar el botón de “exportar resultados de análisis de medidas”.

Al finalizar el proceso se obtienen dos archivos \*.txt con el valor o los valores de resultados. Para el caso donde el modo de análisis es global sólo se obtiene un valor por archivo. Para el caso donde el análisis es por clases, se tiene un listado de resultados por cada clase.

### **B.2.3.El módulo de recuperación**

Para visualizar el módulo, hacer clic en la pestaña de “Recuperación”.

- a) Para realizar recuperaciones y visualizarlas en 3D con información asociada.
  1. a) Seleccionar la carpeta que contiene a las figuras en formato \*.off.
  2. a) Seleccionar el archivo de matriz de distancias \*.matrix.
  3. a) Seleccionar el archivo de clasificación \*.cla.
  4. a) Presionar el botón de “cargar datos de matriz y clasificación”.
  5. a) Cargando la información permite la recuperación de las figuras con los valores de distancia más pequeños ordenados de menor a mayor. Para ello presionar el botón de “cargar figura off”, se selecciona el archivo y automáticamente se cargan los modelos más similares.

Mediante la interfaz gráfica no es posible visualizar todos los resultados. Para visualizar todos los resultados se ejecuta un proceso que exporta todas las recuperaciones de cada figura en formato *html*. Para generar los formatos html se utilizan los archivos seleccionados en los pasos anteriores.

- b) Para exportar las recuperaciones en *html* se siguen los siguientes pasos.

1. b) Verificar que se haya seleccionado el archivo de matriz de distancias \*.matrix.
2. b) Verificar que se haya seleccionado el archivo de clasificación \*.cla.
3. b) Seleccionar la carpeta donde se escribirán los archivos \*.html.
4. b) Presionar el botón de “exportar recuperación de figuras en listados”.

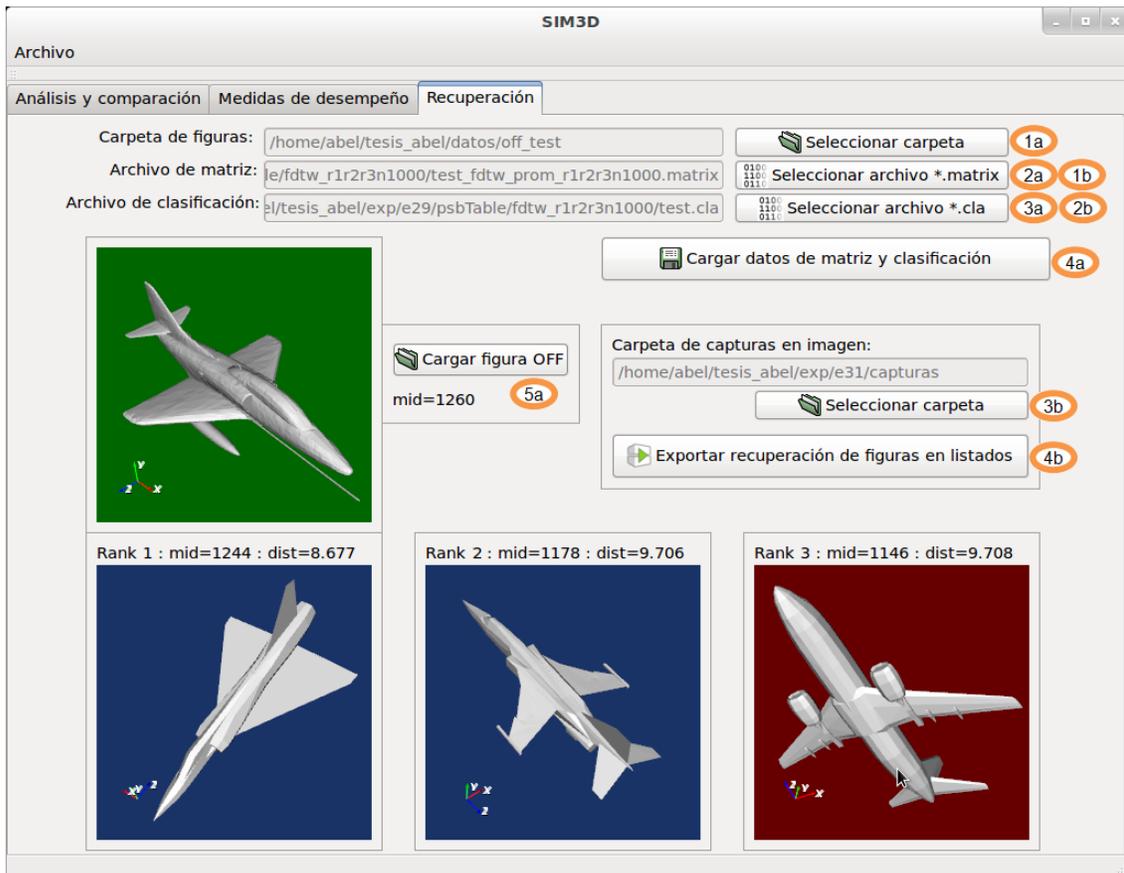


Figura 78: Módulo de recuperación.



## REFERENCIAS

---

- [1] A. Angeles-Yreta. Cómputo de la similitud entre figuras geométricas. Master's thesis, Centro de Investigación en Computación del Instituto Politécnico Nacional, 2006. (Cita en páginas 2 y 32.)
- [2] A. Angeles-Yreta and J. Figueroa-Nazuno. Semejanza entre objetos tridimensionales aplicando redes neuronales fuzzy art. *Advances in Computer Science in Mexico, Research on Computer Science*, 13:127–134, 2005. ISSN 1665-9899. (Cita en página 2.)
- [3] M. Ankerst, G. Kastenmuller, H. Kriegel, and T. Seidl. Nearest neighbor classification in 3d protein databases. In AAAI Press, editor, *7th International Conference on Intelligent Systems for Molecular Biology*, pages 34–43, 1999. (Cita en página 7.)
- [4] R.A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA., 1999. (Cita en página 38.)
- [5] Princeton Shape Benchmark. <http://shape.cs.princeton.edu/benchmark/index.cgi>, 2003. URL <http://shape.cs.princeton.edu/benchmark/index.cgi>. (Cita en página 38.)
- [6] C. Böhm, S. Berchtold, and D.A. Keim. Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. In *ACM Comput. Survey.*, volume 33, pages 322–373, 2001. (Cita en página 21.)
- [7] B. Bustos, D. Keim, D. Saupe, T. Scheck, and D.V. Vranic. An experimental effectiveness comparison of methods for 3d similarity search. *Internat. J. Digital Libraries*, 2005. (Cita en páginas xxviii, 15, 22, 84, 85 y 97.)
- [8] R. J. Campbell and P.J. Flynn. A survey of free-form object representation and recognition techniques. In *Computer Vision Image Understand*, volume 81, pages 166–210, 2001. (Cita en página 8.)

- [9] Computational Geometry Algorithms Library CGAL. <http://www.cgal.org>. URL <http://www.cgal.org>. (Cita en páginas 72 y 115.)
- [10] E. Chávez, G. Navarro, R. Baeza-Yates, and J.L. Marroquín. Searching in metric spaces. In *ACM Comput. Surv.*, volume 33, pages 273–321, 2001. (Cita en página 21.)
- [11] K. Devlin. Mathematics: The sciences of patterns. *Scientific American Library, New York*, page 215, 1994. (Cita en página 5.)
- [12] J. Figueroa-Nazuno E.A. Santos-Camacho. Comparación de diferentes índices de similitud. *17th International Congress on Computer Science Research CIICC'11*, 2011. (Cita en páginas 24, 27 y 35.)
- [13] J. Figueroa-Nazuno, A. Angeles-Yreta, J. Medina-Apodaca, V. Ortega-González, K. Ramírez-Amaro, M. Mirón-Bernal, and V. Landassuri-Moreno. Sobre el problema de semejanza. In *Reporte Técnico*, Mexico, DF, 2006. Instituto Politecnico Nacional. doi: ISBN:970-36-0343-2. (Cita en página 23.)
- [14] Z. Geradts, H. Hardy, A. Poortman, and J. Bijhold. Evaluation of contents-based image retrieval methods for a database of logos on drug tablets. In *SPIE Enabling Technologies for Law Enforcement and Security*, volume 4232, pages 553–562, 2001. (Cita en página 7.)
- [15] D. Giorgi, S. Biasotti, and L. Paraboschi. Shape retrieval contest 2007: Watertight models track. *SHREC 2007: 3D Shape Retrieval Contest, Technical Report*, art. UU-CS-2007-015, 2007. (Cita en página 40.)
- [16] S. Goodall, P.H. Lewis, K. Martinez, P.A..S. Sinclair, F. Giorgini, M. Addis, M.J. Boniface, C. Lahanier, and J. Stevenson. Sculpteur: Multimedia retrieval for museums. *Proc. Third Int'l Conf. Image and Video Retrieval*, pages 638–646, 2004. (Cita en página 40.)
- [17] A. Guzmán. *Computer Recognition of Three-dimensional Objects in a Visual Scene*. PhD thesis, Electrical Engineering Department, M. I. T., December 1968. Project MAC Technical Report MAC TR 59. AD-692-200. (Cita en páginas xxv y 13.)
- [18] M. Hilaga, Y. Shinagawa, T. Kohmura, and T.L. Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In *Proc. ACM SIGGRAPH*, pages 203–212, August 2001. (Cita en página 16.)
- [19] Ilknur Icke. Content based 3d shape retrieval a survey of state of the art. Technical report, City University of New York, 2004. (Cita en página 2.)

- [20] S. Jayanti, K. Kalyanaraman, N. Iyer, and K. Ramani. Developing an engineering shape benchmark for cad models. *Computer Aided Design*, 38(9):939–953, 2006. (Cita en página 40.)
- [21] D. Keim. Efficient geometry-based similarity search of 3d spatial databases. In *1999 ACM SIGMOD International Conference on Management of Data (SIG-MOD'99)*, pages 419–430, New York, NY, 1999. ACM Press. (Cita en página 7.)
- [22] E. Keogh. Exact indexing of dynamic time warping. In *28th International Conference on Very Large Data Bases*, pages 406–417, 2002. (Cita en página 32.)
- [23] D. Keysers and W. Unger. Elastic image matching is np-complete. In *Pattern Recognition Letters*, volume 24, pages 445–453, January 2003. (Cita en página 26.)
- [24] Donald E. Knuth. Computer Programming as an Art. *Communications of the ACM*, 17(12):667–673, December 1974. (Cita en página xiii.)
- [25] Loncaric. A survey of shape analysis techniques. In *Pattern Recognition*, volume 31, pages 983–1001, 1998. (Cita en página 14.)
- [26] T. Müller and B. Trumbore. Fast, minimum. storage ray-triangle intersection. *Journal of Graphics Tools*, pages 37–46, 22–28, 1997. (Cita en páginas 71 y 107.)
- [27] M. Novotni and R. Klein. Geometric 3d comparison - an application. In *ECDL WS Generalized Documents*, 2001. (Cita en página 8.)
- [28] E.V. Ortega-González. Una técnica para el análisis de similitud entre imágenes. Master's thesis, Centro de Investigación en Computación del Instituto Politécnico Nacional, 2007. (Cita en páginas 23, 34 y 47.)
- [29] V. Ortega-González, A. Valle-Chávez, A. Avilés-Gamboa, and D. Araujo-Díaz J. Ferrer-Tenorio E. Morales-Hernández, F. Vázquez-Velásquez. Ervin, espacios didácticos de realidad virtual para niños. *II Congreso Nacional Estudiantil en Investigación, Querétaro, México*, 2006. (Cita en página 2.)
- [30] L.G. Roberts. Machine perception of three-dimensional solids. In J.T. Tippett et al., editor, *Optical and Electrooptical information processing*, pages 159–197. MIT Press, 1965. (Cita en página 13.)
- [31] 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 IEEE Computer Society, editor, *16th International Conference on Pattern Recognition*, volume 2, pages 290–295, 2002. (Cita en página 7.)

- [32] H. Sundar. Skeleton based shape matching and retrieval. *Proc. Shape Modeling Int'l*, page 130, 2003. (Cita en página 16.)
- [33] K. Teknomo. Similarity measurement. 2007. (Cita en página 26.)
- [34] Eigen Template Library for Linear Algebra. <http://eigen.tuxfamily.org>. URL <http://eigen.tuxfamily.org>. (Cita en páginas 72 y 115.)
- [35] The Visualization ToolKit. <http://www.vtk.org>, 2011. URL <http://www.vtk.org>. (Cita en página 115.)
- [36] Gino van den Bergen. *Collision Detection in Interactive 3D Environments*. Elsevier, 2004. doi: ISBN-10:155860801X. (Cita en página 62.)
- [37] R.C. Veltkamp. Shape matching: Similarity measure and algorithms. In *Proceedings Shape Modelling International*, pages 188–197, 2001. (Cita en páginas 27 y 28.)
- [38] D.V. Vranic. *3D Model Retrieval*. PhD thesis, Univ. of Leipzig, 2004. (Cita en página 2.)
- [39] D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. In *IEEE Transactions on Evolutionary Computation*, volume 1, pages 67–82, 1997. doi: ISSN:1089-778X. (Cita en página 61.)