

00327

2



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE CIENCIAS

**UNA MEDIDA DE SIMILITUD PARA CURVAS
TRIDIMENSIONALES REPRESENTADAS POR MEDIO DEL
CÓDIGO DE CADENAS DE CAMBIOS DE DIRECCIÓN
ORTOGONAL.**

T E S I S

**QUE PARA OBTENER EL TÍTULO DE
LICENCIADA EN CIENCIAS DE LA COMPUTACIÓN**

EJEMPLAR UNICO

**P R E S E N T A:
WENDY ELIZABETH AGUILAR MARTÍNEZ**



**FACULTAD DE CIENCIAS
UNAM**

DIRECTOR DE TESIS: DR. ERNESTO BRIBIESCA CORREA



**FACULTAD DE CIENCIAS
SECCION ESCOLAR**

2003

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

Autorizo a la Dirección General de Bibliotecas de la UNAM a difundir en formato electrónico e impreso el contenido de mi trabajo referencial.

NOMBRE: Wendy Elizabeth Aguilar Martínez

FECHA: 10 - Septiembre - 2003

FIRMA: [Signature]

DRA. MARÍA DE LOURDES ESTEVA PERALTA
Jefa de la División de Estudios Profesionales de la
Facultad de Ciencias
Presente

Comunicamos a usted que hemos revisado el trabajo escrito: Una medida de similitud para curvas tridimensionales representadas por medio del código de cadenas de cambios de dirección ortogonal.

realizado por Wendy Elizabeth Aguilar Martínez

con número de cuenta 9330505-3 , quien cubrió los créditos de la carrera de: Lic. en Ciencias de la Computación

Dicho trabajo cuenta con nuestro voto aprobatorio.

Atentamente

Director de Tesis
Propietario

DR. ERNESTO BRIBIESCA CORREA

E. Bribiesca

Propietario

M. en I. JAIME AYALA PEREZ

[Signature]

Propietario

DR. HERMILO SANCHEZ CRUZ

[Signature]

Suplente

M. en C. MIGUEL MIRANDA MIRANDA

[Signature]

Suplente

DR. PEDRO MIRAMONTES VIDAL

[Signature]

Consejo Departamental de Matemáticas



FACULTAD DE CIENCIAS
CONSEJO DEPARTAMENTAL

MAT. ANA LUISA SOLÍS GONZÁLEZ COSÍO

TESIS CON FALLA DE ORIGEN

1.a

AGRADECIMIENTOS Y DEDICATORIAS

A Dios, por iluminar mi camino.

A mis padres, porque este triunfo es de ustedes. Mis dos grandes pilares, mis dos grandes amores. Mil gracias por darme tanto, por no dejarme caer nunca, por darme una familia tan unida y tan increíble, por construir un verdadero hogar en donde pudiera crecer en armonía. Gracias por confiar en mí, por apoyarme en todas mis decisiones y por darme la libertad de elegir mi camino.

Mami, gracias por ser un ángel en mi vida. Admiro tu fortaleza, tu decisión para lograr tus objetivos, tu perseverancia y tu entrega en todo lo que haces. Siempre tienes un remedio para curarlo todo, aunque la mayoría de las veces basta con un abrazo tuyo para sentirme mejor. Jamás hubiera podido llegar hasta aquí si no hubiera sido por ti.

Papi, gracias por festejarme todos mis triunfos, por motivarme a ser mejor cada día y a dar siempre lo mejor de mí. Gracias por respaldar todos mis proyectos que emprendo, el hecho de saber que estas conmigo me llena de fortaleza. Gracias por haber apoyado mis mil y un cursos que desde niña me interesaron, estos han sido fundamentales en mi desarrollo personal y académico. Finalmente, gracias por cuidarme siempre con tanto amor.

Nancy, gracias por ser un ejemplo para mí y por quererme tanto a pesar de ser tan distintas y de no estar de acuerdo con mis ideas. Gracias por motivar mi curiosidad desde que era un bebé, porque has sido parte de mi formación desde siempre. Eres una mujer admirable de quien he aprendido mucho. Ojalá algún día pueda leer la mitad de todo lo que has leído, con el mismo entusiasmo e interés que lo haces.

Héctor, gracias por caminar a mi lado, por llenar mi vida con esa alegría, sencillez y tranquilidad que te caracterizan. Siempre que te he necesitado, has estado ahí con esa gran sonrisa. Gracias por todos tus comentarios, ayuda y tips que fueron de mucha ayuda para concluir este trabajo. Finalmente, gracias por compartir tu vida y por dejarme compartir la mía, por tantos sueños, por tanto amor.

A toda mi familia (Aguilar y Martínez), que siempre me ha apoyado.

A mi director de tesis, el Dr. Bribiesca. Durante la realización de esta tesis aprendí mucho de usted. Gracias por contagiarme de su entusiasmo por la investigación y por creer en mis capacidades.

Gracias a todos mis sinodales por sus comentarios y correcciones a este trabajo.

Agradezco de manera muy especial a José Luis Villarreal, por su apoyo y consejos como jefe y como amigo, a lo largo de mi estancia en el Departamento de Visualización en DGSCA.

Esta tesis también va para todos mis amigos. Me siento muy afortunada de que formen parte de mi vida y de alguna forma de yo formar parte de la suya. A mis amigas y amigos de la secundaria, prepa, DGSCA y demás lugares por los que me los he encontrado. Muchas gracias por compartir este momento tan importante para mí.

INDICE

Introducción	i
--------------------	---

Parte I. FUNDAMENTOS

1 Percepción visual de la forma y similitud	1
1.1 Principales teorías de la percepción visual de la forma	2
1.1.1 La escuela Gestalt	2
1.1.2 Teoría de Hebb	4
1.1.3 Teoría de Gibson	5
1.1.4 Teorías modernas de percepción visual de la forma	6
1.2 Similitud	8
1.2.1 Aproximación de elementos comunes	9
1.2.2 Modelos de plantillas	10
1.2.3 Aproximación geométrica	11
1.2.4 Modelos por características	12
1.2.5 Medidas de similitud basadas en funciones de distancia	13
1.2.6 Basados en alineaciones	14
1.2.7 Transformacionales	14
1.2.8 Teoría Geon	15
1.2.9 Comentarios finales	15
2 Correspondencia de patrones geométricos	17
2.1 Correspondencia de patrones espacial	17
2.1.1 Correspondencia de patrones fotométrico y geométrico	19
2.1.2 Correspondencia total y correspondencia parcial	20
2.1.3 Correspondencia exacta y aproximada	21
2.1.4 Problemas relacionados con la correspondencia de patrones geométricos	21
2.1.5 Obteniendo patrones geométricos	22
2.2 Principales técnicas en correspondencia de patrones geométricos	23
2.2.1 Métodos globales	24
2.2.2 Correspondencia de gráficas	24
2.2.3 Hashing geométrico	25
2.2.4 Correspondencia de patrones basada en medidas de similitud	25
2.3 Correspondencia de patrones basado en medidas de similitud	26
2.3.1 Métodos de correspondencia de congruencia exacta	26
2.3.2 Propiedades deseables en las medidas de disimilitud	27
2.3.3 Medidas de disimilitud para conjuntos finitos de puntos	30

2.3.3.1	Distancia L_p , Distancia de Minkowski	30
2.3.3.2	Distancia uniforme	31
2.3.3.3	Distancia de Hausdorff	31
2.3.4	Medidas de disimilitud para curvas	32
2.3.4.1	Función de ángulo	32
2.3.4.2	Función de firma	34
2.3.4.3	Longitud de arco afin	35
2.3.4.4	Distancia de Hausdorff	35
2.3.4.5	Distancia de Fréchet	35
2.3.5	Construcciones	37
2.3.5.1	Remapeo	37
2.3.5.2	Normalización	37
2.3.5.3	De Semi-métrica a Métrica	37
2.3.5.4	Ventaja	37
3	Representación de curvas tridimensionales.	39
3.1	Propiedades deseables	40
3.2	Principales representaciones de curvas	41
3.2.1	Representación explícita	41
3.2.2	Representación implícita	42
3.2.3	Forma paramétrica	43
3.2.3.1	Curvas paramétricas polinomiales	44
3.2.3.2	Curvas paramétricas polinomiales cúbicas	45
3.2.4	Códigos de cadenas para curvas discretas tridimensionales.	47
3.2.4.1	Código de cadenas de Freeman	47
3.2.4.2	Código de cadenas de cambio de dirección ortogonal.	48
3.2.5	Otras representaciones	48
3.3	Código de cadenas de cambio de dirección ortogonal	48
3.3.1	Conceptos y definiciones	49
3.3.2	Independencia de rotación	50
3.3.3	Inverso de una cadena	51
3.3.4	Independencia del punto de inicio para curvas abiertas	53
3.3.5	Independencia del punto de inicio para curvas cerradas	53
3.3.6	Invarianza bajo la transformación de espejo	54
3.3.7	Comparación de curvas	56
3.3.8	Operación de ligado	56
3.3.9	Comparación entre los códigos de cadenas de Freeman y de Bribiesca.	57

Parte II. DESARROLLO

4 Presentación teórica de la medida de similitud	59
4.1 Definiciones	59
4.2 Reconocimiento y disimilitud de subcurvas	62
4.2.1 Reconocimiento de las subcurvas que tienen en común dos curvas	62
4.2.2 Disimilitud de una subcurva común a dos curvas	63
4.2.2.1 Dirección acumulada	64
4.2.2.2 Pseudométrica de Dirección Acumulada	66
4.2.2.3 Disimilitud entre dos subcadenas de una pareja común máxima	67
4.2.2.4 Disimilitud acotada entre dos subcadenas de una pareja común máxima	71
4.4 Medida de similitud entre dos curvas	74
4.3.1 Propiedades	77
4.3.2 Medida de similitud acotada, entre dos códigos de cadenas	80
5 Implementación	83
5.1 Algoritmos	83
5.1.1 Algoritmo general	83
5.1.2 Algoritmo para encontrar todas las parejas comunes máximas de A y B. ...	85
5.1.3 Algoritmo para elegir las parejas comunes máximas de A y B sin conflictos entre sí.	87
5.2 Tipos abstractos de datos	95
5.2.1 Clase CódigoDeCadenas3D	95
5.2.1.2 Clase ElementoDeCadena	96
5.2.1.3 Clase ParejaComunMaxima	96
5.2.1.4 Clase Subcadena3D	96
5.2.1.5 Clase Interseccion	96
5.2.1.6 Clase CorrespondenciaSubcadenas3D	96
5.3 Aplicación	97
6 Resultados	99
6.1 Curvas con cadenas muy parecidas	99
6.2 Curvas con correspondencias parciales	100
6.3 Espirales	101
6.4 Curvas cerradas de longitud 18	102
6.5 Autosimilitud en la curva de Peano	103
6.6 Autosimilitud en la curva de hilbert en 3d	104
Conclusiones	107
Referencias	109

INTRODUCCIÓN

La similitud es un concepto importante y ampliamente usado. Esto se debe a que existen muchos procesos en los cuales la similitud juega un papel importante, como se ejemplifica a continuación. El éxito de las personas en la solución de problemas, depende de la similitud de problemas previamente resueltos con los problemas actuales. El razonamiento inductivo esta basado en el principio de que si un evento es similar a un evento previo, entonces se pueden predecir resultados similares. La clasificación depende de la similitud de los objetos para que éstos sean clasificados como miembros de cierta categoría. Es por esto que la similitud, el reconocimiento y la clasificación son cruciales para muchas de las investigaciones recientes.

Sin embargo, la pregunta: "¿Cómo decidimos que dos cosas son parecidas o diferentes?", es aún una pregunta abierta. Esto se debe a que existen muchos factores que influyen en el proceso de similitud, es decir, los juicios de similitud están influenciados por el contexto, la perspectiva, las alternativas de elección y la experiencia. Mas aún, es probable que se usen diferentes procesos para realizar un juicio de similitud para diferentes tareas, dominios y estímulos. Por estas razones, la decisión de qué tan similares son dos cosas es aún una opinión subjetiva y personal, y es un proceso que aún no esta entendido por completo, ni por los psicólogos, ni por la medicina, ni por los científicos que intentan crear modelos matemáticos y computacionales del proceso de similitud.

Actualmente es de interés el desarrollo de sistemas computacionales artificiales de percepción visual, que sean capaces de comparar, reconocer y clasificar patrones geométricos, como por ejemplo curvas. Esto no es tan sencillo y actualmente es un reto para los investigadores de las áreas de visión por computadora, reconocimiento de patrones, análisis de imágenes médicas, procesamiento de imágenes de rango, análisis de secuencias de imágenes, navegación robótica, etc.

Por ejemplo, intente contestar a la pregunta: ¿qué tan parecidas son las dos curvas de la Figura 1 en el intervalo del 0 al 1, en donde 1 indica que son idénticas y 0 que son completamente distintas?

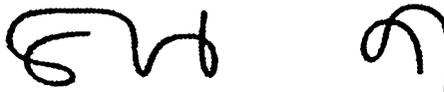


Figura 1 Dos curvas tridimensionales.

Pregúntese ahora, ¿qué juicios tomó en cuenta para poder tomar una decisión? y ¿cómo hacer que una computadora pueda realizar el mismo proceso?

TESIS CON
FALLA DE ORIGEN

Decidir qué tan similares son dos curvas se complica cuando éstas sufren transformaciones como rotaciones, cambios de escala, oclusiones parciales, cuando son vistas desde distintos ángulos, etc. Algunos ejemplos de estas transformaciones se muestran en las curvas de la Figura 2.



Figura 2 Curvas tridimensionales con variaciones en rotaciones y escalas.

Por otro lado, el estudio de las curvas es importante ya que existe una cantidad considerable de comportamientos y estructuras en la naturaleza y en el mundo en general, que pueden ser modelados por medio de curvas. Pensemos por ejemplo en la trayectoria del vuelo de una mosca o de una gota de agua, en las huellas digitales, caracteres, en un electrocardiograma, en contornos de objetos, en las órbitas de los planetas y hasta en secuencias de ADN. El proveer de métodos que cuantifiquen la similitud de curvas, permite realizar artificialmente el proceso de reconocimiento, clasificación y finalmente el análisis de los entes representados por dichas curvas.

Se han desarrollado diferentes métodos para comparar curvas, aunque el mayor esfuerzo se ha centrado en curvas bidimensionales. Para este caso, destacan las técnicas de correspondencia de contornos, correspondencia estructural (el cual esta basado en características estructurales específicas) y correspondencia de conjuntos de puntos. Los descriptores de Fourier, que se derivan de las representaciones de contornos, también han sido usados por varios autores para describir la similitud de formas.

Sin embargo, existe muy poca investigación reportada con lo que respecta a similitud de curvas tridimensionales. Se han hecho propuestas utilizando operaciones de cadenas [Rodriguez et al, 2001], usando medidas de similitud [Hagedoorn, 2000] y extrayendo puntos característicos (por ejemplo puntos de mayor curvatura) [Li, 1993], entre otros [Lo Chong-Huah y Don Hon-Son, 2000], [Kong y Kimia, 2001]. Pero cada uno de estos con sus limitantes, lo cual hace que sea actualmente un tema de investigación activo.

En la presente tesis se propone una medida de similitud para curvas tridimensionales representadas con un código de cadenas de cambios de dirección ortogonal. Dicha representación es invariante a las transformaciones de translación, rotación y espejo, y puede hacerse invariante al punto de inicio. Además, es una representación que preserva la información y que permite la reducción de datos considerablemente.

El objetivo de esta tesis es:

Dar una medida de similitud para comparar dos curvas tridimensionales representadas con un código de cadenas de cambios de dirección ortogonal, de tal forma que permita contestar a la pregunta: ¿Qué tan similar (disimilar) es una curva tridimensional A a una curva tridimensional B?

Para contestar a esta pregunta se presenta una función que asocia a cada pareja de curvas con un valor numérico en el intervalo de 0 a 1, en donde 1 indica que son idénticas y 0 que son completamente diferentes. Esta función se basa en cuantificar las similitudes y diferencias entre las curvas dadas. Por un lado, busca las subcurvas en común a ambas curvas y elimina las intersecciones entre éstas, obteniendo una asociación uno a uno. Para eliminar las intersecciones usa el criterio de preferir a aquellas subcurvas cuya disimilitud D sea menor. D es una medida de disimilitud entre subcurvas que también se propone en esta tesis. Por otro lado, una vez que se tengan las correspondencias uno a uno, se miden las longitudes de las subcurvas que quedaron sin correspondencia, lo cual cuantifica las diferencias entre las curvas. Finalmente, se calcula la similitud entre ambas curvas de acuerdo a una función Sim que conjunta las similitudes y diferencias.

Este trabajo de investigación pretende dar una propuesta nueva al problema de similitud de curvas 3D. Es importante mencionar que actualmente no existe ninguna otra propuesta para comparar dos curvas representadas con éste código.

La tesis se encuentra dividida en dos partes principales, con la finalidad de separar claramente los capítulos que fundamentan y ponen en contexto el trabajo de aquellos en donde se realizan las aportaciones.

La parte I, esta compuesta de tres capítulos en donde se presentan los fundamentos y antecedentes del tema. En el capítulo 1 se define el concepto de similitud y se presentan sus diferentes aproximaciones. En el capítulo 2 se desarrolla el tema de correspondencia de patrones geométricos, en particular de curvas. Finalmente, en el capítulo 3 se muestran las principales representaciones de curvas tridimensionales y se introduce la representación de código de cadenas de cambios de dirección ortogonal en el cual se basa ésta tesis.

La parte II es el desarrollo de la tesis. Está compuesta de tres capítulos en donde se desarrolla la propuesta. El capítulo 4 da una presentación teórica a la medida de similitud, el capítulo 5 muestra los algoritmos e implementación de dicha medida y en el capítulo 5 se muestran los resultados de aplicar la medida de similitud a diferentes conjuntos de curvas tridimensionales generadas sintéticamente.

PARTE I

FUNDAMENTOS

En la parte I se presentan los fundamentos que pondrán en contexto y darán soporte a la medida de similitud propuesta.

Esta compuesta de 3 capítulos:

Capítulo 1. Percepción visual de la forma y similitud.

Capítulo 2. Correspondencia de patrones geométricos.

Capítulo 3. Representación de curvas tridimensionales.

PAGINACION

DISCONTINUA

CAPITULO 1

PERCEPCIÓN VISUAL DE LA FORMA Y SIMILITUD

El sistema visual humano (HVS, por sus siglas en inglés) es uno de los sistemas mas sofisticados y versátiles que existen en la naturaleza. Tan es así, que a pesar de la gran cantidad de investigación que se ha desarrollado en torno al estudio del HVS, sus habilidades no han podido ser mejoradas por los sistemas de percepción visual creados artificialmente. Por ejemplo, actualmente sigue siendo un reto el reconocimiento y clasificación de rostros, de objetos y de estructuras visuales en general (en particular, las curvas) por medio de un sistema computarizado autónomo.

Por otro lado, la Similitud es uno de los problemas centrales de la ciencia, en particular de la psicología. Esto se debe a que existen muchos procesos en los cuales la similitud juega un papel importante, como se ejemplifica a continuación. El éxito de las personas en la solución de problemas, depende de la similitud de problemas previamente resueltos con los problemas actuales. El razonamiento inductivo está basado en el principio de que si un evento es similar a un evento previo, entonces se pueden predecir resultados similares. La clasificación depende de la similitud de los objetos para que estos sean clasificados como miembros de cierta categoría. El reconocimiento y la clasificación de objetos son cruciales para muchas de las investigaciones recientes. El entendimiento de estos procesos cognitivos requiere de que se comprenda cómo los humanos llevan a cabo el proceso de similitud.

En las siguientes secciones se desarrolla brevemente un panorama de las teorías y aproximaciones que se han propuesto a la modelación de la percepción visual de la forma y similitud. Esto con la finalidad de dar sustento a la medida de similitud entre dos curvas que se propone en la presente tesis.

1.1 PRINCIPALES TEORÍAS DE LA PERCEPCIÓN VISUAL DE LA FORMA

El tema de la forma y la materia es tan viejo como la misma filosofía, pero el uso filosófico del término "forma" es considerablemente más amplio que el que se le da en el lenguaje común. Por ejemplo, Aristóteles la definió como: *"Toda la realidad esta compuesta de materia y forma. La materia es aquello de lo que algo esta hecho; la forma es aquello por lo cual algo es lo que es. La forma es el principio activo que actualiza la potencialidad de la materia, principio pasivo. De ahí que la relación materia-forma se corresponde con la de potencia-acto"*. Por otro lado, Sócrates dio una definición de forma en términos empleados en problemas geométricos (en esta definición la palabra "Figura"

es usada en el sentido de "forma"): "*Figura es el límite de un sólido*". Este tema fue discutido por los filósofos y teólogos a lo largo de la edad media (por ejemplo, Thomas Aquinas), y continuó en escritos de Hobbes, Bacon, Descartes, Spinoza, Locke y otros.

Sin embargo, entre los investigadores de la forma visual, aún no existe un acuerdo en lo que significa "forma". En cada caso, uno necesita saber qué operaciones en específico se efectuaron para llevar al investigador a usar el término "forma", antes de que pueda ser decidido cuándo una forma es: (a) la cualidad corporal de un objeto en un espacio tridimensional, (b) la proyección de tal objeto sobre una superficie bidimensional, (c) una representación pictórica plana (en dos dimensiones), (d) una distribución no representativa de contornos en un plano, o (e) los valores de coordenadas en un espacio euclidiano.

Varias escuelas de psicología se han esforzado en comprender y describir a los mecanismos de comportamiento, en general, y a los aspectos específicos de la percepción visual. En las siguientes subsecciones se presentan las principales teorías que proponen dichas escuelas con respecto a la percepción visual de la forma.

1.1.1 La escuela Gestalt

La escuela Gestalt de psicología ha jugado un papel revolucionario con su nueva aproximación a la forma visual. Se puede encontrar una exposición detallada de este tema en [Helson, 1926], [Koehler, 1929] y [Koffka, 1935]. La teoría de Gestalt es una teoría no computacional de la forma visual, lo cual es una desventaja para aplicaciones prácticas del cómputo.

La Figura 1.1 resume las principales categorías de agrupamiento perceptual desarrolladas por la escuela Gestalt, las cuales se explican a continuación:

- **Proximidad.** Cuando los objetos están cercanos unos de otros éstos tienden a agruparse. Por ejemplo, los arreglos que se muestran en la Figura 1.1a y 1.1b contienen el mismo número de elementos, sin embargo, el primer arreglo tiene un espaciamiento horizontal mas pequeño que hace que se perciba como renglones de puntos negros mientras que en el segundo existe un espaciamiento vertical más pequeño dando la percepción de columnas de puntos.
- **Similitud.** Los objetos que resultan ser similares en tamaño, orientación y color tienden a ser agrupados juntos. En la Figura 1.1c el arreglo de puntos será agrupado en cuatro regiones, dos blancas y dos negras.
- **Cerradura.** Una colección de objetos o curvas tienden a percibirse como formas cerradas en vez de formas abiertas. Por ejemplo, en la Figura 1.1d se percibirán cuadrados incompletos, en vez de corchetes debido a la propiedad de cerradura.

- **Continuidad.** La Figura 1.1e se percibe como dos líneas cruzadas en vez de dos formas que no se cruzan. Esto indica que la percepción de formas continuas se prefiere a cambios abruptos.
- **Simetría.** El significado perceptual está unido a las formas que están reflejadas a lo largo de un eje, como en la Figura 1.1f.
- **Separación de fondo.** En general, el área más grande se percibirá como el fondo mientras que la más pequeña será percibida como el objeto o Figura en donde el área más grande rodea a la pequeña. En la Figura 1.1g, la imagen de la derecha muestra un diamante negro sobre un fondo cuadrado blanco, pero la imagen de la izquierda tiene áreas iguales por lo cual puede ser interpretada de otras maneras.

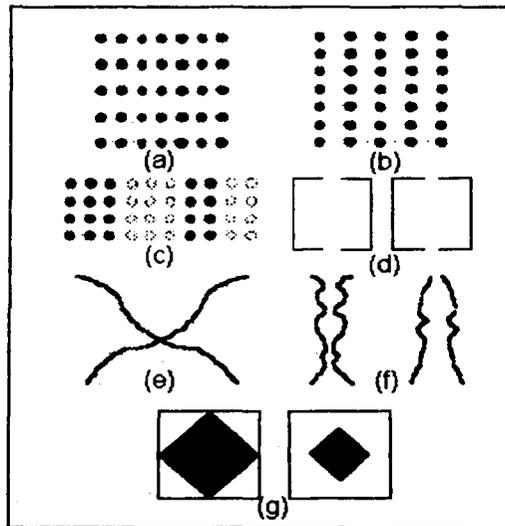


Figura 1.1 Categorías de agrupamiento perceptual: (a) y (b) proximidad; (c) similitud; (d) cerradura; (e) continuación; (f) simetría; (g) separación de fondo

Dos psicólogos destacados de la escuela de Gestalt son Hake y Zusne. Hake, discutió varias aproximaciones a la representación de formas naturales. Zusne presentó una

visión general de las teorías contemporáneas de la forma visual. Además, propuso varias leyes para las formas visuales, entre las cuales destacan:

- La forma visual es la propiedad más importante de una configuración.
- Las formas visuales se pueden trasponer (con respecto a la translación, tamaño, orientación y color), sin pérdida de su identidad.
- Las formas pueden fusionarse con otras para producir nuevas formas.
- Un cambio en alguna parte de la forma afecta a otras partes de la forma (ley de compensación).
- Las formas visuales tienden a aparecer y a desaparecer como un todo.
- Las formas visuales dejan un efecto secundario que las hace fáciles de recordar (ley de reproducción).
- Las formas visuales resisten el cambio. Éstas tienden a mantener su estructura en contra de fuerzas disturbantes.
- Todas las formas visuales poseen al menos dos aspectos, una porción de figura y una de fondo.

1.1.2 Teoría de Hebb

Otra aproximación a la teoría de formas visuales se encuentra en el trabajo de [Hebb, 1949]. Hebb presentó una teoría neuro-psicológica del comportamiento en su libro "La organización del conocimiento". En su teoría, Hebb enfatizó el papel que juegan las estructuras neuronales en el mecanismo de percepción visual. Su trabajo influyó a un gran número de investigaciones en el campo de las redes neuronales artificiales. A diferencia de la escuela Gestalt, Hebb argumenta que la forma se percibe como un todo, sino que consiste de partes. La organización y relación espacial mutua de las partes debe ser aprendida para que se pueda llevar a cabo un reconocimiento exitoso. Este aspecto del aprendizaje de la percepción es el punto central en la teoría de Hebb. El movimiento de los ojos es el mecanismo principal que integra los elementos más simples de la percepción. Los elementos más simples son ángulos y líneas. La teoría de Hebb es en su mayor parte cualitativa y no computacional, lo cual nuevamente es una desventaja para aplicaciones prácticas del cómputo.

1.1.3 Teoría de Gibson

[Gibson, 1950] desarrolló otra teoría importante acerca de la percepción visual basada en dos principios. El primer principio de esta teoría consiste en que el espacio no es una entidad geométrica ni abstracta, sino una entidad visual real caracterizada por las formas que se encuentran en ella. La teoría de Gibson esta centrada en la percepción de objetos reales tridimensionales, y no en sus proyecciones bidimensionales. El segundo principio es que existe un estímulo del mundo real detrás de cualquier percepción visual simple o compleja. Este estímulo está en la forma de un gradiente, el cual es una propiedad de una superficie. Ejemplos de gradientes físicos son los cambios en el tamaño de los elementos de textura, en el grado de convergencia de bordes paralelos (perspectiva), intensidad y saturación de los colores, y la iluminación y sombreado. Gibson menciona que la escuela de Gestalt ha estado ocupada con el estudio de proyecciones bidimensionales del mundo tridimensional y que su dinamismo no es mas que la ambigüedad de la interpretación de las imágenes proyectadas. En su clasificación existen diez diferentes tipos de formas:

- Forma sólida (ver un objeto significa ver una forma sólida).
- Forma de superficie (formas con bordes).
- Forma de borde (un dibujo de las orillas de una forma sólida).
- Forma pictórica (representaciones que son dibujos, fotografías, pinturas, etc.).
- Formas planas (un dibujo de las orillas de la proyección de una superficie sobre una superficie plana).
- Forma perspectiva (un dibujo en perspectiva de una forma).
- Forma sin sentido (dibujos que no representan objetos reales).
- Forma geométrica plana (una forma geométrica abstracta no derivada de un intento de hacer una forma sólida visible).
- Forma geométrica sólida (una parte abstracta de un espacio tridimensional limitado con superficies).
- Formas proyectadas. (Una forma geométrica plana la cual es una proyección de una forma).

Estas formas se agrupan en las siguientes tres clases:

- Objetos reales: formas sólidas y de superficies.

- Representaciones de objetos reales: formas delineadas, pictóricas, planas y de perspectiva.
- Objetos abstractos (no reales): Formas geométricas planas, formas geométricas sólidas y formas proyectadas.

La primer clase es la clase "real" que consiste de objetos del mundo real. La segunda clase son representaciones de objetos reales. La tercera clase son abstracciones que pueden ser representadas usando símbolos pero que no corresponden a objetos reales (debido a que no tienen representaciones correspondientes en el mundo real).

1.1.4 Teorías modernas de percepción visual de la forma

[Marr y Hildreth, 1980] realizaron contribuciones importantes al estudio del sistema de percepción visual humano. En el paradigma de Marr, el punto central de su investigación va desde aplicaciones hasta temas correspondientes a módulos del sistema visual humano. Sus contribuciones representan una parte importante de toda la investigación en visión por computadora. Artículos que tratan de la forma a partir de cierta técnica incluyen: forma a partir de sombreado [Horn, 1990], forma a partir de contornos [Horn y Brady, 1988] y [Ulfupinar y Nevatia, 1990], forma a partir de textura [Gibson, 1950], forma a partir de estereo [Hoff y Ahuja, 1989] y forma a partir de geometría fractal [Chen et al, 1990].

Koenderink, Koenderink y van Doorn han estudiado los aspectos psicológicos de la percepción visual y propusieron varios paradigmas interesantes. Las aproximaciones convencionales de la forma son normalmente estáticas en el sentido de que tratan a todos los detalles de la forma de igual manera como si fueran características globales. Un modelo dinámico de forma fue desarrollado en donde la percepción visual se ejecuta a diferentes escalas de resolución. Tales nociones de orden y de relación, están presentes en la psicología visual y están ausentes en las teorías geométricas convencionales de forma. En el arte así como en percepción, una forma es vista como una estructura jerárquica. Se desarrolló un procedimiento para morfogénesis basado en múltiples niveles de resolución.

Varios autores están de acuerdo en la importancia que tienen los puntos de gran curvatura en la percepción visual. Attneave y Arnoult [Attneave, 1954], [Attneave y Arnoult, 1966] realizaron experimentos psicológicos para investigar el significado de las esquinas en la percepción. En el famoso experimento del gato de Attneave (Figura 1.2), se usó el dibujo de un gato para localizar puntos de gran curvatura, los cuales enseguida fueron conectados para crear un dibujo simplificado del gato. Después de una breve presentación, el gato pudo reconocerse en el dibujo. Esto ha sugerido que tales puntos contienen información importante de la forma. El trabajo de Attneave ha conducido a la investigación en el tema de partición de curvas [Rosenfeld y Johnston, 1973], [Pavlidis y

Horowitz, 1974] y [Fischler y Bolles, 1986], por ejemplo. Para aproximar las curvas con líneas rectas, los puntos de gran curvatura son las mejores posiciones para dividir las líneas, obteniendo una imagen que retiene la cantidad máxima de información necesaria para el reconocimiento exitoso de la forma. Para el propósito de descripción de formas, las esquinas se usan como puntos de gran curvatura y la forma puede ser aproximada con un polígono. Se han desarrollado una gran variedad de algoritmos para aproximaciones poligonales de la forma [Pavlidis, 1978]. [Davis, 1977] combinó el uso de puntos de gran curvatura y las aproximaciones de segmentos de líneas para realizar aproximaciones poligonales de las formas. [Stokely y Wu, 1992] investigaron métodos para medir la curvatura de superficies en tres dimensiones que evolucionaron en varias aplicaciones (por ejemplo, en imágenes médicas de tomografía).

[Hoffman y Richards, 1984] argumentan que cuando el sistema visual descompone los objetos, éste lo hace en los puntos de gran curvatura.

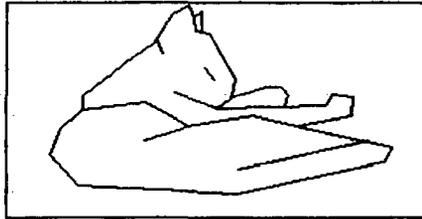


Figura 1.2 Gato de Attneave. Dibujo en donde las líneas de conexión de poca curvatura han sido reemplazadas por líneas rectas, preservando solo los puntos de gran curvatura.

[Leyton, 1987] desarrolló una teoría que dice que todas las formas son básicamente círculos que cambiaron su forma como resultado de varias deformaciones causadas por fuerzas externas como empujar, jalar, estirar, etc.

[Pentland, 1986] investigó métodos para la representación de formas naturales por medio de geometría fractal. Argumenta que las funciones fractales son más apropiadas para la representación de formas naturales debido a que muchos de los procesos físicos producen formas de superficies fractales.

[Lowe, 1987] propuso un sistema de visión por computadora que puede reconocer objetos tridimensionales desde puntos de vista no conocidos y a partir de una sola imagen bidimensional. El procedimiento usa tres mecanismos de agrupamiento perceptual para determinar conocimiento tridimensional acerca del objeto. Las condiciones que deben de satisfacerse por las operaciones de los grupos perceptuales son:

- La condición de invarianza del punto de vista. Las características primitivas observadas deben permanecer estables a lo largo de un rango de puntos de vista.
- La condición de detección. Debe existir suficiente información disponible para evitar mal interpretaciones accidentales.

1.2 SIMILITUD

El proceso de similitud ha sido estudiado por distinguidos pensadores que van desde Aristóteles hasta psicólogos modernos como [Shepard, 1962], [Tversky, 1977], y [Nosofsky, 1992].

En esta sección el término "objeto" significará cualquier estímulo, incluyendo objetos reales como una piedra o un árbol, pero también puede ser un punto moviéndose o una curva.

Intuitivamente:

- 1) La similitud entre A y B está relacionada con aquello que tienen en común. Entre más tengan en común, más se parecen. De la misma forma, entre más diferencias existan, serán más distintos.
- 2) La máxima similitud entre A y B se alcanza cuando A y B son idénticos, sin importar cuanto tengan en común.

Además, cuando A y B no tienen algo en común, su similitud es la mínima sin importar qué tan distintos son. Por ejemplo, la similitud entre una búsqueda binaria y un sillón no es menor ni mayor que la similitud entre un planeta y una televisión.

Desde el punto de vista de la psicología, la similitud parece ser una primitiva psicológica irreducible, como por ejemplo el color "rojo", sin embargo varios teóricos han tratado de mostrar como ésta se relaciona con otras consideraciones fundamentales. Asumamos, como la mayoría de los autores lo hacen, que los estímulos de los objetos están representados internamente, y que la similitud entre los objetos proviene de algún tipo de comparación entre sus representaciones. Entonces surgen varias preguntas, por ejemplo: (1) ¿Qué información es guardada en las representaciones de los estímulos? (2) ¿Cómo se combina o se estructura esta información dentro de la representación? (3) ¿Cómo son comparadas las representaciones para obtener la similitud? y (4) Dado un conjunto de estímulos, ¿cómo son sus similitudes determinadas y representadas de la mejor manera?

Existen ocho principales aproximaciones teóricas a la similitud:

- 1.- Aproximación de elementos comunes

- 2.- Modelos de plantillas
- 3.- Aproximación geométrica
- 4.- Modelos por características
- 5.- Medidas de similitud basadas en funciones de distancia
- 6.- Basados en alineaciones
- 7.- Transformacionales
- 8.- Teoría Gqon

En las siguientes subsecciones se describen brevemente cada una de las aproximaciones.

1.2.1 Aproximación de elementos comunes

Uno de los intentos más conocidos para sugerir los procesos que hay de fondo a la similitud se deriva de que los estímulos se representen como colecciones de elementos. Si se desea, la similitud puede ser calculada contando el número de elementos en común relativos a otros elementos y/o sumando sus valores.

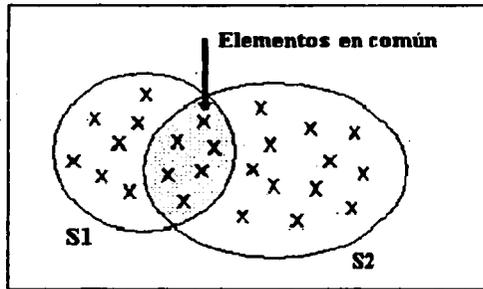


Figura 1.3 Aproximación de elementos comunes

En la Figura 1.3 se muestra esta aproximación, en donde los elementos de dos estímulos se representan con una "X" y de esta forma la proporción de los elementos en común del estímulo (sombreados en gris) determina su similitud.

Este esquema tan simple no representa adecuadamente las similitudes entre estí-

mulos complejos en la mayoría de los casos. Las siguientes aproximaciones son representaciones de estímulos que contienen más información.

1.2.2 Modelos de plantillas

Los modelos de plantillas fueron desarrollados como una respuesta al problema de reconocimiento de objetos, y éstos incorporan, al menos implícitamente, la idea de comparación con similitud. Las representaciones que se asumen en los modelos de plantillas contienen mucha más información detallada acerca de la estructura del estímulo de lo que la tiene la representación de elementos antes descrita. Estos modelos son normalmente aplicados a objetos visuales extendidos espacialmente, y su representación puede pensarse en términos de su organización espacial.

La similitud está basada en el grado de correspondencia en una comparación espacial punto a punto entre las representaciones a ser comparadas, como sugiere la Figura 1.4.

En este ejemplo, la oportunidad de que sean reconocidas como Figuras similares se puede mejorar utilizando técnicas de filtrado de imágenes, en donde la representación marcada con líneas más fuertes podría ser la entrada mientras que la representación borrosa podría ser la representación obtenida de la memoria.

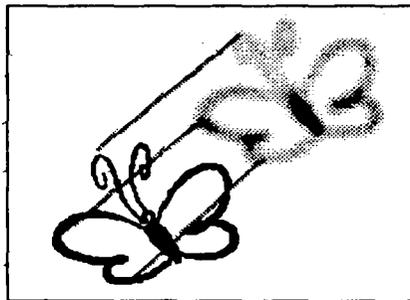


Figura 1.4 Aproximación con Plantillas. Usan una correspondencia punto a punto (ver las líneas) entre las representaciones.

Tales modelos han sido comúnmente descartados porque parece que son incapaces de detectar similitudes entre formas que se encuentran desplazadas, rotadas o con distinto tamaño. De cualquier forma, tales restricciones han sido superadas cuando se aplican operaciones de pre-procesamiento que puedan transformar a las formas en unas equivalentes que puedan ser comparadas en orientación y tamaño.

1.2.3 Aproximación geométrica.

La aproximación con modelos geométricos hace énfasis en la representación de las relaciones de similitud entre los miembros de un conjunto de objetos. El estímulo de un objeto es representado simplemente por sus coordenadas en un "espacio de similitud". La similitud es dada por una distancia entre los objetos en este espacio; entre más cercanos se encuentren dos objetos, más similares son. La aproximación asume (1) que los objetos pueden ser representados por valores en pocas dimensiones continuas y (2) que la similitud puede ser representada por la distancia en un espacio coordinado. La Ilustración I.1 muestra un ejemplo de estímulos de objetos colocados en un espacio definido por las dimensiones de tamaño y color. En este caso, se ordena al conjunto de pájaros en un espacio de similitud de dos dimensiones. Los pájaros pequeños van cerca de la parte superior mientras que los más grandes van cerca la parte inferior. Los pájaros rojos van hacia la izquierda y los amarillos hacia la derecha. En el caso más simple, se traza una línea recta entre los dos pájaros a comparar y su longitud determinará su similitud.

Los modelos geométricos han sido de las aproximaciones más importantes para analizar la similitud, y se ejemplifican por modelos de escala multidimensionales (MDS) [Nosofsky, 1992] y [Shepard, 1962]. La entrada a las rutinas MDS pueden ser juicios de similitud, matrices de confusión (una tabla de qué tan seguido cada entidad es confundida con cada una de las otras entidades), las probabilidades de que las entidades sean agrupadas juntas, o cualquier otra medida de similitud subjetiva entre todos los pares de entidades en un conjunto. La salida de una rutina MDS es un modelo geométrico de similitud de las entidades, en donde cada entidad del conjunto es representada como un punto en un espacio N-dimensional. La similitud entre dos entidades i y j , está inversamente relacionada a su distancia $D(i,j)$, la cual puede ser calculada por la distancia L_p . La distancia L_p se explica en el capítulo 2.

Una métrica Euclidiana (L_2) comúnmente provee de un buen modelo para los juicios de similitud de los humanos cuando las entidades son bien percibidas, mientras que una métrica "City-block" (L_1) normalmente provee un mejor modelo cuando las entidades son claramente divisibles en dimensiones separadas [Garner, 1974].

Los modelos geométricos normalmente asumen la minimalidad [$D(A,B) \geq D(A,A) = 0$], la simetría [$D(A,B) = D(B,A)$] y la desigualdad del triángulo [$D(A,B) + D(B,C) \geq D(A,C)$]. [Tversky, 1977] criticó los modelos geométricos argumentando que éstos violan las tres suposiciones que fueron observadas empíricamente. La minimalidad puede ser violada porque no todos los objetos idénticos parecen igualmente similares, los objetos complejos que son idénticos (por ejemplo los gemelos) pueden ser mas similares entre sí que objetos idénticos simples (por ejemplo dos cuadrados). La similitud asimétrica ocurre cuando un objeto con varias características es juzgado como menos similar a un objeto con escasas características que viceversa; por ejemplo, Corea del Norte es juzgada a ser más parecida a China que China a Corea del Norte [Tversky, 1977]. La desigualdad del triángulo puede ser violada cuando A (por ejemplo una lámpara) y B (por ejemplo la luna) comparten una característica idéntica (ambas proveen luz),

y B(luna) y C(pelota) comparten una característica idéntica, pero A y C no comparten ninguna característica [Tversky y Gati, 1982]. Los modelos geométricos pueden ser modificados para corregir esas suposiciones [Nosofsky, 1991].

1.2.4 Modelos por características.

A pesar de que la aproximación geométrica tiene ventajas prácticas, sus suposiciones limitan su aplicabilidad. Primero, si la disimilitud va a ser representada como una distancia métrica, ésta debe cumplir los tres axiomas de métricas, pero se han encontrado varios ejemplos a través de situaciones experimentales con humanos en los que se contradicen estos tres axiomas. Segundo, pocos estímulos difieren entre sí en sólo algunas dimensiones continuas tales como tamaño y color. La mayoría de los estímulos parecen ser más efectivamente descritos por la presencia o ausencia de características cualitativas.

[Tversky, 1977] propuso un modelo de similitud basado en características, en el cual las características comunes tienden a incrementar la similitud percibida en los dos conceptos, y en donde las características diferentes tienden a decrementar la similitud percibida. Por ejemplo, un jitomate y una cereza son similares en virtud de que tienen ciertas características en común: redonda y roja. Por otro lado, son diferentes porque tienen algunas características distintas: tamaño (grande contra pequeño). Ésta es una afirmación que no sorprende mucho y que además es intuitiva; además, el modelo de Tversky afirma que las características que tienen en común tienden a incrementar la similitud percibida más de lo que las características distintas la decrementan. Esto es, cuando tratamos de establecer una similitud le damos más importancia a aquellas características que tienen en común que a aquellas que los distinguen.

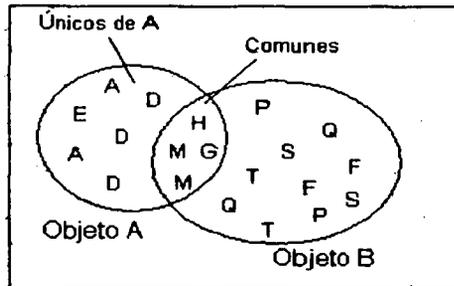


Figura 1.5 Representación de dos objetos en donde cada uno contiene sus propias y únicas características y que también comparten ciertas características entre sí.

Tversky y otros han asumido que un objeto es representado por un conjunto de características o atributos. Normalmente éstas son variables binarias (por ejemplo consonante con voz o sin voz) o partes que están presentes o no (por ejemplo ojos, cola, barra horizontal), pero también podrían ser conjuntos ordenados de propiedades como color o tamaño. Esto se ejemplifica en la Figura 1.5.

En este modelo la similitud es determinada por la correspondencia de características de las entidades en comparación, e integrando estas características en la fórmula:

$$S(A, B) = a f(A \text{ y } B) - b f(A - B) - c f(B - A) \quad (1.1)$$

La similitud entre A y B , $S(A, B)$, se expresa como una combinación lineal de la medida de lo que tienen en común y las características en las que difieren. El término $(A \text{ y } B)$ representa las características que A y B tienen en común. $(A - B)$ representa las características que A tiene pero que B no tiene. $(B - A)$ representa las características que tiene B pero no A .

Los términos a , b y c reflejan los pesos dados a los componentes que tienen en común y a los que difieren, y la función f normalmente se asume que es aditiva. Otros modelos de características calculan la similitud tomando la proporción de lo que tienen en común con lo que tienen de distinto [Sjoberg, 1972].

1.2.5 Medidas de similitud basadas en funciones de distancia

Una de las aproximaciones que ha sido ampliamente usada, es la basada en medidas de similitud. Esta aproximación consiste en dar una función que asocie un valor numérico con una pareja de objetos, con la idea de que un valor grande indica una gran disimilitud entre los objetos dados y un valor pequeño indica que son muy parecidos.

Es deseable que dicha función cumpla con ciertas propiedades, entre las más importantes son las propiedades métricas: positividad, identidad, unicidad y desigualdad del triángulo. Otras propiedades incluyen la invarianza a un conjunto dado de transformaciones y propiedades de continuidad, entre otras.

La aproximación que se propone en esta tesis para comparar dos curvas tridimensionales, en una de sus partes esta basada en una medida de disimilitud, por lo cual se describirá mas ampliamente esta aproximación en el capítulo 2. Por lo pronto, se seguirá dando el panorama del resto de las aproximaciones a la similitud que se han desarrollado.

1.2.6 Basados en alineaciones

[Markman y Gentner, 1993] propusieron un concepto de similitud interesante. A diferencia de la propuesta de Tversky, la cual trata simplemente de características comunes y características diferentes, el modelo basado en estructura de Markman y Gentner asume una base tripartita para los juicios de similitud: características en común alineables, diferencias alineables y diferencias no alineables. Desde este punto de vista, la similitud entre estructuras conceptuales es percibida relativa a una alineación coherente, o a un mapeo estructural, entre aquellas representaciones conceptuales. Para que una característica sea considerada como común a ambos conceptos, ésta no sólo tiene que existir en ambas representaciones, sino que debe ocupar la posición relativa en cada uno (es decir, las características deben ser alineables bajo un mapeo de estructura o una operación de isomorfismo-gráfico). Las características que comparten que no ocupan la misma posición relativa son consideradas no alineables.

Por ejemplo, supongamos que se compara la universidad X con otra universidad Y. Ambas universidades cuentan con un departamento de Ciencias de la Computación cuyos edificios principales son excéntricos y cuentan con nuevas supercomputadoras. Al intentar obtener la similitud entre X y Y, la alineación de departamentos de Ciencias de la Computación, edificios principales excéntricos y supercomputadoras nos dan tres aspectos de comunidad estructurada. Pero si sabemos que X tiene un programa de posgrado en desarrollo mientras que Y tiene un posgrado ya establecido, entonces se obtiene una diferencia alineable. Mas aún, si se sabe que X cuenta con un departamento de Arte pero Y no, entonces este cuenta como una diferencia no alineable en la comparación.

Markman y Gentner afirman, con soporte empírico, que las diferencias alineables juegan un mayor papel en los juicios de similitud que las diferencias no alineables.

1.2.7 Transformacionales

Esta aproximación al modelado de la similitud esta basada en una distancia de transformación. La similitud entre dos entidades se asume ser inversamente proporcional al número de operaciones requeridas para transformar una entidad para que esta sea idéntica a la otra. Por ejemplo, "XXXXO" requiere solo una transformación para convertirse en "XXXOO" (cambiar una O por una X), pero se requiere de dos transformaciones para convertirse en "OOXXX" (cambiar una O por una X, y hacer el reverso de la cadena. Por lo tanto "XXXXO" es mas similar a "XXXOO". Un ejemplo de éste tipo de aproximación a la similitud lo presentan [Sánchez-Cruz y Bribiesca], en el cuál miden la similitud entre dos objetos tridimensionales calculando el esfuerzo de transformar uno en otro..

1.2.8 Teoría Geon

Al igual que la aproximación con plantillas, la teoría Geon de Biederman, por ejemplo [Biederman, 1987] está relacionada principalmente con el reconocimiento de objetos y está centrada en la representación de la forma visual. De acuerdo a la teoría Geon, los estímulos de los objetos son representados por formas primitivas o partes elementales, como cilindros o conos, que se encuentran bajo una cierta relación entre sí. De acuerdo con la teoría, la generalización entre dos objetos ocurrirá si las mismas partes y relaciones son visibles en ambos, incluso si los detalles de varias partes cambian considerablemente. Por ejemplo, si se rota un objeto pero esto no causa que se oculten algunas de sus partes o relaciones, el objeto es aún reconocible y presumiblemente, las imágenes rotadas y no rotadas son similares. Para una extensa discusión de la teoría Geon, ver [Kirkpatrick, 2001].

1.2.9 Comentarios finales

Estas aproximaciones han tenido cierto éxito en predecir cuantitativamente el juicio de las personas acerca de la similitud entre dos entidades, aunque el probar estos modelos para comparar entidades es aún un tema de investigación.

Otra cuestión importante es referente al papel que juega la similitud en otros procesos cognitivos. Por ejemplo, mientras que varios modelos de categorización están completamente basados en la similitud, otros investigadores han argumentado que también depende de conocimiento abstracto y teórico. Otra complicación para la explicación del papel de la similitud, es que ésta podría no ser un fenómeno unitario. Los juicios de similitud están influenciados por el contexto, la perspectiva, las alternativas de elección y la experiencia. Es probable que se usen diferentes procesos para realizar un juicio de similitud para diferentes tareas, dominios y estímulos. La elección de características, transformaciones y descripciones estructurales usadas para describir entidades guiarán las predicciones hechas por modelos similares tanto como lo hacen los mecanismos de modelos para comparar e integrar estas representaciones.

Finalmente, la pregunta: "¿Qué hace que las cosas se parezcan o se vean diferentes?", es aún una de las preguntas fundamentales de la psicología y del reconocimiento de patrones.

En el siguiente capítulo se presenta un panorama general de las diferentes propuestas que se han dado al problema de la similitud de patrones geométricos, desde un punto de vista computacional.

**FALTA
PAGINA**

16

CAPITULO 2

CORRESPONDENCIA DE PATRONES GEOMÉTRICOS

De acuerdo a [Hagedoorn, 2000] existen dos tipos de correspondencia de patrones:

- i) Combinatorial
- ii) Espacial

En el caso de correspondencia de patrones combinatorial tenemos los siguientes ejemplos: correspondencia de cadenas, correspondencia de patrones de DNA y correspondencia de patrones de árboles. Por otro lado, la correspondencia espacial de patrones es el problema de encontrar la correspondencia entre dos imágenes de intensidad de color o de modelos geométricos. La correspondencia podría ser una similitud o una transformación geométrica.

En este capítulo solo se considerará la correspondencia de patrones espacial, ya que son el tipo de patrones con los que se trabajará en esta tesis.

2.1 CORRESPONDENCIA DE PATRONES ESPACIAL

La correspondencia de patrones espacial trata de transformar un patrón, y medir el parecido con otro, usando por ejemplo alguna medida de similitud. De tal forma que las medidas de similitud son una parte esencial en la correspondencia de patrones espaciales. A pesar de que se usa comúnmente el término similitud, la disimilitud corresponde a la noción de distancia: pequeñas distancias significan poca disimilitud y gran similitud.

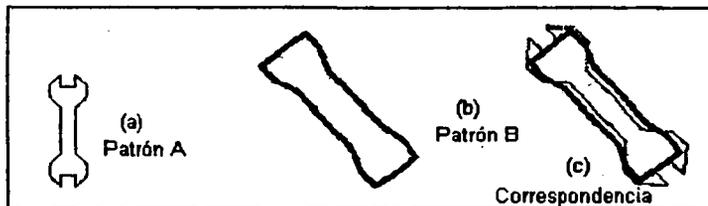


Figura 2.1

Un ejemplo simple de correspondencia de patrones espacial es el siguiente. Dados dos patrones de entrada A y B, que se muestran en la Figura 2.1(a) y 2.1(b) respectivamente, el problema es encontrar una transformación g para la cual $g(A)$ sea similar a B. En este caso, g puede ser una combinación de transformaciones de escala, rotación y translación. La salida esperada de un buen algoritmo de correspondencia de patrones espacial, podría ser una transformación g como la que se presenta en la Figura 2.1(c) en donde $g(A)$ esta superpuesta sobre B.

La correspondencia de patrones puede ser estudiada de varias formas. Dados dos patrones y una medida de disimilitud, podemos:

- Problema de cálculo: calcular la disimilitud entre los dos patrones.
- Problema de decisión: para un umbral dado, decidir cuándo la disimilitud es más pequeña que el umbral.
- Problema de decisión: para un umbral dado, decidir cuándo existe una transformación tal que la disimilitud entre el patrón transformado y el otro patrón es más pequeña que el umbral.
- Problema de optimización: encontrar la transformación que minimice la disimilitud entre el patrón transformado y el otro patrón.

Algunas veces las complejidades en tiempo para resolver estos problemas son muy grandes, de tal forma que tiene sentido utilizar algoritmos de aproximación:

- Problema de optimización aproximada: encontrar una transformación que de una disimilitud entre los dos patrones que se encuentre entre un factor multiplicativo constante de la disimilitud mínima.

En el caso general, la entrada de los algoritmos de correspondencia de patrones espacial es una pareja de patrones, y la salida es una transformación. Los problemas de correspondencia de patrones pueden ser clasificados por tres componentes:

- 1.- La colección de patrones.
- 2.- La clase de transformaciones.
- 3.- El criterio usado para seleccionar una transformación.

Algunos ejemplos de colecciones de patrones son:

- Imágenes a color,
- Imágenes en escala de grises,
- Modelos CAD y
- Gráficas vectorizadas.

Algunos ejemplos de clases de transformaciones son:

- Translaciones,
- Isometrías Euclidianas (las cuales preservan la distancia euclidiana entre cada pareja de puntos),
- Transformaciones afines (las cuales son composiciones de transformaciones lineales y translaciones),
- Transformaciones elásticas (las cuales no preservan necesariamente las líneas rectas) y
- Las correspondencias (las cuales son relaciones entre características particulares de ambos patrones).

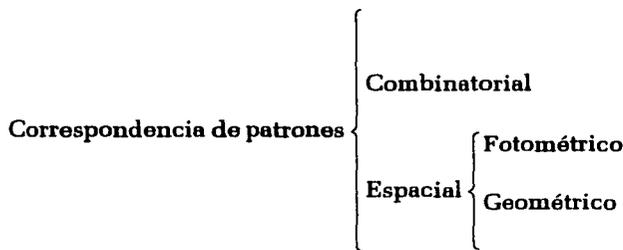
Un ejemplo de un criterio de selección es que una transformación deba minimizar el valor de una medida de similitud (la cual es una función que asigna un valor real no negativo a cada pareja de patrones). El criterio de selección no se hace explícito siempre en la descripción del método.

2.1.1 Correspondencia de patrones fotométrico y geométrico

La correspondencia de patrones espacial puede ser subdividida en dos categorías, dependiendo del tipo de patrones de entrada:

- i) Fotométrico
- ii) Geométrico

De tal forma que la correspondencia de patrones se clasifica como se muestra en el siguiente esquema.



Los métodos fotométricos trabajan directamente sobre imágenes que son consideradas como arreglos de valores de intensidad o funciones evaluadas en los reales. En otras palabras, se refiere a la información de intensidad medida por cámaras, digitalizadores, MRI, CT, etc.

Los métodos geométricos trabajan sobre datos geométricos tales como conjuntos finitos de puntos o de polígonos. Estos datos geométricos pueden ser obtenidos directamente de una representación de objetos basada en vectores. También pueden ser obtenidos a partir de imágenes usando técnicas de extracción de características. En este contexto, la correspondencia de patrones geométricos podría ser llamada correspondencia de patrones basada en características.

Algunas veces los métodos fotométricos y los métodos geométricos son usados en combinación. Por ejemplo cuando las características son usadas para generar candidatos de correspondencia y la fotometría es usada para verificar las correspondencia de los candidatos.

2.1.2 Correspondencia total y correspondencia parcial

Existe una diferencia entre la correspondencia total y la correspondencia parcial. En la correspondencia total, se realiza la correspondencia de un patrón completo con otro patrón completo. Mientras que en la correspondencia parcial, se realiza la correspondencia de un patrón (o de parte de este) con una parte del otro patrón.

La correspondencia parcial puede ser vista como el análogo geométrico de la correspondencia de subcadenas. Se requiere tener cuidado con la formulación de los problemas de correspondencia parcial. Por ejemplo, si se define la correspondencia parcial como la transformación de un patrón de tal forma que éste se convierta en un subconjunto de otro patrón, podría llevar a resultados indeseables si se implementa, especialmente si las transformaciones que se consideran incluyen a la escala.

Existen varias formas de definir la correspondencia parcial en términos de la correspondencia total. Un ejemplo es identificar los subpatrones de un patrón que corresponden a distintos objetos; cada uno de estos subpatrones da lugar a un problema de correspondencia total.

2.1.3 Correspondencia exacta y aproximada

La correspondencia de patrones exacta normalmente significa encontrar una transformación bajo la cual un patrón se convierta en una copia idéntica del otro. La correspondencia aproximada significa encontrar la transformación bajo la cual un patrón se vuelva similar (pero no necesariamente idéntico) a otro patrón. La confusión de estos dos términos radica en el hecho de que el adjetivo "aproxima" es también usado en relación con algoritmos de optimización; y muchos tipos de correspondencia de patrones pueden ser vistos como problemas de optimización. Un algoritmo de aproximación calcula una solución que no es necesariamente la óptima. En este contexto, la aproximación no tiene nada que ver con el tipo de correspondencia de patrones.

2.1.4 Problemas relacionados con la correspondencia de patrones geométricos.

El reconocimiento de objetos es muy similar a la correspondencia de patrones geométricos, aunque no hay consenso acerca del significado exacto de reconocimiento de objetos. Existen tres usos distintos para el término reconocimiento de objetos.

Primero, el término reconocimiento de objetos es usado para la **segmentación** de regiones. La segmentación de regiones es el problema de determinar las regiones en una imagen que corresponden a objetos físicos distintos.

Segundo, el término reconocimiento de objetos se usa para un tipo de correspondencia de patrones parcial. En este caso, el problema es **detectar las ocurrencias de un objeto dado en una imagen**.

Tercero, el término reconocimiento de objetos es usado para el **reconocimiento de formas**. El reconocimiento de formas juega un papel importante en las siguientes categorías:

- **Recuperación de formas.** Consiste en buscar en una base de datos de formas (que normalmente es considerablemente grande) todas las formas que son similares a una dada en la petición. Normalmente se encuentran todas las formas que cumplen con estar dentro de cierta distancia con respecto a la de la petición (problema de decisión), o en encontrar las primeras formas cuya distancia sea la mas pequeña (problema de calculo). Si la base de datos es grande, no sería adecuado calcular la similitud de la forma dada y todas las que se encuentran en la base de datos. En este caso se puede usar una estructura de indexado para excluir grandes partes de la base de datos en una etapa temprana de la búsqueda, en donde se usa normalmente alguna propiedad de tipo desigualdad del triángulo.

- **Reconocimiento de formas y clasificación.** Consiste en determinar cuándo una forma dada corresponde lo suficientemente cerca a un modelo (problema de decisión), o a cuál de los representantes de k clases es más similar.
- **Alineación y registro de formas.** Consiste en transformar una forma de tal manera que se obtenga su mejor correspondencia con otra forma (problema de optimización), por completo o en alguna parte.
- **Aproximación de formas y simplificación.** Consiste en construir una forma de menos elementos (puntos, segmentos, triángulos, etc.), de tal forma que ésta aún sea similar a la original. Existen muchas heurísticas para aproximar curvas poligonales y superficies poliédricas. Los métodos óptimos construyen una aproximación con los menores elementos dada una medida de disimilitud, o con la menor disimilitud dado el número máximo de elementos.

2.1.5 Obteniendo patrones geométricos

Los algoritmos de correspondencia de patrones geométricos trabajan con patrones geométricos. Un patrón geométrico es un subconjunto de algún espacio base (el cual es normalmente R^2 o R^3). En esta tesis, la palabra patrón será usada como una abreviación de patrón geométrico.

Cada algoritmo de correspondencia de patrones geométricos trabaja sobre una colección específica de patrones. Un ejemplo importante es la colección de todos los subconjuntos finitos en el plano. Otros ejemplos son la colección de todas las poli líneas en el plano y la colección de todos los polígonos en el plano. Para cada algoritmo considerado en este capítulo, la colección de patrones consiste de uniones finitas de primitivas geométricas. Ejemplos de primitivas geométricas en dos dimensiones son segmentos de líneas, segmentos de curvas y triángulos. Ejemplos de primitivas geométricas en otras dimensiones son puntos y parches de hipersuperficies.

Los patrones se obtienen de dos fuentes: imágenes y modelos geométricos. Las imágenes son comúnmente almacenadas como arreglos de valores de intensidad. Un modelo geométrico podría ser almacenado como una lista de primitivas geométricas, una representación de borde o una representación jerárquica como un CSG (Constructive Solid Geometry). Los algoritmos de correspondencia de patrones normalmente necesitan representaciones simples como entrada, por ejemplo, listas de primitivas geométricas. Para imágenes y modelos geométricos, se deben realizar varios pasos previos para obtener representaciones de patrones que sean adecuadas para ser entradas de un algoritmo de correspondencia de patrones geométricos.

Para obtener patrones a partir de modelos geométricos, podrían ser necesarias conversiones de representaciones. Si el modelo geométrico es almacenado como una lista de primitivas geométricas o con su representación de borde, la conversión tiende a ser

directa. Por otro lado, si el modelo geométrico es almacenado como una representación jerárquica, la conversión puede no ser trivial.

Para obtener patrones a partir de imágenes, es necesario realizar el proceso de extracción de características. Un paso preliminar a la extracción de características es la aplicación de técnicas de procesamiento de imágenes para mejorar la calidad de la imagen. Para algunos algoritmos de extracción de características, la salida es una imagen binaria (en blanco y negro). En estos casos, se requiere de un paso adicional de vectorización para obtener primitivas geométricas.

La extracción de características es una de las dos grandes partes del reconocimiento de patrones y es un campo de investigación por sí mismo. Este está relacionado con disciplinas como procesamiento de señales, procesamiento de imágenes, visión por computadora e inteligencia artificial. Ejemplos de extracción de características son: detección de esquinas, detección de bordes y detección de regiones.

La detección de esquinas es la tarea de encontrar esquinas de objetos físicos en objetos. Las esquinas son puntos sobre el contorno de un objeto en el cual la curvatura es alta.

La detección de bordes es el proceso de encontrar los bordes entre regiones en una imagen que corresponden a distintos objetos físicos. La aproximación más simple calcula las diferencias en intensidades entre los píxeles vecinos. Estas diferencias son después aplicadas a un proceso de umbral con lo cual se obtienen los bordes.

La detección de regiones es el problema de determinar las regiones en una imagen que corresponden a un solo objeto físico. Los métodos más primitivos de detección de regiones aplican un umbral sobre el color o intensidad. Las técnicas más avanzadas trabajan sobre diferentes niveles de detalle.

2.2 PRINCIPALES TÉCNICAS DE CORRESPONDENCIA DE PATRONES GEOMÉTRICOS

Las técnicas más usadas para realizar correspondencia de patrones geométricos son:

- Métodos globales,
- Correspondencia de gráficas,
- Hashing geométrico y
- Correspondencia de patrones basados en medidas de similitud.

En las siguientes subsecciones se presenta una breve descripción de cada técnica así como sus principales ventajas y desventajas.

2.2.1 Métodos globales

Los métodos globales son métodos que usan información geométrica global para encontrar las correspondencias. Un ejemplo es el mapeo de la caja envolvente de un patrón en la caja envolvente de otro patrón. Otro ejemplo es la translación del centroide de un patrón en el centroide de otro patrón. Las técnicas basadas en momentos [Chen, 1993] generalizan este último ejemplo. Además éstas técnicas pueden ser usadas para realizar correspondencias bajo isometrías euclidianas y transformaciones de similitud.

Su principal ventaja es:

- Son métodos muy rápidos debido a su simplicidad.

Sus principales desventajas son:

- No son muy adecuados para encontrar correspondencias admisibles entre dos patrones que son muy diferentes.
- Tampoco son muy adecuados para encontrar correspondencias parciales.

2.2.2 Correspondencia de gráficas

La correspondencia de gráficas es un método en donde la estructura de un patrón se describe como una gráfica. Una vez que las gráficas han sido construidas, la correspondencia se realiza entre las gráficas buscando isomorfismos de gráficas o isomorfismos de subgráficas.

Su principal ventaja es:

- Existen algoritmos eficientes para clases en particular de gráficas.

Sus principales desventajas son:

- Debido a su naturaleza discreta, la correspondencia de gráficas no es muy robusta a errores en el patrón.
- El éxito de la técnica depende de la generación correcta de las gráficas a partir de las entradas.
- Existe una falta de diferenciación: varias clases de patrones comparten la misma gráfica.

Un ejemplo de correspondencia de gráficas se da en [Gigus et al, 1991].

2.2.3 Hashing geométrico

El hashing geométrico, [Wolfson, 1990] y [Wolfson y Rigoutsos, 1997], es otro tipo de método de correspondencia. Esta técnica consiste en que, dado un patrón geométrico desconocido encontrar la mejor correspondencia con otros patrones geométricos conocidos que ya han sido definidos previamente en una base de datos de patrones.

En este método las primitivas geométricas que forman el patrón, son usadas para generar una descripción normalizada del patrón como un todo. Por ejemplo, para conjuntos finitos de puntos en un plano, se genera una descripción invariante a transformaciones afines cuando el conjunto de puntos se expresa en coordenadas del sistema. Todas las descripciones normalizadas son usadas para construir una tabla hash. Supongamos que se requieren detectar eficientemente las ocurrencias de un patrón fijo A como un subconjunto de varios patrones B . En este caso, se construye una estructura hash para A , y se insertan descripciones normalizadas de B en esa estructura hash.

Sus principales ventajas son:

- Es robusto cuando existen puntos faltantes.
- Es robusto cuando se presentan oclusiones parciales.
- Es eficiente en tiempo cuando se desea reconocer un conjunto fijo de patrones, después de que la estructura haya sido construida.
- Es inherentemente paralelizable.

Sus principales desventajas son:

- La estructura de datos puede crecer demasiado, dependiendo del grupo de transformaciones.
- Es una aproximación muy dependiente de la representación y topología de los patrones, lo cual es una desventaja si los patrones de entrada se obtienen por medio de un proceso de extracción de características.

2.2.4 Correspondencia de patrones basada en medidas de similitud

Como ya se había mencionado, una función de similitud es una función que asigna un número real no negativo a cada pareja de patrones. Un ejemplo típico de correspondencia de patrones basada en medidas de similitud es encontrar una transformación geométrica que minimice una medida de similitud. En este ejemplo, el problema preciso es, dados los patrones A y B , encontrar una transformación g (de una clase específica de transformaciones) que minimice el valor de la medida de similitud entre $g(A)$ y B .

Sus principales ventajas son:

- Este tipo de correspondencia puede ser usado para encontrar correspondencias admisibles aunque los patrones difieran mucho.
- Esta aproximación puede ser usada en correspondencias parciales.
- Siempre y cuando la medida de similitud usada esté definida sobre los patrones como conjuntos (en vez de sobre su representación), los métodos basados en medidas de similitud son independientes de la representación.

Su principal desventaja es:

- La eficiencia de estos métodos depende mucho de la elección de la medida de similitud. Aunque por lo general éstos métodos tienden a ser más lentos que los globales.

La correspondencia de patrones basada en medidas de similitud se discute con más detalle en la siguiente sección, en donde se presentan varios ejemplos de medidas de similitud.

2.3 CORRESPONDENCIA DE PATRONES BASADO EN MEDIDAS DE SIMILITUD

En esta sección se discuten varios métodos de correspondencia de patrones geométricos que están (implícitamente o explícitamente) basados en medidas de similitud. Primero se discute el problema fundamental de la correspondencia de congruencia exacta. Este tipo de correspondencia de patrones está basado implícitamente en la métrica discreta. Enseguida se discuten las propiedades deseables en las medidas de similitud y los métodos que están basados explícitamente en estas, los cuales incluyen la distancia de embotellamiento (bottleneck), la métrica Hausdorff y la distancia Frechet, entre otros.

2.3.1 Métodos de correspondencia de congruencia exacta

El problema de correspondencia de congruencia exacta es el siguiente: dados dos patrones A y B , encontrar una transformación g en G bajo la cual la imagen $g(A)$ sea igual a B . Normalmente, A y B son conjuntos finitos de R^k , y el grupo G está compuesto de las isometrías Euclidianas en R^k .

La métrica discreta, denotada por c , está definida como:

$$c(A, B) = \begin{cases} 0 & \text{si } A = B \\ 1 & \text{en otro caso} \end{cases} \quad (2.1)$$

Los algoritmos de correspondencia de congruencia exacta realizan los siguientes pasos normalizados. Ambos patrones son trasladados de tal forma que sus centroides caigan en el origen. El conjunto resultante es después proyectado sobre una esfera unitaria en R^k . Cada punto proyectado es etiquetado con la distancia del punto original al origen. Estos algoritmos proceden a realizar la correspondencia entre los conjuntos de puntos etiquetados. En dos dimensiones, el problema de realizar la correspondencia del conjunto de puntos etiquetados se reduce a realizar una correspondencia de cadenas. En tres dimensiones se reduce a una prueba de isomorfismos de gráficas. En dimensiones más grandes, se usan técnicas de reducción de dimensión.

Los algoritmos de correspondencia de congruencia exacta sólo encuentran las correspondencias si los patrones son iguales bajo una transformación. Por lo tanto, el uso práctico de estos algoritmos es limitado. Este tipo de algoritmos no se presta para la generalización a otras medidas de similitud además de la métrica discreta.

2.3.2 Propiedades deseables en las medidas de similitud

En esta sección se presenta una lista de las posibles propiedades que puede cumplir una medida de similitud. La decisión de cuando o no son deseadas ciertas propiedades dependerá de la aplicación en particular, algunas veces una propiedad será útil, mientras que algunas otras será indeseable. Algunas combinaciones de las propiedades son contradictorias, de tal forma que no se puede encontrar una función de similitud que satisfaga a todas ellas.

Una medida de disimilitud de patrones geométricos, o función de distancia, sobre una colección de patrones geométricos S es una función $d: S \times S \rightarrow R$. Las siguientes propiedades aplican para todos los patrones geométricos A, B y C en S .

- 1.- No negatividad: $d(A, B) \geq 0$
- 2.- Identidad: $d(A, A) = 0$ para todas las formas en A
- 3.- Unicidad: $d(A, B) = 0$ implica que $A = B$
- 4.- Desigualdad del triangulo fuerte: $d(A, B) + d(A, C) \geq d(B, C)$

Una función de similitud que satisfaga (2), (3) y (4) es llamada una métrica. Si una función satisface solo (2) y (4), entonces es llamada una semimétrica.

Una formulación más común de la desigualdad del triángulo es la siguiente:

- 5.- Desigualdad del triangulo: $d(A, B) + d(B, C) \geq d(A, C)$

Las propiedades (2) y (5) no implican la simetría.

Las medidas de similitud para correspondencias parciales (las cuales dan una pequeña distancia $d(A, B)$ si una parte de A hace correspondencia con una parte de B) en

general no obedecen la desigualdad del triángulo. Un contraejemplo es el que se muestra en la Figura 2.2, en donde la distancia del *hombre* al *centauro* es pequeña (se parecen mucho), la distancia del *centauro* al *caballo* es pequeña (se parecen mucho), pero la distancia del *hombre* al *caballo* es grande (son muy distintos).



Figura 2.2 Contraejemplo de la desigualdad del triángulo.

Entonces, $d(\text{hombre}, \text{centauro}) + d(\text{centauro}, \text{caballo}) > d(\text{hombre}, \text{caballo})$ no es coherente. En este caso es mejor formular una forma más débil de la desigualdad del triángulo:

6.- Desigualdad del triángulo relajada: $c(d(A, B) + d(B, C)) \geq d(A, C)$, para alguna constante $c > 1$

7.- Simetría: $d(A, B) = d(B, A)$

La simetría es una propiedad que no siempre es deseada. Esto se debe a que la percepción humana no siempre encuentra que por ejemplo una forma A es igualmente similar a B , como B es a A .

8.- Invarianza: d es invariante bajo un grupo elegido de transformaciones G , si para toda $g \in G$, $d(g(A), g(B)) = d(A, B)$.

Para el reconocimiento de objetos, normalmente es deseable que la medida de similitud sea invariante bajo las transformaciones afines, ilustradas en la Figura 2.3; aunque depende de la aplicación y algunas veces se desea un grupo más grande de transformaciones.

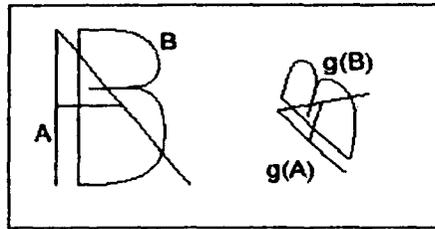


Figura 2.3 Transformaciones afines.

Las siguientes cuatro propiedades se refieren a la robustez, una forma de continuidad. Tales propiedades son útiles para que la medida sea robusta a los efectos de discretización, ver Figura 2.4.

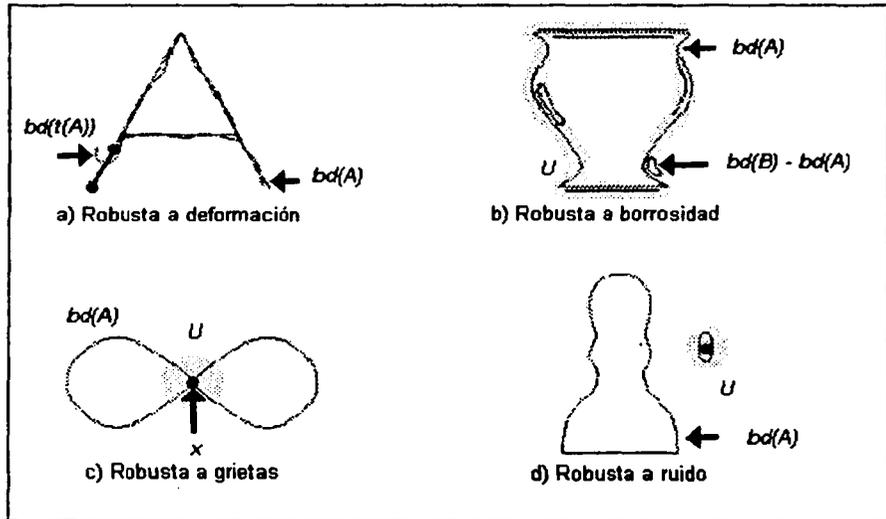


Figura 2.4 Efectos de discretización: (a)deformación, (b)borrosidad, (c)grietas y (d)ruido.

9.- Robusta a deformación: Para cada $\epsilon > 0$, existe un conjunto abierto F de deformaciones lo suficientemente cerca de la identidad, tales que $d(f(A), A) < \epsilon$ para toda $f \in F$.

10.- Robusto a la borrosidad: Para cada $\epsilon > 0$, una vecindad abierta U de $bd(A)$, el borde de A existe, tal que $d(A, B) < \epsilon$ para toda B que satisfaga $B - U = A - U$ y $bd(A) \subseteq bd(B)$.

11.- Robusto a grietas (crack): Para cada $\epsilon > 0$, y cada grieta x en el contorno de A , existe una vecindad abierta U de x tal que para toda B que satisfaga $B - U = A - U$ y $bd(A) \subseteq bd(B)$.

12.- Robusto al ruido: Para cada $x \in R^2 - A$, y cada $\epsilon > 0$, existe una vecindad abierta U de x tal que para toda B , $B - U = A - U$ implica que $d(A, B) < \epsilon$.

13.- Distributividad: Para toda A y $B \cup C$, $d(A, B \cup C) \leq d(A, B) + d(A, C)$.

La siguiente propiedad dice que el cambio de A a $A \cup B$ es más pequeño que el cambio de $A \cup C$ si B es más pequeño que C :

14.- Proporcionalidad: Para toda $A \cap B = \emptyset$ y $A \cap C = \emptyset$, si $B \subseteq C$, entonces $d(A, A \cup B) < d(A, A \cup C)$.

Finalmente, la función de distancia es estrictamente monótona si al menos uno de los pasos intermedios de sumar $B - A$ con A , y $C - B$ con B es más pequeño que los dos pasos combinados:

15.- Monotonía: Para toda $A \subseteq B \subseteq C$, $d(A, C) > d(A, B)$, o $d(A, C) > d(B, C)$.

2.3.3 Medidas de similitud para conjuntos finitos de puntos

Sean A y B dos conjuntos de puntos de tamaños n y m respectivamente. La correspondencia de puntos significa encontrar una correspondencia entre los puntos de A y los puntos de B . Una correspondencia óptima minimiza alguna medida de disimilitud entre los conjuntos de puntos. La correspondencia puede ser muchos a muchos, pero también uno a uno, y ambos casos tienen sus aplicaciones.

Las principales medidas de similitud para conjuntos finitos de puntos se describen a continuación.

2.3.3.1 Distancia L_p , Distancia de Minkowski

Muchas de las medidas de similitud sobre patrones geométricos están basadas en la distancia L_p entre dos puntos. Para dos puntos (x, y) en R^k , la distancia L_p esta definida

como $L_p(x, y) = \left(\sum_{i=0}^k |x_i - y_i|^p \right)^{1/p}$. A esta distancia también se le conoce como la distancia

de Minkowski. Para $p=2$, ésta se convierte en la distancia Euclidiana L_2 . Para $p=1$, se obtiene la distancia de Manhattan L_1 . En el caso en que p se aproxima a ∞ , se obtiene la métrica máxima: $\max_i(|x_i - y_i|)$.

Para toda $p \geq 1$, las distancias L_p son métricas. Para $p < 1$ no es una métrica, ya que la desigualdad del triángulo no se cumple.

2.3.3.2 Distancia uniforme

La distancia uniforme es el mínimo de todas las correspondencias uno a uno fentre A y B de la diferencia entre el máximo y el mínimo de $d(a, f(a))$. También es conocida como la distancia balanceada o la distancia justa. Esta distancia se puede calcular en tiempo $O(n^{10/3} \log n)$ [Efrat y Katz, 1996]. El problema de encontrar la distancia uniforme más pequeña de todas las correspondencias uno a uno entre k puntos en A y k puntos en B puede ser calculada con la misma complejidad en tiempo.

2.3.3.3 Distancia de Hausdorff

En muchas aplicaciones no todos los puntos de A necesariamente tienen un punto correspondiente en B , debido a oclusión y a ruido.

En este tipo de aplicaciones normalmente los dos conjuntos de puntos son de tamaños distintos, de tal forma que no existe una correspondencia uno a uno entre todos los puntos. En este caso, se utiliza frecuentemente la distancia de Hausdorff.

La distancia de Hausdorff para conjuntos finitos de puntos se define como sigue. La distancia directa de Hausdorff $\vec{d}(A, B)$ se define como el máximo de todos los puntos en A de las distancias a un punto de B . La distancia de Hausdorff $d(A, B)$ es el máximo de $\vec{d}(A, B)$ y $\vec{d}(B, A)$:

$$d(A, B) = \max\{\vec{d}(A, B), \vec{d}(B, A)\}, \quad \vec{d}(A, B) = \max_{a \in A} \min_{b \in B} d(a, b),$$

en donde $d(a, b)$ es la distancia en la cual se basa la distancia de Hausdorff, por ejemplo puede ser la distancia euclidiana.

La distancia parcial de Hausdorff es el máximo de las dos distancias directas parciales de Hausdorff $\vec{d}_k(A, B)$ y $\vec{d}_k(B, A)$:

$$d_k(A, B) = \max\{\vec{d}_k(A, B), \vec{d}_k(B, A)\}, \quad \vec{d}_k(A, B) = \max_{q \in A} \min_{r \in B} d(a, b)^{k^{\text{th}}}$$

La distancia parcial de Hausdorff no es una métrica ya que no cumple la propiedad de la desigualdad del triángulo.

2.3.4 Medidas de disimilitud para curvas

La forma más directa para representar curvas es con su función de posición, la cual define todas las posiciones de la curva. Una curva paramétrica A esta definida en términos de un parámetro: $A(t) = (x(t), y(t))$. En general, muchas de las parametrizaciones resultan en la misma forma de la curva, pero tienen diferentes vectores de derivadas a lo largo de la curva. Las curvas poligonales (poli-líneas) se representan normalmente por sus secuencia de vértices. También se pueden representar las curvas por medio de una definición implícita, $A: f(x,y) = 0$, pero no es muy usada para realizar correspondencia de curvas. Las principales representaciones de curvas se describen con mas detalle en el capítulo 3.

Las poli-líneas que provienen de aplicaciones del mundo real es común que contengan vértices falsos, los cuales pueden ser eliminados al aproximar el polígono. Existen muchas heurísticas para aproximar curvas poligonales, ver por ejemplo [Rosin, 1997] para una comparación. Dos métodos para una aproximación óptima son los siguientes:

- Dada una poli-línea A y un número k , construir una poli-línea aproximada A_k de k vértices, minimizando el error de aproximación, o la disimilitud, $d(A, A_k)$.
- Dada una poli-línea y una cota de error ϵ , construir una poli-línea aproximada A_ϵ con disimilitud $d(A, A_\epsilon) < \epsilon$, minimizando el número de vértices.

Ambas aproximaciones pueden ser calculadas en un tiempo de $O(n^2 \log n)$ para varias medidas de error. Por otro lado, estas aproximaciones óptimas no son adecuadas para construir aproximaciones jerárquicas, en el sentido de que cada segmento en un nivel pueda ser refinado en el siguiente nivel de la aproximación. La aproximación de polígonos en varios niveles permite que se lleve a cabo el procesamiento de curvas jerárquico [Veltkamp, 1998].

A continuación se describen brevemente las medidas de similitud de curvas mas representativas.

2.3.4.1 Función de ángulo

A partir de la representación de curvas con funciones paramétricas, se pueden derivar otras representaciones tal como la función de ángulo acumulado [Otterloo, 1992].

La función de ángulo acumulado, o función de ángulo, $\Theta_A(s)$ de un polígono A da el ángulo entre la tangente en sentido contrario de las manecillas del reloj y el eje x como una función de la longitud del arco s . $\Theta_A(s)$ mantiene el rastro de los ángulos que

se han registrado, incrementándose con vueltas a la izquierda y decrementándose con vueltas a la derecha. Claramente esta función es invariante bajo la translación de la poli-línea. La rotación de la poli-línea en un ángulo θ resulta en un corrimiento vertical de la función con una cantidad θ .

Para las poli-líneas, la función de ángulo es una función constante a tajos, que se incrementa o decrementa en los vértices, y que es constante entre dos vértices consecutivos, ver Figuras 2.5 y 2.6.

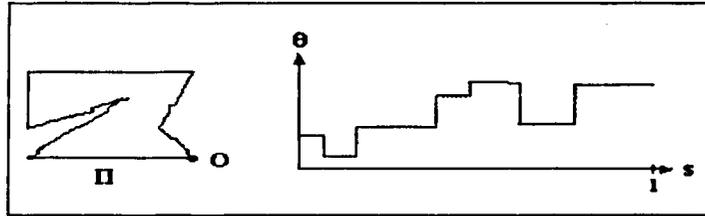


Figura 2.5 Una curva poligonal simple y su función de ángulo

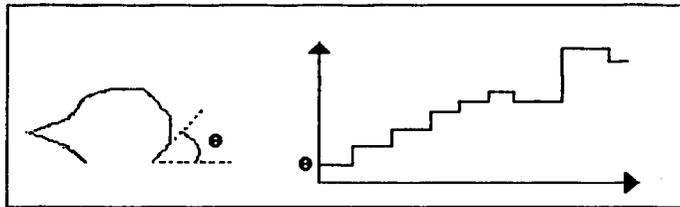


Figura 2.6 Una curva poligonal y su función de ángulo

La correspondencia de poli-líneas basada en la función de ángulo puede realizarse como sigue. Por simplicidad, primero asumamos que las dos curvas tienen la misma longitud. Para realizar la correspondencia de las curvas usando la función de ángulo, por ejemplo, primero se escala el tamaño de los polígonos de tal forma que tengan el mismo perímetro. Además, supongamos que la métrica L_p en espacios de funciones, aplicado a Θ_A y a Θ_B , da la medida de disimilitud entre A y B (ver Figura 2.7):

$$d_{A,B} = \left(\int |\Theta_A(s) - \Theta_B(s)|^p ds \right)^{1/p}$$

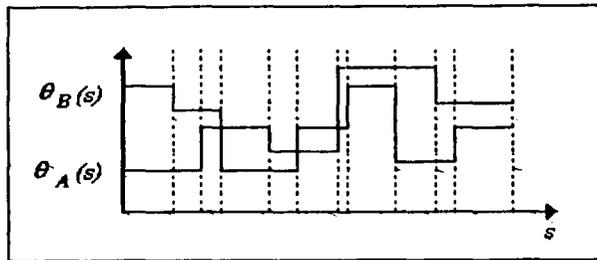


Figura 2.7 Los rectángulos encerrados por Θ_A y Θ_B , y las líneas punteadas son usadas para la evaluación de la disimilitud.

2.3.4.2 Función de firma

En cada punto a lo largo de la curva, el valor σ de la función de firma es la longitud del arco de la curva a la izquierda o sobre la línea tangente en ese punto, ver Figura 2.8.

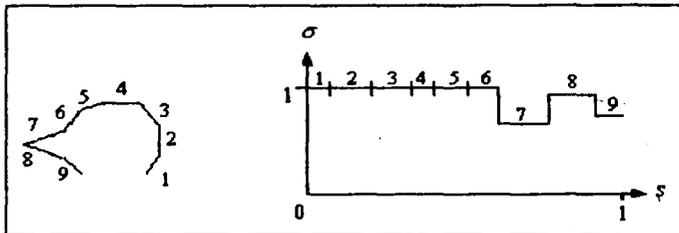


Figura 2.8 Una curva y su función de firma

Esta función es invariante a las transformaciones de similitud: combinaciones de rotaciones, translaciones y escalas. Para las curvas convexas, la función de firma es uno en todas partes, porque en cada punto, toda la curva se encuentra a la izquierda de la tangente. Para una sola curva poli-línea, la función de firma puede ser calculada en tiempo de $O(n^2)$ [O'Rourke, 1985].

Para poli-líneas, las medidas de disimilitud que pueden ser usadas son las que están basadas en "alteración de tiempos" de secuencias de elementos (vértices o segmentos), apareando los elementos de A con los elementos de B . El apareamiento no necesita ser uno a uno: el apareamiento del elemento i de A al elemento j de B , podría ser seguido por un apareamiento de i con $j+1$, $i+1$ con j , o $i+1$ con $j+1$.

2.3.4.3 Longitud de arco afín

En vez de usar funciones de ángulo, se pueden usar representaciones de curvas invariantes a transformaciones afines como base para la correspondencia de formas. Mientras que las funciones de ángulo son invariantes únicamente bajo transformaciones de similitud, esta representación es invariante a todas las transformaciones afines. Huttenlocher y Kedem [Huttenlocher y Kedem, 1990] usan la distancia unidimensional de Hausdorff para comparar dos descripciones de longitud de arco afín de curvas.

Sea $A: [0,1] \rightarrow R$ una curva dos veces continuamente diferenciable, y sea A' y A'' la primera y segunda derivada, respectivamente. La longitud de arco afín es la función $\sigma: R \rightarrow R$ dada por:

$$\sigma(t) \rightarrow \int_0^t |\det(A'(x), A''(x))|^{1/3} dx$$

La longitud de arco normalizada esta definida como sigue:

$$\sigma^*(t) = \frac{\sigma(t)}{\sigma(1)}$$

En vez de estas definiciones, Huttenlocher y Kedem usan una versión discretizada de la longitud de arco afín para representar el borde con un polígono simple. Esta representación discretizada es un conjunto finito de números entre 0 y 1, un número para cada vértice del borde. Dos polígonos simples son iguales si las longitudes de arco discretizadas son iguales bajo translación modulo 1. Este problema puede ser resuelto de tal forma que sea robusto a la perturbación al minimizar la distancia de Hausdorff entre dos representaciones (vistas como conjuntos de puntos finitos unidimensionales). El último problema puede ser resuelto en tiempo $O(mn \log(mn))$.

2.3.4.4 Distancia de Hausdorff

La distancia de Hausdorff no esta definida sólo para conjuntos finitos de puntos, sino también para cualesquiera dos conjuntos compactos. Un caso especial son los conjuntos de poli-líneas. Los resultados para poli-líneas son los mismos que para polígonos.

Dados dos polígonos A y B , la distancia directa de Hausdorff de A a B puede ser calculada usando el diagrama de Voronoi de B , el cual le asigna a cada vértice y arista de A una región de puntos que se encuentran mas cerca de ese vértice o arista que cualquier otro. [Belogay et al, 1997] presenta cómo calcular la distancia de Hausdorff entre curvas.

2.3.4.5 Distancia de Fréchet

La distancia de Hausdorff normalmente no es apropiada para medir la disimilitud entre curvas. Para todos los puntos en A , la distancia al punto mas cercano en B podría

ser pequeña, pero si caminamos hacia adelante a lo largo de las curvas A y B simultáneamente, y medimos la distancia entre los puntos correspondientes, la máxima de estas distancias podría ser mas grande, ver Figura 2.11. Esto es a lo que se llama la distancia de Fréchet. Formalmente, sean A y B dos curvas parametrizadas $A(\alpha(t))$ y $B(\beta(t))$, y sean sus parametrizaciones α y β funciones continuas del mismo parámetro $t \in [0,1]$, tales que $\alpha(0) = \beta(0) = 0$ y $\alpha(1) = \beta(1) = 1$. La distancia de Fréchet es el mínimo de todas las parametrizaciones monótonas incrementales $\alpha(t)$ y $\beta(t)$ de la distancia máxima $d(A(\alpha(t)), B(\beta(t)))$, $t \in [0,1]$.

[Alt y Godeau, 1995] considera el calculo de la distancia Fréchet para el caso especial de poli-líneas. El decidir cuando la distancia de Fréchet es mas pequeña que una constante dada, puede realizarse en tiempo $O(mn)$. Basándose en este resultado, y en la técnica de búsqueda paramétrica se deriva que el calculo de la distancia de Fréchet puede calcularse en tiempo $O(mn \log(mn))$. A pesar de que el algoritmo tiene baja complejidad asintótica, no es realmente práctico. La técnica de búsqueda paramétrica usada aquí usa una red ordenada con constantes muy grandes en tiempo de ejecución. Si se usa un algoritmo de ordenamiento mas simple se puede obtener un tiempo de ejecución asintótico de $O(mn(\log mn)^3)$. Aún así, la búsqueda paramétrica no es fácil de implementar.

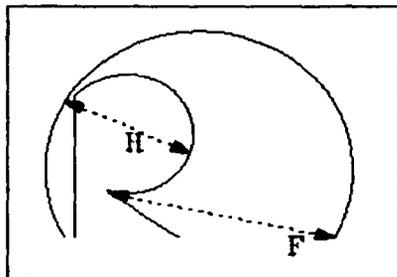


Figura 2.11 La distancia Hausdorff (H) y la distancia de Fréchet (F) entre dos curvas.

Se obtiene una variación de la distancia de Fréchet al eliminar la condición de la monotoneidad de la parametrización. La distancia de Fréchet resultante $d(A,B)$ es una semimétrica: la distancia cero no significa necesariamente que los objetos son el mismo.

Otra variación consiste en considerar la correspondencia parcial: encontrar la parte de una curva con la cual la otra tiene la distancia de Fréchet mas pequeña. El problema de decisión correspondiente puede ser resuelto en tiempo $O(mn \log(mn))$, y el problema de calculo en tiempo $O(mn(\log(mn))^2)$.

2.3.5 Construcciones

En esta sección se discutirán varias construcciones que pueden ser usadas para manipular las medidas de similitud, con la finalidad de que estas cumplan con ciertas propiedades.

2.3.5.1 Remapeo

Sea $w: [0, \infty) \rightarrow [0, \infty)$ una función monótona continua con $w(x) = 0$ si $x = 0$, y subaditiva: $w(x + y) \leq w(x) + w(y)$. Ejemplos de ésta incluyen mapeos a $x/(1+x)$, $\tan^{-1}(x)$, $\log(x)$, $x^{1/p}$, para alguna $p \geq 1$, y $\min(x, c)$, para alguna constante positiva c . Si $d(A, B)$ es una métrica, entonces también lo es $\bar{d}(A, B) = w(d(A, B))$. Con alguna de estas funciones, se puede mapear una métrica no acotada a una métrica acotada.

2.3.5.2 Normalización

La normalización se usa comúnmente para escalar el rango de valores de una medida de similitud a $[0, 1]$, pero al realizar una normalización puede también cambiar otras propiedades. Por ejemplo podría hacerla invariante a un grupo de transformaciones mas grande.

2.3.5.3 De Semi-métrica a Métrica

Sea S un espacio de objetos, y d una semimétrica. La identificación de todos los elementos A, B de S para los cuales $d(A, B) = 0$ y considerar a estos como un solo objeto, produce otro espacio S' . La semimétrica en S entonces da una métrica $d'(C, D) = d(C, D)$ en S' .

2.3.5.4 Ventaja

Sea d alguna función de distancia en un espacio S de patrones, $d: S \times S \rightarrow \mathbb{R}$. Para algún patrón fijo $C \in S$ (objeto de ventaja), la función $\bar{d}_C(A, B) = |d(A, C) - d(B, C)|$ es una semimétrica, incluso si d no cumple las propiedades de no-negatividad, identidad, desigualdad del triángulo débil y simetría.

Como se vio, la mayoría de las medidas de similitud para realizar correspondencias de curvas se basan en las representaciones de éstas. En el siguiente capítulo se da un panorama general de las principales representaciones de curvas tridimensionales, haciendo énfasis en la representación del código de cadenas de cambios de dirección ortogonal, que es la representación con la que se trabajó en esta tesis.

CAPITULO 3

REPRESENTACIÓN DE CURVAS TRIDIMENSIONALES.

La representación de las curvas es la base sobre la cual se realiza su correspondencia, por esta razón en este capítulo se presenta una breve visión de las principales representaciones de curvas tridimensionales que existen hasta el momento, incluyendo la representación de curvas en la que se basa el presente trabajo de tesis. Además, se presentan una serie de características deseables de una representación.

El uso de una representación invariante permite que la comparación entre éstas sea directa, de tal forma que la correspondencia de las curvas pueda realizarse independientemente de las posiciones de éstas. Una representación invariante de curvas puede ser clasificada como global, local o semi-local. En la representación local, se descompone a la curva en segmentos determinados por puntos de características locales como por ejemplo en la curvatura máxima; y después se representa por los puntos y por sus atributos (deseablemente invariantes). Basándose en esta representación, la correspondencia de curvas se reduce a realizar la correspondencia de los patrones de puntos con atributos. Tales representaciones son adecuadas para manejar curvas parciales y no segmentadas (conectadas). Una desventaja es que trata a la curva como un patrón de los puntos característicos y por lo tanto no hace uso de la información contenida en los segmentos de la curva que se encuentran entre los puntos característicos. Esto no es bueno, ya que tal información provee restricciones sustanciales para realizar la correspondencia de curvas. Otra desventaja es que el cálculo de invariantes locales normalmente requiere de derivadas de orden alto.

Una representación global puede no estar basada en características, por ejemplo usando polinomios, [Schwartz y Sharir, 1987] y [Besl y McKay, 1992]. La correspondencia se realiza normalmente usando el método de ajuste de mínimos cuadrados. La ventaja de tal esquema es que éste no requiere de la extracción de características locales ni de la estimación de derivadas. Una desventaja es que éste es menos adecuado para manejar curvas parciales y cambios de escala. Mas aún, éste requiere que los datos de la curva forzosamente contengan solo una curva modelada antes de que se pueda realizar la correspondencia con el modelo. Cuando los datos contienen varias curvas, las distintas curvas tienen que ser separadas previamente.

Una representación semi-local requiere derivadas de menor orden en el cálculo de invariantes. También usa por completo la información de la curva y por lo tanto provee mayores restricciones a la correspondencia. Un ejemplo son las coordenadas invariantes afines para objetos planos [Lamdan et al, 1988], [Wolfson, 1990] y [Rothwell et al, 1992]. En este caso, un segmento de curva plana se mapea a un sistema coordenado invariante

en el cual se pueden calcular los invariantes. Una desventaja de tal representación es que es menos local que una representación verdaderamente local.

3.1 PROPIEDADES DESEABLES

Un problema común en la investigación de representaciones de formas, es el cómo juzgar su calidad. No todas las representaciones son apropiadas para cada uno de los tipos de forma y para todas las aplicaciones. La elección de la representación depende de las propiedades de la forma que se deseen describir y de la presencia de ruido, por ejemplo. Aún no existe un criterio de evaluación consistente, sin embargo, varios autores han propuesto criterios de evaluación en forma de listas de cualidades que una buena representación debería tener.

[Marr y Nishihara, 1978] proponen los siguientes criterios:

- **Accesibilidad.** La accesibilidad describe qué tan fácil (o difícil) es calcular la representación de la forma en términos de requerimientos de memoria y tiempo de procesamiento.
- **Alcance.** El alcance se refiere a la cantidad de tipos de forma que pueden ser representadas con ésta.
- **Unicidad.** La unicidad describe cuándo existe un mapeo uno a uno entre las formas y sus representaciones.
- **Estabilidad y sensibilidad.** Se refiere a qué tan sensible es una representación a pequeños cambios en la forma.

[Brady, 1983] propone los siguientes criterios:

- **Rico soporte local.** Este criterio requiere que la representación sea preservante de la información y que pueda ser calculada localmente. El soporte local es importante para la representación de objetos parcialmente ocultos.
- **Extensión suave.** Este criterio asegura que las descripciones locales puedan fácilmente producir descripciones globales. Esta es una especie de continuidad de la representación.
- **Propagación.** Este criterio agrega una propiedad jerárquica a la representación, en el sentido de que las subpartes perceptivas son propagadas del nivel de descripción local al nivel global.

3.2 PRINCIPALES REPRESENTACIONES DE CURVAS

Una curva puede representarse con su función implícita, explícita o paramétrica, principalmente; pero existen otras representaciones (por ejemplo, códigos de cadenas) las cuales han llevado a resultados interesantes.

Estas representaciones se presentan brevemente en las siguientes secciones.

3.2.1 Representación explícita

La forma explícita de una curva en dos dimensiones da el valor de una variable, la variable dependiente, en términos de la otra, la variable independiente. En el espacio (x,y) , se puede escribir:

$$y = f(x), \quad (3.1)$$

o se podría invertir la relación y expresar a x como una función de y :

$$x = g(y). \quad (3.2)$$

No existe ninguna garantía de que esta forma exista para una curva dada. Para representar una línea, normalmente se escribe su ecuación en términos de su pendiente m , a pesar de que sabemos que esta ecuación no funciona para líneas verticales. Este problema es uno de muchos efectos de dependencia de sistemas de coordenadas que causan problemas para todos los campos en donde se trabaja con curvas. Las líneas y los círculos existen independientemente de cualquier representación, y cualquier representación que falle para ciertas orientaciones, como para líneas verticales, tiene serias deficiencias.

Los círculos proveen un ejemplo mas ilustrativo. Consideremos un círculo de radio r centrado en el origen. Un círculo tiene curvatura constante (una medida de que tan rápido se dobla una curva en un punto). No existe una curva bidimensional cerrada mas simétrica que el círculo. Aún así, lo mejor que podemos hacer, usando una representación explícita, es escribir una ecuación para la mitad de éste:

$$y = \sqrt{r^2 - x^2} \quad (3.3)$$

y una segunda ecuación para la otra mitad.

En tres dimensiones, la representación de una curva requiere de dos ecuaciones.

Por ejemplo, si x es la variable independiente, se tienen dos variables dependientes:

$$\begin{aligned}y &= f(x) \\z &= g(x)\end{aligned}\tag{3.4}$$

Al igual que en dos dimensiones, una curva podría no tener una representación explícita. Por ejemplo, las ecuaciones

$$\begin{aligned}y &= ax + b \\z &= cx + d\end{aligned}\tag{3.5}$$

describen una línea en tres dimensiones, pero estas ecuaciones no pueden representar una línea en un plano de constante x .

3.2.2 Representación implícita

En dos dimensiones, una curva implícita puede ser representada por la ecuación

$$f(x, y) = 0\tag{3.6}$$

Los dos ejemplos anteriores, la línea y el círculo centrado en el origen, tienen las siguientes representaciones respectivamente:

$$ax + by + c = 0\tag{3.7}$$

$$x^2 + y^2 - r^2 = 0\tag{3.8}$$

La función f , es realmente un probador de pertenencia. En otras palabras, es una función que divide el espacio en aquellos puntos que pertenecen a la curva de aquellos que no. Esto nos permite tomar una pareja de (x, y) y evaluar f para determinar cuando este punto pertenece a la curva. En general, esta no proporciona una manera analítica de encontrar un valor y sobre la curva que corresponde para una x dada, o viceversa. La forma implícita es menos dependiente del sistema de coordenadas que la forma explícita.

En tres dimensiones, la forma implícita

$$f(x, y, z) = 0\tag{3.9}$$

describe una superficie.

Por ejemplo, cualquier plano puede ser escrito como:

$$ax + by + cz + d = 0 \quad (3.10)$$

para a , b , c y d constantes. Y una esfera de radio r centrada en el origen puede ser descrita por:

$$x^2 + y^2 + z^2 - r^2 = 0 \quad (3.11)$$

Las curvas en tres dimensiones no son tan fácilmente representadas de forma implícita. Podemos representar una curva como la intersección, si es que ésta existe, de las dos superficies:

$$f(x, y, z) = 0 \quad (3.12)$$

$$g(x, y, z) = 0 \quad (3.13)$$

De esta forma, si se prueba un punto (x, y, z) y éste pertenece a ambas superficies, entonces éste debe pertenecer a la curva de intersección.

3.2.3 Forma paramétrica

La forma paramétrica de una curva expresa el valor de cada variable espacial para puntos de la curva en términos de una variable independiente, u : el parámetro. En tres dimensiones, se tienen tres funciones explícitas:

$$x = x(u)$$

$$y = y(u)$$

$$z = z(u) \quad (3.14)$$

Una de las ventajas de la forma paramétrica es que ésta es la misma en dos y tres dimensiones. En dos dimensiones simplemente se elimina la ecuación para z . Una interpretación útil de la forma paramétrica es visualizar la posición de los puntos dibujados conforme u varía, como se muestra en la Figura 3.1. Podemos pensar en la derivada

$$\frac{dp(u)}{du} = \begin{bmatrix} \frac{dx(u)}{du} \\ \frac{dy(u)}{du} \\ \frac{dz(u)}{du} \end{bmatrix} \quad (3.15)$$

como la velocidad en que la curva se dibuja en la dirección tangente a la curva.

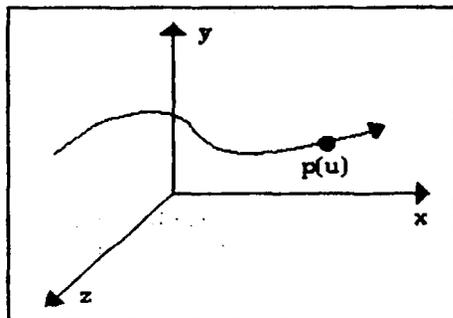


Figura 3.1 Curva paramétrica

La forma paramétrica es más flexible y más robusta, aunque aún existen las dependencias de un sistema coordenado, ya que aún se usa x , y y z para una representación en particular.

3.2.3.1 Curvas paramétricas polinomiales

Las formas paramétricas no son únicas. Una curva dada puede ser representada de muchas formas, pero las más usadas son las paramétricas en las cuales las funciones son polinomios en u .

Consideremos una curva de la forma

$$p(u) = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix} \quad (3.16)$$

Una curva paramétrica polinomial de grado n es de la forma

$$p(u) = \sum_{k=0}^n u^k c_k \quad (3.17)$$

donde cada c_k tiene componentes independientes x , y y z ; que es,

$$c_k = \begin{bmatrix} c_{xk} \\ c_{yk} \\ c_{zk} \end{bmatrix} \quad (3.18)$$

Las matrices de $n+1$ columnas $\{c_k\}$ son los coeficientes de p ; lo cual nos da $3(n+1)$ grados de libertad en la manera en que se eligen los coeficientes de una p en particular. No existe ninguna relación entre los componentes x , y y z , de tal forma que se puede trabajar con tres ecuaciones independientes, cada una de la forma

$$p(u) = \sum_{k=0}^n u^k c_k \quad (3.19)$$

y cada una con $n+1$ grados de libertad. Podemos definir las curvas para cualquier rango $u_{\min} \leq u \leq u_{\max}$, pero sin pérdida de generalidad se puede asumir que $0 \leq u \leq 1$.

Conforme el valor de u varía sobre su rango, se va definiendo un segmento de curva, como se muestra en la Figura 3.2.

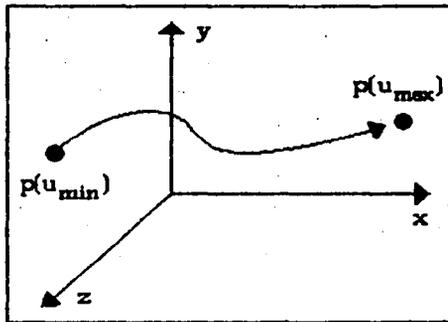


Figura 3.2 Segmento de curva

3.2.3.2 Curvas paramétricas polinomiales cúbicas

Si se decidiera usar las curvas paramétricas polinomiales para representar las curvas, en seguida se debe elegir el grado de la curva. Si se elige un grado alto, se tendrán muchos parámetros que se pueden elegir para formar la forma deseada, pero la evalua-

ción de los puntos sobre la curva sería costosa. Además, conforme el grado de un polinomio se hace mayor, existe mayor peligro de que la curva se haga mas rugosa. Por otro lado, si se elige un grado demasiado pequeño, no se tendrán suficientes parámetros con los cuales trabajar. Por lo tanto, si se diseña cada segmento de curva sobre un intervalo pequeño, se pueden realizar la mayoría de los propósitos tan solo con curvas de grado pequeño. A pesar de que se tendrán sólo unos pocos grados de libertad, éstos son suficientes para permitir crear la forma deseada en una región pequeña.

Se puede escribir un polinomio paramétrico cúbico usando una matriz como

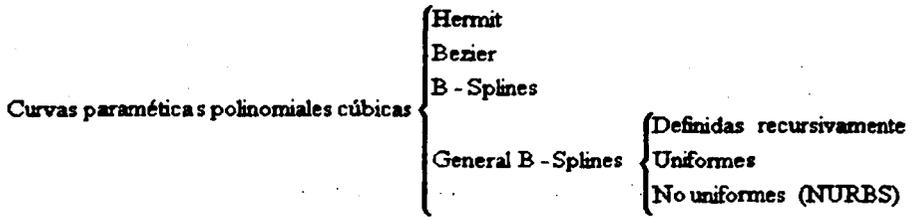
$$p(u) = c_0 + c_1u + c_2u^2 + c_3u^3 = \sum_{k=0}^3 x_k u^k = u^T c \quad (3.20)$$

$$c = \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}, \quad u = \begin{bmatrix} 1 \\ u \\ u^2 \\ u^3 \end{bmatrix}, \quad c_k = \begin{bmatrix} c_{kx} \\ c_{ky} \\ c_{kz} \end{bmatrix}. \quad (3.21)$$

De esta forma, c es una matriz columna que contiene los coeficientes del polinomio; esto es lo que se desea determinar a partir de los puntos de control. De esta forma se pueden determinar varios tipos de curvas cúbicas. Los tipos difieren en la forma en que se usan los puntos de control. Se busca encontrar 12 ecuaciones en 12 incógnitas para cada tipo, pero como x, y y z son independientes, se pueden agrupar estas ecuaciones en tres conjuntos independientes de cuatro ecuaciones con cuatro incógnitas.

El diseño de un tipo en particular de curva cúbica se basa en los datos dados en algunos valores del parámetro u . Estos datos podrían tomar la forma de condiciones de interpolación en los cuales el polinomio debe corresponder con los datos en algunos puntos. Los datos también podrían requerir que el polinomio interpolara algunas derivadas en ciertos valores del parámetro. También se podrían tener condiciones de suavidad que forcen varias condiciones de continuidad en los puntos de unión que se comparten por dos segmentos de curva. Finalmente, se pueden tener condiciones que no son tan estrictas, que solo requieran que la curva pase cerca de varios puntos conocidos. Cada tipo de condición define un tipo diferente de curva, y dependiendo de como se usen algunos de los datos dados, y los mismos datos pueden definir mas de una sola curva.

Los tipos de curvas paramétricas polinomiales cúbicas más comunes son:



3.2.4 Códigos de cadenas para curvas discretas tridimensionales.

Otra alternativa surge al considerar que cualquier curva continua en tres dimensiones puede ser digitalizada y representada como una curva tridimensional discreta. La primera aproximación para representar curvas discretas tridimensionales usando cadenas de códigos fué introducida por [Freeman, 1974]. Una segunda aproximación fué propuesta por [Bribiesca, 2000] y [Bribiesca y Velarde, 2001]. Cada una se presenta a continuación por separado y al final se da una comparación entre ambas propuestas.

3.2.4.1 Código de cadenas de Freeman

Un cadena de Freeman para representar una curva discreta 3D es una secuencia en la cual cada elemento es un símbolo que representa el vector que une dos voxels vecinos de la curva discreta 3D [Haralick y Shapiro, 1991]. Por lo tanto, existen 26 posibles direcciones que corresponden a los siguientes vectores: $(0,1,0)$, $(0,1,1)$, $(0,0,1)$, $(0,-1,1)$, $(0,-1,0)$, $(0,-1,-1)$, $(0,0,-1)$, $(0,1,-1)$, $(1,1,0)$, $(1,1,1)$, $(1,0,1)$, $(1,-1,1)$, $(1,-1,0)$, $(1,-1,-1)$, $(1,0,-1)$, $(1,1,-1)$, $(1,0,0)$, $(-1,1,0)$, $(-1,1,1)$, $(-1,0,1)$, $(-1,-1,1)$, $(-1,-1,0)$, $(-1,-1,-1)$, $(-1,0,-1)$, $(-1,1,-1)$ y $(-1,0,0)$.



Figura 3.3 Código de cadenas de Freeman. (a) Las 6 posibles direcciones. (b) Una curva discreta representada con el código de Freeman.

Con la finalidad de simplificar la descripción anterior, sólo se usarán direcciones ortogonales. De esta forma, las posibles direcciones se reducen a seis vectores (0,1,0), (0,0,1), (0,-1,0), (0,0,-1), (1,0,0) y (-1,0,0). Estas direcciones se representan con los dígitos del uno al seis, respectivamente, como se muestra en la Figura 3.3 (a). Finalmente, la Figura 3.3 (b) ilustra una curva 3D discreta y su representación con la cadena de Freeman usando este código simplificado.

3.2.4.2 Código de cadenas de cambio de dirección ortogonal

[Bribiesca, 2000] propuso un código de cadenas para representar curvas tridimensionales discretas. El código de cadenas es invariante bajo las transformaciones de rotación y translación. Opcionalmente puede hacerse invariante al punto de origen y a la transformación espejo. Además, se genera un descriptor de curva único, con lo cual la comparación de curvas se simplifica. Bribiesca representó un DEM (Modelo Digital de Elevaciones) como un sólido binario por medio de una sola cadena. La presente tesis está basada en esta representación, por lo cual se explicará con mas detalle en la sección 3.3.

3.2.5 Otras representaciones

[Li, 1993] propuso una representación invariante basada en un sistema de coordenadas invariante a transformaciones de similitud (SICS). Este sistema fue usado para realizar correspondencias de curvas tridimensionales bajo el grupo de transformaciones de similitud. De esta forma, se establece un SICS para cada segmento de curva limitado por dos puntos característicos invariantes. En el SICS, la geometría de un segmento de curva es único.

Li hace ciertas suposiciones. Supone que los datos 3D están disponibles. En la escena, existe al menos un segmento de curva continuo que contiene al menos dos puntos característicos invariantes. Las curvas bajo consideración excluyen a las regulares, tales como un círculo.

3.3 CÓDIGO DE CADENAS DE CAMBIO DE DIRECCIÓN ORTOGONAL

La Figura 3.4(a) muestra un ejemplo de una curva continua 3D y (b) ilustra la representación discreta de la curva mostrada en (a). Nótese que la curva 3D discreta en la Figura 3.4(b) está compuesta de segmentos de línea rectos de la misma longitud en posiciones ortogonales, el cual se asumirá que es igual a uno.

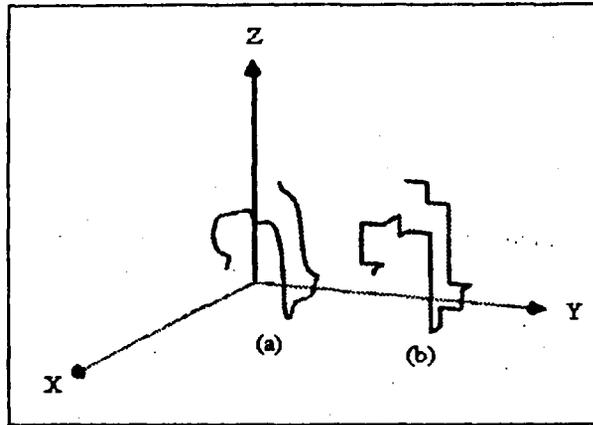


Figura 3.4 Un ejemplo de una curva discreta 3D: (a) una curva 3D continua; (b) la representación discreta de la curva mostrada en (a).

3.3.1 Conceptos y definiciones

Las curvas discretas están compuestas de segmentos de líneas rectas de longitud constante, en donde dos segmentos de líneas contiguos definen un cambio de dirección y dos cambios de dirección definen un elemento de la cadena. Existen sólo cinco posibles direcciones ortogonales para representar a cualquier curva discreta tridimensional. Esta cadena de códigos sólo considera cambios de dirección relativos, lo cual permite que la cadena sea una representación invariante a la rotación.

Definición 1. Elemento de una cadena.

Un elemento a_i de una cadena indica los cambios de dirección ortogonales de los segmentos rectos contiguos de una curva discreta en esa posición del elemento.

Existen únicamente cinco cambios de dirección posibles (0,1,2,3 y 4) para representar cualquier curva discreta en tres dimensiones, los cuales indican cambios de dirección relativos. Cada elemento de la cadena etiqueta a un vértice de la curva 3D discreta e indica el cambio de dirección ortogonal en dicho vértice. Se asume que los dos primeros segmentos de la curva forman un ángulo recto y que su vértice correspondiente se deja sin etiqueta.

A continuación se presentan las reglas para el etiquetado de los vértices, en donde a y b son las direcciones de los dos segmentos anteriores de referencia y c es la dirección del vértice a ser etiquetado. Nótese que x denota el producto cruz.

$$label(a, b, c) = \begin{cases} 0, & \text{si } c = b & \text{(igual)} \\ 1, & \text{si } c = a \times b & \text{(derecha)} \\ 2, & \text{si } c = a & \text{(adentro)} \\ 3, & \text{si } c = -(a \times b) & \text{(izquierda)} \\ 4, & \text{si } c = -a & \text{(frente)} \end{cases}$$

De esta forma, una etiqueta es un elemento del conjunto $\{0, 1, 2, 3, 4\}$.

La Ilustración I.2 muestra los 5 elementos de la cadena y la Ilustración I.3 muestra un ejemplo de una curva 3D discreta representada por una cadena.

Definición 2. Cadena

Una cadena A es una secuencia ordenada de elementos y esta representada por $A = a_1 a_2 \dots a_n = \{a_i; 1 \leq i \leq n\}$ donde n indica el número de elementos de la cadena.

En términos de etiquetas, una cadena es una secuencia no vacía y finita de etiquetas. En lenguajes formales, las cadenas son las palabras no vacías del lenguaje sobre el alfabeto $\{0, 1, 2, 3, 4\}$.

Definición 3. Longitud

La longitud L de una cadena es la suma de las longitudes de sus elementos. L puede ser expresada como:

$$L = (n+2) * l$$

donde l es la longitud de cada segmento de línea recto, el cual es considerado ser igual a uno.

3.3.2 Independencia de rotación

La cadena de códigos es invariante bajo la transformación de rotación, esto se debe al hecho de que se usan cambios relativos de dirección.

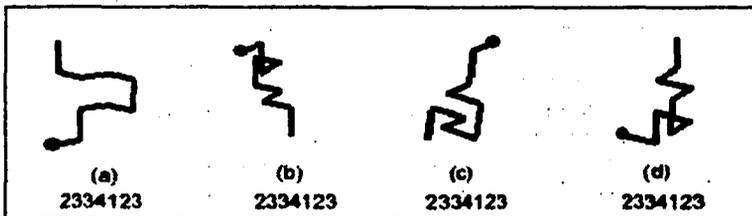


Figura 3.5 Independencia de rotación: (a) una curva discreta y su cadena; (b)-(d) rotaciones de la misma cadena sobre el eje, X, Y y Z respectivamente.

TESIS CON FALLA DE ORIGEN

La Figura 3.5 muestra la invarianza bajo rotación usando esta cadena. Nótese que todas las cadenas son iguales, por lo tanto, son invariantes bajo la transformación de rotación.

3.3.3 Inverso de una cadena

El inverso de una cadena es otra cadena formada de los elementos de la primera cadena ordenados en orden inverso. La Figura 3.6(b) muestra el inverso de la cadena presentada en 3.6(a), nótese que los elementos de la cadena inversa mostrada en 3.6(a) están organizados en orden inverso. La Figura 3.6(c) muestra una curva discreta y a su cadena, la cual contiene algunos elementos iguales a cero.

Cuando se recorre una curva con la finalidad de obtener sus elementos y se encuentran elementos cero, necesitamos saber cuál fue el último elemento distinto de cero para poder determinar el siguiente elemento. En el caso que se muestra en la Figura 3.6(c), el primer elemento encontrado "4" se obtuvo con referencia al elemento previo "2" el cual no es un "0". De esta forma la orientación no se pierde. Entonces, la cadena inversa de la cadena que se muestra en la Figura 3.6(c) corresponde a la cadena presentada en 3.6(d), nótese que el orden de un elemento de la cadena inversa se mueve cuando existen elementos cero.

Se define la concatenación y reversa de cadenas de la misma forma en que se define la concatenación de listas en la programación funcional, denotadas por ++ y reverse respectivamente. Se asumen las propiedades básicas de estas funciones, tales como la asociatividad de ++, así como:

$$\begin{aligned} \text{reverse}(\text{reverse } x) &= x, \\ \text{reverse}(x ++ y) &= \text{reverse } y ++ \text{reverse } x. \end{aligned}$$

Definición 4. Invertibilidad

Una cadena c es invertible si su último elemento es distinto de 0. Su cadena inversa, $\text{inv } c$, se define como sigue:

- (i) Si 0 no ocurre en c , entonces $\text{inv } c = \text{reverse } c$.
- (ii) Si c tiene todos sus elementos iguales a 0, excepto el último, entonces $\text{inv } c = c$.
- (iii) De otra forma, c puede expresarse como una concatenación de dos o más cadenas, $c = c_1 ++ c_2 ++ \dots ++ c_n$, donde cada $c_k (k=1, \dots, n)$ pertenece a alguna de las formas presentadas en las clases anteriores. Entonces:

$$\text{inv } c = \text{inv } c_n ++ \dots ++ \text{inv } c_2 ++ \text{inv } c_1.$$

Nótese que la descomposición de c puede no ser única, pero esto no afecta la definición.

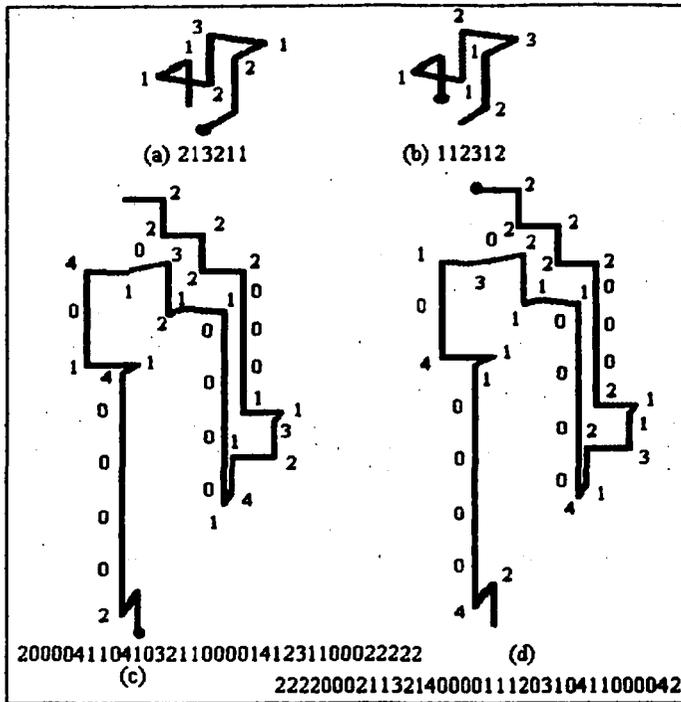


Figura 3.6 El inverso de una cadena: (a) una curva discreta y su cadena; (b) la cadena inversa de (a); (c) una curva discreta y su cadena, la cual contiene algunos elementos cero; (d) el inverso de la cadena en (c).

Teorema 1. Si c es invertible, entonces $inv(inv\ c) = c$.

Demostración. Considerese en turno cada uno de los casos anteriores en la definición de invertibilidad. Si c tiene la forma de (i), entonces también tiene esta forma (*reverse* c), y por lo tanto se tiene que $inv(inv\ c) = reverse(reverse\ c) = c$. Si c tiene la forma de (ii), entonces $inv(inv\ c) = inv\ c = c$. En otro caso, si c tiene la forma de (iii), también la tiene (*inv* c) y entonces:

$$\begin{aligned}
 \text{inv}(\text{inv } c) &= \text{inv}(\text{inv } c_n ++ \dots ++ \text{inv } c_2 ++ \text{inv } c_1) \\
 &= \text{inv}(\text{inv } c_1) ++ \text{inv } c_2 (\text{inv } c_3) ++ \dots ++ \text{inv}(\text{inv } c_n) \\
 &= \text{inv}(\text{inv } c_1) ++ \text{inv } c_2 (\text{inv } c_3) ++ \dots ++ \text{inv}(\text{inv } c_n) \\
 &= c_1 ++ c_2 ++ \dots ++ c_n = c,
 \end{aligned}$$

Teorema 2. Si a y b son invertibles, entonces $\text{inv}(a ++ b) = \text{inv } b ++ \text{inv } a$.

Demostración. Si $a ++ b$ es ya una concatenación de la forma requerida en (iii), la igualdad se implica de tal definición. De otra forma, tanto a como b pueden ser descompuestas como se requiere, de lo cual se obtiene:

$$\begin{aligned}
 \text{inv}(a ++ b) &= \text{inv}(a_1 ++ \dots ++ a_n ++ b_1 ++ \dots ++ b_m) \\
 &= \text{inv } b_m ++ \dots ++ \text{inv } b_1 ++ \text{inv } a_n ++ \dots ++ \text{inv } a_1 \\
 &= \text{inv } b ++ \text{inv } a
 \end{aligned}$$

Por simple inducción, este resultado puede ser generalizado como sigue.

Teorema 3. Si c_1, \dots, c_n son invertibles, entonces $\text{inv}(c_1 ++ \dots ++ c_n) = \text{inv } c_n ++ \dots ++ \text{inv } c_1$.

3.3.4 Independencia del punto de inicio para curvas abiertas

Usando el concepto de la inversa de una cadena, esta notación puede ser el punto inicial normalizado al elegir el punto inicial de tal forma que la secuencia resultante de elementos forme un entero de valor mínimo. Por ejemplo, la cadena de la curva abierta que se muestra en la Figura 3.6(a) representa el número entero: 213211; y la cadena que se muestra en 3.6(b) representa el número 112312, respectivamente. Por lo tanto, el entero mínimo corresponde a la cadena que se muestra en la Figura 3.6(b). Por lo tanto este es el punto inicial normalizado de la cadena.

3.3.5 Independencia del punto de inicio para curvas cerradas

Las curvas cerradas que son descritas con esta cadena pueden hacerse invariantes al punto de inicio, eligiendo el punto de inicio de tal forma que la secuencia resultante de elementos forme el mínimo entero. Por lo tanto, la cadena de la curva discreta 3D que se presenta en la Figura 3.7(a) puede hacerse invariante bajo el punto de inicio al rotar los dígitos hasta que el número sea mínimo. Finalmente, la Figura 3.7(b), ya es invariante al punto de inicio.

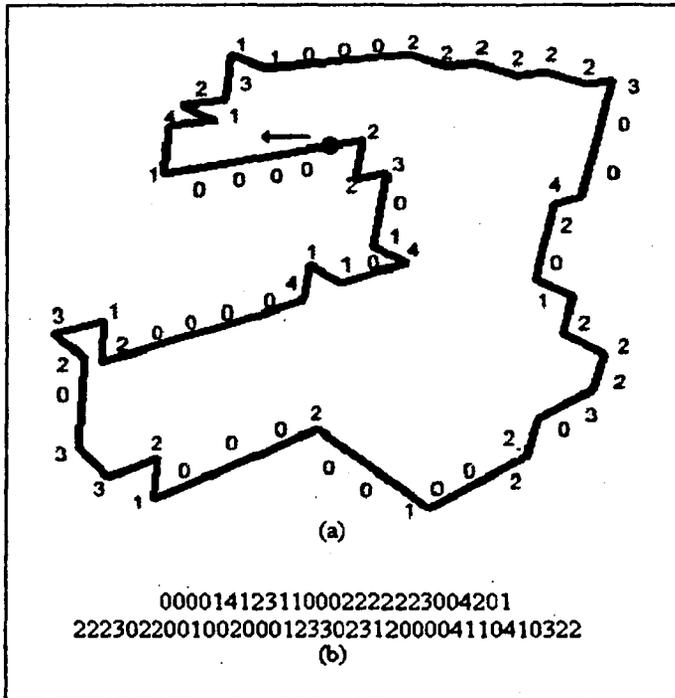


Figura 3.7 Independencia del punto de inicio para curvas cerradas.
Un ejemplo de una curva discreta 3D y su cadena correspondiente. Su cadena ya es invariante al punto de inicio.

3.3.6 Invarianza bajo la transformación de espejo

Definición 5. Espejo

La cadena del espejo de una curva discreta 3D es otra cadena (llamada la cadena espejo) cuyos elementos "1" son reemplazados por elementos "3" y viceversa.

La Figura 3.8 ilustra la transformación espejo. En la Figura 3.8 (a) el plano espejo esta alineado con el plano estándar "XY", nótese que los elementos "1" y "3" de la cadena espejo fueron cambiados. En la Figura 3.8 (b) el plano espejo esta alineado con el plano "XZ" y en (c) con el plano "YZ" respectivamente.

Teorema 4. Sean c, c_1, \dots, c_n cadenas, entonces:

- (i) $\text{mirror}(\text{mirror } c) = c$,
- (ii) si no ocurren los elementos 1 y 3 en c , entonces $\text{mirror } c = c$,
- (iii) $\text{mirror}(c_1 ++ \dots ++ c_n) = \text{mirror } c_1 ++ \dots ++ \text{mirror } c_n$,
- (iv) si c es invertible, entonces $\text{inv}(\text{mirror } c) = \text{mirror}(\text{inv } c)$.

Estas propiedades se deducen directamente de las definiciones. En términos geométricos, la condición en (ii) significa que el camino poligonal c es planar.

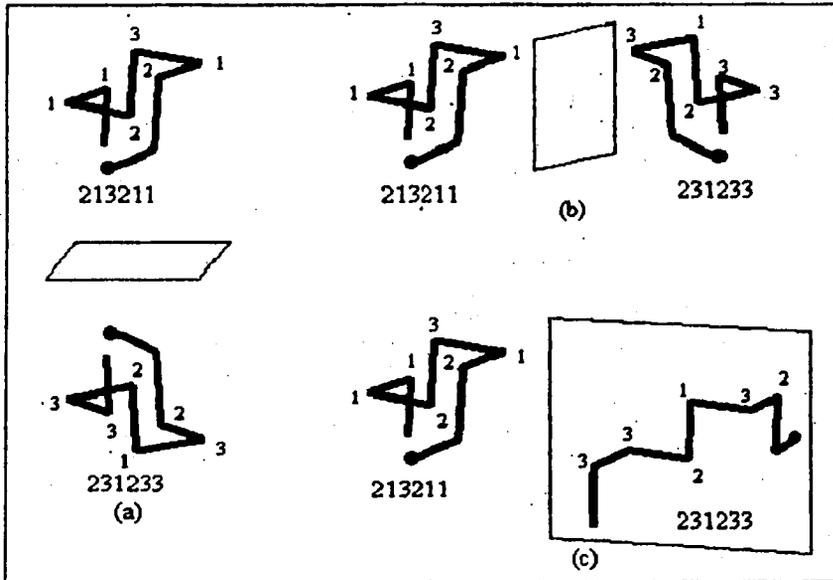


Figura 3.8 La invarianza bajo la transformación espejo: (a) el plano espejo está alineado con el plano estándar XY, (b) con el plano XZ, y (c) con el plano YZ, respectivamente.

3.3.7 Comparación de curvas

Definición 6. Isoperimetría

Dos curvas discretas en \mathbb{R}^3 son isoperimétricas si éstas tienen la misma longitud o perímetro.

Debido a que esta representación es invariante a las transformaciones de translación, rotación, inversa y al punto de inicio, se puede determinar si dos curvas isoperimétricas tienen la misma forma si sus cadenas son iguales.

También es posible decidir cuando una cierta forma local ocurre dentro de otra forma al comparar sus cadenas o partes de ellas.

3.3.8 Operación de ligado

Supongamos que se nos dan dos caminos poligonales con sus correspondientes cadenas, como se ilustra en la Figura 3.9 (a). De estos caminos poligonales, se puede obtener uno nuevo al conectarlos por medio de cero o mas lados adicionales, de tal forma que el camino poligonal resultante pueda ser representado por una cadena. Las figuras 3.9 (b) y 3.9 (c) ilustran esta operación geométrica. En ese ejemplo, se liga un camino poligonal $a = 1$ con un camino poligonal $b = 2$ al añadir un lado de ligado.

Una vez que la conexión esta realizada, para obtener la cadena del nuevo camino poligonal, se deben etiquetar los nuevos vértices intermedios, como se muestra en la Figura 3.9, obteniendo 10442. Por lo tanto, en términos de cadenas, la concatenación $a ++ 044 ++ b$ corresponde a la operación geométrica. Por si misma, 044 es una cadena, pero observe que ésta no es la cadena del lado agregado, ésta describe la forma en la cual a y b fueron ligadas; bajo esta interpretación, se dirá que 044 es una liga. Nótese que una liga debe consistir de al menos dos elementos, uno para etiquetar el último vértice de a y otro para etiquetar el segundo vértice de b . Este último elemento no puede ser 0 porque éste etiqueta a un vértice de ángulo recto. Formalizando estas observaciones, se obtienen las siguientes definiciones.

Definición 7. Liga

Una liga es una cadena con al menos dos etiquetas y que no termina en 0. Sean a y b cadenas y sea l una liga. La l -liga de a y b es la cadena $a ++ l ++ b$.

Teorema 5. Sean a y b cadenas invertibles y sea l una liga. Entonces $inv(a ++ l ++ b) = inv b ++ inv l ++ inv a$

Demostración.

Como l es una cadena invertible, la igualdad se deduce del teorema 3. ■

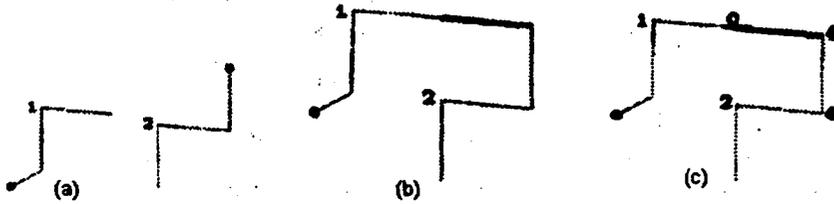


Figura 3.9 Operación de ligado.

3.3.9 Comparación entre los códigos de cadenas de Freeman y de Bribiesca

La principal diferencia entre la cadena de códigos de Freeman y la de Bribiesca es que las cadenas de Freeman usan direcciones absolutas y las de Bribiesca usan cambios de dirección relativos.

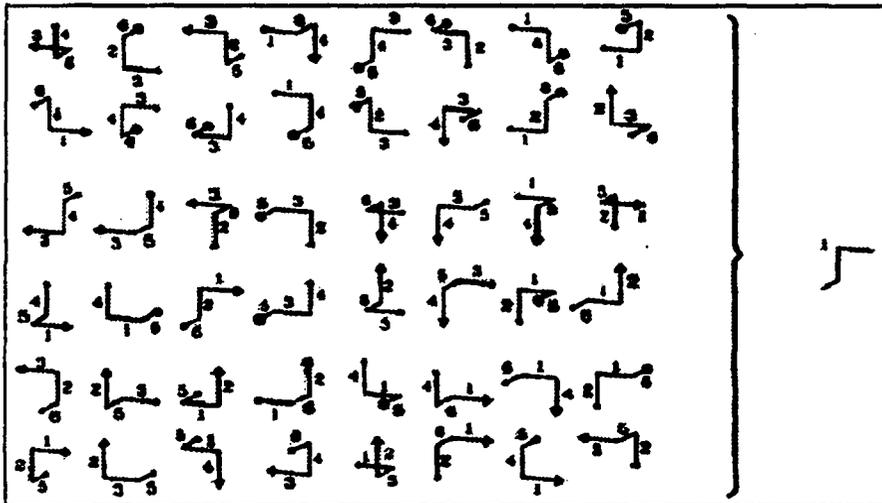


Figura 3.10 Del lado izquierdo las 46 cadenas distintas en código de Freeman que representan a la curva del lado derecho. Del lado derecho, la curva representada por un solo elemento en el código de Bribiesca.

TESIS CON
FALLA DE ORIGEN

Esto da como consecuencia que la propuesta de Freeman no sea invariante a la transformación de rotación mientras que la de Bribiesca si lo es. La Figura 3.10 muestra una curva discreta 3D y sus 48 posibles cadenas de Freeman dependiendo de las 24 posibles rotaciones y sus cadenas inversas. Por el otro lado, el código de cadenas de Bribiesca representa la misma curva con sólo un elemento.

Resumiendo, el código de Bribiesca utiliza únicamente 5 elementos para representar cualquier curva tridimensional discreta, es preservante de la información, genera descripciones únicas, cuenta con un rico soporte local, extensión suave y propagación. Además es invariante a las transformaciones de translación y rotación. También, si se desea puede hacerse invariante a la transformación espejo y al punto de inicio. Estas características hacen que éste código sea una alternativa adecuada para realizar correspondencias de curvas.

Aquí termina la revisión de los fundamentos. En los siguientes capítulos se presenta la propuesta para medir la similitud entre dos curvas 3D representadas con el código de cadenas de cambios de dirección ortogonal.

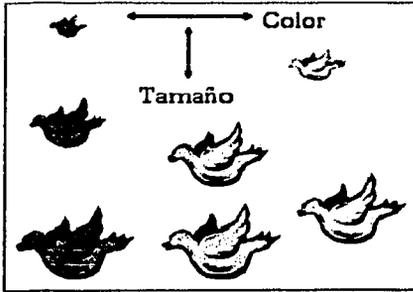


Ilustración I.1. Aproximación geométrica.

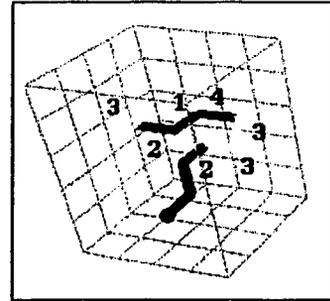


Ilustración I.3. Un ejemplo de una curva 3D discreta: 2334123.

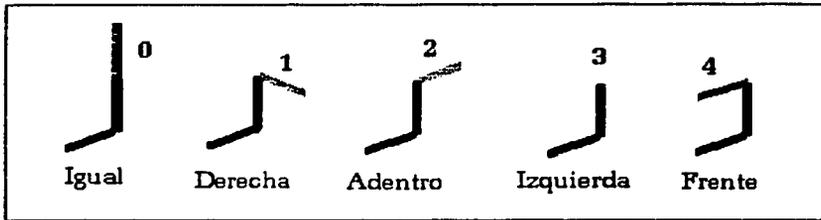


Ilustración I.2. Los cinco cambios de dirección posibles para representar curvas discretas 3D.

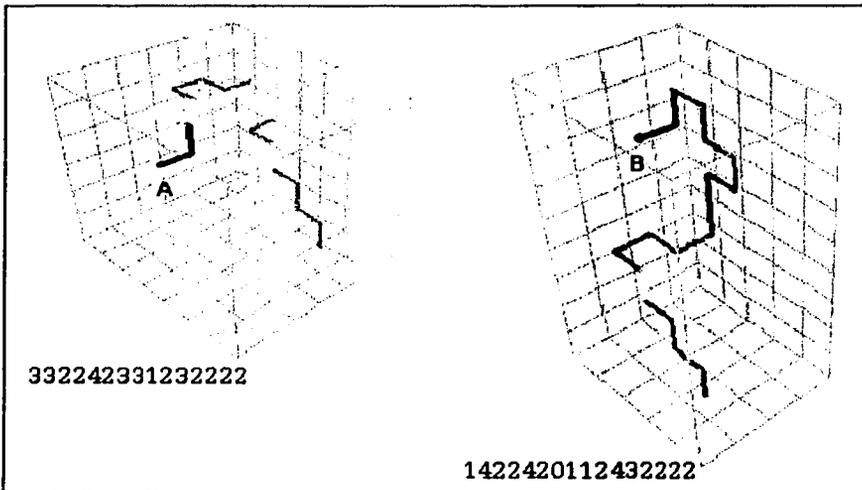


Ilustración I.4. Dos curvas tridimensionales con subcurvas en común, mostradas en las ilustraciones I.5 y I.6.

TESIS CON FALLA DE ORIGEN

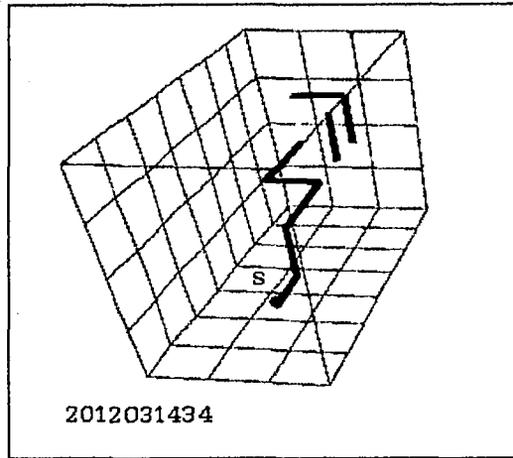


Ilustración I.8. Una curva tridimensional (s).

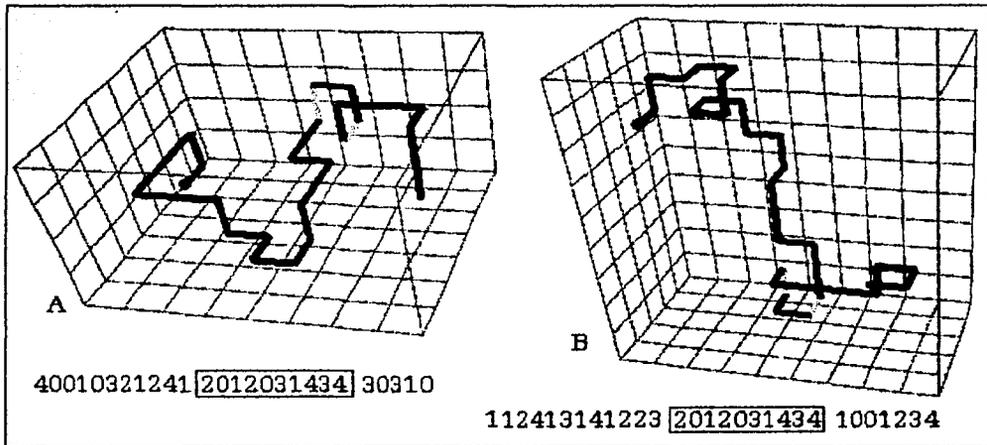


Ilustración I.9. Dos curvas distintas A y B que contienen a la misma subcurva 2012031434, I.8 (s).

TESIS CON
BALLA DE ORIGEN

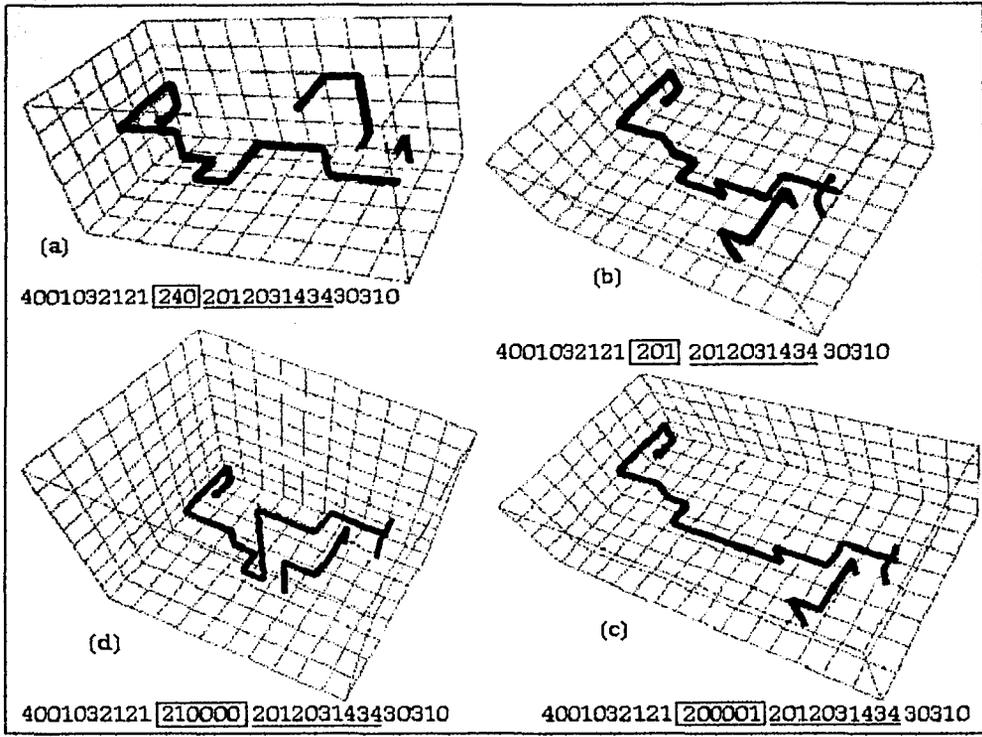


Ilustración I.10. Cuatro curvas con el patrón s, en donde sus dos elementos anteriores contienen elementos cero.

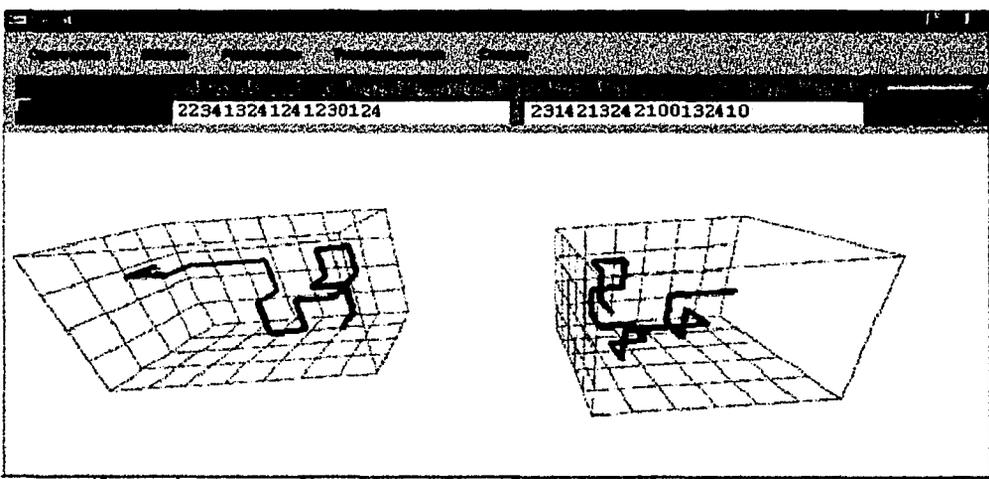


Ilustración I.11. Aplicación escrita en C++, OpenGL y Glow que visualiza curvas 3D y mide su similitud.

**TESIS CON
FALLA DE ORIGEN**

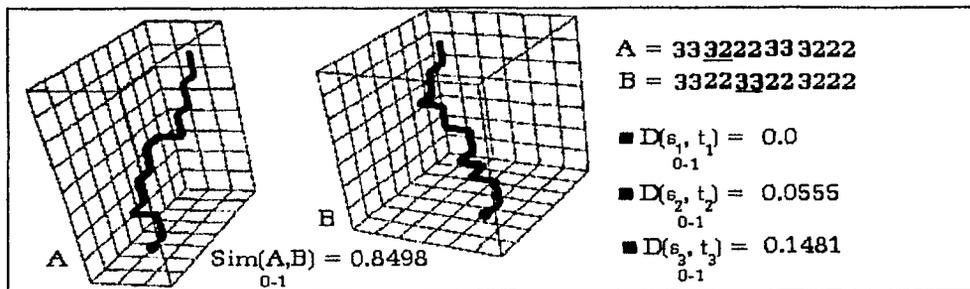


Ilustración I.12. Similitud de dos curvas (A y B) con cadenas muy parecidas.

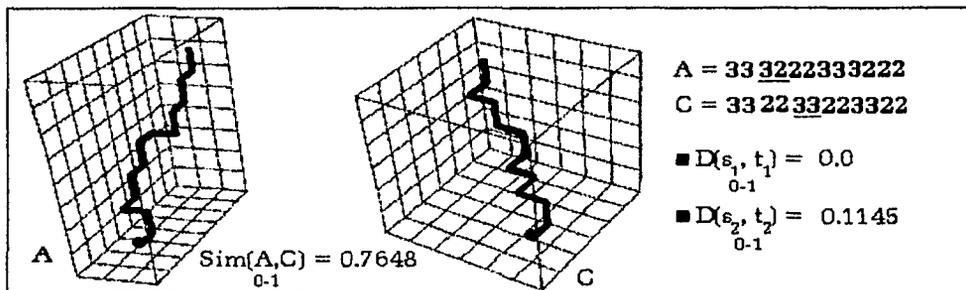


Ilustración I.13. Similitud de dos curvas (A y C) con cadenas muy parecidas.

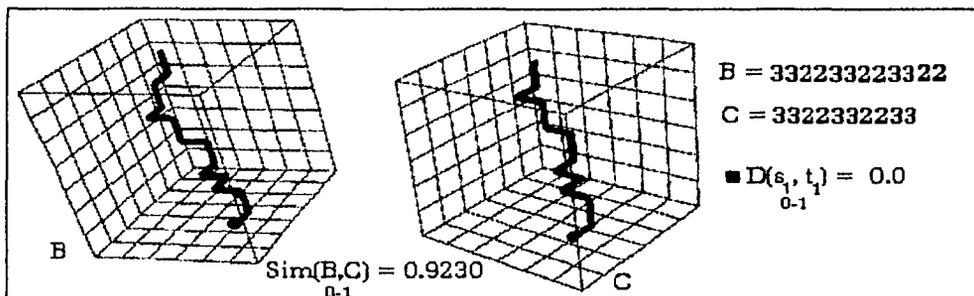


Ilustración I.14. Similitud de dos curvas (B y C) con cadenas muy parecidas.

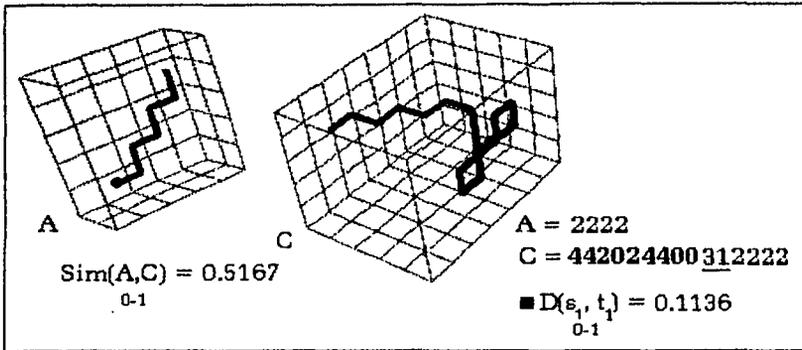


Ilustración I.15. Similitud de dos curvas (A y C) con correspondencias parciales.

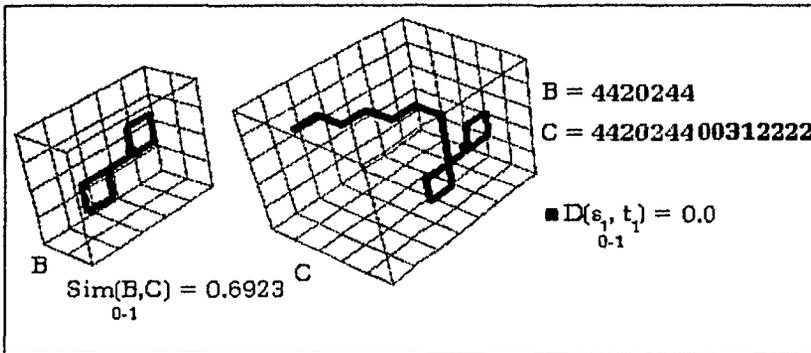


Ilustración I.16. Similitud de dos curvas (B y C) con correspondencias parciales.

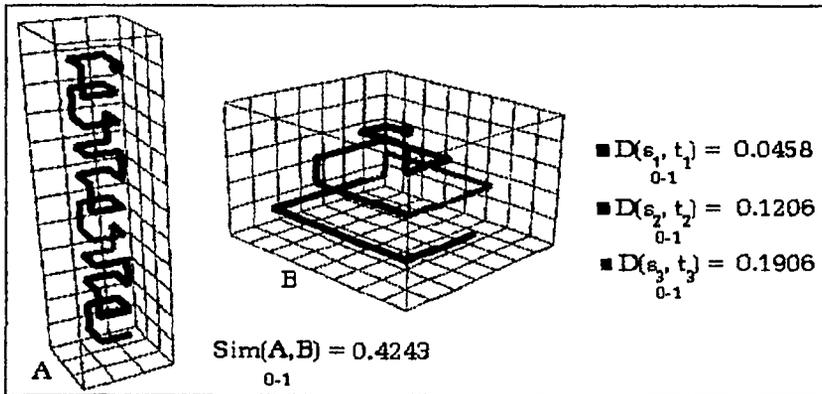


Ilustración I.17. Similitud de dos curvas (A y B) en forma de espiral.

TESIS CON
 FALLA DE ORIGEN

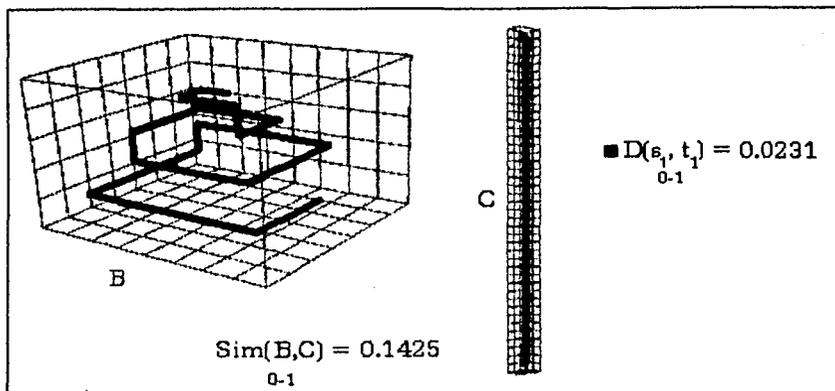


Ilustración I.18. Similitud entre dos curvas, B y C.

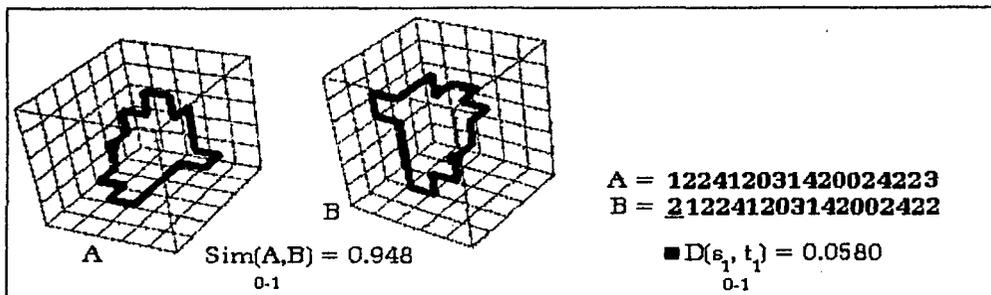


Ilustración I.19. Similitud de dos curvas cerradas (A y B) de longitud 18.

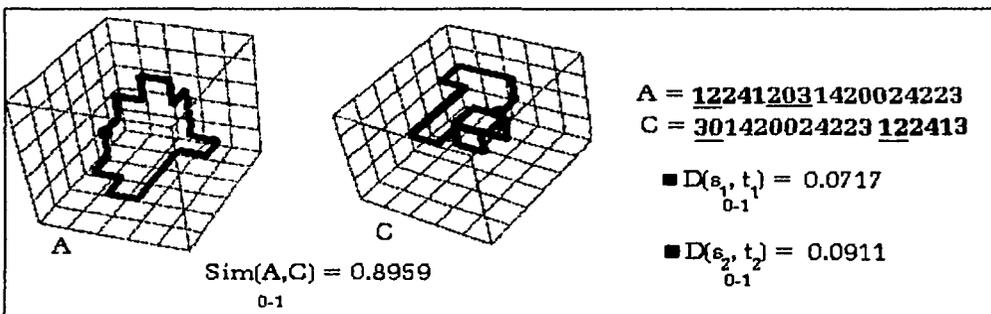


Ilustración I.20. Similitud de dos curvas cerradas (A y C) de longitud 18.

**TESIS CON
 FALLA DE ORIGEN**

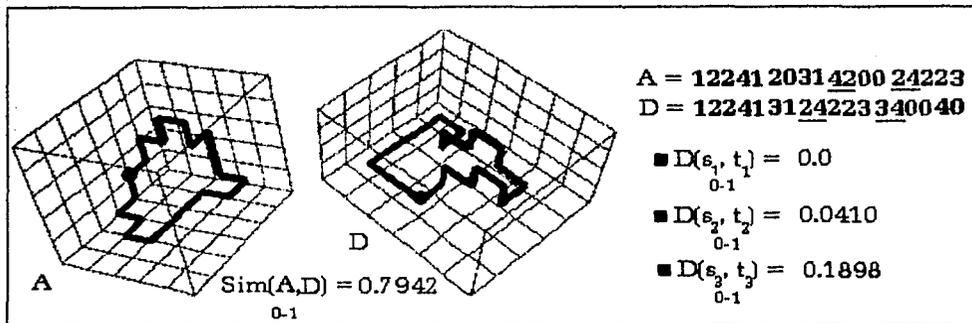


Ilustración I.21. Similitud de dos curvas cerradas (A y D) de longitud 18.

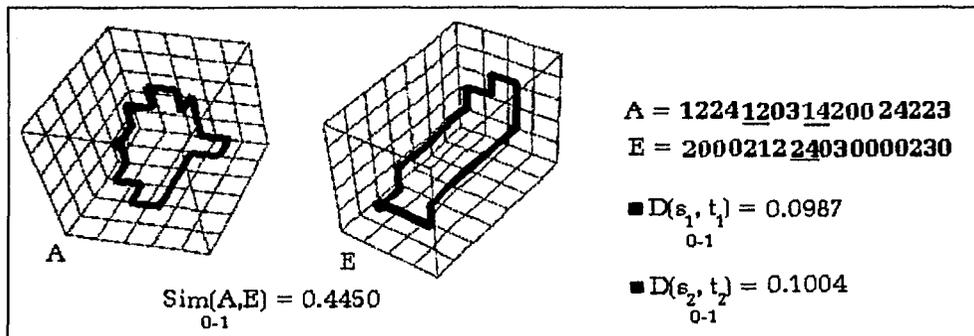


Ilustración I.22. Similitud de dos curvas cerradas (A y E) de longitud 18.

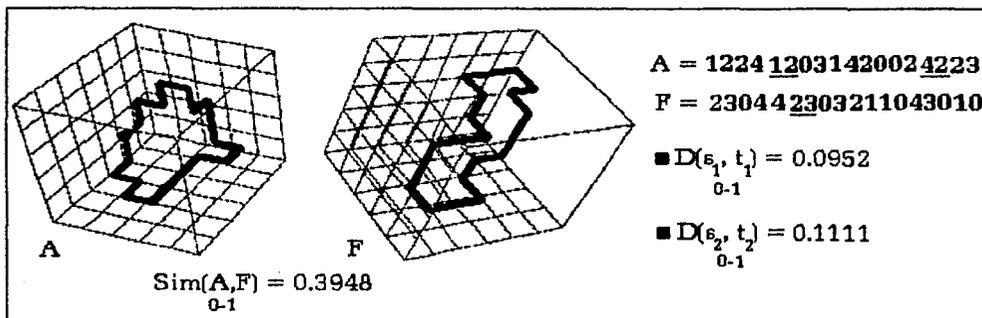


Ilustración I.23. Similitud de dos curvas cerradas (A y F) de longitud 18.

TESIS CON
 FALLA DE ORIGEN

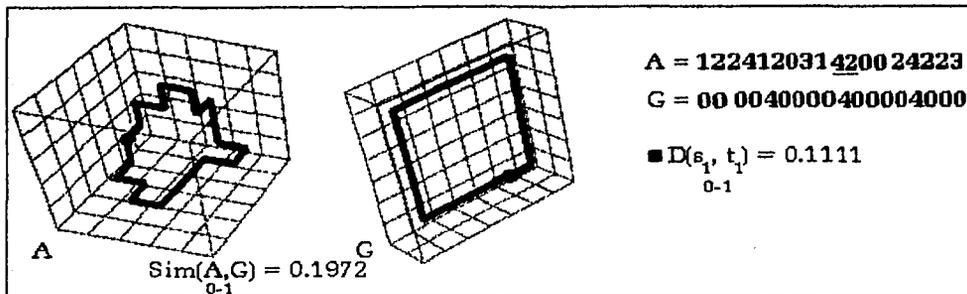


Ilustración I.24. Similitud de dos curvas cerradas (A y G) de longitud 18.

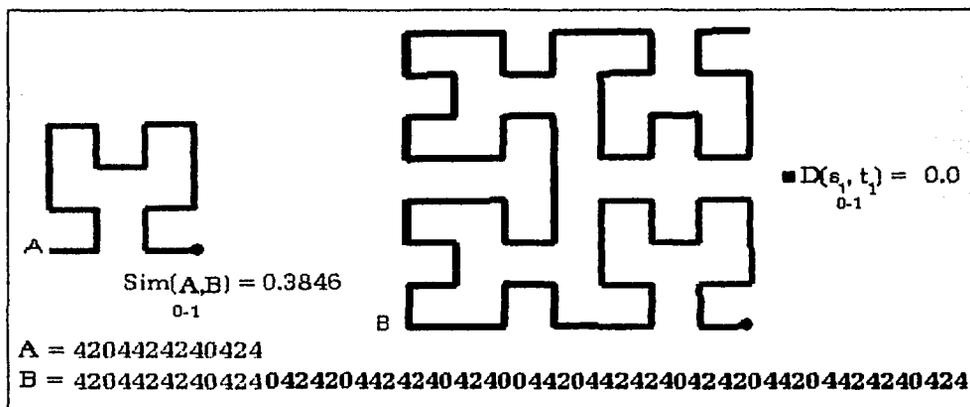


Ilustración I.25. Similitud de las curvas de Peano para los niveles 1 y 2.

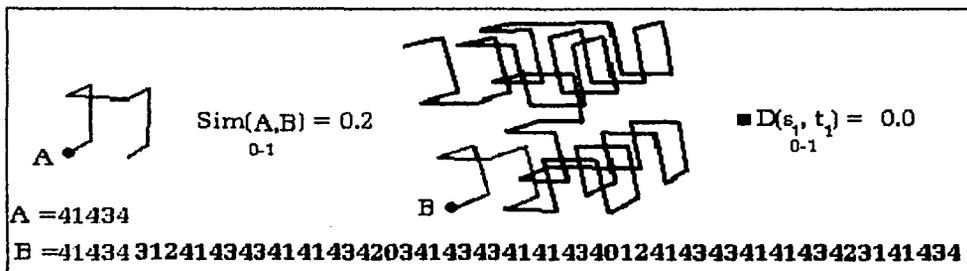


Ilustración I.26. Similitud entre las curvas de Hilbert para el nivel 0 y 1.

PARTE II

DESARROLLO

En la parte II se presenta el desarrollo de la medida de similitud entre dos curvas tridimensionales. Es una propuesta nueva que pretende dar una aportación al problema de similitud de curvas. Se da una presentación formal de la medida de similitud y en seguida se presentan los algoritmos para calcularla. Finalmente se presentan los resultados de aplicar esta medida a diferentes conjuntos de curvas.

Esta compuesta de 3 capítulos:

Capítulo 4. Presentación teórica de la medida de similitud.

Capítulo 5. Implementación.

Capítulo 6. Resultados.

CAPITULO 4

PRESENTACIÓN TEÓRICA DE LA MEDIDA DE SIMILITUD

La medida de similitud entre dos curvas 3D propuesta en la presente tesis, está basada en que las curvas se encuentran representadas por un código de cadenas de cambio de dirección ortogonal como se explicó en el capítulo 3. Además supone que las curvas han sido generadas con la misma escala.

De aquí en adelante, el término cadena hará referencia a una curva tridimensional representada con dicho código, y subcadena hará referencia a una subcurva.

Adicionalmente a las definiciones dadas en (Bribiesca, 2000) y en (Bribiesca y Velarde, 2001) que se presentan en el capítulo 3, definiré los siguientes conceptos.

4.1 DEFINICIONES

Definición 4.1: Medida de disimilitud

Una medida de disimilitud es una función que asocia a cada pareja de secuencias (por ejemplo, dos cadenas) con un valor numérico, con la idea de que entre más grande sea su valor más grande es la disimilitud. Normalmente las medidas de disimilitud son métricas.

Definición 4.2: Medida de similitud

Una medida de similitud es una función que asocia a cada pareja de secuencias (por ejemplo, dos cadenas) con un valor numérico, con la idea de que entre más grande sea su valor más grande es su similitud (lo contrario que la disimilitud).

Definición 4.3: Similitud entre dos curvas tridimensionales

Intuitivamente dos curvas tridimensionales son más parecidas entre más subcurvas tengan en común (asociación uno a uno) y entre más parecidas sean las orientaciones y posiciones dentro de las curvas.

Una subcurva en común tiene la misma orientación si se encuentra rotada en la misma dirección y tiene la misma posición si inicia en el mismo vértice, en ambas curvas.

Por ejemplo, consideremos las curvas de la Ilustración I.4. Ambas curvas contienen a las subcurvas (a) y (b) de la Ilustración I.5.

Sin embargo, sus orientaciones y posiciones son distintas. Para el caso de la subcurva 1.5 (a), la posición es la misma en las dos curvas, pero su orientación es distinta. Para el caso de la subcurva 1.5 (b), la posición es distinta (esta desplazada por un vértice), pero la orientación es la misma. Estas diferencias en posición y orientación, junto con el resto de las subcurvas en las que difieren (marcadas en color negro) hacen que las curvas *A* y *B* no sean tan parecidas.

Que las subcurvas en común tengan una asociación uno a uno, significa que a la subcurva que se encuentra en la curva 1 se le asocia sólo con una subcurva en la curva 2. Esto es debido a que no importa cuantas veces se repita la subcurva en la curva 2, ésto no influye en que las curvas sean mas parecidas.

Por ejemplo, considere la misma curva *A* del ejemplo anterior y la curva *C* de la Ilustración 1.7. En este caso, la subcurva (en morado) se encuentra una sola vez en *A* mientras que se repite 4 veces en *C*. El hecho de que se repita la subcurva mas veces en la curva *C*, no hace que *A* y *C* se parezcan mas, por el contrario, al elegir cual de las repeticiones de *C* se asociará con la subcurva de *A*, provocará que el resto de las repeticiones queden sin una asociación y esto hace que las curvas sean más distintas.

El punto de interés en este caso, es decidir con cual de todas las repeticiones se hará la asociación, es decir, elegir la mejor asociación basándose en la disimilitud de las subcurvas.

Definición 4.4: Correspondencia de una subcurva común a dos curvas

Se define la correspondencia de una subcurva común a dos curvas como: dada una subcurva que se encuentra tanto en la curva 1 como en la curva 2, calcular su disimilitud.

Las siguientes definiciones hacen referencia al código de cadenas de cambio de dirección ortogonal y se usan en el resto del capítulo.

Definición 4.5: Subcadena

Sea *A* una cadena:

$$A = a_1 a_2 \dots a_n = \{a_i : 1 \leq i \leq n\}$$

Se define una subcadena *S* de *A* con índice inicial *i* como:

$$S = a_i a_{i+1} \dots a_m \text{ tal que } 1 \leq i \leq n \text{ y } m \leq n$$

Además se define la función *sub* como:

$$sub : CH \times N \times N \rightarrow CH$$

$$(a_1 \dots a_i a_{i+1} \dots a_j \dots a_n, i, j) \mapsto a_i \dots a_j$$

donde, *CH* es el conjunto de las cadenas y *N* es el conjunto de los números naturales.

Por ejemplo:

Si *A* = 02340123241002

entonces *sub*(*A*, 3, 6) = 3401, la cual es una subcadena de *A* con índice inicial 3.

Definición 4.6: Subcadena máximaSean A y B dos cadenas:

$$A = a_1 a_2 \cdots a_n$$

$$B = b_1 b_2 \cdots b_m$$

Se define una subcadena máxima S de A y B con índices iniciales q y r

$$1 \leq q \leq n$$

$$1 \leq r \leq m$$

como:

$$S = s_1 s_2 \cdots s_l \text{ tal que:}$$

$$i) \quad 1 \leq l \leq \min(n, m)$$

ii) S es una subcadena tanto de A , con índice inicial q , como de B con índice inicial r .

$$iii) \quad a_{q-1} \neq b_{r-1} \quad \text{y} \quad a_{q+L(S)} \neq b_{r+L(S)}$$

Por ejemplo:

Si $A = 01231201300120321$ y $B = 0124201312400$ Entonces $S = 2013$ es una subcadena máxima de A y B con índices iniciales 6 y 5.**Definición 4.7: Función de maximización por la izquierda**

Sean

$$A = a_1 a_2 \cdots a_m$$

$$B = b_1 b_2 \cdots b_n$$

dos cadenas.

Sea u una subcadena tanto de A , con índice inicial i , como de B con índice inicial j .

De tal forma que:

$$A = a_1 a_2 \cdots a_i a_{i+1} \cdots a_{i+L(u)-1} \cdots a_m \text{ tal que } a_i = u_1, a_{i+1} = u_2, \dots, a_{i+L(u)-1} = u_{L(u)}$$

$$B = b_1 b_2 \cdots b_j b_{j+1} \cdots b_{j+L(u)-1} \cdots b_n \text{ tal que } b_j = u_1, b_{j+1} = u_2, \dots, b_{j+L(u)-1} = u_{L(u)}$$

Se define la función $lmax$ como:

$$lmax: (CH \times CH \times CH) \rightarrow CH$$

$$lmax(A, B, u) = a_k a_{k+1} \cdots a_i a_{i+1} \cdots a_{i+L(u)-1}$$

$$= b_k b_{k+1} \cdots b_j b_{j+1} \cdots b_{j+L(u)-1} = u_{lmax}$$

donde u_{lmax} es una subcadena máxima de A y B .En otras palabras, la función $lmax$ hace crecer a la subcadena u por la izquierda hasta que u sea una subcadena máxima de A y B .Se le llama al índice k , índice de maximización por la izquierda en A , y al índice k' , índice de maximización por la izquierda en B .

Con estas definiciones en mente, en las siguientes secciones se presenta cómo se reconocen las subcurvas que tienen en común dos curvas, cómo se realiza su correspondencia (calcular su disimilitud), cómo se hace la asociación uno a uno y finalmente cómo se mide la similitud entre ambas curvas.

4.2 RECONOCIMIENTO Y DISIMILITUD DE SUBCURVAS

Para poder medir la similitud global entre dos curvas, se requiere localizar y medir la similitud de aquellas partes que tienen en común ambas curvas, es decir, reconocer y medir de manera local qué tanto se parecen dos subcurvas en común.

En la sección 4.2.1 se tratará primeramente el problema de reconocer las subcurvas en común y en la sección 4.2.2 el de disimilitud.

4.2.1 Reconocimiento de las subcurvas que tienen en común dos curvas

Partiendo del hecho de que la representación de las curvas es por medio de un código de cadenas de cambio de dirección ortogonal, podemos asegurar que cualquier curva dada se encuentra representada de manera única. Esto es cierto siempre y cuando la cadena se haya hecho invariante al punto de inicio, como se muestra en el capítulo 3 sección 3.3. Por otro lado, esta representación es invariante a las transformaciones de translación y rotación, lo cual implica que no importa en que orientación y posición se encuentre la curva, su representación siempre será la misma.

Estas propiedades, también se cumplen cuando se habla de subcurvas. Por ejemplo, considérese la curva 2012031434 que se muestra en la Ilustración I.8.

Si esta curva se encuentra ahora como una subcurva de cualquier otra curva, su representación será la misma sin importar la posición ni la orientación en la que se encuentre, como se muestra en la Ilustración I.9.

Con respecto a la posición de la subcurva, ésta está dada por los índices en donde inicia la subcadena. Por ejemplo, en la curva *A* la subcadena inicia en el índice 12 y en el índice 13 en la curva *B*.

Los dos elementos que le preceden al patrón 2012031434 (41 para el caso de la curva *A* y 23 para el caso de la curva *B*) son los elementos ortogonales necesarios para poder dar la orientación a la subcurva. Recordemos que cada elemento representa un cambio de dirección relativo a los dos elementos ortogonales anteriores. Estos dos elementos se encuentran en negro en la curva *s* y en morado en las curvas *A* y *B*.

Como los dos elementos anteriores deben indicar un cambio de dirección ortogonal, éstos deben ser distintos de cero. Las curvas (a) y (b) de la Ilustración I.10 son ejemplos de casos en que es cero el primer o el segundo elemento predecesor y las curvas (c) y (d) de casos en que se encuentran intercalados valores de cero entre los dos elementos ortogonales. Notemos que las subcurvas ya no son idénticas a *s*, ya que su inicio difiere un poco, lo cual hay que tenerlo presente cuando se mida su disimilitud.

Debido a que los dos elementos ortogonales predecesores a la subcadena dependen a su vez de sus propios dos elementos ortogonales predecesores, la orientación de la subcurva es afectada por todos los elementos que le preceden, esto se debe nuevamente al hecho de que los cambios de dirección son relativos.

En conclusión, dadas dos curvas invariantes al punto de origen, se puede reconocer a una subcurva que se encuentra tanto en la curva 1 como en la curva 2 porque tiene los mismos elementos como cadena mas dos elementos predecesores (distintos de 0) que definen un cambio de dirección ortogonal.

Con esto, el problema se reduce a resolver correspondencia de cadenas (string matching). Sin embargo, no se desea encontrar a todas las subcadenas que tengan en común dos cadenas, si no que se desea encontrar a todas las subcadenas de *mayor longitud* que se encuentren tanto en la cadena 1 como en la cadena 2.

Debido a que se pide que las cadenas sean invariantes al punto de inicio, la similitud entre A e $inv A$ será la misma (donde A es una cadena). Ésta es una propiedad importante, ya que como se mostró en el capítulo 3, A y su inversa tienen la misma forma.

Para el caso de la transformación de espejo, su interpretación es dependiente de la aplicación. Por ejemplo, si se consideran a las curvas 3D como trayectorias, entonces el espejo de una cierta curva diría que todos los movimientos que se hicieron a la izquierda ahora son a la derecha y viceversa. Desde este punto de vista las dos trayectorias serían muy distintas. Sin embargo, si las curvas representan la silueta de una forma 3D, la transformación espejo generaría simetrías, lo cual hace que las curvas sean mas parecidas.

Con esto se quiere decir que dependiendo del problema que se quiera resolver, una curva 3D y su correspondiente espejo, serán más parecidas o más distintas. Por estas razones se le considera como un caso especial y por lo tanto no será considerada en la medida de similitud.

4.2.2 Disimilitud de una subcurva común a dos curvas

De acuerdo a la definición 4.3, dos curvas son más parecidas entre más subcurvas tengan en común (asociación uno a uno) y entre más parecidas sean las orientaciones y posiciones de éstas. La forma en que se detectan las subcurvas en común se acaba de presentar en la sección anterior y su posición está dada por sus índices de inicio dentro de las cadenas. Por lo tanto, lo único que resta definir es la manera en que se calcula su orientación. Para esto se definirá la *dirección acumulada*, la cual calcula la orientación final de la subcurva después de haber sido afectada por todos sus elementos predecesores. Finalmente, haciendo uso de estas definiciones, se definirá la disimilitud de una subcurva común a dos curvas.

4.2.2.1 Dirección acumulada

La dirección acumulada calcula la orientación final de una subcurva después de haber sido afectada por todos sus elementos predecesores, es decir, calcula los dos vectores de dirección (u y d) que son la base de referencia para poder definir el siguiente elemento.

Para el primer elemento de una cadena, las dos direcciones ortogonales previas son dadas arbitrariamente. Por ejemplo, $u'=(0,1,0)$ y $d'=(-1,0,0)$, o cualesquiera dos vectores unitarios y ortogonales, como se muestra en la Figura 4.1. Además de estos dos vectores se requiere agregar dos elementos auxiliares (distintos de cero) cualesquiera antes de que inicie la cadena para poder usarlos como referencia del primer elemento, por ejemplo podemos agregar dos elementos 3.

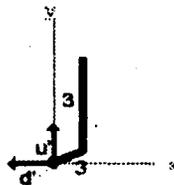


Figura 4.1 Los dos primeros vectores de dirección u' y d' , y dos elementos ortogonales cualesquiera necesarios para empezar a definir una cadena (33).

Dependiendo del elemento que se quiera agregar a la cadena, los nuevos vectores de dirección u y d se calculan en términos de los vectores de dirección predecesores u' y d' , de acuerdo a lo siguiente y como se muestra en la Figura 4.2.

Elemento 0	$u = u'$	$d = d'$
Elemento 1	$u = d'$	$d = u' \times d'$
Elemento 2	$u = d'$	$d = u'$
Elemento 3	$u = d'$	$d = -(u' \times d')$
Elemento 4	$u = d'$	$d = -u$

Donde $u' \times d'$ es el producto exterior usual de los vectores u' y d' .

De esta forma, la orientación final de la subcurva esta dada por los vectores finales u y d resultantes del cálculo de los vectores de dirección por cada elemento que compone a la subcadena.

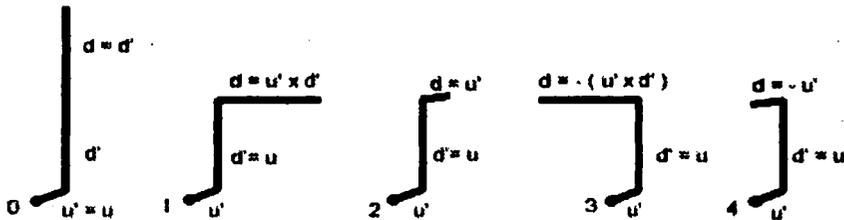


Figura 4.2 Muestra como calcular los nuevos vectores de dirección u y d (en términos de los dos vectores de dirección previos u' y d') dependiendo del último elemento de la cadena que se agregó.

Por ejemplo, tomemos a la cadena $A = 01234$, su dirección acumulada se calcula de la siguiente manera.

- 1) Para iniciar se dan arbitrariamente dos vectores unitarios y ortogonales.
 $u' = (0,1,0)$ y $d' = (-1,0,0)$
- 2) Se agrega el elemento auxiliar 3:
 $u = d' = (-1,0,0)$ y $d = -(u' \times d') = (0,0,-1)$
- 3) Se agrega el elemento auxiliar 3:
 $u' = (-1,0,0)$ y $d' = (0,0,-1)$
 $u = d' = (0,0,-1)$ y $d = -(u' \times d') = (0,1,0)$
- 4) Se agrega el elemento 0, y ahora:
 $u' = (0,0,-1)$ y $d' = (0,1,0)$
 $u = u' = (0,0,-1)$ y $d = d' = (0,1,0)$
- 5) Se agrega el elemento 1, y ahora:
 $u' = (0,0,-1)$ y $d' = (0,1,0)$
 $u = d' = (0,1,0)$ y $d = u' \times d' = (1,0,0)$
- 6) Se agrega el elemento 2, y ahora:
 $u' = (0,1,0)$ y $d' = (1,0,0)$
 $u = d' = (1,0,0)$ y $d = u' = (0,1,0)$
- 7) Se agrega el elemento 3, y ahora:
 $u' = (1,0,0)$ y $d' = (0,1,0)$
 $u = d' = (0,1,0)$ y $d = -(u' \times d') = (0,0,-1)$
- 8) Se agrega el elemento 4, y ahora:
 $u' = (0,1,0)$ y $d' = (0,0,-1)$
 $u = d' = (0,0,-1)$ y $d = -u' = (0,-1,0)$

Por lo tanto la dirección acumulada de la cadena 01234 es $u = (0,0,-1)$ y $d = (0,-1,0)$.

4.2.2.2 Pseudométrica de Dirección Acumulada

Definición 4.8: Pseudométrica de Dirección Acumulada

Sean A, B y C dos cadenas con direcciones acumuladas (u, d) , (x, w) y (v, z) respectivamente.

Se define la función de dirección acumulada como:

$$\Delta(A, B) = \begin{cases} 0 & \text{si la dirección acumulada de } A \text{ y } B \text{ es la misma } (u = x \text{ y } d = w) \\ 1 & \text{en otro caso} \end{cases} \quad (4.1)$$

La dirección acumulada es una pseudométrica.

Demostración.

i) Positividad. Por demostrar que $\Delta(A, B) \geq 0$.

Por definición $\Delta(A, B) \geq 0$.

ii) Simetría. Por demostrar que $\Delta(A, B) = \Delta(B, A)$.

Si $\Delta(A, B) = 0$,

entonces $u = x$ y $d = w$, que es lo mismo que $x = u$ y $w = d$. Por lo tanto, $\Delta(A, B) = \Delta(B, A)$

Si $\Delta(A, B) = 1$,

entonces $u \neq x$ o $d \neq w$, que es lo mismo que $x \neq u$ o $w \neq d$. Por lo tanto, $\Delta(A, B) = \Delta(B, A)$.

iii) Identidad. Por demostrar que $\Delta(A, A) = 0$.

Si la dirección acumulada de A es u y d , entonces $u = u$ y $d = d$. Por lo tanto $\Delta(A, A) = 0$.

iv) Desigualdad del triángulo. Por demostrar que $\Delta(A, C) \leq \Delta(A, B) + \Delta(B, C)$.

Supongamos que la desigualdad del triángulo no se cumple:

$$\Delta(A, C) > \Delta(A, B) + \Delta(B, C)$$

Si $\Delta(A, C) = 0$,

entonces (por definición) $\Delta(A, B) + \Delta(B, C)$ no puede tomar un valor mas chico que 0. Por lo tanto, la pseudométrica de dirección acumulada entre A y C no puede ser cero.

Si $\Delta(A, C) = 1$,

entonces $u \neq v$ y/o $d \neq z$, y $\Delta(A, B) + \Delta(B, C) = 0$.

Entonces, $\Delta(A, B) = 0$ y $\Delta(B, C) = 0$.

Por lo tanto, $u = x$, $d = w$, $x = v$ y $w = z$,

es decir, $u = x = v$ y $d = w = z$. Lo cual es una contradicción.

Por lo tanto, al cumplir estas cuatro propiedades, se tiene que la función de dirección acumulada es una pseudométrica. ■

Notemos que no se cumple la propiedad de unicidad porque existen cadenas distintas cuya dirección acumulada es la misma, y por lo tanto no es una métrica. Un ejemplo de dos cadenas distintas cuya dirección acumulada es la misma se muestra en la Figura 4.3.



Figura 4.3 Dos cadenas distintas con la misma dirección acumulada.

4.2.2.3 Disimilitud entre dos subcadenas de una pareja común máxima

Definición 4.9: Pareja común máxima

Sean A y B dos cadenas.

$$A = a_1 a_2 \cdots a_m$$

$$B = b_1 b_2 \cdots b_n$$

Sea P' una subcadena de A y de B con índices iniciales i_1 e i_2 respectivamente y $L(P') = k$, para $2 \leq k \leq \min(m, n)$. Nótese que se pide que la longitud de la subcadena sea por lo menos igual a dos.

$$P' = p'_1 p'_2 \cdots p'_k$$

Sea $P = lmax(A, B, P')$, con índice de maximización por la izquierda en A i_1 e índice de maximización por la izquierda en B i_2 .

$$\begin{aligned}
 P &= p_1 p_2 \cdots p_k \\
 &= a_n a_{n+1} \cdots a_{L(P)+n-1} \\
 &= b_{i_2} b_{i_2+1} \cdots b_{L(P)+i_2-1}
 \end{aligned}$$

Sea s una subcadena de A con índice inicial 1 y $L(s) = i_1 + L(P) - 1$.

$$s = a_1 a_2 \cdots a_n a_{n+1} \cdots a_{n+L(P)-1} = a_1 a_2 \cdots p_1 p_2 \cdots p_k,$$

Sea t una subcadena de B con índice inicial 1 y $L(t) = i_2 + L(P) - 1$.

$$t = b_{i_2} b_{i_2+1} \cdots b_{i_2+L(P)-1} = b_{i_2} b_{i_2+1} \cdots p_1 p_2 \cdots p_k$$

Por conveniencia se le llamará a la pareja (s, t) pareja común máxima.

Definición 4.10: Distancia entre dos subcadenas de una pareja común máxima.
 Se define la distancia entre dos subcadenas de una pareja común máxima como:

$$D(s, t) = \begin{cases} 0 & \text{si } s = t \\ \left[\frac{|L(s) - L(t)| + \left(\frac{L(s) + L(t)}{L(P)} - 2 \right)}{\frac{3 \max(L(s), L(t))}{2} - 3} \right] + \Delta(s, t) + |ne_s - ne_t| & \text{si } s \neq t \end{cases} \quad (4.2)$$

donde,

$|L(s) - L(t)|$: Mide el desplazamiento de las dos subcadenas, de tal forma que si la correspondencia en posición es exacta éste parámetro se hace 0.

$\left(\frac{L(s)}{L(P)} + \frac{L(t)}{L(P)} - 2 \right)$: Mide qué tan grandes son las subcadenas con respecto al segmento de curva en el cual están contenidas.

$\Delta(s, t)$: Es la pseudométrica de dirección acumulada de s y t . En otras palabras, mide si la orientación final de las subcurvas s y t es la misma, en cuyo caso éste término se hace cero.

$|ne_s - ne_t|$: Mide el número de elementos predecesores a P en s y t necesarios para definir un cambio de dirección ortogonal como se explica en la sección 4.2.1. Por ejemplo, si $P=431$,

$s=2301431$ y $t=140032431$, entonces $ne_s=3$ (la subcadena 301 predecesora a P que contiene a los dos primeros elementos distintos de cero: 1 y 3) y $ne_t=2$. Para el mismo patrón P , ahora supongamos que $s=431$ y $t=1431$, entonces $ne_s=2$ y $ne_t=2$, ya que hay que considerar que antes del elemento 4 existen otros dos elementos imaginarios, de la misma manera en la que se hizo la construcción de la dirección acumulada. Éste término se hace cero cuando el número de elementos necesarios para definir un cambio de dirección ortogonal es el mismo en ambas subcadenas:

$$\frac{3 \max(L(s), L(t)) - 3}{2} : \text{Acota entre 0 y 1 a la suma de } |L(s) - L(t)| + \left(\frac{L(s)}{L(P)} + \frac{L(t)}{L(P)} - 2 \right),$$

con la finalidad de darle menos importancia al desplazamiento y a la longitud de las subcurvas. Se puede ver que ésta suma toma su valor máximo cuando $L(P)=2$, y $L(s)=2$ ó $L(t)=2$, y que el valor máximo está dado por éste término. También se puede ver que su valor es cero únicamente cuando $L(s)=L(t)=L(P)=2$, lo cual implica que $s=t$ y por lo tanto la disimilitud es cero.

La medida de disimilitud entre dos subcadenas de una pareja común máxima es una métrica.

Demostración.

i) Positividad: Por demostrar que $D(s, t) \geq 0$.

Si $s=t$, entonces por definición $D(s, t) = 0$

Si $s \neq t$, entonces

$$D(s, t) = \left[\frac{|L(s) - L(t)| + \left(\frac{L(s) + L(t)}{L(P)} - 2 \right)}{\frac{3 \max(L(s), L(t)) - 3}{2}} \right] + \Delta(s, t) + |ne_s - ne_t|$$

Veamos que cada término es mayor o igual a cero:

$|L(s) - L(t)| \geq 0$ por definición del valor absoluto.

Ahora, $\frac{L(s)}{L(P)} \geq 1$ y $\frac{L(t)}{L(P)} \geq 1$ por la construcción de s, t y P . Entonces

$$\left(\frac{L(s)}{L(P)} + \frac{L(t)}{L(P)} \geq 2 \right) \text{ y por lo tanto } \left(\frac{L(s)}{L(P)} + \frac{L(t)}{L(P)} - 2 \right) \geq 0$$

Como por definición $L(s) \geq 2$ y $L(t) \geq 2$, entonces se tienen las siguientes desigualdades:

$$\max(L(s), L(t)) \geq 2,$$

$$\frac{\max(L(s), L(t))}{2} \geq 1,$$

$$\frac{3 \max(L(s), L(t))}{2} \geq 3,$$

$$\frac{3 \max(L(s), L(t))}{2} - 3 \geq 0.$$

$\Delta(s, t) \geq 0$ por definición.

$|ne_s - ne_t| \geq 0$ por definición del valor absoluto.

Por lo tanto $D(s, t) \geq 0$.

ii) Simetría. Por demostrar que $D(s, t) = D(t, s)$.

Si $s = t$,

entonces $D(s, t) = 0 = D(t, s)$, por lo tanto $D(s, t) = D(t, s)$.

Si $s \neq t$

se tienen las siguientes igualdades:

$$|L(s) - L(t)| = |L(t) - L(s)|$$

$$\frac{L(s) + L(t)}{L(P)} - 2 = \frac{L(t) + L(s)}{L(P)} - 2$$

$$\frac{3 \max(L(s), L(t))}{2} - 3 = \frac{3 \max(L(t), L(s))}{2} - 3$$

$$\Delta(s, t) = \Delta(t, s)$$

$$|ne_s - ne_t| = |ne_t - ne_s|$$

Por lo tanto, $D(s, t) = D(t, s)$

iii) **Identidad.** Por demostrar que $D(s, s) = 0$.

Por definición $D(s, s) = 0$.

iv) **Unicidad.** Como ya se demostró la identidad, entonces solo resta demostrar que $D(s, t) > 0$ para $s \neq t$.

Supongamos que $D(s, t) = 0$, entonces

$$|L(s) - L(t)| = 0, \text{ lo cual implica que } L(s) = L(t)$$

$$\text{Además, } \frac{L(s) + L(t)}{L(P)} - 2 = 0$$

$$\frac{2L(s)}{L(P)} = 2, \text{ y por lo tanto } L(s) = L(P) = L(t).$$

Por la construcción de s , t y P se concluye que $s=t$ lo cual contradice la hipótesis $s \neq t$.

v) **Desigualdad del triángulo.** Por demostrar que $D(s, t) \leq D(s, r) + D(r, t)$.

Se intuye que se cumple la desigualdad del triángulo. Se probó computacionalmente para $2 \leq L(s) \leq 1000$, $2 \leq L(t) \leq 1000$ y $2 \leq L(P) \leq \min(L(s), L(t))$, sin embargo, resta dar una demostración formal.

4.2.2.4 Disimilitud acotada entre dos subcadenas de una pareja común máxima

Afirmación: $0 \leq D(s, t) \leq \max(m, n)$

Demostración.

Si $s=t$ entonces $0 = D(s, t) \leq \max(m, n)$

Si $s \neq t$ entonces

$$\text{como ya se había visto, } 0 < \frac{|L(s) - L(t)| + \left(\frac{L(s) + L(t)}{L(P)} - 2 \right)}{\frac{3 \max(L(s), L(t))}{2} - 3} \leq 1, \text{ y toma el valor}$$

de 1 cuando $\min(L(s), L(t)) = 2$ y $L(s) \neq L(t)$.

Por definición, $0 \leq \Delta(s, t) \leq 1$

Finalmente,

$$0 \leq |ne_s - ne_t| \leq (\max(m, n) - 2 + 2) - 2 = \max(m, n) - 2. \text{ Recordemos que}$$

existen dos elementos imaginarios antes del primer elemento de las subcadenas s y t . Notemos que este valor máximo se da cuando $L(P)=2$. Un ejemplo para mostrar cuando este término toma su valor máximo es si $A=12, B=0000012, P=12, s=12$ y $t=0000012$, entonces

$$|ne_s - ne_t| = |2 - 7| = 5 = \max(2, 7) - 2 = \max(m, n) - L(P) = \max(m, n) - 2.$$

Por lo tanto:

$$0 \leq D(s, t) \leq 2 + \max(m, n) - 2 = \max(m, n)$$

■

Definición 4.11 Disimilitud acotada entre dos subcadenas de una pareja común máxima

Se define la disimilitud acotada entre dos subcadenas de una pareja común máxima como:

$$D_{0-1}(s, t) = \frac{D(s, t)}{\max(m, n)} \quad (4.3)$$

de tal forma que:

$$0 \leq D_{0-1}(s, t) \leq 1$$

El término $\max(L(m), L(n)) \neq 0$ por definición de s y t .

La disimilitud acotada entre dos subcadenas de una pareja común máxima es una métrica.

Demostración.

i) Positividad. Por demostrar que $D_{0-1}(s, t) \geq 0$.

Ya se demostró que $D(s, t) \geq 0$. Además por construcción $\max(L(s), L(t)) \geq 0$.

Por lo tanto, $D_{0-1}(s, t) \geq 0$.

ii) Simetría. Por demostrar que $D_{0-1}(s, t) = D_{0-1}(t, s)$.

Como ya se demostró anteriormente $D(s, t) = D(t, s)$ y por lo tanto:

$$D_{0-1}(s, t) = \frac{D(s, t)}{\max(m, n)} = \frac{D(t, s)}{\max(n, m)} = D_{0-1}(t, s)$$

iii) Identidad. Por demostrar que $D_{0-1}(s, s) = 0$.

Por definición $D(s, s) = 0$ y por lo tanto, $D_{0-1}(s, s) = \frac{0}{\max(m, m)} = 0$.

iv) Unicidad. Como ya se demostró la identidad, entonces solo resta demostrar que

$$D_{0-1}(s, t) > 0 \text{ para } s \neq t.$$

Sabemos que $D(s, t) > 0$ para $s \neq t$ y que $\max(L(m) - L(n)) > 0$, por lo tanto:

$$\frac{D(s, t)}{\max(m, n)} > 0$$

v) Desigualdad del triángulo. Por demostrar que $D_{0-1}(r, t) \leq D_{0-1}(r, s) + D_{0-1}(s, t)$.

Se intuye que se cumple la desigualdad del triángulo. Se probó computacionalmente para $2 \leq L(s) \leq 1000$, $2 \leq L(t) \leq 1000$ y $2 \leq L(P) \leq \min(L(s), L(t))$, sin embargo, resta dar una demostración formal. ■

Ejemplo.

El siguiente es un ejemplo que muestra como calcular la disimilitud acotada entre dos subcadenas de una pareja común máxima.

Si:

A = 0123102441021 y

B = 1241200404421

Entonces: P=44

$L(P)=2$

$s = 012310244$

$L(s)=9$

$ne_s = 3$

$\Delta(s, t) = 0$

$t = 12412004044$

$L(t)=11$

$ne_t = 5$

$m = 13$

$n = 13$

$$D(s, t) = \frac{|9-11| + \left(\frac{9+11}{2} - 2\right)}{\frac{3(11)}{2} - 3} + 0 + |3-5| = \frac{20}{27} + 2$$

$$D_{0-1}(s, t) = \frac{\frac{20}{27} + 2}{13+13} = 0.105$$

Lo cual indica que las subcurvas son muy parecidas. Estas curvas se muestran en la Figura 4.4. La pareja común máxima (s,t) se muestra en color gris fuerte, e incluye a la subcadena máxima P que se muestra en negro. Notemos que la dirección acumulada de s y t es la misma. Esto lo podemos notar al observar que la subcadena máxima P tiene la misma orientación en la curva A y en la curva B .

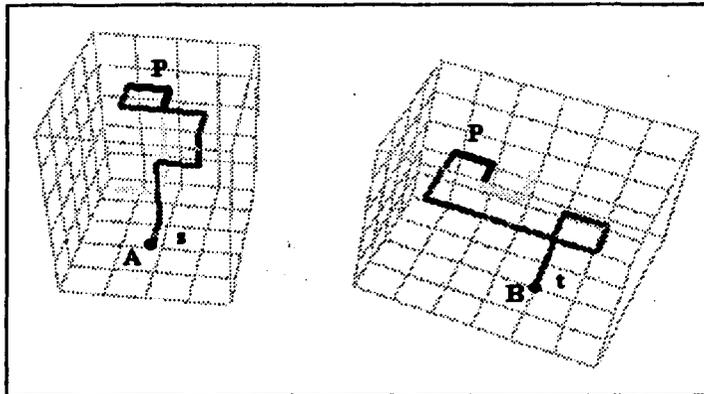


Figura 4.4 Dos curvas A y B . Una pareja común máxima (s,t) , la cual se muestra en gris fuerte. Una subcadena máxima P común a A y B , la cual se muestra en negro.

4.3 MEDIDA DE SIMILITUD ENTRE DOS CURVAS

En base a la definición 2.3, que dice que dos curvas tridimensionales son más parecidas entre más subcurvas tengan en común (asociación uno a uno) y entre más parecidas sean las orientaciones y posiciones de éstas, se define formalmente la similitud entre dos curvas.

Definición 4.12: Similitud entre dos curvas

Sean:

$$A = a_1 a_2 a_3 \dots a_m \text{ y } B = b_1 b_2 \dots b_n \text{ dos cadenas.}$$

Se define la similitud Sim entre A y B como:

$$Sim(A, B) = \sum_{j=1}^l D(s_j, t_j) + \left[m + n - \left(\sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j} \right) + 4 \right] \quad (4.4)$$

donde,

- (s_j, t_j) : Es la j -ésima pareja común máxima encontrada en A y B sin conflicto con otras parejas comunes máximas elegidas.
- l : Es el número total de parejas comunes máximas encontradas en A y B sin conflictos entre sí.
- P'_j : Es la j -ésima subcadena correspondiente a la j -ésima pareja común máxima, antes de la maximización por la izquierda.
- ne_{s_j}, ne_{t_j} : Son el número de elementos en s_j y t_j necesarios para definir un cambio de dirección ortogonal con respecto a P'_j .

Notemos que el segundo término mide el número de elementos sin correspondencia, mientras que el primer término mide la disimilitud (acotada entre 0 y 1) de las subcadenas máximas sin conflictos que tienen en común A y B . En otras palabras, el primer término mide lo que tienen en común la cadena A y la cadena B mientras que el segundo término mide sus diferencias.

Finalmente se suma un cuatro para realizar una translación y hacer que la $Sim(A, B)$ sea mayor o igual a cero, como se verá mas adelante.

Dos parejas comunes máximas (s, t) y (s', t') de A y B , con subcadenas máximas:

$$P = p_1 p_2 \cdots p_k = a_{i_1} a_{i_1+1} \cdots a_{i_1+k-1} = b_{i_2} b_{i_2+1} \cdots b_{i_2+k-1}$$

$$Q = q_1 q_2 \cdots q_r = a_{i_3} a_{i_3+1} \cdots a_{i_3+r-1} = b_{i_4} b_{i_4+1} \cdots b_{i_4+r-1}$$

respectivamente, e índices iniciales i_1 en A , i_2 en B para P e índices iniciales i_3 en A , i_4 en B para Q , no tienen conflictos entre sí, si las subcadenas:

$$P' = p'_1 p'_2 \cdots p'_k = a_{i'_1} a_{i'_1+1} \cdots a_{i'_1+k'-1} = b_{i'_2} b_{i'_2+1} \cdots b_{i'_2+k'-1}$$

$$Q' = q'_1 q'_2 \cdots q'_{r'} = a_{i'_3} a_{i'_3+1} \cdots a_{i'_3+r'-1} = b_{i'_4} b_{i'_4+1} \cdots b_{i'_4+r'-1}$$

antes de la maximización por la izquierda:

$$P = l \max(A, B, P')$$

$$Q = l \max(A, B, Q')$$

cumplen con que:

- i) $\{i'_1 - ne_s, \dots, i'_1 - 1, i'_1, i'_1 + 1, \dots, i'_1 + k' - 1\} \cap \{i'_3 - ne_{s'}, \dots, i'_3 - 1, i'_3, i'_3 + 1, \dots, i'_3 + r' - 1\} = \phi$
- ii) $\{i'_2 - ne_t, \dots, i'_2 - 1, i'_2, i'_2 + 1, \dots, i'_2 + k' - 1\} \cap \{i'_4 - ne_{t'}, \dots, i'_4 - 1, i'_4, i'_4 + 1, \dots, i'_4 + r' - 1\} = \phi$

En donde los números de elementos necesarios para definir un cambio de dirección ortogonal, ne_s , ne_s' , ne_t y ne_t' , son con respecto a P' y a Q' .

Por ejemplo, si las parejas comunes máximas (s, t) y (s', t') son:

$$\begin{array}{ll}
 s=132001431 & s'=132001 \\
 t=24001431 & t'=24001 \\
 P=001431 & i_1=4 \quad i_2=3 & Q=001 & i_3=4 \quad i_4=3 \\
 P'=431 & i'_1=7 \quad i'_2=6 & Q'=001 & i'_3=4 \quad i'_4=3
 \end{array}$$

Entonces

$$ne_s = 4 \quad ne_s' = 4 \quad ne_t = 2 \quad ne_t' = 2$$

i) $\{3,4,5,6,7,8,9\} \cap \{2,3,4,5,6\} = \{3,4,5,6\}$ lo cual implica que hay conflicto entre s y s' .

ii) $\{2,3,4,5,6,7,8\} \cap \{1,2,3,4,5\} = \{2,3,4,5\}$ lo cual implica que hay conflicto entre t y t' .

Si tienen conflictos entre sí, se elige a la pareja cuya disimilitud D sea menor. En el caso en que existieran dos parejas con la misma disimilitud, entonces se elige cualquiera de ellas arbitrariamente.

Las subcadenas P'_j deben cumplir con que:

i) Son subcadenas máximas de A y B

o

ii) Son subcadenas de subcadenas máximas de A y B que tuvieron que ser divididas porque tenían conflictos entre sí.

Por ejemplo, si:

$$\begin{array}{l}
 A=12343333 \\
 B=333312343
 \end{array}$$

Encontramos dos subcadenas máximas:

$P=12343$, con índices iniciales 1 y 5 en A y B respectivamente.

$Q=3333$, con índices iniciales 5 y 1 en A y B respectivamente.

Como tienen conflicto entre sí al compartir el elemento en el índice 5, se calcula su disimilitud acotada $D_{0-1}(s, t)$.

Para el caso de P :

$$s=12343 \quad t=333312343 \quad D_{0-1}(s, t) = 0.093$$

Para el caso de Q:

$$s = 12343333$$

$$t = 3333$$

$$D_{0-1}(s, t) = 0.172$$

Por lo tanto se elige P y nos quedamos con la subcadena 333 con índices iniciales 6 y 2 en A y B respectivamente.

4.3.1 Propiedades

i) Positividad. Por demostrar que $Sim(A, B) \geq 0$.

$D_{0-1}(s_j, t_j) \geq 0$, como se demostró anteriormente en la sección 4.2.2.4.

Por lo tanto, $\sum_{j=1}^l D_{0-1}(s_j, t_j) \geq 0$

Por otro lado, por construcción se sabe que $\{P^j; 1 \leq j \leq l\}$ son subcadenas de A y de B que no tienen intersecciones entre sí, tomando en cuenta a los elementos necesarios para definir un cambio de dirección ortogonal para cada P^j .

Por lo tanto,

$$m+n+4 \geq \sum_{j=1}^l 2L(P^j) + ne_{s_j} + ne_{t_j}$$

El cuatro se debe a los dos elementos imaginarios predecesores al primer elemento en A y en B necesarios para definir un cambio de dirección ortogonal.

Con lo cual se tiene que,

$$m+n - \left(\sum_{j=1}^l 2L(P^j) + ne_{s_j} + ne_{t_j} \right) + 4 \geq 0$$

Por lo tanto $Sim(A, B) \geq 0$.

ii) Simetría. Por demostrar que $Sim(A, B) = Sim(B, A)$.

$$\sum_{j=1}^l D_{0-1}(s_j, t_j) = \sum_{j=1}^l D_{0-1}(t_j, s_j) \text{ y}$$

$$m+n - \left(\sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j} \right) + 4 = n+m - \left(\sum_{j=1}^l 2L(P'_j) + ne_{t_j} + ne_{s_j} \right) + 4$$

Por lo tanto,

$$Sim(A, B) = Sim(B, A)$$

iii) Identidad. Por demostrar que $Sim(A, A) = 0$.

Como $A=A$, entonces $l=1$, ya que existe una sólo subcadena máxima $P_1=P'_1$ y por lo tanto una sólo pareja común máxima (s_1, t_1) tales que $s_1=A$ y $t_1=A$. Lo cual implica que, $s_1=t_1=P_1=P'_1$. Además, $ne_{s_1} = ne_{t_1} = 2$, que son los dos elementos imaginarios predecesores al primer elemento de A .
Por lo tanto,

$$Sim(A, A) = D(s_1, s_1) + 2m - (2L(A) + 2 + 2) + 4$$

$$Sim(A, A) = 0 + 2m - 2m - 2 - 2 + 4 = 0$$

iv) Unicidad. Por demostrar que $Sim(A, B) > 0$ para $A \neq B$.

Supongamos que $Sim(A, B) = 0$ para $A \neq B$, entonces:

$$\sum_{j=1}^l D(s_j, t_j) + \left[m+n - \left(\sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j} \right) + 4 \right] = 0$$

Lo cual implica que

$$\sum_{j=1}^l D(s_j, t_j) = 0 \text{ y que } \left[m+n - \left(\sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j} \right) + 4 \right] = 0$$

Si $l=0$, entonces

$$Sim(A, B) = m+n+4 \neq 0 \text{ y por lo tanto } l \geq 1$$

Si $l > 1$, entonces

Por lo menos existen dos parejas comunes máximas distintas: (s_1, t_1) y (s_2, t_2) , con P'_1 y P'_2 respectivamente, sin conflictos entre sí.

Por lo tanto $D(s_1, t_1) + D(s_2, t_2) = 0$, lo cual implica que

$$D(s_1, t_1) = 0 \text{ y } D(s_2, t_2) = 0$$

Lo cual implica que $s_1=t_1$ y $s_2=t_2$.

Pero por construcción ésto no puede ser posible. Y por lo tanto:

$$\sum_{j=1}^l D(s_j, t_j) > 0$$

Lo cual implica que l no puede ser mayor que uno y por consiguiente, para que $Sim(A, B) = 0$ l tiene que ser 1.

Si $l=1$, entonces

$$D(s_1, t_1) = 0 \text{ implica que } s_1 = t_1 = P'_1 \text{ y que } ne_{s_1} = 2.$$

Además, por la hipótesis,

$$m + n + 4 = 2L(P'_1) + ne_{s_1} + ne_{t_1}$$

$$m + n + 4 = 2L(P'_1) + 2ne_{s_1}$$

$$m + n = 2L(P'_1) + 2(2) - 4$$

$$m + n = 2L(P'_1)$$

$$m + n = L(s_1) + L(t_1)$$

Lo cual implica que

$$L(s_1) = m \text{ y } L(t_1) = n$$

ya que $L(s_1) \leq m$ y $L(t_1) \leq n$

y por lo tanto

$$A = B.$$

Que contradice a $A \neq B$, y por lo tanto $Sim(A, B) > 0$ para $A \neq B$. ■

Una propiedad que no se cumple es la desigualdad del triángulo, es decir, no es cierto que:

$$Sim(A, C) \leq Sim(A, B) + Sim(B, C).$$

Contra ejemplo: Sean

A=4123242333333333

B=111110000

C=412324211111

$Sim(A, C) = 0.5625$, existe una sola subcadena máxima: 4123242.

$Sim(A, B) = 0$, ya que no existe ninguna subcadena máxima en A y B.

$Sim(B, C) = 0.5548$, existe una sola subcadena máxima: 11111.

$$0.5625 > 0.5548$$

$$0.5625 > 0 + 0.5548$$

$$\text{Sim}(A,C) > \text{Sim}(A,B) + \text{Sim}(B,C)$$



El que no se cumpla la desigualdad del triángulo es deseable, ya que esto permite que se midan adecuadamente las correspondencias parciales, según lo explicado en la sección 2.3.2.

4.3.2 Medida de similitud acotada, entre dos códigos de cadenas

Afirmación. $0 \leq \text{Sim}(A,B) \leq m+n+4$

Demostración.

i) Por demostrar que $0 \leq \text{Sim}(A,B)$.

Es una propiedad de la medida de similitud.

ii) Por demostrar que $\text{Sim}(A,B) \leq m+n+4$, es decir:

$$\sum_{j=1}^l D(s_j, t_j) + \left[m+n - \left(\sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j} \right) + 4 \right] \leq m+n+4$$

Por definición:

$$2 \leq L(P'_j) \text{ y } 0 \leq D(s_j, t_j) \leq 1, \text{ para toda } 1 \leq j \leq l$$

Con lo cual tenemos que:

$$2 \leq L(P'_j)$$

$$4 \leq 2L(P'_j)$$

Por lo tanto,

$$D(s_j, t_j) \leq 1 < 4 \leq 2L(P'_j)$$

$$D(s_j, t_j) < 2L(P'_j)$$

$$D(s_j, t_j) < 2L(P'_j) + ne_{s_j} + ne_{t_j}$$

$$\sum_{j=1}^l D(s_j, t_j) < \sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j}$$

$$\sum_{j=1}^l D(s_j, t_j) - \sum_{j=1}^l 2L(P^j) - ne_{s_j} - ne_{t_j} < 0$$

$$\sum_{j=1}^l D(s_j, t_j) - \sum_{j=1}^l 2L(P^j) - ne_{s_j} - ne_{t_j} + m + n + 4 < m + n + 4$$

$$\sum_{j=1}^l D(s_j, t_j) + \left[m + n - \left(\sum_{j=1}^l 2L(P^j) + ne_{s_j} + ne_{t_j} \right) + 4 \right] < m + n + 4$$

Observación. Si $a_i \neq b_i$ para toda $1 \leq i \leq \min(m, n)$ entonces $l=0$ y por lo tanto:

$$Sim(A, B) = 0 + m + n - 0 + 4 = m + n + 4$$

■

Definición 4.13: Similitud acotada entre dos curvas

Se define la medida de similitud acotada entre dos códigos de cadenas como:

$$Sim(A, B) = 1 - \frac{Sim(A, B)}{m + n + 4} \quad (4.5)$$

de tal forma que:

$$0 \leq Sim(A, B) \leq 1$$

Notemos que la medida de similitud entre dos curvas es cero cuando las dos curvas son completamente distintas y es uno cuando son idénticas.

**FALTA
PAGINA**

82

CAPITULO 5

IMPLEMENTACIÓN

En este capítulo se presentan los algoritmos para calcular la similitud entre dos curvas de acuerdo a las definiciones del capítulo anterior. Adicionalmente, se describe brevemente su implementación bajo el paradigma de orientación a objetos. Finalmente, se presenta una aplicación gráfica implementada en C++, OpenGL (para el soporte del despliegue gráfico 3D) y Glow (para el soporte de la interface de usuario gráfica), la cual permite desplegar y comparar curvas 3D a partir de su código de cadenas de cambio de dirección ortogonal.

5.1 ALGORITMOS

De aquí en adelante sean A y B dos cadenas cualesquiera con longitudes m y n respectivamente.

En las siguientes subsecciones se presentan los algoritmos principales para calcular la similitud entre dos curvas.

5.1.1 Algoritmo general

Para calcular la similitud entre dos curvas, se propone una aproximación que tome en cuenta aquellos aspectos que tienen en común y aquellos en los que difieren, como se ejemplifica en la Figura 5.1. Las funciones de disimilitud acotada entre dos subcurvas y la función *Sim*, se presentan en el capítulo 4. El algoritmo general para calcular la similitud entre dos curvas se presenta a continuación.

Sim(A, B)

- 1.- Encontrar todas las parejas comunes máximas de A y B .
- 2.- Elegir las parejas comunes máximas que no tengan conflictos entre sí de acuerdo a la definición 4.12 y que corresponden a las parejas: (s_j, t_j) .

$$\sum_{j=1}^l D(s_j, t_j) + \left[m + n - \left(\sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j} \right) + 4 \right]$$

3.- Calcular $Sim(A, B) = 1 - \frac{\sum_{j=1}^l D(s_j, t_j) + \left[m + n - \left(\sum_{j=1}^l 2L(P'_j) + ne_{s_j} + ne_{t_j} \right) + 4 \right]}{m + n}$

de acuerdo a las parejas (s_j, t_j) seleccionadas.

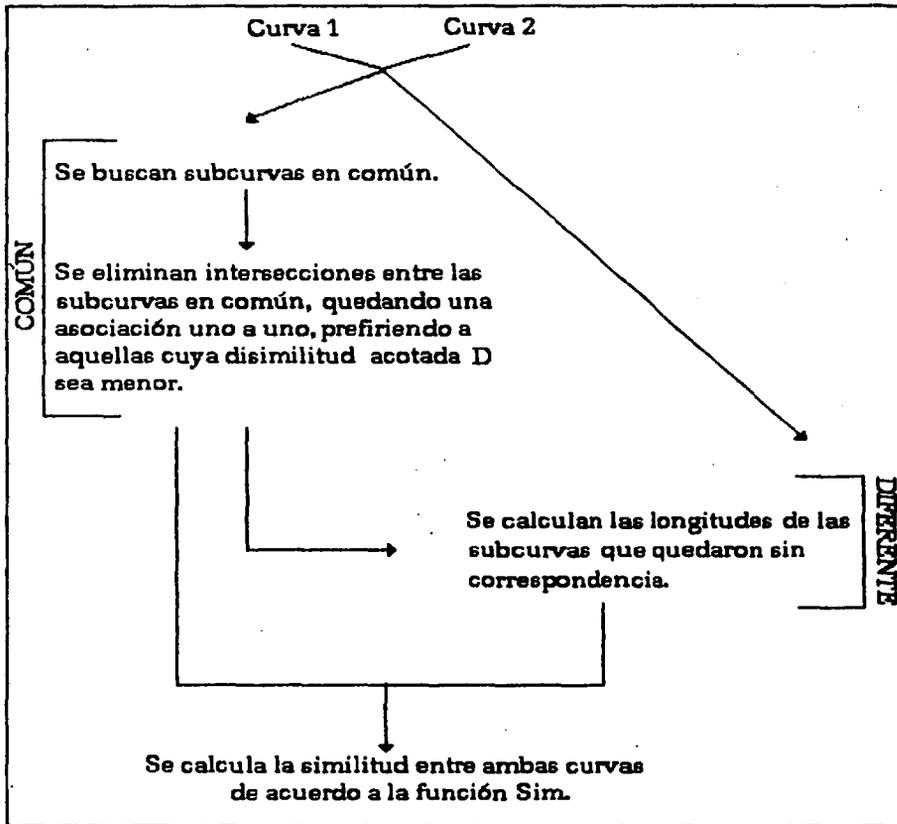


Figura 5.1 Algoritmo general para calcular la similitud entre dos curvas.

5.1.2 Algoritmo para encontrar todas las parejas comunes máximas de A y B.

Este algoritmo encuentra todas las parejas comunes máximas (definición 4.9) de A y B, o en otras palabras, encuentra todas las subcurvas de mayor longitud que tienen en común la curva A y la curva B.

encuentraParejasComunesMaximas (A, B)

(parejas_maximas es una estructura de datos, por ejemplo un vector. i es una variable de tipo entero)

1. Repetir con *i* desde 1 hasta *m*

1.1 Si *i*=1

entonces

Llamar a *encuentraPatronesMaximos* con argumentos:

- i) La subcadena de A que va del índice *i* al índice *m*
- ii) B
- iii) -1
- iv) *i*
- v) *parejas_maximas*

si no

Llamar a *encuentraPatronesMaximos* con argumentos:

- i) La subcadena de A que va del índice *i* al índice *m*.
- ii) B
- iii) el elemento *i*-1 de A
- iv) *i*
- v) *parejas_maximas*

1.2 {Fin del condicional del paso 1.1}

2. {Fin del ciclo del paso 1}

La condicional 1.1 se realiza para verificar que la variable *i* no sea el primer elemento de la cadena A, de serlo se envía el valor -1. Este valor le indica a la llamada a *encuentraPatronesMaximos* que no existe un elemento anterior.

Los elementos de las cadenas A y B son enteros en el conjunto {0,1,2,3,4}.

Al terminar el algoritmo la estructura de datos *parejas_maximas* contiene todas las parejas comunes máximas de A y B.

**encuentraPatronesMaximos(A,B,elemento_anterior, indx,
parejas_maximas)**

{elemento anterior, indx, ind e i son variables de tipo entero.
parejas_maximas es una estructura de datos, p.e un vector.
m es un adt (abstract data type) que representa una correspondencia y que esta compuesta de dos subcadenas.}

1. Repetir con *i* desde 1 hasta $n - 1$

1.1 Si (el elemento *i* de *B* = el elemento 1 de *A*) y
(el elemento *i*+1 de *B* = el elemento 2 de *A*)

entonces

1.1.1 Si (elemento_anterior!=-1) y (*i*!=1) y
(el elemento *i*-1 de *B* = elemento_anterior)

entonces

Pasar a la siguiente iteración;

si no

Hacer *ind* ← llamar a *encuentraIndiceMaximo* con:

i) *A*

ii) La subcadena de *B* que va del índice *i* al
índice *n*.

Hacer *m* ← Una nueva correspondencia con las
subcadenas:

i) La subcadena de *A* que va del índice *indx*
al índice *ind*.

ii) La subcadena de *B* que va del índice *i* al
al índice *ind*.

Agregar *m* a parejas_maximas;

1.1.2 {Fin del condicional del paso 1.1.1}

1.2 {Fin del condicional del paso 1.1}

2. {Fin del ciclo del paso 1}

encuentraIndiceMaximo(A,B)

{indice es una variable de tipo entero. Al final regresa el valor
de indice.}

1. Hacer *indice* ← 1;

2. Repetir mientras (*indice* <= *m*) y (*indice* <= *n*) y
(el elemento *indice* de *A* = el elemento *indice* de *B*)

Hacer *indice* ← *indice* + 1;
3. {Fin del ciclo del paso 2}

5.1.3 Algoritmo para elegir las parejas comunes máximas de A y B sin conflictos entre sí.

En el paso 2 se eligen las mejores correspondencias de subcurvas de acuerdo al criterio de disimilitud acotada entre dos subcadenas de una pareja común máxima (definición 4.11).

eligeParejasComunesMaximas (parejas_maximas)

{*parejas_maximas* es una estructura de datos, p.e un vector.
Cada elemento de *parejas_maximas* es una estructura que representa una correspondencia y que esta compuesta de dos subcadenas.
seleccionada es una estructura como la anterior.
seleccionadas es una estructura de datos como *parejas_maximas*.
nuevas_parejas es una estructura de datos como *parejas_maximas*.
Al final regresa *seleccionadas*.}

1. Ordenar ascendentemente la estructura de datos *parejas_maximas* de acuerdo a la disimilitud acotada;
 2. Repetir mientras *parejas_maximas* tenga elementos

{Selecciona la primera pareja y la copia a *seleccionadas*}
Hacer *seleccionada* ← elemento 0 de *parejas_maximas*;

Agregar *seleccionada* a *seleccionadas*;

Eliminar *seleccionada* de *parejas_maximas*;

Hacer *nuevas_parejas* ← llamar a *resuelveConflictos* con:
i) *seleccionada*
ii) *parejas_maximas*
- 2.1 Si *nuevas_parejas* tiene elementos

Agregar todos los elementos de *nuevas_parejas* a *parejas_maximas*;

Ordenar ascendentemente la estructura de datos *parejas_maximas* de acuerdo a la disimilitud acotada;

2.2 {Fin del condicional del paso 2.1}

3. {Fin del ciclo del paso 2}

Al terminar la estructura de datos *seleccionadas* contiene a todas las parejas seleccionadas sin conflictos entre sí.

La función *resuelveConflictos* se encarga de resolver los conflictos entre la pareja seleccionada y el resto de las parejas comunes máximas. El resolver los conflictos significa eliminar aquellas parejas comunes máximas que se intersecten completamente con la pareja seleccionada o en dividir el patron en aquellas partes que haya conflicto. Al final regresa las nuevas correspondencias generadas en caso de que se haya tenido que dividir el patron.

resuelveConflictos(pareja_seleccionada, parejas_maximas)

{pareja seleccionada es un adt que representa una correspondencia y que esta compuesta de dos subcadenas.

parejas_maximas es una estructura de datos, p.e un vector, en donde cada elemento de *parejas_maximas* es una estructura como la de *pareja_seleccionada*.

InterseccionA e *InterseccionB* son un adt que representa los índices de interseccion entre dos subcadenas.

nuevas y *nuevas_correspondencias* son estructuras de datos, p.e un vector, igual que *parejas_maximas*.

i es una variable de tipo entero.

Al final regresa *nuevas_correspondencias*.)

1. Repetir para *i=0* hasta *i = número de elementos de parejas_maximas*

{Calcula la intersección en A}

1.1 Hacer *interseccionA* ← llamar a *obtenInterseccion* con:

- i) primer subcadena de *pareja_seleccionada*
- ii) primer subcadena del elemento actual de *parejas_maximas*

{Calcula la intersección en B}

1.2 Hacer *interseccionB* ← llamar a *obtenInterseccion* con:

- i) segunda subcadena de *pareja_seleccionada*
- ii) segunda subcadena del elemento actual de *parejas_maximas*

```

1.3 Si (es vacia interseccionA) y (es vacia interseccionB)
    entonces
        Hacer  $i \leftarrow i+1$ ;
    si no {La intersección puede provocar que se
        divida el patron o que se elimine}
        Hacer nuevas  $\leftarrow$  llamar a resuelveConflictos2 con:
            i) elemento actual de parejas_maximas
            ii) interseccionA
            iii) interseccionB

        Eliminar el elemento actual de parejas_maximas;

        Copiar todos los elementos de nuevas a
        nuevas_correspondencias;

1.4 { Fin del condicional del paso 1.3}

```

```

2. {Fin del ciclo del paso 1}

```

La función *obtenInterseccion* calcula el índice inicial y el índice final de intersección entre dos subcadenas. Si no existe ninguna intersección, entonces regresa los índices -1, -1.

obtenInterseccion(r, s)

{*r* y *s* son un adt que representa una subcadena.

interseccion es un adt que representa los índices de interseccion entre dos subcadenas.

ir, ner, il, lr, el, is, nes, i2, ls, e2 son variables de tipo entero. Al final regresa *interseccion*.)

```

1. Hacer  $ir \leftarrow$  índice inicial de r;
2. Hacer  $ner \leftarrow$  número de elementos predecesores a r necesarios
    para definir un cambio de dirección ortogonal;
3. Hacer  $il \leftarrow ir - ner$ ;
4. Si  $il \leq 0$ 
    entonces
        Hacer  $il \leftarrow 1$ ;
5. Hacer  $lr \leftarrow$  longitud de r;
6. Hacer  $el \leftarrow ir + lr - 1$ ;

```

7. Hacer $i_s \leftarrow$ índice inicial de s ;
8. Hacer $n_{es} \leftarrow$ número de elementos predecesores a s necesarios para definir un cambio de dirección ortogonal;
9. Hacer $i_2 \leftarrow i_s - n_{es}$;
10. Si $i_2 \leq 0$
entonces
 Hacer $i_2 \leftarrow 1$;
11. Hacer $l_s \leftarrow$ longitud de s ;
12. Hacer $e_2 \leftarrow i_s + l_s - 1$;
- 13.A Si $(i_2 < i_1)$ y $(e_2 \geq i_1)$ y $(e_2 \leq e_1)$ {por la izquierda}
entonces
 Hacer $interseccion \leftarrow$ la intersección inicia en i_1 y termina en e_2 ;
- 13.B Si $(i_2 > i_1)$ y $(i_2 \leq e_1)$ y $(e_2 > e_1)$ {por la derecha}
entonces
 Hacer $interseccion \leftarrow$ la intersección inicia en i_2 y termina en e_1 ;
- 13.C Si $(i_2 \geq i_1)$ y $(e_2 \leq e_1)$ {contenido}
entonces
 Hacer $interseccion \leftarrow$ la intersección inicia en i_2 y termina en e_2 ;
- 13.D Si $(i_2 \leq i_1)$ y $(e_2 \geq e_1)$ {conteniendo}
entonces
 Hacer $interseccion \leftarrow$ la intersección inicia en i_1 y termina en e_1 ;

resuelveConflictos2(pareja_con_conflicto, interseccionA, interseccionB)

{pareja con conflicto es un adt que representa una correspondencia y que esta compuesta de dos subcadenas.

interseccionA e interseccionB son un adt que representa los índices de intersección entre dos subcadenas.

niA, sliA, desplazamientoA, s2iA, nuevoi1, nueva longitudA, niB, sliB, desplazamientoB, nuevoi2 y nueva longitudB son variables de tipo entero.

nuevoB, nA y nB son un adt que representa una subcadena. nuevosA y aux son una estructura de datos, p.e un vector, en donde cada elemento es una subcadena.

nuevas correspondencias es una estructura de datos, p.e un vector, en donde cada elemento es un adt que representa una correspondencia y que esta compuesta de dos subcadenas.

Al final regresa nuevas correspondencias)

1. Hacer nuevosA ← llamar a *divide* con:
 - i) primer subcadena de pareja_con_conflicto
 - ii) interseccionA
2. Repetir para cada elemento de nuevosA
 - 2.1 Hacer niA ← índice inicial de la subcadena actual de nuevosA;
 - 2.2 Hacer s1iA ← índice inicial de la primer subcadena de pareja_con_conflicto;
 - 2.3 Hacer desplazamientoA ← niA - s1iA;
 - 2.4 Hacer s2iA ← índice inicial de la segunda subcadena de pareja_con_conflicto;
 - 2.5 Hacer nuevoi1 ← s2iA + desplazamientoA;
 - 2.6 Hacer nueva_longitudA ← longitud de la subcadena actual de nuevosA;
 - 2.7 Hacer nuevoB ← una nueva subcadena con índice inicial nuevoi1 y longitud nueva_longitudA;
 - 2.8 Hacer aux ← llamar a *divide* con:
 - i) nuevoB
 - ii) interseccionB
 - 2.9 Repetir para cada elemento de aux
 - Hacer niB ← índice inicial de la subcadena actual de aux;
 - Hacer s1iB ← índice inicial de la segunda subcadena de pareja_con_conflicto;
 - Hacer desplazamientoB ← niB - s1iB;

```
Hacer nuevoi2 ← sliA + desplazamientoB;  
Hacer nueva_longitudB ← longitud de la subcadena  
actual de aux;  
Hacer nA ← Una nueva subcadena con índice inicial  
nuevoi2 y longitud nueva_longitudB;  
Hacer nB ← elemento actual de aux;  
Agregar a nuevas_correspondencias una nueva corres-  
pondencia con su primer subcadena nA y su segunda  
subcadena nB;
```

2.10 (Fin del ciclo del paso 2.9)

3. (Fin del ciclo del paso 2)

La función *divide* se encarga de partir una subcadena de acuerdo a la intersección dada. La división puede generar 2, 1 o ninguna nueva subcadena.

divide(subcadena, interseccion)

{*subcadena* es un adt que representa una subcadena.
interseccion es un adt que representa los índices de intersección entre dos subcadenas.
il, *indl interseccion*, *indf interseccion*, *ls*, *el*, *nuevo indicei*,
nueva long, *ind verdadero* y *num elem* son variables de tipo entero.
nuevos patrones es una estructura de datos (p.e un vector) en donde cada uno de sus elementos es de un adt que representa una subcadena.
Al final regresa *nuevos patrones*.}

1. Si *interseccion* no es vacía entonces

```
Hacer il ← el índice inicial de subcadena;  
Hacer indi_interseccion ← el índice inicial de  
interseccion;  
Hacer indf_interseccion ← el índice final de  
interseccion;  
Hacer ls ← la longitud de subcadena;  
Hacer el ← il + ls - 1;
```

```
{Caso 1: Si la intersección se da en la subcadena P}
1.1 Si indi_interseccion >= il
    entonces
        1.1.1 Si (indi_interseccion - il) >= 2
            entonces
                Hacer nuevo_indice_i ← il;
                Hacer nueva_long ← indi_interseccion - il;
                Agregar a nuevos_patrones una nueva
                subcadena con índice inicial
                nuevo_indice_i y longitud nueva_long;
            1.1.2 {Fin del condicional del paso 1.1.1}
        1.1.3 Si (el - indf_interseccion) >= 2
            entonces
                Hacer nuevo_indice_i ← indf_interseccion + 1;
                Hacer nueva_long ← el - indf_interseccion;
                Hacer ind_verdadero ← nuevo_indice_i;
                Hacer num_elem ← 0;
                1.1.3.1 Repite mientras (num_elem < 2) y
                (ind_verdadero < nuevo_indice_i + nueva_long)
                    1.1.3.1.1 Si el elemento de la cadena
                    a la que pertenece subcadena en el
                    índice ind_verdadero != 0
                        entonces
                            Hacer num_elem ← num_elem + 1;
                            1.1.3.1.2 {Fin condicional 1.1.3.1.1}
                            Hacer ind_verdadero ← ind_verdadero + 1;
                            1.1.3.2 {Fin del ciclo del paso 1.1.3.1}
                            1.1.3.3 Si num_elem = 2
                                entonces
                                    1.1.3.3.1 Si (nuevo_indice_i +
                                    nueva_long - 1 - ind_verdadero) >= 2
                                        entonces
                                            Hacer nuevo_indice_i ←
                                            ind_verdadero;
                                            Hacer nueva_long ← el -
                                            ind_verdadero + 1;
                                            Agregar a nuevos_patrones
                                            una nueva subcadena con
                                            índice inicial nuevo_indice_i
                                            y longitud nueva_long;
                                        1.1.3.3.2 {Fin paso 1.1.3.3.1}
                                    1.1.3.4 {Fin condicional del paso 1.1.3.3}
                            1.1.4 {Fin del condicional del paso 1.1.3}
```

{Caso 2: Si la intersección se da en los elementos ne, o parte en ne y parte en la subcadena P}

1.2 si no

Hacer nuevo_indice_i ← ind_f interseccion + 1;
Hacer nueva_long ← el - ind_f interseccion;
Hacer ind_verdadero ← nuevo_indice_i;
Hacer num_elem ← 0

1.2.1 Repetir mientras (num_elem < 2) y
(ind_verdadero < nuevo_indice_i + nueva_long)

1.2.1.1 Si el elemento de la cadena a
la que pertenece subcadena en el índice
ind_verdadero ≠ 0
entonces

Hacer num_elem ← num_elem + 1;

1.2.1.2 {Fin del condicional 1.2.1.1}

Hacer ind_verdadero ← ind_verdadero + 1;

1.2.2 {Fin del ciclo del paso 1.2.1}

1.2.3 Si num_elem = 2

entonces

1.2.3.1 Si (nuevo_indice_i + nueva_long - 1 -
ind_verdadero) ≥ 2
entonces

Hacer nuevo_indice_i ← ind_verdadero;

Hacer nueva_long ← el - ind_verdadero + 1

Agregar a nuevos_patrones una nueva
subcadena con índice inicial

nuevo_indice_i y longitud nueva_long;

1.2.3.2 {Fin del condicional 1.2.3.1}

1.2.4 {Fin del condicional 1.2.3}

1.3 {Fin del condicional 1.1}

Si no

Agregar a nuevos_patrones una nueva subcadena con
índice inicial i_l y longitud l_s;

2. {Fin del condicional del paso 1}

5.2 TIPOS ABSTRACTOS DE DATOS

Los algoritmos de la sección 5.1 se implementaron bajo un diseño orientado a objetos en donde las principales clases programadas son:

CodigoDeCadenas3D
ElementoDeCadena
ElementoDeCadena0
ElementoDeCadena1
ElementoDeCadena2
ElementoDeCadena3
ElementoDeCadena4
ParejaComunMaxima
Subcadena3D
Interseccion
CorrespondenciaSubcadenas3D

5.2.1 Clase CodigoDeCadenas3D

La clase *CodigoDeCadenas3D* representa a una curva 3D representada por medio de un código de cadenas de cambio de dirección ortogonal, en donde sus elementos son objetos de tipo *ElementoDeCadena*.

El método principal de ésta clase es aquel que calcula la similitud acotada entre dos cadenas, y que corresponde a la implementación del paso 3 del algoritmo general descrito en la sección 5.1.1.

También incluye métodos privados auxiliares, entre los cuales destacan las implementaciones de los siguientes algoritmos:

```
encuentraParejasComunesMaximas (A, B)  
  
encuentraPatronesMaximos (A, B, elemento_anterior, indx,  
                           parejas_maximas)
```

```
encuentraIndiceMaximo(A, B)
eligeParejasComunesMaximas(parejas_maximas)
resuelveConflictos(pareja_seleccionada, parejas_maximas)
```

5.2.1.2 Clase ElementoDeCadena

La clase *ElementoDeCadena* es una clase abstracta que representa un elemento de una cadena. De esta clase heredan las clases concretas: *ElementoDeCadena0*, *ElementoDeCadena1*, *ElementoDeCadena2*, *ElementoDeCadena3* y *ElementoDeCadena4*, las cuales representan cada uno de los cuatro posibles elementos de una cadena.

5.2.1.3 Clase ParejaComunMaxima

La clase *ParejaComunMaxima* representa a una pareja común máxima como se define en la definición 4.9. Su principal método calcula la disimilitud acotada entre dos subcadenas de una pareja común máxima.

5.2.1.4 Clase Subcadena3D

La clase *Subcadena3D* es la abstracción de una subcadena. Sus principales métodos son las implementaciones de los algoritmos:

```
obtenInterseccion(r, s)
divide(subcadena, interseccion)
```

5.2.1.5 Clase Interseccion

La clase *Interseccion* representa los índices de intersección entre dos subcadenas. Esta clase es usada en el algoritmo que elimina conflictos entre las parejas comunes máximas.

5.2.1.6 Clase CorrespondenciaSubcadenas3D

Esta clase representa una correspondencia de una subcadena en dos cadenas. Su método principal es la implementación del algoritmo:

```
resuelveConflictos2(pareja_con_conflicto, interseccionA,
                    interseccionB)
```

5.3 APLICACIÓN

El contar con una aplicación computacional que permitiera el despliegue gráfico de las dos curvas 3D a comparar y que realizara su comparación, era de gran importancia para la realización de esta tesis. Gracias a su implementación se pudo experimentar, detectar patrones y crear hipótesis acerca del comportamiento de la representación.

Actualmente existen varios programas de despliegue tridimensional, entre los cuales destacan Autocad y 3D Studio Max. Sin embargo, su enfoque es para realizar diseño asistido por computadora y para animación, respectivamente. Esto hace que la visualización y exploración de las curvas 3D representadas con un código de cadenas no sea directa ni flexible.

Como una respuesta a estas necesidades, se programó una aplicación en C++, OpenGL (para el soporte del despliegue gráfico 3D y Glow (para el soporte de la interface de usuario gráfica).

El sistema cuenta con las siguientes características:

- Permite introducir las dos curvas a comparar en su forma de cadenas.
- Visualiza en tres dimensiones las curvas dadas.
- Permite manipular en tres dimensiones a las curvas. Esto incluye rotaciones, translaciones y escalas.
- Se pueden modificar propiedades de la visualización, como el ancho de las líneas, si se desea que se muestren los voxels por donde pasa la curva y si se desea que se muestren los ejes y planos del espacio de la curva.
- Cuenta con dos tipos de cámaras, una con proyección en perspectiva y otra con proyección ortográfica.
- Los elementos de la curva se pueden visualizar con un código de colores: el elemento 0 en azul, el elemento 1 en verde, el elemento 2 en rojo, el elemento 3 en amarillo y el elemento 4 en gris. Esto facilita la detección de patrones.
- Calcula su similitud mostrando con colores las asociaciones uno a uno de las subcurvas en común. Además muestra la disimilitud entre cada pareja común máxima.

La ilustración I.11 muestra una imagen de la apariencia de esta aplicación.

**FALTA
PAGINA**

98

CAPITULO 6

RESULTADOS

En este capítulo se muestran y discuten los resultados obtenidos al aplicar la medida de similitud propuesta, a diferentes conjuntos de curvas 3D generadas artificialmente.

6.1 CURVAS CON CADENAS MUY PARECIDAS

El primer conjunto de prueba esta formado por tres curvas tridimensionales cuyas cadenas son muy parecidas. Se espera que las curvas que tengan cadenas muy parecidas representen también curvas muy parecidas y por lo tanto que la medida de similitud entre éstas sea muy cercana a 1.0. Las curvas se muestran en la Figura 6.1.

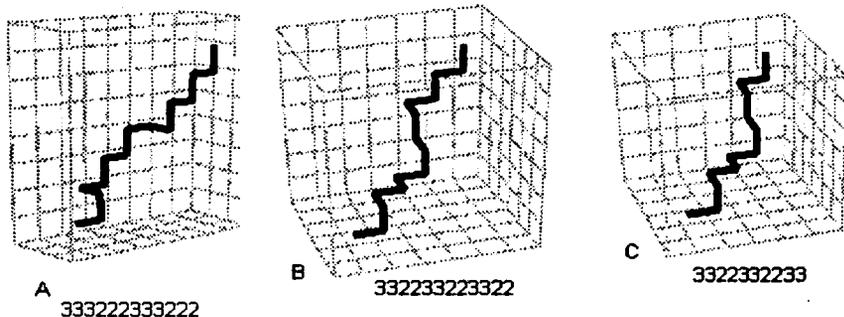


Figura 6.1 Curvas con cadenas muy parecidas

La tabla 6.1 muestra los resultados de la similitud entre las curvas A, B y C de la Figura 6.1. En la diagonal de esta tabla se puede observar que la similitud de una curva dada a sí misma es 1.0, es decir, que son idénticas. También se puede observar la simetría de la función *Sim*. Las ilustraciones 1.12, 1.13 e 1.14 muestran en colores las correspondencias uno a uno elegidas, así como las disimilitudes entre las parejas comunes máxi-

REPOSICIONADO
 INSTITUTO DE QUÍMICA

TESIS CON
 FALLA DE ORIGEN

mas. En color negro se muestran las subcadenas que quedaron sin correspondencia. Los elementos subrayados son los elementos distintos de cero necesarios para definir un cambio de dirección ortogonal.

	A	B	C
A	1.0	0.8498	0.7648
B	0.8498	1.0	0.9230
C	0.7648	0.9230	1.0

Tabla 6.1 Medidas de similitud entre curvas 3D con cadenas muy parecidas.

Este ejemplo evidencia que las cadenas que son muy parecidas representan curvas que visualmente también son muy parecidas, y la medida de similitud también refleja esta relación.

6.2 CURVAS CON CORRESPONDENCIAS PARCIALES

El segundo conjunto de prueba esta formado por tres curvas 3D con correspondencias parciales. Las curvas se muestran en la Figura 6.3.

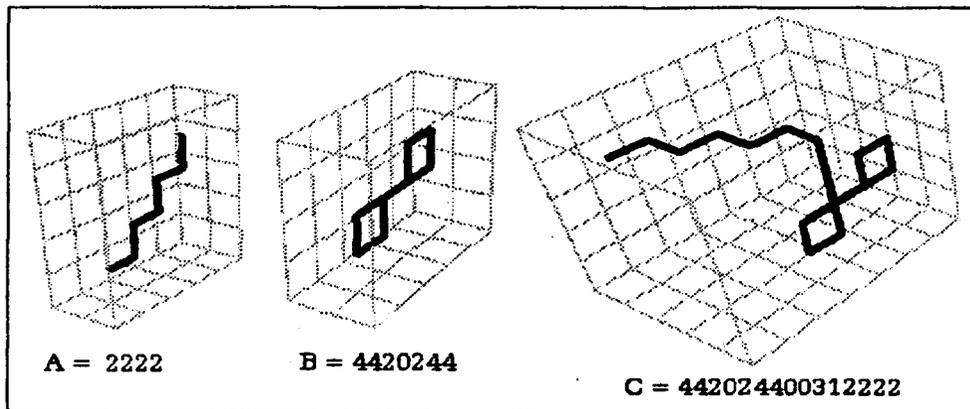


Figura 6.2 Curvas 3D con correspondencias parciales.

La tabla 6.2 muestra las similitudes entre estas tres curvas.

Las curvas fueron construidas de tal forma que la curva *C* estuviera formada la mitad de la curva *A* y la otra mitad de la curva *B*. La medida de similitud reconoce las correspondencias parciales y las mide correctamente dando valores cercanos al 0.5. La Figura 6.3 y las ilustraciones I.15 e I.16 muestran las correspondencias uno a uno encontradas, así como las disimilitudes entre éstas.

	A	B	C
A	1.0	0.0	0.5167
B	0.0	1.0	0.6923
C	0.5167	0.6923	1.0

Tabla 6.2 Medidas de similitud entre curvas 3D con correspondencias parciales.

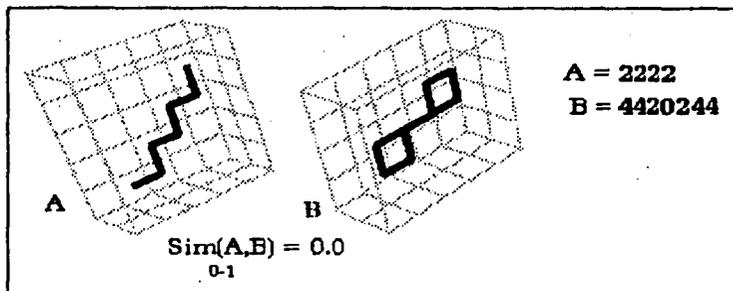


Figura 6.3 Similitud de dos curvas 3D sin nada en común.

6.3 ESPIRALES

El tercer conjunto de prueba esta compuesto de tres curvas 3D, dos espirales y una línea recta con dos pestañas. Estas curvas se muestran en la Figura 6.4. Sus similitudes se muestran en la tabla 6.3.

	A	B	C
A	1.0	0.4243	0.0
B	0.4243	1.0	0.1425
C	0.0	0.1425	1.0

Tabla 6.4 Medidas de similitud entre las curvas 3D de la Figura 6.4.

Como era de esperarse los dos espirales son más parecidos que cualquiera de ellos con la línea recta, de hecho el espiral A y la línea recta son completamente distintos.

Las ilustraciones I.17 e I.18 muestran las correspondencias uno a uno encontradas. Las cadenas correspondientes a éstas curvas son:

A = 43134313431343134313431343134313431343134313

B = 4031030040031000300040000310000300000400

C = 004

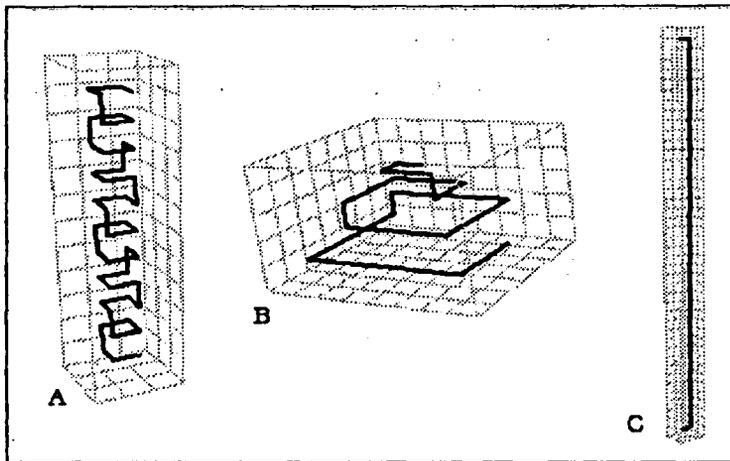


Figura 6.4 Tres curvas 3D a comparar.

6.4 CURVAS CERRADAS DE LONGITUD 18

El cuarto conjunto de prueba contiene 7 curvas cerradas de longitud 18. Se comparó la curva A con el resto de las curvas. La Figura 6.5 muestra las curvas ordenadas por similitud. Para el caso de las curvas cerradas se puede observar que encuentra correctamente las subcurvas en común y que la medida de similitud se comporta aceptable. El ejemplo más claro es la comparación de la curva A con la curva G, en cuyo caso solo comparten una pequeña porción de las curvas y esto se ve reflejado en la similitud obtenida.

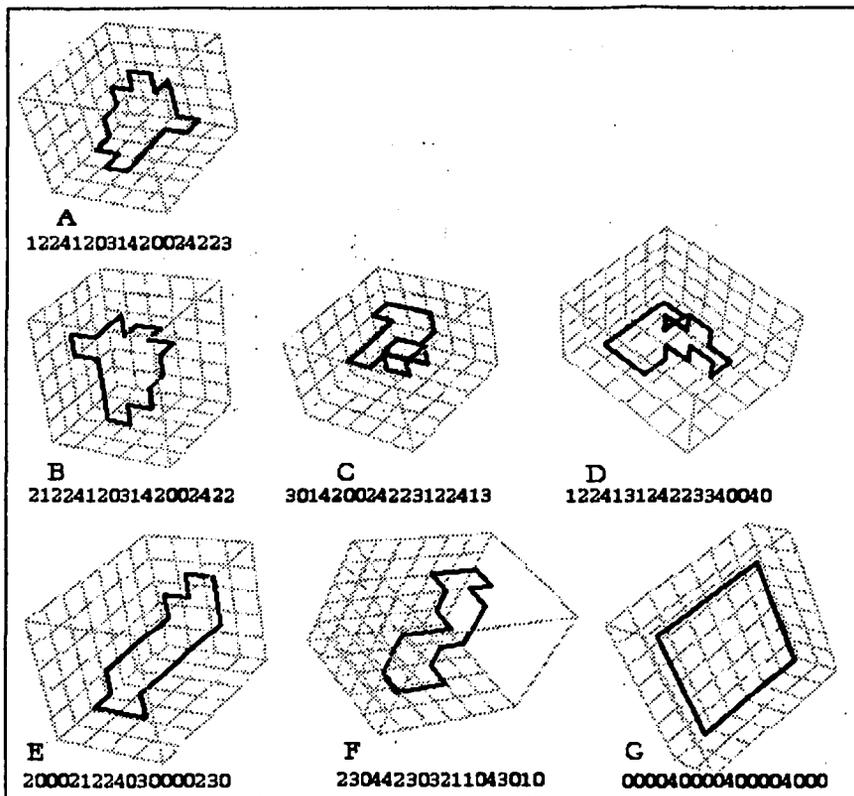


Figura 6.5 Comparación de las curvas [B-G] con la curva A. Las curvas se encuentran ordenadas por similitud de la más parecida a la más distinta.

6.5 AUTOSIMILITUD EN LA CURVA DE PEANO

Este ejemplo contiene dos curvas. La primera curva es el nivel uno correspondiente a la curva de Peano. La segunda curva corresponde al nivel dos de la curva de Peano (ver Figura 6.6).

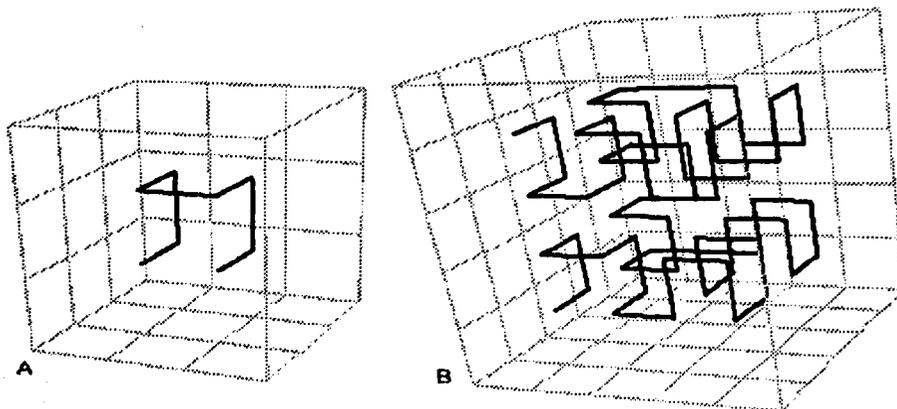


Figura 6.7 Curvas de Hilbert. La curva A corresponde al nivel 0 y B al nivel 1.

TESIS CON
TALA DE ORIGEN

**FALTA
PAGINA**

106

CONCLUSIONES

Se presentó una nueva propuesta de similitud para curvas tridimensionales basada en la representación del código de cadenas de cambio de dirección ortogonal [Bribiesca, 2000] y [Bribiesca y Velarde, 2001]. Gracias a que esta representación es invariante a las transformaciones de translación, rotación y espejo, la medida de similitud también es invariante a estas transformaciones. Mas aún, estas propiedades se extienden a un nivel local, de tal forma que las subcurvas y sus disimilitudes también son invariantes a estas transformaciones. Sin embargo, aún no existe algún método que permita que las curvas representadas con éste código sean invariantes a la transformación de escala. Durante el desarrollo de esta tesis se observó que para escalas enteras es posible realizar la invarianza, pero aún es un tema abierto. Debido a que la medida de similitud se basa en esta representación, si se llegara a encontrar algún método para realizar la invarianza a la escala, esta medida seguiría funcionando y sería también invariante a esta nueva transformación.

Se propusieron dos medidas: una medida de disimilitud entre subcurvas y una medida de similitud entre curvas. La medida de disimilitud es una métrica, que mide disimilitudes locales, mientras que la medida de similitud no es una métrica y utiliza las disimilitudes locales para definir una similitud global entre dos curvas. Estas medidas se normalizaron a escala del 0 al 1. Para el primer caso, 0 significa que son idénticas y 1 que son completamente distintas. Por el contrario, para el segundo caso, 0 significa que son completamente distintas y 1 que son idénticas.

Gracias a que la representación cumple con la propiedad de unicidad, que cuenta con un rico soporte local y que las propiedades de invarianzas se extienden a un nivel local, es posible la representación y comparación de curvas con oclusiones o con similitudes parciales.

Se propone también un método para realizar correspondencias uno a uno entre las subcurvas en común, el cual se basa en dar preferencia a aquellas subcurvas cuya disimilitud sea mas pequeña.

Adicionalmente se presentaron los algoritmos para calcular la similitud de dos curvas que implementan éstas medidas.

El reconocimiento de subcurvas se redujo a un problema de correspondencia de cadenas, cuyo algoritmo puede ser paralelizado.

También se realizó una aplicación que implementa los algoritmos propuestos. Esta aplicación fue programada en C++, OpenGL (para el soporte gráfico) y Glow (para la interfase de usuario gráfica).

Permite dar dos cadenas, compararlas (con la medida de similitud) y visualizarlas en tres dimensiones con algunas propiedades configurables como: un código de color, ancho de las líneas, y mostrar y ocultar ejes.

Las medidas propuestas (entre subcurvas y curvas) están definidas para cadenas que pueden ser de longitudes distintas. Gracias a esto, se pudo aplicar a curvas con autosemejanza, por ejemplo la curva de Hilbert.

Los conjuntos de curvas con los que se probó la medida de similitud fueron contruidos sintéticamente. Estos conjuntos incluyen curvas abiertas y cerradas, curvas con correspondencias parciales, curvas con cadenas muy parecidas y curvas con autosemejanza.

Si la cadena de una curva no contiene elementos 3 ni 1, entonces es una curva en dos dimensiones. Se probó también la medida de similitud para curvas en dos dimensiones dando resultados favorables.

Una limitante de esta representación es su sensibilidad al ruido, y por lo tanto se asumió que las curvas no contenían ruido.

Como trabajo a futuro se propone dar una solución al problema de escala para escalas no enteras, estudiar formalmente las funciones de similitud propuestas y optimizar los algoritmos.

REFERENCIAS

[Alt y Godeau, 1995]. Alt Helmut y Godeau Michael. Computing the Fréchet distance between two polygonal curves. *International Journal of Computational Geometry & Applications*, páginas 75-91, 1995.

[Alt y Guibas, 1999]. Alt Helmut y Guibas Leonidas. Discrete Geometric Shapes: Matching, Interpolation and Aproximation. In J. Sack and J. Urrutia, editors, *Handbook of Computational Geometry*, pp. 121-153. North Holland, Amsterdam, 1999.

[Attneave, 1954]. Attneave F, Some informational aspects of visual perception. *Psychol. Rev.* 61, 183-193, 1954.

[Attneave y Arnoult, 1966]. Attneave F y Arnoult M. D. , The quantitative study of shape and pattern perception. *Pattern recognition*, L Uhr, ed., pp. 123-141, Wiley, New York, 1966.

[Belogay et al, 1997]. E. Belogay, C. Cabrelli, U. Molter, R. Shonkwiler. Calculating the Hausdorff distance between curves. *Information Processing Letters*, v. 64, pp. 17-22, 1997.

[Besl y McKay, 1992]. Besl P J. y McKay N. D. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14:239-256, 1992.

[Biederman, 1987]. Biederman, I. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94, 115-147, 1987.

[Brady, 1983]. Brady M. Criteria for representations of shape, in *Human and Machine Vision*. J. Beck, B. Hope y A. Rosenfeld, eds, pp. 39-84. Academic Press, New York, 1993.

[Bribiesca, 2000]. Bribiesca E. A chain code for representing 3D curves. *Pattern Recognition*, 33, 755-765, 2000.

[Bribiesca y Velarde, 2001]. Bribiesca E. y Velarde C. A Formal Language Approach for 3D Curve Representation. *Computers & Mathematics with applications*, 42, 1571-1584, 2001.

[Chen, 1990]. Chen S. , Keller J. y Crownover R. , Shape from fractal geometry, *Artificial Intell.* 43, 199-218, 1990.

[Chen, 1993]. Chen C. Improved moment invariants for shape discrimination. *Pattern Recognition*, 26(5): 683-686, 1993.

- [Davis, 1977]. Davis L. Understanding shape: Angles and sides, IEEE. Trans. Comput. 26, 236-242, 1977.
- [Efrat y Katz, 1996]. Efrat A. y Katz M. J. Computing fair and bottleneck matchings in geometric graphs. International Symposium on Algorithms and Computation, páginas 115-125, 1996.
- [Fischler y Bolles, 1986]. Fischler M. A y Bolles R. C. Perceptual organization and curve partitioning, IEEE Trans. Pattern Anal. Mech, Intell, 8, 100-105, 1986.
- [Freeman, 1974]. Freeman H. Computer processing of line drawing images. ACM Computing Surveys 6, 57-97, 1974.
- [Garner, 1974]. Garner, W. R. The processing of information and structure. New York: Wiley. 1974.
- [Gibson, 1950]. Gibson J. J. The Perception of the Visual World. Houghton, 1950.
- [Gigus et al, 1991]. Gigus Z et al. Efficiently computing and representing aspects graphs and polyhedral objects. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13:542-551, 1991.
- [Hagedoorn, 2000]. Hagedoorn M. Pattern matching using similarity measures. Utrecht University. Phd Thesis, 2000.
- [Haralick y Shapiro, 1991]. Haralick R. y Shapiro L. Glossary of computer vision terms. Pattern Recognition, 24, 69-93, 1991.
- [Hebb, 1949]. Hebb, D. O. The Organization of Behavior. Wiley, New York, 1949.
- [Helson, 1926]. Helson, H.H. The psychology of Gestalt. Amer. J. Psychol. 1926.
- [Hoff y Ahuja, 1989]. Hoff W. y Ahuja N. Surfaces from stereo: Integrating feature matching, disparity estimation, and contour detection, IEEE Trans. Pattern Anal. Mech. Intell. 11, 121-136, 1989.
- [Hoffman y Richards, 1984]. Hoffman D.D. y Richards W.A. Parts of recognition, cognition 18, 65-96, 1984.
- [Horaud y Brady, 1988]. Horaud R. y Brady M. On the geometric interpretation of image contours, Artificial Intell. 37, 333-353, 1988.

-
- [Horn, 1990]. Horn B., Height and gradient from shading, *Int. J. Compu. Vision*, 5, pp. 37-75, 1990.
- [Huttenlocher y Kedem, 1990]. Huttenlocher D. P. y Kedem H. Computing the minimum Hausdorff distance for point sets under translation. In *Proc. 6th Annual ACM Symp. Computational Geometry*, pages 340-349, 1990.
- [Kirkpatrick, 2001]. Kirkpatrick, K. Object perception. In R. G. Cook, (Ed.), *Avian visual cognition [On-line]*. Disponible: www.pigeon.psy.tufts.edu/avc/kirkpatrick/. 2001.
- [Koehler, 1929]. Koehler, W. *Gestalt psychology*. New York: Liveright, 1929.
- [Koffka, 1935]. Koffka, K. *Principles of Gestalt psychology*. New York: Harcourt, Brace, 1935.
- [Kong y Kimia, 2001]. Kong W. y Kimia B. On Solving 2D and 3D Puzzles Using Curve Matching. *Proc. IEEE Computer Vision and Pattern Recognition*, pp. 583-590, 2001.
- [Leyton, 1987]. Leyton M. Symmetry-curvature duality, *Comput. Vision Graphics Image Process*, 38, 327-341, 1987.
- [Li, 1993]. Li S. Z. Similarity Invariants for 3D Space Curve Matching. In *Proceedings of the First Asian Conference on Computer Vision*, pages 454-457, Osaka, Japan. 1993.
- [Lo Chong-Huah y Don Hon-Son, 2000]. Invariant Representation and Matching of Space Curves. *Journal of Intelligent and Robotic Systems*, 28 (1/2), pp. 125-149, June 2000.
- [Lowe, 1987]. Lowe D. G. Three-dimensional object recognition from single two-dimensional images, *Artificial Intell.* 31, 355-395, 1987.
- [Markman y Gentner, 1993]. Markman, A. B. y Gentner, D. Structural alignment during similarity comparisons. *Cognitive Psychology*, 25, 431-467. 1993.
- [Marr y Hildreth, 1980]. Marr D. y Hildreth E. Theory of edge detection, *Proc. Roy. Soc. London B207*, 187-217, 1980.
- [Marr y Nishihara, 1978]. D. Marr y H. Nishihara. Representation and recognition of the spatial organization of three-dimensional shapes. *Proc. Roy. Soc. London, B200*, 269-294, 1978.
- [Nosofsky, 1991]. Nosofsky, R. M. Stimulus bias, asymmetric similarity, and clasification. *Cognitive Psychology*, 23, 94-140. 1991.
-

- [Nosofsky, 1992]. Nosofsky, R. M. Similarity scaling and cognitive process models. *Annual Review of Psychology*, 43, 25-53. 1992.
- [O'Rourke, 1985]. O'Rourke Joseph Curve Similarity via Signatures, pages 295-317. North-Holland, 1985.
- [Otterloo, 1992]. P. J. van Otterloo. A Contour-Oriented Approach to Shape Analysis. Hemel Hempstead, Prentice Hall, 1992.
- [Pavlidis y Horowitz, 1974]. Pavlidis T. y Horowitz S. L. Segmentation of plane curves, *IEEE Trans. Comput*, 23, 860-870, 1974.
- [Pavlidis, 1978]. Pavlidis T. A review of algorithms for shape analysis, *Comput. Graphics Image Process.* 7, 243-258, 1978.
- [Pentland, 1986]. Pentland A. Perceptual organization and the representation of natural form, *Artificial Intell.* 28, 293-331, 1986.
- [Rodriguez et al, 2001]. Rodriguez, Last, Kandel y Bunke. Geometric Approach To Data Mining. *International Journal of Image and Graphics*, Vol. 1, No. 2, pp. 363-386, World Scientific Publishing Company, 2001.
- [Rosenfeld y Johnston, 1973]. Rosenfeld A. y Johnston E. Angle detection in digital curves, *IEEE Trans. Comput.* 22, 875-878, 1973.
- [Rosin, 1997]. Rosin Paul L. Techniques for assessing polygonal approximations of curves. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 19(5):659-666, 1997.
- [Sánchez-Cruz y Bribiesca]. Hermilo Sanchez-Cruz y Ernesto Bribiesca. A Method of Optimum Transformation of 3D Objects Used as a Measure of Shape Dissimilarity. Por publicarse en la revista *Image and Vision Computing*.
- [Schwartz y Sharir, 1987]. Schwartz J. T. y Sharir M. . Identification of partially obscured objects in two and three dimensions by matching noisy characteristic curves. *International Journal of Robotic Research*, 6(2):29-44, 1987.
- [Shepard, 1962]. Shepard, R. N. The analysis of proximities: Multidimensional scaling with an unknown distance function. Part I. *Psychometrika*, 27, 125-140. 1962.
- [Sjoberg, 1972]. Sjoberg, L. A cognitive theory of similarity. *Goteborg Psychological Reports*, 2(10). 1972.

-
- [Stokely y Wu, 1992]. Surface parametrization and curvature measurement of arbitrary 3-D objects: five practical methods, IEEE. Trans. Pattern Anal. Mech. Intell. 14, 833-840, 1992.
- [Tversky, 1977]. Tversky, A. Features of similarity. Psychological Review, 84, 327-352. 1977.
- [Tversky y Gati, 1982]. Tversky, A. y Gati, I. Similarity, separability, and the triangle inequality. Psychological Review, 89, 123-154. 1982.
- [Ulupinar y Nevatia, 1990]. Ulupinar F. y Nevatia R. Inferring shape from contour for curved surfaces, in: Proc. Int. Conf. on Pattern Recognition, pp. 147-154, 1990.
- [Veltkamp, 1998]. Veltkamp Remco C. Hierarchical approximation and localization. The Visual Computer, 14(10):471-487, 1998.
- [Wolfson, 1990]. Wolfson H. J. Model based object recognition by geometric hashing. In Proc. 1st European Conference on Computer Vision. 526-536, 1990.
- [Wolfson, 1990]. Wolfson H. J. On Curve Matching. IEEE Transactions On Pattern Analysis and Machine Intelligence. Vol. 12. No. 5. May, 1990.
- [Wolfson y Rigoutsos, 1997]. Wolfson H. J. y Rigoutsos I. Geometric hashing: An overview. IEEE Computational Science & Engineering. 4(4) 1997.