



INAOE

**Representación de Formas Articuladas por
medio de Grafos Pesados basados en
Esqueletos**

por

Laura Alejandra Pinilla Buitrago

Tesis sometida como requisito parcial
para obtener el grado de

**MAESTRA EN CIENCIAS EN EL ÁREA DE
CIENCIAS COMPUTACIONALES**

en el

Instituto Nacional de Astrofísica, Óptica y Electrónica

Noviembre 2014

Tonantzintla, Puebla

Supervisada por:

Dr. Jesús Ariel Carrasco Ochoa

Dr. José Francisco Martínez Trinidad

©INAOE 2014

El autor otorga al INAOE el permiso de reproducir
y distribuir copias en su totalidad o en
partes de esta tesis



Resumen

Partiendo de la noción de que el cerebro humano puede reconocer imágenes haciendo uso únicamente de las siluetas de los objetos, han surgido líneas de investigación interesadas en abstraer información de la forma de los mismos. Los dos enfoques principales para el reconocimiento de formas son: basados en contorno y basados en esqueleto. Los métodos basados en esqueleto han demostrado ser más útiles que los basados en contorno cuando los objetos son formas articuladas. Sin embargo, aunque existen diferentes métodos para la construcción de esqueletos a partir de formas $2D$, estos métodos requieren de diferentes parámetros de entrada los cuales son sensibles a las características de las formas o requieren demasiado tiempo de ejecución. Por otra parte, aunque son numerosos los métodos para la obtención de esqueletos a partir de formas, son pocos los trabajos que presentan estrategias para la obtención de características a partir de los esqueletos, de tal manera que puedan ser utilizadas en tareas de identificación, clasificación y reconocimiento de formas.

En esta tesis se propone un nuevo método para la obtención de esqueletos sin pérdida de ramas importantes y libre de ramas no deseadas. Posteriormente, son construidos grafos pesados a partir de características obtenidas desde el esqueleto y éstos son a su vez utilizados en tareas de clasificación.

El método de clasificación propuesto en este trabajo utiliza *OSB* (del inglés *Optimal Subsequence Bijection*) para obtener un valor de similitud en función de las correspon-

dencias encontradas entre pares de píxeles finales para dos esqueletos representados como grafos pesados. La estrategia de clasificación propuesta en este trabajo permite una alta reducción del costo computacional y produce mejores resultados de clasificación respecto a métodos presentados en la literatura que utilizan *OSB* para comparar grafos.

Abstract

Based on the notion that human brain can recognize images only using the silhouettes of objects, several researches related to obtaining information from the shapes have been developed. There are two main approaches for shape recognition: based on contour and based on skeleton. Methods based on skeleton have shown be more useful on articulated shapes than those based on contour. However, although there are different methods for building skeletons from 2D shapes, these methods need several input parameter values, which are sensitive to shape characteristics or these methods spend too much execution time. In spite that there are a lot of methods for obtaining skeletons from shapes, there are few studies presenting strategies for obtaining characteristics from the skeletons, such that these can be used on tasks like identification, classification and recognition.

In this thesis, we propose a new method for computing skeletons from articulated shapes without losing significant branches and free of unwanted branches. Then, using the characteristics obtained from the skeletons, weighted graphs are constructed and used on classification tasks.

The classification method proposed in this thesis use *OSB* (Optimal Subsequence Bijection) for obtaining a similarity value according the matching between pairs of endpoints of two skeleton represented as weighted graphs. The classification strategy proposed in this work allows reducing the runtime and produce better classification results than other methods in the literature, which used the *OSB* method for comparing graphs.

Índice general

Resumen	3
1. Introducción	13
1.1. Problemática actual	15
1.2. Motivación	16
1.3. Objetivo general	17
1.4. Objetivos específicos	17
1.5. Aportaciones	18
1.6. Organización de esta tesis	18
2. Marco Teórico	19
2.1. Conceptos básicos utilizados en la construcción de esqueletos	19
2.2. Teoría de grafos	27
3. Trabajo Relacionado	30
3.1. Métodos para cálculo del esqueleto sin poda	30
3.2. Métodos para el cálculo del esqueleto con poda	33
3.3. Discusión	41
4. Método Propuesto	44
4.1. Construcción de esqueletos	45

4.1.1.	Segmentación del contorno a trozos	45
4.1.2.	Construcción del esqueleto	48
4.1.3.	Podado del esqueleto	49
4.2.	Módulo de extracción de características y construcción de grafos pesados	53
4.3.	Módulo de clasificación	58
4.3.1.	Evaluación del parecido entre grafos	60
4.3.2.	Clasificación de formas articuladas	65
5.	Resultados Experimentales	66
5.1.	Construcción de esqueletos	68
5.2.	Clasificación	85
5.3.	Discusión	88
6.	Conclusiones	91
6.1.	Conclusiones	91
6.2.	Aportaciones del trabajo de investigación	93
6.2.1.	Publicaciones derivadas de este trabajo de tesis	93
6.3.	Trabajo futuro	94
	Notación	95
	Bibliografía	96

Índice de figuras

2.1. Ejemplos de formas articuladas	20
2.2. Vecindad $N_8(p)$ de un pixel p en una imagen articulada binaria	21
2.3. Mapa de distancia utilizando la distancia Euclidiana	23
2.4. Disco máximo y centro de disco máximo	24
2.5. Esqueleto de una imagen binaria	25
2.6. Punto cóncavo	26
2.7. Punto convexo	27
2.8. Camino entre dos vértices en un grafo	28
2.9. Camino geodésico entre dos vértices en un grafo	28
3.1. Esqueleto obtenido por el algoritmo de adelgazamiento secuencial pro- puesto por Zhou et al. (1995).	31
3.2. Esqueleto obtenido por el algoritmo de adelgazamiento paralelo propuesto por Zhang & Suen (1984).	32
3.3. Esqueletos de una estrella obtenidos mediante diagramas de Voronoi y mapa de distancia	33
3.4. Esqueleto obtenido mediante operaciones morfológicas	34
3.5. Esqueleto de un camello obtenido a partir de un algoritmo basado en el mapa de distancia Euclidiana (Hesselink & Roerdink, 2008)	35

3.6. Pixeles del contorno obtenido con <i>DCE</i>	37
4.1. Método de Representación de formas articuladas por medio de grafos pesados basados en esqueleto	44
4.2. Método para la obtención de esqueletos a partir de formas articuladas 2D	45
4.3. Segmentación del contorno a trozos	47
4.4. Ramas incidentes en los puntos dominantes cóncavos	48
4.5. Construcción del esqueleto en función del contorno segmentado a trozos.	49
4.6. Esqueleto de un elefante con ramas finales incidentes en los puntos dominantes convexos del contorno	50
4.7. Contribución visual de una protuberancia en función del área local	51
4.8. Podado de ramas no deseadas	52
4.9. Esqueletos de formas articuladas y su topología	54
4.10. Centroides de formas articuladas y centroides ajustados al camino geodésico más largo de los esqueletos	55
4.11. Secuencia de radios obtenidos desde el centroide hacia pixeles finales del esqueleto de un elefante	57
4.12. Obtención de grafos a partir del esqueleto	58
4.13. Correspondencias entre dos grafos G y G'	61
4.14. Correspondencias entre grafos para dos imágenes con diferente rotación	64
4.15. Ordenamiento de la matriz $\gamma(G, G')$	64
5.1. Base de datos Mio et al. (2007)	68
5.2. Base de datos Aslan & Tari (2005)	68
5.3. Base de datos Aslan et al. (2008)	69
5.4. Base de datos Baseski et al. (2009)	70
5.5. Base de datos Latecki & Lakämper (2000)	71

5.6. Esqueletos generados por el método propuesto para formas articuladas bajo transformaciones geométricas y escalamiento	72
5.7. Esqueletos obtenidos para diferentes formas articuladas por el método propuesto	73
5.8. Esqueletos obtenidos en diferentes etapas por el método propuesto	74
5.9. Esqueletos obtenidos para diferentes formas articuladas por los métodos Zhang & Suen (1984), Bai et al. (2007), Montero & Lang (2012) y el método propuesto	75
5.10. Tiempo de ejecución y calidad del esqueleto (<i>RER</i>) para cada clase en la base de datos (Mio et al., 2007)	79
5.11. Tiempo de ejecución y calidad del esqueleto (<i>RER</i>) para cada clase en la base de datos Aslan & Tari (2005)	81
5.12. Tiempo de ejecución y calidad del esqueleto (<i>RER</i>) para cada clase en la base de datos Aslan et al. (2008)	82
5.13. Tiempo de ejecución y calidad del esqueleto (<i>RER</i>) para cada clase en la base de datos Baseski et al. (2009)	83
5.14. Tiempo de ejecución y calidad del esqueleto (<i>RER</i>) para cada clase en la base de datos Latecki & Lakämper (2000)	84

Índice de tablas

3.1. Comparación entre los métodos de eskeletonización	43
5.1. Comparación de los resultados obtenidos mediante el método propuesto y los métodos Zhang & Suen (1984), Bai et al. (2007) y Montero & Lang (2012) en función del tiempo de ejecución y la calidad del esqueleto (<i>RER</i>) para una sola forma articulada	77
5.2. Información sobre el tamaño de las imágenes en las bases de datos utilizadas en la construcción de esqueletos	78
5.3. Promedio en tiempo de ejecución y error de reconstrucción (<i>RER</i>) para la base de datos Mio et al. (2007)	79
5.4. Promedio en tiempo de ejecución y error de reconstrucción (<i>RER</i>) para la base de datos Aslan & Tari (2005)	81
5.5. Promedio en tiempo de ejecución y error de reconstrucción (<i>RER</i>) para la base de datos Aslan et al. (2008)	82
5.6. Tiempo promedio de ejecución y error de reconstrucción (<i>RER</i>) para la base de datos Baseski et al. (2009)	83
5.7. Tiempo promedio de ejecución y el error de reconstrucción (<i>RER</i>) para la base de datos Latecki & Lakämper (2000)	84
5.8. Bases de datos utilizadas en el proceso de clasificación	85

5.9. Resultados de la clasificación, en términos del porcentaje de aciertos (<i>exactitud</i>), utilizando el método propuesto y el método de Bai & Latecki (2008) en varias colecciones de imágenes	87
--	----

Capítulo 1

Introducción

El reconocimiento de imágenes ha sido, desde hace varios años, un problema de investigación abierto en la literatura. Las técnicas más utilizadas para el reconocimiento de imágenes binarias 2D de formas articuladas tienen dos enfoques principales: métodos basados en contorno y métodos basados en esqueleto. Estudios recientes (Belongie et al., 2002; Sebastian & Kimia, 2005; Bai & Latecki, 2008; Bai et al., 2009) han demostrado que los métodos basados en contorno resultan ser adecuados cuando las variaciones entre objetos al interior de una clase son pequeñas. Por el contrario, los métodos basados en esqueleto son los más apropiados cuando hay variación entre las figuras que pertenecen a una misma clase; producto del movimiento en las articulaciones, oclusión, cambio de posición (perspectiva del objeto), entre otras (Sebastian & Kimia, 2005; Bai & Latecki, 2008; Bai et al., 2009; Macrini et al., 2011).

Los esqueletos son una herramienta útil y poderosa, puesto que resumen, sintetizan y ayudan a entender el objeto, debido a que preservan la información estructural que integra las características geométricas y topológicas de las figuras (Rodríguez et al., 2002; Prasad, 2007; Bai & Latecki, 2008). Por lo anterior, dicha representación ha sido ampliamente utilizada en aplicaciones basadas en video e imágenes para el análisis y reconocimiento

de objetos, tales como recuperación de imágenes y computación gráfica (Sundar et al., 2003), reconocimiento de caracteres (Bag & Harit, 2011; Bag et al., 2014), procesamiento de imágenes y análisis de imágenes biomédicas (Wang et al., 2012). A su vez, han ido ganando importancia en áreas como animación por computadora (Song et al., 2013), geometría diferencial y biometría (Wu et al., 2014).

Existen diferentes métodos para la construcción de esqueletos (Brady & Asada, 1984; Martínez-Pérez et al., 1987; Ogniewicz & Kübler, 1995; Ge & Fitzpatrick, 1996; Dimitrov et al., 2003; Krinidis & Krinidis, 2013); sin embargo, la construcción del esqueleto es sensible al ruido y a deformaciones en el contorno que generan ramas no deseadas, provocando que la obtención de un esqueleto sencillo y consistente ¹ sea un problema no trivial.

Usualmente, el esqueleto debe ser podado para obtener un esqueleto sencillo que represente a la figura. Muchos métodos de podado han sido propuestos (Bai et al., 2007; Liu et al., 2012; Montero & Lang, 2012; Krinidis & Krinidis, 2013). En estos trabajos, los autores han obtenido buenos resultados, pero sus métodos requieren de varios parámetros de entrada que deben ser cambiados para cada figura o grupo de figuras similares, según la percepción humana.

Posteriormente, el esqueleto puede ser traducido a un grafo, donde ciertos puntos del esqueleto corresponden con los nodos del grafo, mientras que las aristas se obtienen a partir de los conjuntos de puntos que conectan entre sí a estos puntos (Rodríguez et al., 2002). No obstante, los grafos por sí solos no son útiles en tareas de clasificación o reconocimiento, puesto que objetos que pertenecen a la misma clase pueden tener mayor o menor número de nodos (producto de las articulaciones, el ruido, la posición, las características del objeto, entre otras), haciendo que los grafos sean topológicamente ² diferentes,

¹Relación entre las partes importantes del objeto y las ramas que las representan en el esqueleto

²En teoría de grafos, la topología determina la manera en que los elementos de un grafo (vértices y aristas) están interconectados. Se dice que dos grafos son topológicamente equivalentes, si son isomorfos; es decir si tienen el mismo número de vértices y si las relaciones de adyacencia entre los vértices son las mismas para ambos grafos.

mientras que objetos de diferentes clases pueden estar representados por medio de grafos con la misma topología (Bai & Latecki, 2008). Teniendo en cuenta lo anterior, es necesario extraer información a partir del objeto y su topología y relacionarla con el grafo, generando un grafo pesado. Por otro lado, la mayoría de los métodos determinan la similitud entre los grafos a partir de la correspondencia entre vértices o aristas; pero en este caso, debido a la inestabilidad en la topología (formas articuladas visualmente similares son representadas por diferentes grafos, mientras que formas articuladas diferentes son representadas con grafos muy parecidos) es necesario desarrollar estrategias diferentes, que permitan obtener valores de similaridad para objetos representados por grafos, obtenidos a partir de la construcción de esqueletos.

Teniendo en cuenta los aspectos anteriormente mencionados, esta tesis está enfocada en dos problemas principales: proponer un método que genere esqueletos sencillos y que requiera un único parámetro de entrada; de tal manera que tal parámetro genere buenos esqueletos respecto a los reportados en la literatura, para un amplio conjunto de figuras articuladas. Además, se estudiarán diferentes características a obtener a partir del esqueleto, con el fin de generar grafos con pesos asignados en sus vértices y/o aristas. La estructura topológica de los grafos y los pesos asignados a estos, deben proporcionar información útil para llegar a cabo tareas de clasificación.

1.1. Problemática actual

Un problema vigente en áreas como visión por computadora y reconocimiento de patrones es la necesidad de sintetizar y abstraer la información contenida en los objetos, de tal manera que sea útil en tareas como la clasificación de imágenes. Una técnica ampliamente utilizada para cubrir esta necesidad es la construcción de esqueletos (Sebastian & Kimia, 2005). Sin embargo, los esqueletos obtenidos con métodos convencionales vienen acompañados de ramas no deseadas que contienen información redundante o ruido, di-

ficultando el reconocimiento de los objetos. Por lo tanto, técnicas de podado deben ser incluidas con el objetivo de obtener un esqueleto simple que represente y preserve las características que describen al objeto.

Por otra parte, los métodos de reconocimiento basados en esqueletos deben atender a un desafío mayor, el cual radica en encontrar correspondencias entre esqueletos con topologías diferentes para imágenes que pertenecen a una misma clase o esqueletos topológicamente iguales para objetos de diferentes clases (Bai & Latecki, 2008); esto hace que no sea posible calcular valores de similitud a partir de correspondencias uno a uno entre los esqueletos, los cuales usualmente son traducidos a árboles o grafos.

En la literatura se han propuesto diferentes métodos para la poda de los esqueletos y se han presentado diferentes métodos de clasificación de imágenes basados en esqueletos, los cuales han reportado buenos resultados. Sin embargo, los métodos actuales presentan algunos problemas descritos a continuación.

Problema 1. Los algoritmos para la obtención de esqueletos podados requieren parámetros de entrada dependientes de la percepción humana, los cuales deben ser modificados para cada imagen de entrada o para conjuntos de imágenes con características similares, lo cual los hace poco prácticos.

Problema 2. Los métodos que utilizan esqueletos para la clasificación de imágenes binarias 2D de formas articuladas utilizan información obtenida principalmente del esqueleto; lo cual, debido a la inconsistencia topológica entre los grafos de esqueletos, puede ser poco útil para tareas más complejas como la clasificación.

1.2. Motivación

Como se mencionó anteriormente, disponer de un algoritmo de podado de esqueletos que requiera pocos parámetros de entrada, sin requerir constantemente de la intervención

humana, es una necesidad en aplicaciones prácticas, tales como recuperación y clasificación de imágenes basadas en esqueleto. Por lo tanto, el desarrollo de un método no paramétrico o que requiera de pocos parámetros de entrada y cuyos valores no sean dependientes de cada imagen, para la obtención de esqueletos podados, es una línea abierta de investigación.

Por otra parte, para la clasificación de imágenes basada en esqueletos es necesario estudiar qué tipo de información se puede obtener a partir del esqueleto, y de la figura, y de qué manera dicha información pueda ser combinada y relacionada de tal modo que sea útil para encontrar la correspondencia entre grafos obtenidos a partir de los esqueletos así como en el cálculo de su similitud, permitiendo obtener buenos resultados en tareas de clasificación.

1.3. Objetivo general

Desarrollar un método que permita la construcción de grafos pesados a partir de esqueletos obtenidos de formas articuladas de imágenes 2D. Los grafos mantendrán información útil de las imágenes, de tal manera que se obtengan mejores resultados en tareas de clasificación, respecto a los reportados en la literatura.

1.4. Objetivos específicos

- Desarrollar un método para la construcción de esqueletos a partir de formas articuladas.
- Proponer una representación del esqueleto, así como de la información que se pueda obtener del mismo, mediante grafos pesados.
- Evaluar la utilidad de los grafos obtenidos a partir de figuras 2D, en tareas de clasificación.

1.5. Aportaciones

- Un método para generar esqueletos a partir de imágenes binarias 2D de formas articuladas, el cual requiera pocos parámetros y genere resultados consistentes para conjuntos de datos que contienen formas articuladas con diferentes características.
- Un método para construir grafos pesados, a partir de las características obtenidas del esqueleto, que sean útiles en tareas de clasificación.

1.6. Organización de esta tesis

En el capítulo 2 se presentan los conceptos básicos requeridos para entender el resto del documento.

En el capítulo 3 se describen los trabajos más relevantes relacionados con la construcción de esqueletos y con técnicas de podado propuestas en la literatura, así como trabajos sobre clasificación de imágenes basada en esqueletos.

En el capítulo 4 se describe el método propuesto para la obtención de grafos pesados mediante la construcción de esqueletos a partir de formas articuladas 2D.

En el capítulo 5 se muestran los resultados experimentales obtenidos al evaluar la utilidad de los grafos pesados, obtenidos a partir de los esqueletos, en tareas de clasificación.

Por último, en el capítulo 6 se presentan las conclusiones de este trabajo y posibles direcciones para trabajo futuro.

Capítulo 2

Marco Teórico

En este capítulo se dan los conceptos básicos necesarios para entender el resto del documento.

2.1. Conceptos básicos utilizados en la construcción de esqueletos

En esta tesis se pretende obtener grafos pesados a partir de imágenes binarias 2D de formas articuladas, por lo tanto empezaremos presentando el concepto de forma articulada.

Definición 2.1.1 (Forma articulada). Una forma O se dice que es articulada si tiene partes articuladas, esto significa que:

1. O puede ser descompuesta en varias partes O_i ($i = 1, 2, \dots, n$).
2. Dos partes adyacentes O_i y O_j están conectadas por una parte de unión J_{ij} .
3. Las uniones J_{ij} entre partes adyacentes son más pequeñas en comparación con las partes adyacentes O_i y O_j .

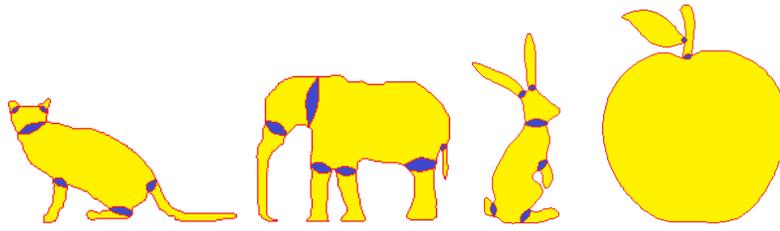


Figura 2.1: Ejemplos de formas articuladas. Los segmentos amarillos son partes articuladas y los segmentos en azul son conexiones

Teniendo en cuenta lo anterior, una forma articulada O de n partes es definida como:

$$O = \{\bigcup_{i=1}^n O_i\} \cup \{\bigcup_{i \neq j} J_{ij}\}, \text{ donde:}$$

- $\forall i, 1 \leq i \leq n, O_i \subset O$ con $O_i \cap O_j = \emptyset, \forall i \neq j, 1 \leq i, j \leq n$.
- $\forall i, \forall j, i \neq j, 1 \leq i, j \leq n, J_{ij} \subset O$, es la unión entre O_i y O_j . Si O_i y O_j no están unidos, entonces $J_{ij} = \emptyset$. De otra manera, $J_{ij} \cap O_i \neq \emptyset$ y $J_{ij} \cap O_j \neq \emptyset$

En la *figura 2.1* mostramos algunos ejemplos de formas articuladas. Una forma articulada 2D puede ser convertida en una imagen binaria, lo anterior facilita el procesamiento y el análisis de la imagen.

Definición 2.1.2 (Imagen binaria). Una *imagen binaria* S obtenida a partir de O , se representa mediante una matriz de tamaño $m \times n$ (donde m y n son el alto y el ancho de la imagen respectivamente) que en cada pixel $S(k, l)$ (para $k = 1, 2, \dots, m$ y $l = 1, 2, \dots, n$) almacena un solo bit (0 ó 1). Normalmente, los dos colores utilizados para una imagen binaria son blanco y negro, donde el 1 le corresponde al blanco y representa la información del objeto (en inglés *foreground*) y el 0 al negro y corresponde al fondo del objeto (en inglés *background*)(ver *figura 2.2b*).

En las técnicas de procesamiento y reconocimiento de imágenes existen dos términos ampliamente utilizados para identificar las relaciones entre los pixeles: *vecindad* y *conectividad*.

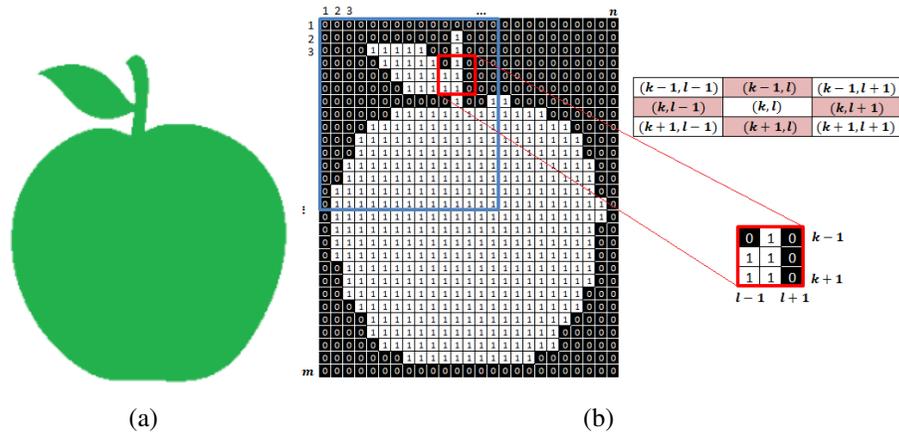


Figura 2.2: *a*). Forma articulada *b*). Imagen binaria obtenida a partir de *(a)* y vecindad $N_8(p)$ para el pixel en la posición (k, l)

Definición 2.1.3 (Vecindad). Un pixel p en las coordenadas (k, l) tiene dos vecinos horizontales y dos vecinos verticales, cuyas coordenadas están dadas por:

$$(k, l - 1), (k, l + 1), (k - 1, l) \text{ y } (k + 1, l)$$

Este grupo de pixeles se denota como $N_4(p)$, que significa, vecindad 4 de p . De la misma forma, un pixel p tiene 4 vecinos diagonales que se denotan como $N_D(p)$, cuyas coordenadas son:

$$(k - 1, l - 1), (k - 1, l + 1), (k + 1, l - 1) \text{ y } (k + 1, l + 1)$$

La unión de los pixeles de $N_4(p)$ junto con los de $N_D(p)$ dan como resultado la vecindad 8 de p , definida como:

$$N_8(p) = N_4(p) \cup N_D(p)$$

En la parte inferior derecha de la *figura 2.2b* podemos ver el pixel (de la imagen binaria en la *figura 2.2b*) en la posición $(5, 12)$ y su vecindad $N_8(p)$ (marcados con el recuadro rojo). En la parte superior derecha de la *figura 2.2b*, podemos observar las vecindades $N_4(p)$ y $N_D(p)$ de un pixel en la posición (k, l) .

Definición 2.1.4 (Conectividad). En una imagen binaria S , dos pixeles están conectados si son adyacentes y satisfacen un criterio especificado de similitud. De esta manera, podemos decir que en una imagen binaria con posibles valores $\{0, 1\}$, en donde 1 representa a los pixeles que forman parte del objeto (*foreground*), existen dos tipos de conectividad:

- **Conectividad 4:** Dos pixeles p y q están conectados si $p = 1$ y $q = 1$, y además q pertenece a $N_4(p)$.
- **Conectividad 8:** Dos pixeles p y q están conectados si $p = 1$ y $q = 1$, y además q pertenece a $N_8(p)$.

Teniendo en cuenta lo anterior, el pixel en la posición $(5, 12)$ en la *figura 2.2c* tiene 4 pixeles conectados en su vecindad $N_8(p)$.

Debido a que en este trabajo se utilizan esqueletos (obtenidos desde imágenes 2D) para obtener grafos pesados que serán usados en la clasificación de formas articuladas, se presentan a continuación algunos términos utilizados en la definición formal de esqueleto.

Definición 2.1.5 (Transformada de distancia). Dada una imagen binaria S , la transformación de distancia DT (en inglés *Distance Transform*) asocia a cada pixel en el *foreground* la distancia al punto más cercano del *background* (ver *ecuación 2.1*).

$$\forall (S(k, l) \in S) / (S(k, l) = 1) : DT(k, l) = \min_{\forall (S(i, j) \in S) / S(i, j) = 0} d(S(k, l), S(i, j)) \quad (2.1)$$

Donde $d(\cdot, \cdot)$ es la distancia de un pixel a otro. Cualquier métrica de distancia puede ser utilizada, entre las más comunes están la distancia Euclidiana y la distancia de Manhattan.

En la *figura 2.3* podemos ver una sub matriz del mapa de distancia construido a partir de la *figura 2.2b* (parte marcada con el recuadro azul) utilizando la distancia Euclidiana.

Veamos a continuación cómo se obtiene el valor asociado en el *mapa de distancia* del pixel en la posición $(10, 7)$ (marcado en rojo) en la *figura 2.3*.

	1	2	3	4	5	6	7	8	...
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	1	1	1	1	1
4	0	0	0	0	0	1	1.4	2	1.4
5	0	0	0	0	0	0	1	1.4	2
6	0	0	0	0	0	0	0	1	1
7	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	1	1
9	0	0	0	0	1	1	1.4	2	2
10	0	0	0	1	1.4	2	2.2	2.8	3
11	0	0	0	1	2	2.8	3.1	3.6	4
12	0	0	1	1.4	2.2	3.1	4.1	4.4	5
13	0	0	1	2	2.8	3.6	4.4	5.3	5.8
14	0	1	1.4	2.2	3.1	4.1	5	5.8	6.7
15	0	1	2	2.8	3.6	4.4	5.3	6.3	7.2

Figura 2.3: Mapa de distancia utilizando la distancia Euclidiana

Sea $p = (10, 7)$ un pixel del *foreground* y $q_1 = (8, 6)$, $q_2 = (7, 6)$ y $q_3 = (7, 7)$ pixeles en el *background* (marcados en azul) en la figura 2.3, calculamos la distancia Euclidiana entre p y cada q_i , para $i = 1, 2, 3$.

La *distancia Euclidiana* entre dos pixeles p y q_i en las posiciones (k, l) y (i, j) respectivamente, es:

$$d(p, q_i) = \sqrt{(k - i)^2 + (l - j)^2}$$

Según la formula de la distancia Euclidiana, tenemos:

$$d(p, q_1) = \sqrt{(10 - 8)^2 + (7 - 6)^2} = 2.2$$

$$d(p, q_2) = \sqrt{(10 - 7)^2 + (7 - 6)^2} = 3.2$$

$$d(p, q_3) = \sqrt{(10 - 7)^2 + (7 - 7)^2} = 3$$

Si calculamos la distancia entre p y cualquier otro punto del *background* obtenemos un valor mayor. Por lo tanto:

$$DT(10, 7) = \min_{q_i} d(p, q_i) = 2.2$$

el pixel en la posición $(10, 7)$ tiene un valor asociado en su *mapa de distancia* de 2.2,

el cual corresponde a la distancia Euclidiana entre el pixel en la posición $(10, 7)$ y el pixel en la posición $(8, 6)$ que es el más cercano a p .

Definición 2.1.6 (Transformada de características). Dada una imagen binaria S , la transformada de características FT (en inglés *Feature Transform*) asocia a cada pixel en el *foreground* el par ordenado (x, y) que corresponde a las coordenadas del punto más cercano en el *background*, obtenido a partir de la *transformada de distancia DT*.

Definición 2.1.7 (Disco máximo). Se dice que un disco D es un *disco máximo* en un conjunto A si:

- $D \subseteq A$
- Para cualquier otro disco E , si $D \subset E \Rightarrow E \not\subseteq A$

Definición 2.1.8 (Centro de disco máximo). Un *centro de disco máximo –CMD* (en inglés *Central Maximal Disk*) es el pixel central en un disco máximo D , etiquetado con el radio del disco (Ver figura 2.4).

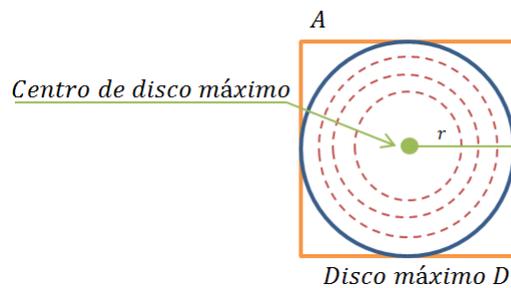


Figura 2.4: Disco máximo (disco en azul) y centro de disco máximo (punto verde) en un conjunto A

Definición 2.1.9 (Esqueleto). El término esqueleto, también conocido como *eje medial* (en inglés *Medial Axis Transform –MAT*), fue inicialmente introducido por Blum (1973). El *esqueleto* es el conjunto de los centros de disco máximo inscritos en una imagen binaria S .

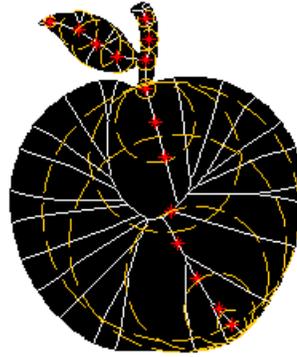


Figura 2.5: Esqueleto de la imagen binaria en la *figura 2.2a*. Algunos discos máximos y centros de disco máximo (línea amarilla discontinua y puntos rojos respectivamente).

$$\text{esqueleto}(S) = \{(p, r) \mid p \in S \text{ y } p \text{ es un centro de disco máximo con radio } r\}$$

Por ejemplo, en la *figura 2.5* se muestra el esqueleto de una manzana (*figura 2.2b*), algunos de sus discos máximos (marcados en azul) y sus respectivos pixeles centrales de disco máximo (marcados en rojo).

La forma original puede ser reconstruida a partir de su esqueleto, a través de la unión de los discos máximos con centro en los pixeles centrales de disco máximo.

Sea F un esqueleto de una imagen binaria S , los pixeles de F pueden dividirse en: pixeles finales, pixeles de unión y pixeles de curva, los cuales se definen como:

Definición 2.1.10 (Pixel final). Un pixel p de un esqueleto F es un *pixel final* si tiene un único pixel o ningún pixel de F en su vecindad $N_8(p)$.

Definición 2.1.11 (Pixel de unión). Un pixel p de un esqueleto F es un *pixel de unión* si tiene más de dos pixeles de F en su vecindad $N_8(p)$.

Definición 2.1.12 (Pixel de curva). Un pixel p de un esqueleto F , es un *pixel de curva* si tiene exactamente dos pixeles de F en su vecindad $N_8(p)$.

Por lo tanto, el esqueleto F de un objeto S puede ser considerado como la unión de los pixeles finales, pixeles de unión y pixeles de curva de F .

$$F = \text{PixelesFinales}(F) \cup \text{PixelesUnion}(F) \cup \text{PixelesCurva}(F)$$

Definición 2.1.13 (Rama). Una *rama* de un esqueleto F , es el conjunto de pixeles de curva adyacentes entre: 1). un pixel final y un pixel de unión, 2). un pixel final y un pixel final; o 3). entre dos pixeles de unión.

Las técnicas para la construcción de esqueletos incluyen etapas tales como obtener el contorno de la forma, identificación de puntos cóncavos y convexos. A continuación presentamos la definición de estos términos, los cuales serán utilizados en el desarrollo de este trabajo.

Definición 2.1.14 (Contorno). El contorno de una forma S , es la curva digital formada por los pixeles que son parte de la región fronteriza o del límite del objeto. Es decir, aquellos puntos $p \in S$ cuya vecindad $N_8(p)$ contiene pixeles del *foreground* y del *background*.

Definición 2.1.15 (Puntos cóncavos). Un pixel $S(i, j)$ perteneciente al contorno de una forma S es un *punto cóncavo* si el ángulo interno formado entre sus pixeles vecinos A y B en el contorno es mayor que π radianes (entre 180° y 360°) (gráficamente, un pixel $S(i, j)$ es cóncavo, si al trazar una recta entre sus puntos vecinos A y B , uno o más pixeles que pertenecen a la recta quedan fuera de S) (ver *figura 2.6*).

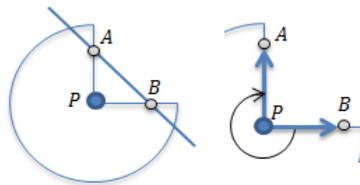


Figura 2.6: Punto cóncavo P respecto a los puntos vecinos A y B

Definición 2.1.16 (Puntos convexos). Un pixel $S(i, j)$ perteneciente al contorno de una forma S es un *punto convexo* si el ángulo interno formado entre sus vecinos A y B en el contorno es menor que π radianes (entre 0° y 180°) (gráficamente, un pixel $S(i, j)$ es convexo, si al trazar una recta entre los puntos A y B , no existe ningún punto de la recta entre A y B que esté fuera de S) (ver figura 2.7).

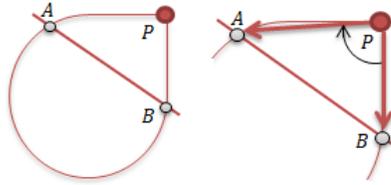


Figura 2.7: Punto convexo P respecto a los puntos vecinos A y B

2.2. Teoría de grafos

Dado que es de nuestro interés representar a los esqueletos como grafos pesados, en esta sección se introducirán algunos conceptos básicos relacionados con la teoría de grafos.

Definición 2.2.1 (Grafo pesado o etiquetado). Un *grafo pesado* es una 4-tupla $G = (V, E, I, J)$, donde:

- V es el conjunto de los elementos conocidos como *vértices*.
- $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$ es el conjunto de los elementos conocidos como aristas. La arista $\{u, v\}$ conecta al vértice u con el vértice v .
- $I : V \rightarrow eVal$ es una *función etiquetadora* encargada de asignar etiquetas a los vértices.
- $J : E \rightarrow eVal$ es un *función etiquetadora* encargada de asignar etiquetas a las aristas.

$eVal$ es el conjunto de posibles valores (etiquetas) que pueden ser asignados a vértices y aristas en un grafo pesado. En esta tesis consideramos $eVal \subseteq \mathbb{R}$

En algunos trabajos, los grafos pesados o ponderados son llamados *ARG* (por sus siglas en inglés *Attributed Relational Graph*).

Definición 2.2.2 (Camino). Un camino en un grafo es una secuencia de vértices conectados por aristas (Ver *figura 2.8*).

Definición 2.2.3 (Longitud de un Camino). En un grafo con pesos asociados a cada arista (grafo pesado), la longitud de un camino es la suma de los pesos asignados a cada arista que forman parte del camino.

Definición 2.2.4 (Camino geodésico). Un *camino geodésico* es el camino más corto o de menor longitud entre todos los posibles caminos que unen dos nodos en un grafo (ver *figura 2.9*).

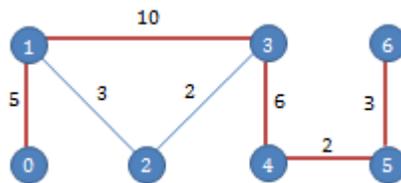


Figura 2.8: Camino desde el vértice 0 al vértice 6 (marcado en rojo)

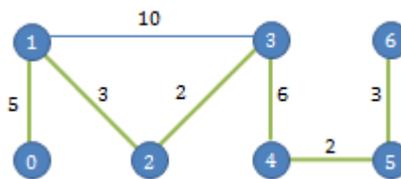


Figura 2.9: Camino geodésico desde el vértice 0 al vértice 6 (marcado en verde)

Definición 2.2.5 (Grafo conexo). Un grafo G se dice *conexo* si, para cualquier par de vértices u y v en G existe al menos un camino de u a v .

Definición 2.2.6 (Ciclo). Un *ciclo* es una sucesión de aristas adyacentes que forman un camino tal que, el vértice inicial y el vértice final son el mismo.

Definición 2.2.7 (Árbol). Un *árbol* es un grafo *conexo* sin *ciclos* en el que dos vértices $\{u, v\}$ están conectados por exactamente un camino.

Definición 2.2.8 (Grafo estrella). En teoría de grafos, una estrella G_k es un grafo *conexo* en el que a lo más un vértice tiene grado mayor que uno (vértice interno) y k hojas.

Capítulo 3

Trabajo Relacionado

En la literatura existen diferentes métodos para la construcción de esqueletos, los cuales pueden ser divididos en dos categorías:

- Métodos para el cálculo del esqueleto sin poda.
- Métodos para el cálculo del esqueleto con poda.

3.1. Métodos para cálculo del esqueleto sin poda

A su vez, los métodos para la construcción de esqueletos sin poda pueden ser divididos en cuatro categorías:

- Algoritmos de adelgazamiento.
- Algoritmos de dominio discreto basados en diagramas de Voronoi.
- Algoritmos basados en transformación de distancia.
- Algoritmos basados en morfología matemática.

Los *algoritmos de adelgazamiento* (Zhang & Suen, 1984; Arcelli & Di Baja, 1985; Martínez-Pérez et al., 1987; Zhou et al., 1995; Leung et al., 2000; Deng et al., 2000; Xie et al., 2003; Huang et al., 2003; Zhu & Zhang, 2008; Bag & Harit, 2011) remueven de forma iterativa pixeles del contorno, lo que permite que la topología de la forma original sea conservada, estos métodos no garantizan que los pixeles del esqueleto final estén ubicados en el eje medial ya que esto depende del orden en que los pixeles del contorno son eliminados. Este tipo de algoritmos pueden ser clasificados a su vez en dos categorías: algoritmos de adelgazamiento *secuencial* (Arcelli & Di Baja, 1985; Martínez-Pérez et al., 1987; Zhou et al., 1995; Zhu & Zhang, 2008; Bag & Harit, 2011) y algoritmos de adelgazamiento *paralelo* (Zhang & Suen, 1984; Leung et al., 2000; Deng et al., 2000; Xie et al., 2003; Huang et al., 2003).

Los algoritmos de *adelgazamiento secuencial* consideran un solo pixel a la vez y visitan cada pixel recorriendo la matriz que contiene la imagen o el contorno de ésta (ver *figura 3.1*).

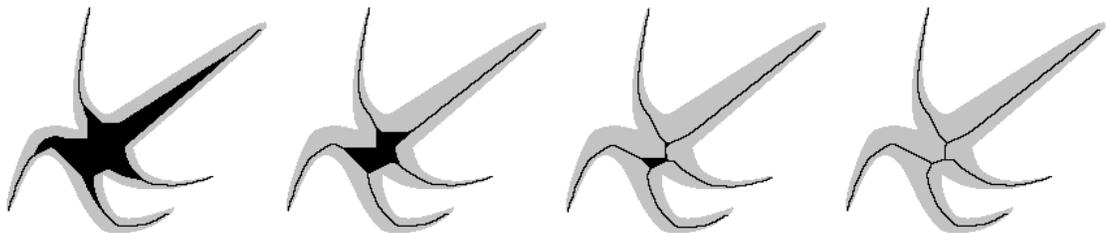


Figura 3.1: Resultados obtenidos por el algoritmo de adelgazamiento secuencial propuesto por Zhou et al. (1995).

Los algoritmos de *adelgazamiento paralelo* consisten en un proceso iterativo, donde la eliminación de cada pixel depende de resultados obtenidos en iteraciones previas. Este tipo de algoritmos, puede operar simultáneamente varios pixeles o dividir la imagen en un número de bloques donde cada bloque es procesado independientemente de forma paralela; sin embargo, aunque este procedimiento puede ser más eficiente computacionalmente, también puede generar esqueletos desconectados o que no preserven la topología

del objeto original (Xie et al., 2003).

En la *figura 3.2* podemos observar el resultado obtenido por el algoritmo de adelgazamiento paralelo propuesto por (Zhang & Suen, 1984), donde cada iteración es dividida en dos sub-iteraciones para ir adelgazando la forma mediante eliminación de píxeles del contorno del objeto.

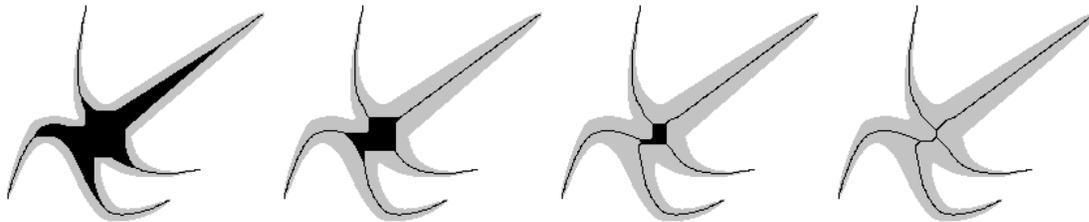


Figura 3.2: Resultados obtenidos por el algoritmo de adelgazamiento paralelo propuesto por Zhang & Suen (1984).

Los *algoritmos de dominio discreto basados en diagramas de Voronoi* (Brandt & Algazi, 1992; Mayya & Rajan, 1994; Ogniewicz & Kübler, 1995; Székely, 2008; Liu et al., 2012) requieren de un conjunto de píxeles de muestreo o píxeles generadores que son parte del contorno de la forma; estos algoritmos asignan a cada píxel generador, los píxeles del plano que están más cercanos a este píxel que a cualquier otro píxel generador. Esto provoca la división del objeto en regiones de Voronoi. Las líneas divisorias entre las regiones que se encuentran en el *foreground* del objeto dan lugar al esqueleto. En la *figura 3.3a* podemos ver los píxeles generadores del esqueleto marcados en rojo y las líneas que forman al esqueleto en negro.

Los *algoritmos basados en transformación de distancia* (Borgefors, 1986; Arcelli & Di Baja, 1993; Ge & Fitzpatrick, 1996; Golland & Leifur Grimson, 2000; Choi et al., 2003) utilizan mapas de distancia para la extracción de los discos máximos (marcados en azul en la *figura 3.3b*), asegurando en la mayoría de los casos la ubicación de los píxeles del esqueleto sobre el eje medial. En la *figura 3.3b* podemos ver el mapa de distancia calculado para una estrella de mar, donde los píxeles más alejados del contorno (más

claros) dan lugar a las ramas del esqueleto (marcado en rojo).

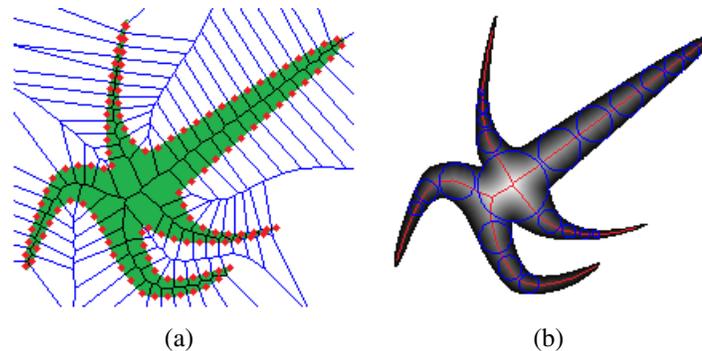


Figura 3.3: *a)* Diagrama de Voronoi obtenido a partir de puntos generadores (marcados en rojo) obtenidos desde el contorno de la estrella. *b)* Mapa de distancia de una estrella y su esqueleto (marcado en rojo)

Por último, los *algoritmos basados en morfología matemática* (Maragos & Schafer, 1986; Zhou & Venetsanopoulos, 1992) realizan operaciones morfológicas sobre un objeto con el objetivo de obtener el esqueleto. Las dos operaciones comúnmente utilizadas son la erosión y la dilatación. La erosión es una operación de contracción, mientras que la dilatación es una operación de expansión, lo anterior se realiza mediante la transformación de la imagen a través de su interacción con otro objeto denominado *elemento estructurante*. En la *figura 3.4d* podemos observar el esqueleto obtenido a partir de una imagen binaria haciendo uso de operadores morfológicos: *erosión* y *dilatación*, con el elemento estructurante *3.4b*.

3.2. Métodos para el cálculo del esqueleto con poda

Desafortunadamente, los métodos descritos anteriormente tienen un inconveniente en común; la construcción del esqueleto es sensible al ruido ¹ y a deformaciones ² presentes

¹En una forma binaria articulada, el ruido es la presencia de información irregular o la variación de píxeles que se encuentran en la imagen, pero que no hacen parte de ella.

²Se considera como deformación, a la porción del contorno de la forma que ha sufrido un cambio de posición, el cual puede ser producto del movimiento o articulación de sus partes.

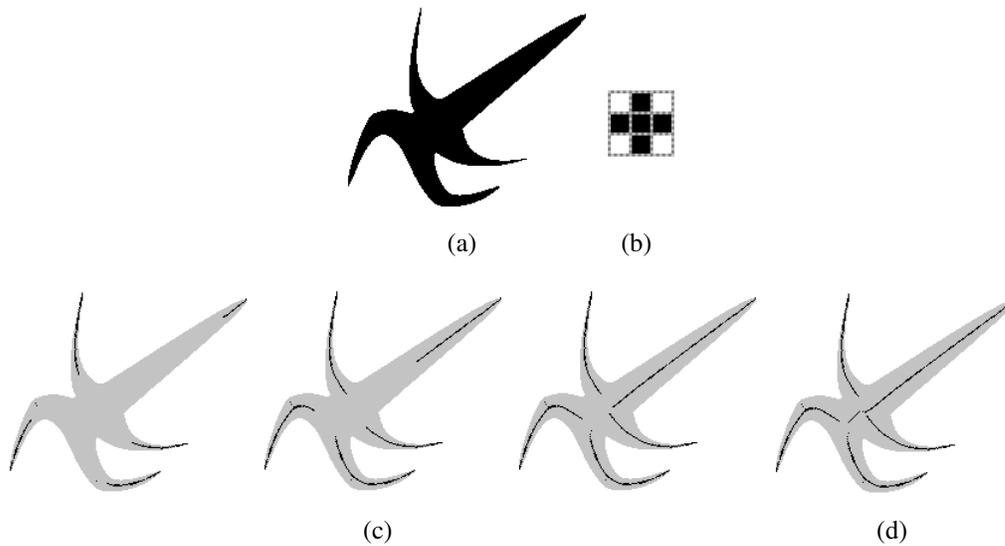


Figura 3.4: *a)* Imagen binaria. *b)* Elemento estructurante para las operaciones de *erosión* y *dilatación*. *c)* Resultados obtenidos en iteraciones intermedias. *d)* Esqueleto final

en las figuras (ver *figura 3.5*), generando ramas redundantes en el esqueleto que distorsionan la topología final (provocando que sean poco útiles en tareas de reconocimiento) (Bai & Latecki, 2007; Shen et al., 2013a). Por esta razón, los métodos propuestos en los últimos años, para la obtención de esqueletos, incluyen diferentes técnicas de podado, con el objetivo de obtener el esqueleto deseado; estas técnicas de podado pueden clasificarse en:

- Suavizado del contorno.
- Asignación de valores de importancia a cada pixel del esqueleto.
- Asignación de valores de importancia a cada rama del esqueleto.

Los métodos que incluyen técnicas de *suavizado de contorno* (Demuth et al., 2010) utilizan este mecanismo para eliminación de ruido presente en la forma, como etapa previa a la construcción del esqueleto; sin embargo, estos métodos pueden cambiar la forma

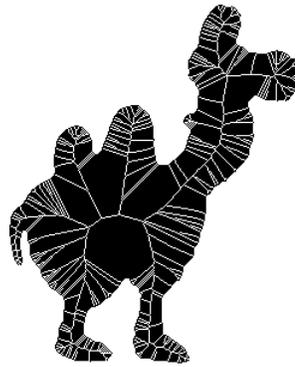


Figura 3.5: Esqueleto de un camello obtenido a partir de un algoritmo basado en el mapa de distancia Euclidiana (Hesselink & Roerdink, 2008)

original del objeto y por lo tanto influir negativamente en el esqueleto resultante (Shaked & Bruckstein, 1998).

La *asignación de valores de importancia a cada pixel del esqueleto* (Duan et al., 2008) permite remover pixeles del esqueleto, los cuales tienen asociado un valor de importancia que no supera un cierto umbral definido. La principal desventaja de estos métodos es que pueden producir esqueletos desconectados. Los *valores de importancia* para cada pixel o rama del esqueleto se calculan con base en su contribución visual, reconstrucción de la región, longitud de la rama, entre otras. De manera análoga, la *asignación de valores de importancia a cada rama del esqueleto* (Ward & Hamarneh, 2010; Liu et al., 2012; Montero & Lang, 2012; Liu et al., 2013; Shen et al., 2013a) permite mantener las ramas más importantes para la representación de la forma a través del esqueleto.

No existen medidas estándar para determinar la importancia de una rama o un pixel en el esqueleto; la importancia puede ser vista desde diferentes perspectivas, provocando diferentes ventajas y limitaciones para cada una de ellas (Attali et al., 2009; Liu et al., 2013). Estas técnicas de podado consideran medidas de importancia locales y/o globales.

Las medidas de importancia *locales* asignan un puntaje a cada rama o pixel del esqueleto, haciendo uso de heurísticas que involucran las partes vecinas de la rama o el pixel en cuestión, provocando que, en algunos casos, ramas o pixeles no importantes sean consi-

derados como importantes (Ward & Hamarneh, 2010; Shen et al., 2011; Montero & Lang, 2012). Por otro lado, las medidas de importancia *globales* calculan la importancia de cada pixel o rama del esqueleto en función de su impacto (al eliminar una rama o pixel) en la apariencia o representación final de la forma (Liu et al., 2012, 2013; Shen et al., 2013a).

A continuación se explican algunos de los métodos más recientes para la construcción de esqueletos que utilizan poda.

En (Bai et al., 2007) se utiliza un método denominado *DCE* (en inglés *Discrete Curve Evolution*). *DCE* remueve de forma iterativa pixeles del contorno, donde cada pixel tiene asociado un valor de importancia respecto a la información global del contorno; los pixeles con los valores de importancia más pequeños se eliminan hasta obtener un subconjunto de pixeles del contorno que mejor representan a la forma (aproximación poligonal) (Latecki & Lakämper, 1999) (ver *figura 3.6*). A partir de los pixeles obtenidos por *DCE*, el contorno es segmentado a trozos y el esqueleto es construido mediante crecimiento de pixeles, partiendo desde el centro de disco máximo inscrito en la forma hacia los pixeles finales convexos sobre el contorno obtenidos por *DCE*. El valor del parámetro de entrada del método de Bai et al. no tiene una relación estricta con el número de pixeles finales del esqueleto, por lo tanto, aunque el método puede remover ramas redundantes, también puede preservar ramas no importantes e ignorar ramas importantes para representar la topología del objeto original.

Bai & Latecki (2007) introducen un método para el podado del esqueleto denominado *DSE* (en inglés *Discrete Skeleton Evolution*). *DSE* remueve ramas finales del esqueleto de manera iterativa, de tal manera que no se alterare la topología del objeto original. En cada iteración, a cada rama se le asigna un peso, el cual se calcula en función de la reconstrucción de la región; posteriormente los pesos son ordenados de menor a mayor, donde la rama con el menor peso es seleccionada y eliminada si su peso es menor a un umbral, de lo contrario el proceso de podado se detiene. El umbral indica el error de reconstrucción máximo permitido. Los esqueletos iniciales son obtenidos con el método propuesto por

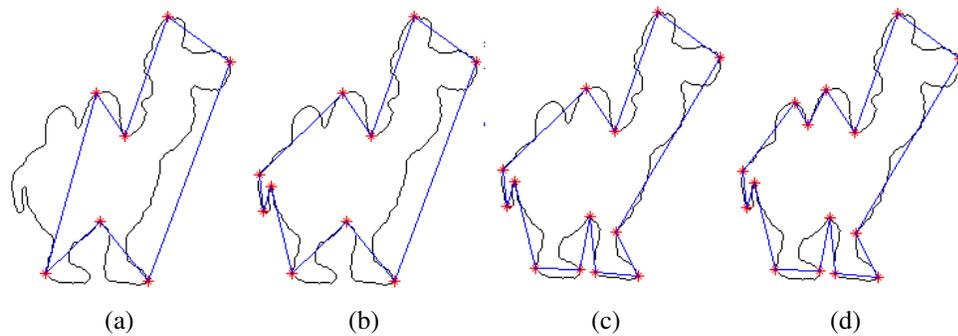


Figura 3.6: Píxeles del contorno obtenido con *DCE*. a) 7 píxeles. b) 10 píxeles. c) 13 píxeles y d) 15 píxeles.

Choi et al. (2003), el cual hace uso del mapa de distancia Euclidiana, construyendo el esqueleto mediante crecimiento de píxeles en relación con un criterio de conectividad en su vecindad 8 (sobre cada píxel que va siendo agregado al esqueleto). El algoritmo propuesto por Bai et al. tiene alto costo computacional, debido a la reconstrucción en cada iteración; sin embargo, para reducir el tiempo de ejecución, los autores obtienen un primer esqueleto podado con el método propuesto en (Bai et al., 2007). En los experimentos, el umbral es definido manualmente para cada conjunto de figuras con características similares.

Aslan et al. (2008) proponen una novedosa representación del esqueleto, mediante la construcción de esqueletos compuestos por ejes desconectados obtenidos a partir del método *TSP* (Tari, Shah y Pien) (Tari et al., 1997) que detecta puntos de simetría a partir de las curvas de nivel. Donde los puntos de simetría representan las partes más importantes del objeto. La principal ventaja de este método es que la construcción no sufre de problemas de inestabilidad, pero en algunas ocasiones se pierde información estructural del objeto.

Yang et al. (2009) utilizan al igual que Bai et al. (2007) *DCE* para obtener el conjunto de puntos sobre el contorno que dan lugar a los píxeles finales del esqueleto. El esqueleto se obtiene mediante crecimiento de píxeles entre pares de píxeles finales, evaluando la vecindad de cada píxel en relación con el mapa de distancia obtenido a partir de la imagen.

Shen et al. (2011) proponen un algoritmo que construye un esqueleto podado mediante crecimiento de píxeles desde el centro de disco máximo inscrito en la forma. Cada píxel candidato (posible píxel a ser agregado al esqueleto) está relacionado con un segmento del contorno. La decisión de agregar o no un píxel al esqueleto es definida mediante una medida de importancia de segmentos del contorno denominada *BRP* (en inglés *Bending Potential Ratio*). El valor de importancia *BPR* para cada píxel candidato es calculado mediante operaciones trigonométricas sobre el píxel en cuestión y su segmento asociado. El algoritmo requiere un parámetro, el cual determina el valor mínimo de importancia que debe tener un píxel, para que éste sea agregado al esqueleto. Aunque el algoritmo propuesto por Shen et al. garantiza la conectividad del esqueleto y genera esqueletos sencillos y descriptivos para figuras con gran cantidad de ruido, el umbral definido para el podado del esqueleto depende estrictamente de la percepción visual humana que se tiene sobre cada figura respecto al esqueleto resultante obtenido.

En (Montero & Lang, 2012) se presenta un método de podado basado en aproximación del contorno y construcción del eje medial, mediante un método denominado IMA (en inglés *Integer Medial Axis*) (Hesselink et al., 2005; Hesselink & Roerdink, 2008). La aproximación del contorno se realiza mediante curvas cúbicas de Bezier (Shao & Zhou, 1996) o aproximación lineal (Horst & Beichel, 1997); estos métodos generan puntos sobre píxeles del contorno, los cuales son utilizados para segmentar el contorno de la forma y generar el esqueleto. Una fase de podado basada en la importancia de cada rama del esqueleto es llevada a cabo para producir el esqueleto final. El método de Motero y Lang a diferencia de otros (Bai et al., 2007; Yang et al., 2009), genera visualmente buenos esqueletos para figuras diferentes, utilizando una misma configuración de los parámetros de entrada.

En (Liu et al., 2012) se calcula un primer esqueleto mediante diagramas de Voronoi a partir de los píxeles sobre el contorno obtenidos por *DCE* (los píxeles obtenidos por *DCE* pertenecen al conjunto de píxeles o puntos generadores). Posteriormente, la importancia de

cada rama es medida a partir de su contribución en la reconstrucción (la cual se determina mediante el área que solo puede ser reconstruida en presencia de la rama que está siendo evaluada, respecto al área total de la forma, medida en píxeles) y su contribución visual (corresponde a la longitud de la rama que no está contenida en el disco máximo con centro en el píxel de unión de la rama en cuestión). Este método tiene tres parámetros que influyen en el esqueleto resultante: el número de píxeles que forman el polígono simplificado por *DCE*, el umbral de error de reconstrucción permitido y el umbral de contribución visual.

Shen et al. (2013a) utilizan reglas Bayesianas para el cálculo de los esqueletos. De manera similar a Bai & Latecki (2007), en cada iteración se asigna un peso a cada rama final del esqueleto (el cual está dado en función de la reconstrucción de la región), donde la rama con el menor peso es eliminada si no supera un umbral. El umbral es un valor calculado utilizando redes Bayesianas, buscando un equilibrio entre reconstrucción de la forma original (la rama que al ser eliminada genere el menor error de reconstrucción) y simplicidad del esqueleto (estructura simple –el menor número de ramas que describan a la forma). Los esqueletos iniciales son generados por el algoritmo basado en el mapa de distancia Euclidiana propuesto por Choi et al. (2003). El método requiere un parámetro de entrada el cual equilibra la precisión de la reconstrucción y la simplicidad del esqueleto. Aunque los esqueletos finales muestran ser estables para diferentes conjuntos de datos. La construcción de los esqueletos tienen un alto costo computacional; a causa de esto, Shen et al. (2013a) proponen partir de esqueletos con varias ramas no deseadas (el número de ramas no deseadas es menor que con (Choi et al., 2003)), los cuales son obtenidos por (Bai et al., 2007) (construcción del esqueleto con ramas finales incidentes en los puntos del contorno obtenidos por *DCE*), para reducir el tiempo de ejecución.

En (Liu et al., 2013) se considera el podado del esqueleto como una decisión multi-objetivo, en donde cada rama del esqueleto es podada mediante fusión de información que involucra: reconstrucción de la región, reconstrucción del contorno y contribución visual (en este método la contribución visual es calculada como la longitud residual de una rama,

que corresponde a los píxeles fuera del disco máximo del píxel de unión de dicha rama); combinando de esta manera medidas de importancia global y local. El algoritmo de Liu et al. elimina gradualmente las ramas del esqueleto, de tal manera que sean conservadas las ramas más importantes que representan al objeto. En una primera fase se eliminan la mayoría de las ramas causadas por el ruido, cuya reconstrucción y contribución visual son muy pequeñas. Posteriormente, se realiza una fase de podado, donde las ramas finales que no cumplen con los parámetros de entrada establecidos, son eliminadas. El método puede trabajar de tres maneras diferentes: (1) Definiendo el número de píxeles finales a ser retenidos en el esqueleto. (2) Utilizando un umbral como condición de paro para la poda de ramas finales, respecto a la medida de importancia calculada para cada una de ellas. (3) Utilizando los dos parámetros de entrada definidos en (1) y (2). El esqueleto inicial es obtenido por un algoritmo de eskeletonización basado en diagramas de Voronoi (Székely, 2008).

Krinidis & Krinidis (2013) proponen un novedoso enfoque de podado del esqueleto, donde el contorno del objeto es descompuesto utilizando un algoritmo denominado *EMD-like*, el cual está basado en la descomposición empírica de modos *EMD* (en inglés *The Empirical Mode Decomposition*) (Wu & Huang, 2009). *EMD* es un método adaptativo que permite el análisis de datos no lineales y no estacionarios mediante la descomposición de los datos en funciones de modo intrínsecas *IMF's* (en inglés *Intrinsic Mode Functions*). El método propuesto por Krinidis & Krinidis (2013) considera a los píxeles del contorno como el conjunto de datos a ser descompuestos por *EMD-like*, donde los *IMF's* se obtienen a partir de la media entre los máximos y mínimos locales, los cuales se distinguen por el ángulo interior formado entre píxeles vecinos (ángulos pequeños corresponden a máximos locales, mientras que ángulos amplios son mínimos locales). Para cada *IMF* un valor de importancia es calculado, basado en el promedio de los ángulos formados entre el *IMF* en cuestión y los *IMF's* vecinos (los vecinos y los vecinos de sus vecinos). Cada *IMF* corresponde a un píxel final en el esqueleto, si su valor de importancia es mayor a un

umbral, el *IMF* es considerado como importante, de lo contrario es descartado y por lo tanto la rama del esqueleto es podada. Este proceso se realiza de forma iterativa hasta obtener visualmente el esqueleto final deseado. El esqueleto inicial es obtenido por el algoritmo basado en mapa de distancia Euclidiana propuesto por Choi et al. (2003).

3.3. Discusión

La mayoría de los métodos de esqueletonización con poda, propuestos recientemente en la literatura, requieren de la asignación de umbrales u otros parámetros de entrada para cada objeto o conjunto de formas con características similares, con el objetivo de representar en el esqueleto partes importantes y eliminar ramas no deseadas. Los valores de los parámetros de entrada dependen de la percepción humana (*i.e.* número de partes visuales importantes en los objetos) o se obtienen mediante prueba y error, convirtiéndose en un proceso tedioso y poco práctico, principalmente cuando el conjunto de prueba contiene muchas imágenes divididas en diferentes clases, donde cada clase contiene imágenes con diferentes variaciones.

Garantizar que un método genere esqueletos descriptivos y consistentes en relación con la percepción visual humana ³ (Feldman & Singh, 2006; Sprote & Fleming, 2013), para todo un conjunto de datos que contiene diferentes formas, es una tarea no trivial y aún no resuelta en la literatura. En la *tabla 3.1* mostramos de manera resumida las principales características de los métodos descritos en este capítulo, donde podemos observar que la mayoría de los métodos propuestos intentan producir esqueletos mediante el manejo de diferentes parámetros para cada conjunto de datos que contiene formas de la misma clase (lo cual reduce en cierto grado el proceso tedioso de asignar umbrales y parámetros manualmente para cada imagen). Sin embargo, los métodos propuestos por Shen et al. (2013a) y Montero & Lang (2012) pueden generar esqueletos para diferentes imágenes mediante

³Los esqueletos obtenidos por el método sean cercanos a los esqueletos construidos manualmente por el ser humano

una sola configuración en los parámetros de entrada. Aunque el método de Montero & Lang (2012) tiene bajo costo computacional, la construcción de los esqueletos dependen de la configuración de dos parámetros de entrada (d y α). Por otro lado, el método de Shen et al. (2013a) puede ser utilizado con dos posibles configuraciones en los parámetros de entrada; una primera configuración requiere un solo parámetro (β) pero la construcción del esqueleto tiene alto costo computacional. Una segunda configuración, aunque reduce el costo computacional, requiere de dos parámetros de entrada (N y β). Por lo tanto, este trabajo de tesis tiene por objetivo desarrollar un método que permita representar formas articuladas mediante esqueletos, manteniendo un compromiso entre tiempo de ejecución y número de parámetros requeridos por el método.

Tabla 3.1: Comparación entre los métodos de esqueletonización

Método	Parámetros	¿parámetro(s) definidos para cada forma?	¿Parámetro(s) definidos para cada clase?	¿Parámetro(s) definidos para todo el conjunto de datos?	¿Garantiza que las partes importantes de la forma se conserven?
(Bai et al., 2007)	N	Si	-	-	No
(Bai & Latecki, 2007)	T_{CR} ó N, T_{CR}	-	Si	-	Si
(Aslan et al., 2008)	ρ	-	Si	-	No
(Yang et al., 2009)	M	-	Si	-	No
(Shen et al., 2011)	t	Si	-	-	Si
(Montero & Lang, 2012)	d, s	-	-	Si	Si
(Liu et al., 2012)	N, T_{CR}, T_{VC}	Si	-	-	Si
(Shen et al., 2013a)	β ó N, β	-	-	Si	Si
(Liu et al., 2013)	M ó t ó M, t	-	Si	-	Si
(Krinidis & Krinidis, 2013)	ϕ	-	Si	-	Si
Método Propuesto	s	-	-	Si	Si

N : número de píxeles en el contorno para aproximación poligonal. T_{CR} : umbral de error de reconstrucción. ρ : efecto en las curvas de nivel. M : Número de píxeles finales en el esqueleto. t : Umbral de importancia de un píxel. d : distancia máxima entre el contorno original y una curva ajustada (aproximación al contorno). s : Factor de peso para determinar la importancia de una rama. T_{VC} : umbral de contribución visual. β : Factor de peso que equilibra error de reconstrucción con simplicidad del esqueleto. ϕ : Factor de peso para determinar la importancia de un píxel final en el esqueleto. - : No cumple con la condición

Capítulo 4

Método Propuesto

En este capítulo se presenta un método para representar formas articuladas de imágenes binarias 2D por medio de grafos pesados basados en esqueletos. El método propuesto para la representación de formas articuladas por medio de grafos pesados está compuesto por dos etapas: 1) representación de formas articuladas 2D mediante la construcción de esqueletos y 2) extracción de características a partir del esqueleto y construcción de grafos pesados (ver figura 4.1).

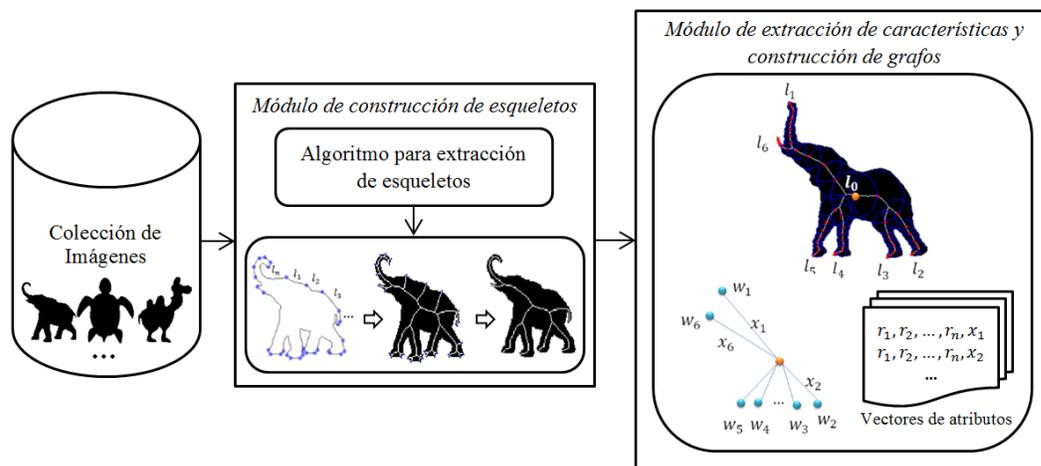


Figura 4.1: Método de Representación de formas articuladas por medio de grafos pesados basados en esqueleto

4.1. Construcción de esqueletos

El método propuesto para la construcción de esqueletos está compuesto por tres etapas: 1) segmentación del contorno a trozos, 2) construcción del esqueleto y 3) podado del esqueleto. Las dos primeras etapas hacen uso de los métodos presentados por Prasad et al. (2011) y Hesselink & Roerdink (2008) con el fin de obtener un esqueleto con un menor número de ramas no deseadas, mediante la construcción del esqueleto *IMA* (del inglés *Integer Medial Axis*) en función de la segmentación a trozos del contorno. Posteriormente, en 3) se propone una etapa de podado que permite eliminar las ramas no deseadas del esqueleto, donde un factor de escalamiento sobre el radio de disco máximo asociado al pixel de unión de la rama que está siendo evaluada, determina su importancia y con ello la decisión de poda. Ver *figura 4.2*

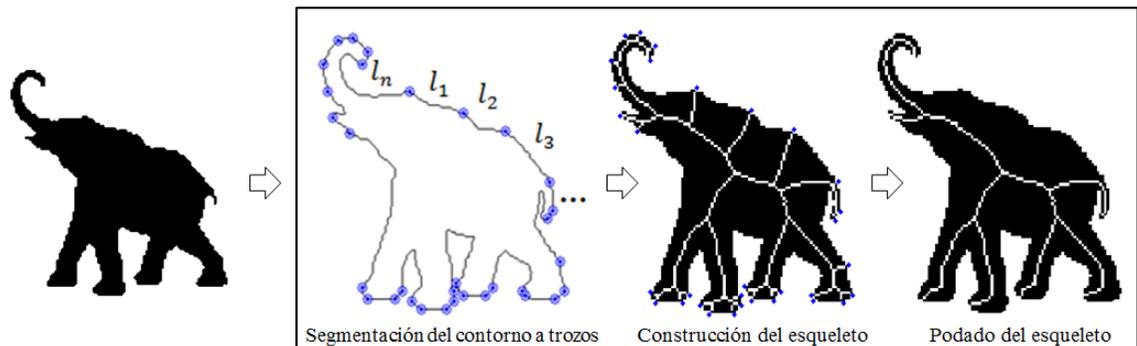


Figura 4.2: Método para la obtención de esqueletos a partir de formas articuladas 2D

4.1.1. Segmentación del contorno a trozos

La segmentación del contorno a trozos es una estrategia que ha sido utilizada en diferentes métodos relacionados con la construcción de esqueletos (Bai et al., 2007; Yang et al., 2009; Montero & Lang, 2012; Shen et al., 2013a), donde el límite de cada segmento se determina generalmente mediante métodos de aproximación poligonal o puntos

dominantes, los cuales son un subconjunto de los píxeles del contorno y representan las principales características del objeto.

En este trabajo se propone realizar la segmentación del contorno a trozos en función de un conjunto reducido de puntos del contorno; puesto que los métodos que hacen uso de esta técnica como etapa previa a la construcción del esqueleto, producen esqueletos con un menor número de ramas no deseadas que otros métodos convencionales para la construcción de esqueletos (Choi et al., 2003). Además, la construcción de un primer esqueleto en función del contorno segmentado a trozos, permite la reducción del costo computacional en etapas posteriores que realizan la poda del esqueleto, ya que un menor número de ramas finales deben ser consideradas como posibles candidatos para eliminación.

Los métodos propuestos hasta el momento que incluyen la segmentación del contorno a trozos, requieren de parámetros de entrada adicionales, los cuales deben ser asignados según la percepción humana (conocimiento *a priori* de la forma) o mediante prueba y error (e.g., DCE, aproximación lineal, aproximación cúbica –Curvas cúbicas de Bezier).

Para la construcción del esqueleto se propone segmentar el contorno haciendo uso del método propuesto por Prasad et al. (2011), el cual es un método no-paramétrico que permite la reducción de puntos en el contorno, donde el conjunto de puntos resultante recibe el nombre de *puntos dominantes*. Los puntos dominantes *DP* (del inglés *Dominant points*), conocidos también como puntos significantes, tienen como principal ventaja la alta reducción de datos, ya que los puntos dominantes representan las principales características de una curva (en este caso, la curva que describe el contorno de un objeto), las cuales se encuentran generalmente en los puntos de máxima curvatura local (Nguyen & Debled-Renneson, 2011; Teh & Chin, 1989), esto produce como resultado un conjunto reducido de puntos que pueden describir una curva, tanto para la percepción visual como para tareas de reconocimiento (Cornic, 1997; Carmona-Poyato et al., 2005).

La segmentación del contorno a trozos está dividida a su vez en tres fases: 1) detección de puntos dominantes (*figura 4.3a*), 2) selección de puntos convexos (*figura 4.3b*) y 3)

segmentación del contorno a trozos de la forma (figura 4.3c).

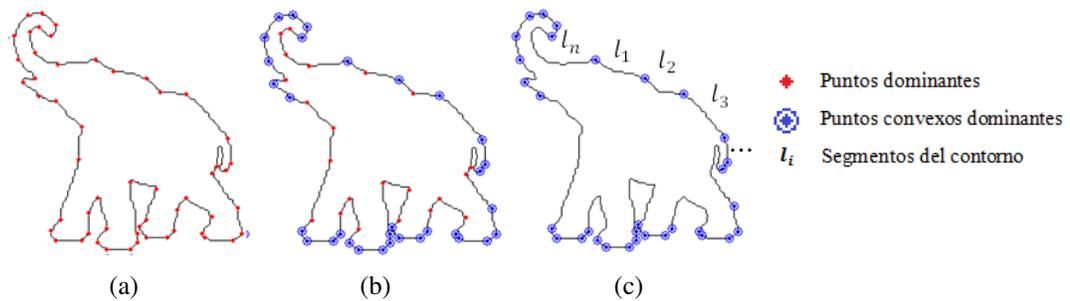


Figura 4.3: Segmentación del contorno a trozos

Ramer (1972), Douglas & Peucker (1973) proponen un método recursivo (*RDP*) para la obtención de puntos dominantes, mediante la reducción de puntos en una curva. Sin embargo, el método *RDP* requiere como parámetro, un umbral que determina la distancia máxima permitida entre la curva original y la curva ajustada. Para hacer este método no paramétrico, Prasad et al. (2011, 2012) consideran la distancia máxima permitida entre la curva original y la curva ajustada como una función relacionada con la calidad de ajuste, permitiendo que el umbral sea obtenido de forma adaptativa y automática. El método propuesto por Prasad et al. (2011, 2012) es considerado como el único método no-paramétrico para la obtención de puntos dominantes en una curva digital.

Debido a que las ramas que inciden en los puntos cóncavos del contorno no proporcionan información estructural para la representación de las formas a través de esqueletos (ver figura 4.4), una vez obtenidos los puntos dominantes que representan el contorno, el conjunto de puntos dominantes es reducido, preservando únicamente los puntos convexos.

Posteriormente, el contorno de la forma es segmentado a trozos, donde el límite de cada segmento es determinado a partir del conjunto de puntos dominantes convexos.

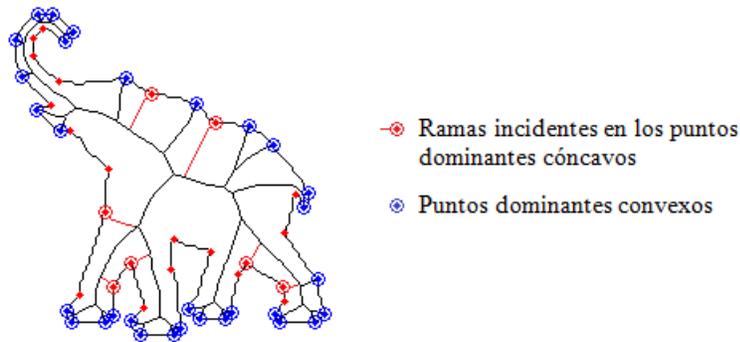


Figura 4.4: Ramas que inciden en los puntos dominantes cóncavos (marcadas en rojo)

4.1.2. Construcción del esqueleto

Como ya se mencionó, existen diferentes métodos para la construcción de esqueletos. El método utilizado en este trabajo pertenece a la categoría de algoritmos basados en transformada de distancia, los cuales usan generalmente mapas de distancia Euclidiana para localizar los centros de disco máximo, ya que tienen como principal característica que son invariantes a rotación.

El algoritmo propuesto por Hesselink & Roerdink (2008) denominado *IMA* (del inglés *Integer Medial Axis*) será utilizado en este trabajo y consta de dos etapas:

- Cálculo de la *transformada de distancia Euclidiana* y la *transformada de características* de una imagen binaria S .
- Obtención del eje medial (*IMA*) mediante selección de puntos del esqueleto en función de la *transformada de características* y la segmentación a trozos del contorno.

IMA realiza una evaluación sobre cada pixel del *foreground* de la imagen respecto a sus pixeles vecinos en la posición $(i - 1, j)$ y $(i, j - 1)$ para determinar si pertenecen o no al esqueleto. Un pixel es considerado como pixel del esqueleto si cumple con las siguientes condiciones:

- a) Es centro de disco máximo.

- b) El disco máximo tiene dos puntos equidistantes y tangentes que pertenecen a dos segmentos diferentes en el contorno de la forma (Montero & Lang, 2012).

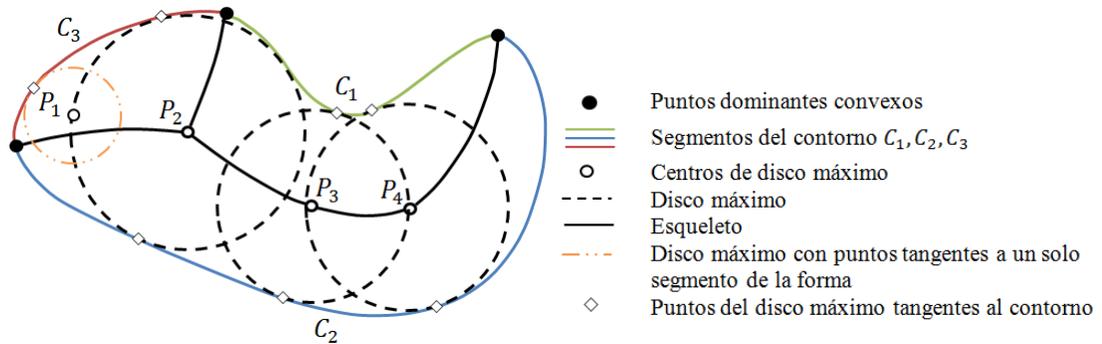


Figura 4.5: Construcción del esqueleto en función del contorno segmentado a trozos.

En la *figura 4.5* están marcados algunos centros de disco máximo y sus respectivos discos máximos, donde se puede apreciar que sólo los píxeles del *foreground* que además de ser centros de disco máximo, tiene dos puntos tangentes a segmentos diferentes en el contorno de la forma son considerados como píxeles del esqueleto. En la *figura 4.5* se ejemplifica un centro de disco máximo (P_1) que no es considerado como píxel del esqueleto, debido a que su disco máximo sólo es tangente a un segmento en el contorno de la forma. Por el contrario, P_2 , P_3 y P_4 son centros de disco máximo, cuyos discos máximos son tangentes a al menos dos segmentos diferentes del contorno y por lo tanto, forman parte del esqueleto.

4.1.3. Podado del esqueleto

Frecuentemente, la simplificación o podado del esqueleto es considerada como una etapa de post-procesamiento para remover del esqueleto puntos o ramas no deseadas, las cuales por lo general son producto del ruido presente en las figuras.

Los enfoques existentes para el podado del esqueleto requieren del ajuste de umbrales o parámetros de entrada para cada imagen o conjuntos de imágenes con características si-

milares, lo cual es tedioso, al ser éste un proceso manual, el cual depende en algunos casos de un conocimiento *a priori* (percepción visual) o de los resultados obtenidos mediante prueba y error.



Figura 4.6: Esqueleto de un elefante con ramas finales incidentes en los puntos dominantes convexos del contorno

Las etapas previas para la segmentación del contorno a trozos y construcción del esqueleto permiten obtener un esqueleto con un pequeño número de ramas no deseadas. Sin embargo, construir el esqueleto con ramas incidentes en los puntos dominantes convexos del contorno no garantiza que el esqueleto esté libre de ramas adicionales (no deseadas) (ver *figura 4.6*). Por lo tanto, es necesaria una etapa que permita la eliminación de dichas ramas.

En este trabajo de tesis proponemos realizar la poda del esqueleto, removiendo iterativamente ramas finales del esqueleto hasta que no se produzcan cambios adicionales en el esqueleto resultante. Las ramas finales son eliminadas en función de la información local de la forma; esta última es obtenida a partir del radio máximo asociado al pixel de unión de la rama que está siendo procesada.

En la *figura 4.7* se presentan una imagen, la cual posee dos protuberancias del mismo tamaño en diferentes partes del contorno de la forma. Las líneas pintadas en azul representan al esqueleto y el disco rojo en línea discontinua corresponde al disco máximo en el pixel de unión, donde el pixel de unión conecta dos o más ramas del esqueleto, las cuales son producto de las articulaciones, deformaciones o ruido presente en la forma. Se puede observar que a medida que el área local del objeto es mayor (etiquetado como 1

en la *figura 4.7a*) la protuberancia provee menor contribución visual en comparación a la protuberancia donde el área local es menor (etiquetado como 2 en la *figura 4.7a*). Lo anterior provoca que ramas que aportan poca contribución respecto al área local en la que se encuentran sean consideradas como poco importantes y en consecuencia sean podadas (ver *figura 4.7b*).

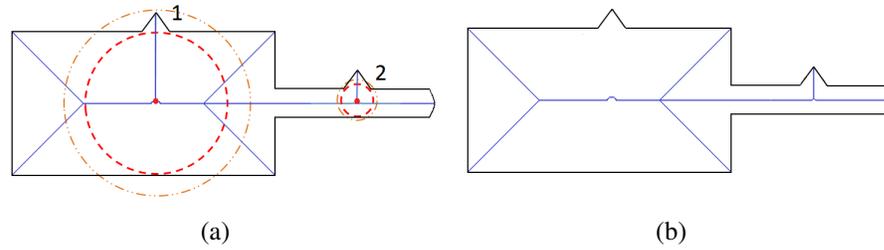


Figura 4.7: a) Contribución visual de una protuberancia en función del área local. b) Esqueleto resultante.

Por lo tanto, la contribución o importancia de una rama se puede determinar al considerar un factor de escalamiento s respecto a la información local, provocando que las ramas deseadas sean distinguidas de las no deseadas en proporción a la región que está siendo procesada. En la *figura 4.7a* se puede apreciar el efecto del factor de escalamiento s ($s = 1.27$) para ambas deformaciones respecto al radio asociado al pixel de unión (circunferencia exterior asociada al pixel de unión). Nótese que en la protuberancia 1 (en la *figura 4.7a*) el factor de escalamiento permite podar la rama que la representa, puesto que la protuberancia es pequeña respecto al tamaño de la forma; mientras que en la protuberancia 2 de la misma figura, el factor de escalamiento no permite eliminar la rama, puesto que su tamaño es relativamente grande respecto al área local de la forma, descrita mediante el radio asociado al pixel de unión.

Consideremos una rama final $\{b_p, \dots, e_p\}$ con un pixel de unión b_p y un pixel final e_p , tal que no contiene otro pixel de unión o pixel final, excepto b_p y e_p . Para el podado del esqueleto en nuestro método se propone que una rama final sea considerada como importante si la suma de las distancias Euclidianas calculadas a partir de las coordenadas

entre cada par de píxeles consecutivos p_i, p_{i+1} que forman a la rama, es mayor que el valor asociado para b_p en el mapa de distancia Euclidiana, ponderado por un factor de escalamiento s , con $s > 1$ (ver ecuación 4.1).

$$\sum_{i=1}^{n-1} |p_i - p_{i+1}|_2 \leq s * DT[b_p] \quad (4.1)$$

Donde $DT[b_p] = |ft[b_p] - b_p|_2$, siendo $ft[b_p]$ la transformada de características para b_p

Como se puede observar en la figura 4.8 (círculo azul), mientras más grande sea el radio de disco máximo en b_p , pequeñas variaciones en el contorno representadas a través de las ramas conectadas a b_p , son consideradas como ramas no importantes y por lo tanto son podadas. Por el contrario, mientras más pequeño sea el radio de disco máximo en b_p , las ramas conectadas al pixel de unión que representan variaciones pueden ser consideradas como importantes y por lo tanto conservadas.

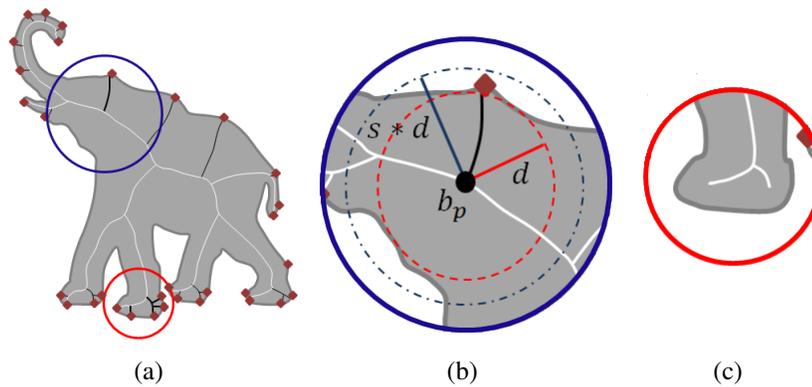


Figura 4.8: Podado de ramas no deseadas: a) ramas no deseadas (marcadas en negro). b) efecto de s sobre $DT[b_p]$ para decisión de poda. c) ramas recortadas.

Para evitar actualizar el estado de los píxeles del esqueleto cada vez que una rama es eliminada y permitir que el método sea más rápido, todas las ramas finales del esqueleto son procesadas y marcadas como ramas no deseadas si cumplen la condición de poda.

En la figura 4.8c se puede observar el resultado obtenido al podar la pata del elefante en una primera iteración (círculo rojo) considerando todas las ramas finales en el proceso

de poda, lo anterior provoca que algunas ramas significantes sean recortadas, lo que en iteraciones posteriores puede dar lugar a pérdida de información estructural. Por lo tanto, todas las ramas finales del esqueleto son procesadas y marcadas como no deseadas si además de cumplir con la condición de poda, no existe otra rama final que comparta el mismo pixel de unión b_p ; en ese caso, sólo la rama de menor longitud es evaluada, debido a que representa la menor cantidad de información respecto a las ramas vecinas con las que comparte el pixel de unión, garantizando que las ramas que aportan menor información sean las primeras en ser procesadas.

4.2. Módulo de extracción de características y construcción de grafos pesados

La mayoría de los métodos existentes en la literatura que permiten construir grafos y/o árboles a partir de esqueletos obtenidos desde formas binarias 2D (He et al., 2006; Jiang et al., 2009; Hingway & Bhurchandi, 2009; Erdem & Tari, 2010; Hingway & Bhurchandi, 2011) no tienen en cuenta la estructura topológica del esqueleto, puesto que objetos similares pueden ser representados mediante esqueletos con estructura topológica diferente (Ver *figura 4.9a* y *figura 4.9b*), mientras que objetos diferentes pueden estar representados por la misma topología (ver *figura 4.9c*); lo anterior se produce principalmente debido a que las articulaciones producen diferentes ubicaciones y ramificaciones en los pixeles de unión (Bai & Latecki, 2008).

Como podemos observar en la *figura 4.9a* y la *figura 4.9b* los pixeles finales son más consistentes ¹ que los pixeles de unión, puesto que forman parte de las ramas finales del esqueleto, las cuales representan a su vez partes articuladas o partes importantes de la forma.

Sin embargo, construir un grafo en función de los pixeles finales del esqueleto no es útil

¹coherentes en una familia de objetos

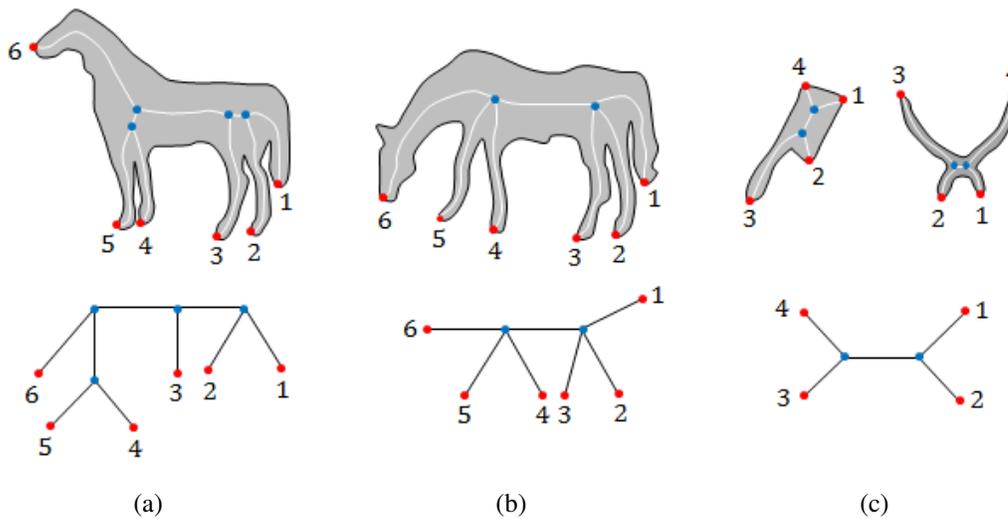


Figura 4.9: (a) y (b) Formas visualmente similares con diferente topología de esqueleto. (c) Formas diferentes con la misma topología de esqueleto

por sí solo, puesto que imágenes pertenecientes a diferentes clases pueden tener el mismo número de píxeles finales (*figura 4.9c*). Por lo tanto, es necesario extraer características que permitan asignar pesos y/o etiquetas a las componentes de los grafos (ya sean vértices, aristas o ambos) de tal manera que esta información permita que imágenes pertenecientes a clases diferentes sean distinguibles.

Cuando se tiene una colección de formas articuladas 2D, es necesario tener en cuenta que para el análisis de objetos se deben considerar aspectos tales como: escala, rotación y traslación; sin olvidar que formas pertenecientes a una misma clase pueden presentar diferentes variaciones, producto del movimiento en las articulaciones. Por lo tanto las características extraídas para generar pesos y/o etiquetas para el grafo que representará a dicha forma, deben tener en cuenta los aspectos anteriormente mencionados.

En este trabajo se propone construir a partir del esqueleto, un grafo pesado tipo estrella, donde el número de aristas corresponde al número de píxeles finales en el esqueleto de la forma articulada. Además, a cada vértice final en el grafo se le asignará un vector de características, que contiene información respecto al área de la forma que está siendo

representada a través de cada uno de los vértices finales en el grafo.

Debido a que es necesario definir qué porción del esqueleto corresponde a cada vértice final en el grafo, en este trabajo se propone obtener el centroide de la forma y ajustarlo al pixel más cercano del camino más largo en el esqueleto. Como podemos observar en la *figura 4.10*, el centroide ajustado al esqueleto está ubicado en una posición similar para imágenes con características similares.

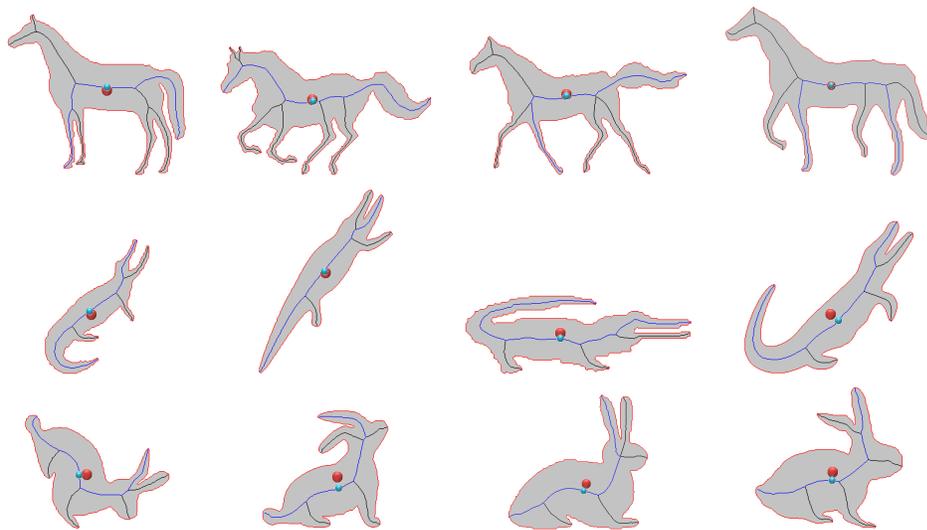


Figura 4.10: Centroide original (marcado en círculo rojo) y centroide ajustado al camino más largo del esqueleto (marcado en círculo azul)

El centroide ajustado al esqueleto nos permite definir el punto de partida para la extracción de características teniendo en cuenta cada pixel final en el esqueleto. Debido a que la forma original puede ser reconstruida a partir de los radios máximos asociados a cada pixel del esqueleto, las características extraídas del esqueleto son:

- Una secuencia de radios normalizados w que describen la variación del contorno de la forma desde el centroide ajustado al esqueleto hacia cada pixel final del esqueleto (ver *figura 4.11*).
- La longitud normalizada desde el centroide hacia cada pixel final, calculada como

la suma de las distancias Euclidianas entre las coordenadas asociadas a cada par de radios consecutivos en w .

En este módulo se obtiene una colección de grafos pesados, donde cada grafo representa una imagen de la colección.

Para construir el grafo a partir del esqueleto que contenga la información de las características extraídas de la forma, se procede de la siguiente manera:

1. Dado un esqueleto con K pixeles finales, sea l_i ($i = 1, 2, \dots, K$) el i -ésimo pixel final a lo largo del contorno de la forma en sentido de las manecillas del reloj (partiendo desde el pixel más a la derecha y más arriba en el contorno de la forma), se construye un *grafo estrella* S_K , donde al vértice interno se le asigna la etiqueta l_0 y a los demás vértices la etiqueta l_i ($i = 1, 2, \dots, K$).
2. A cada vértice l_i en el grafo S_K se le asignan pesos en función del esqueleto que representan las características extraídas de la forma, considerando los siguientes pasos:
 - a) Para un esqueleto con N_0 pixeles, a cada pixel p_g ($g = 1, 2, \dots, N_0$) se asocia su valor correspondiente en el mapa de distancia Euclidiana $DT[p_g]$, el cual contiene el radio de disco máximo inscrito en la forma.
 - b) Se obtiene el centroide de la forma.
 - c) Se obtiene el camino más largo en el esqueleto.
 - d) El centroide es ajustado al pixel más cercano del camino más largo en el esqueleto, el cual corresponde al vértice etiquetado como l_0 en el grafo.
 - e) Para cada l_i en el esqueleto, se obtienen los pixeles que forman el camino geodésico desde l_0 hasta cada pixel final l_i en el esqueleto.
 - f) Para cada camino entre l_0 y l_i obtenemos una secuencia de radios $w_i = (r_1, r_2, \dots, r_t)$, donde r_1 corresponde al radio de disco máximo en el

pixel l_0 . Posteriormente cada r_j ($j = 2, \dots, t$) se obtiene a partir del radio de disco máximo asociado al pixel de intersección entre la circunferencia de radio máximo centrada en r_{j-1} y el camino del esqueleto (ver *figura 4.11*).

- g) Se obtiene la longitud d_i desde el centroide hasta cada pixel final l_i , donde la longitud corresponde a la suma de las distancias Euclidianas calculada entre las coordenadas para cada par de pixeles consecutivos asociados a w_i (ver *ecuación 4.2*)

$$d_i = \sum_{j=1}^{t-1} \sqrt{(r_{j_x} - r_{(j+1)_x})^2 + (r_{j_y} - r_{(j+1)_y})^2} \quad (4.2)$$

- h) A cada vértice l_i en el grafo se asocian sus pesos $w_i = (r_1, r_2, \dots, r_t)$ y d_i , los cuales contienen las características para cada región desde el centroide ajustado al esqueleto hacia cada pixel final en el esqueleto.

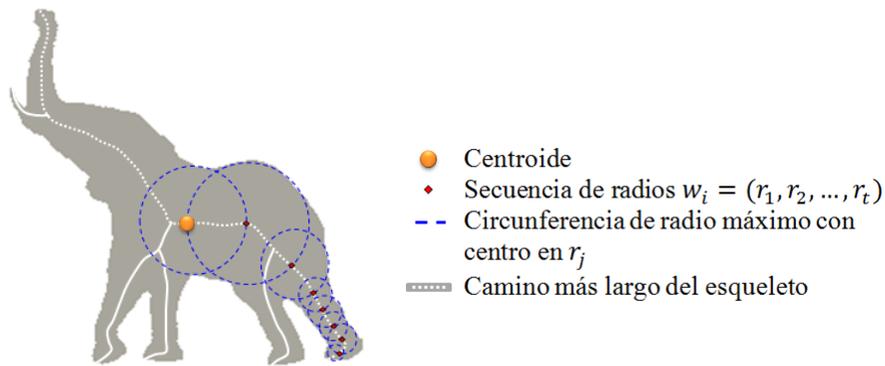


Figura 4.11: Secuencia de radios w_2 para el elefante en la *figura 4.12a*

El grafo para el esqueleto en la *figura 4.12a* se puede observar en la *figura 4.12b*.

Para que la información asociada a los grafos sea invariante a factores tales como el escalamiento, los valores en w_i y d_i son normalizados (ver *ecuación 4.3*).

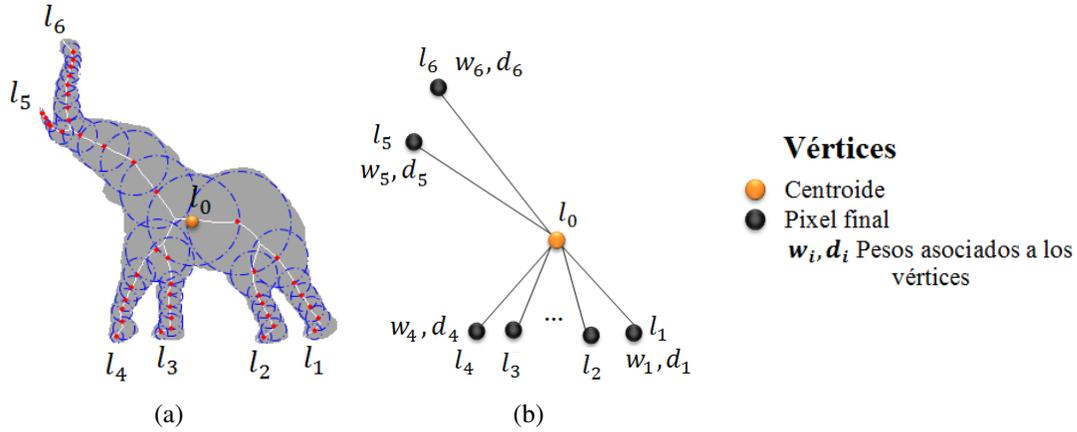


Figura 4.12: Obtención de grafos a partir del esqueleto

$$\forall r_j \in w_i \quad r_j = \frac{r_j}{\frac{1}{N_0} \sum_{g=1}^{N_0} (DT[p_g])}$$

$$d_i = \frac{d_i}{\sum_{j=1}^K d_j} \quad (4.3)$$

Donde $DT[p_g]$ es el valor asociado al pixel p_g en el mapa de distancia Euclidiana; por lo tanto, $\sum_{g=1}^{N_0} (DT[p_g])$ es la suma de todos los radios de disco máximo asociados a cada pixel del esqueleto.

4.3. Módulo de clasificación

Para evaluar la utilidad de los grafos pesados obtenidos de formas articuladas $2D$ en tareas de clasificación, utilizaremos un clasificador del tipo vecino más cercano, por lo cual es necesario evaluar el parecido entre grafos.

Dados dos grafos pesados G y G' que representan dos forma articuladas A y B respectivamente, se obtiene un valor de similitud a partir de las correspondencias encontradas entre sus vértices finales. Sean v_1, v_2, \dots, v_K los vértices finales en G y v'_1, v'_2, \dots, v'_N los

vértices finales en G' , con $K \leq N$. Donde cada v_i ($i = 1, 2, \dots, K$) en G tiene asociado un peso w_i y una longitud d_i ; y cada v'_j ($j = 1, 2, \dots, N$) en G' tiene asociado un peso w'_j y una longitud d'_j . Inicialmente se construye una matriz γ de tamaño $K \times N$ donde K es el número de filas y N es el número de columnas. La matriz γ (ver ecuación 4.4) contiene la similitud entre todos los posibles pares de vértices (que representan a los pixeles finales del esqueleto) en G y G' .

$$\gamma(G, G') = \begin{bmatrix} dc(v_1, v'_1) & dc(v_1, v'_2) & \dots & dc(v_1, v'_N) \\ dc(v_2, v'_1) & dc(v_2, v'_2) & \dots & dc(v_2, v'_N) \\ \vdots & \vdots & \ddots & \vdots \\ dc(v_K, v'_1) & dc(v_K, v'_2) & \dots & dc(v_K, v'_N) \end{bmatrix} \quad (4.4)$$

Donde $dc(., .)$ es la distancia entre dos vértices finales v_i y v'_j en G y G' respectivamente (ver ecuación 4.5)

$$dc(v_i, v'_j) = \sum_{t=1}^h \frac{(r_{it} - r'_{jt})^2}{r_{it} + r'_{jt}} + \alpha \frac{(d_i - d'_j)^2}{d_i + d'_j} \quad (4.5)$$

donde h corresponde al número de elementos del vector de mayor longitud entre w_i y w'_j , α es un factor de peso; y r_{it} es el valor del t -ésimo radio en w_i , en el vértice i ; para esta comparación, al vector de menor longitud le asignamos ceros en las posiciones que se requieran para completar h .

El valor de similitud entre dos grafos es obtenido a partir de la matriz $\gamma(G, G')$, mediante una modificación del método *OSB* (en inglés *Optimal Subsequence Bijection*) (Latecki et al., 2007) el cual consiste en re-ordenar la matriz $\gamma(G, G')$ de tal manera que el mayor número de correspondencias entre los vértices en G y G' sean encontrados. No realizar el re-ordenamiento puede provocar que algunos vértices en G no encuentren su vértice correspondiente en G' y viceversa, puesto que el método *OSB* preserva el orden de las secuencias de entrada; por lo tanto, por efectos de rotación o posición de las formas

articuladas, las etiquetas asignadas en G y G' pudieran no coincidir, como se muestra en la *figura 4.14*.

4.3.1. Evaluación del parecido entre grafos

El método OSB(Latecki et al., 2007) permite encontrar correspondencias entre dos secuencias de datos con longitudes diferentes K y N ($K \leq N$), en este caso, dos secuencias de vértices finales para dos grafos G y G' .

$$G = (v_1, v_2, \dots, v_K) \quad G' = (v'_1, v'_2, \dots, v'_N)$$

El objetivo es encontrar la mejor correspondencia entre los vértices de G y G' saltando algunos elementos en G y G' si es necesario.

La correspondencia óptima entre los vértices en G y G' puede ser encontrada con el algoritmo de camino más corto en un grafo acíclico dirigido *DAG*, considerando como vértices a todos los pares de $(i, j) \in \{1, 2, \dots, K\} \times \{1, 2, \dots, N\}$ y el costo Q de cada arista $\{(i, j), (k, l)\}$ como:

$$Q((i, j), (k, l)) = \begin{cases} dc(v_i, v'_j) & \text{Si } i + 1 = k \text{ y } j + 1 \leq l \\ (k - i - 1) \cdot \text{penalizacion} & \text{Si } i + 1 < k \text{ y } j + 1 \leq l \\ \infty & \text{De otro modo} \end{cases} \quad (4.6)$$

En la *figura 4.13* los vértices en G etiquetados como l_1 , l_2 y l_4 encontraron en G' sus vértices correspondientes l'_1 , l'_2 y l'_5 respectivamente (correspondencias marcadas en línea azul discontinua). Sin embargo para los vértice l_3 en G y los vértices l'_3 y l'_4 en G' no fueron encontradas correspondencias (marcados en círculo rojo). Si las distancias entre los vértices l_1 y l'_1 , l_2 y l'_2 ; y l_4 y l'_5 son pequeñas, el valor total de similitud entre los grafos G y G' será también pequeño, a pesar de que varios vértices tanto en G como en G' han sido excluidos en el valor total de similitud. Lo anterior implica que saltar demasiados vértices

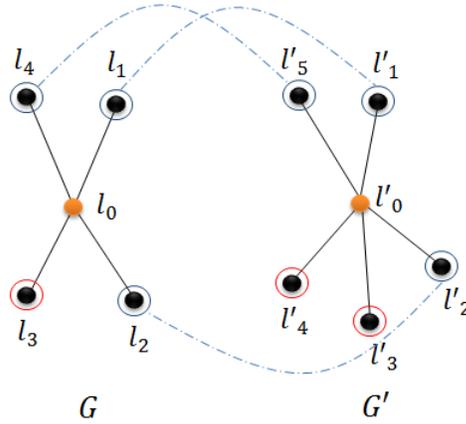


Figura 4.13: Correspondencias entre dos grafos G y G'

(no encontrar para un vértice en G su vértice correspondiente en G' o viceversa) puede influir negativamente en el valor de similaridad final, por lo tanto es necesario incluir una penalización cuando esto suceda (ver ecuación 4.7).

$$\begin{aligned} \text{penalizacion} = & \min(\text{media}(\min_j(dc(v_i, v'_j))) + \text{std}(\min_j(dc(v_i, v'_j))), \\ & \text{media}(\min_i(dc(v_i, v'_j))) + \text{std}(\min_i(dc(v_i, v'_j)))) \quad (4.7) \end{aligned}$$

La *penalización* consiste en obtener las distancias mínimas en $\gamma(G, G')$ (ver ecuación 4.4) para cada fila y columna, denotado en la ecuación 4.7 como $\min_i(dc(v_i, v'_j))$ y $\min_j(dc(v_i, v'_j))$ respectivamente. Llamemos *minFilas* a los distancias obtenidas por $\min_i(dc(v_i, v'_j))$ y *minColumnas* a las distancias obtenidas a partir de $\min_j(dc(v_i, v'_j))$. El valor de penalización es el valor mínimo entre *minFilas* y *minColumnas*. Esto permite asignar una distancia que representa una correspondencia “ficticia” para cada vértice saltado, de tal manera que para todos los vértices en G y G' pueda ser encontrada una correspondencia.

El método *OSB* ha sido utilizado en diferentes trabajos que intentan determinar la similitud entre objetos mediante correspondencias entre sus esqueletos (Bai & Latecki,

2008; Jiang et al., 2009; Xu et al., 2010; Shen et al., 2013b).

El método *OSB* utiliza una penalización constante sobre aquellos vértices en G y G' para las cuales no fue encontrada una correspondencia. La penalización está dada por el cálculo de la media más la desviación estándar sobre un conjunto de datos, los cuales corresponden al valor mínimo de cada fila y de cada columna en la matriz de similitudes

El objetivo es obtener valores de similaridad cercanos a cero para vértices en G y G' que son correspondientes y valores de similaridad alejados de cero para vértices en G y G' que no correspondan.

La penalización constante no permite otorgar una penalización en función del vértice para el cual no fue encontrada una correspondencia, lo cual puede provocar que el valor de penalización sobre dicho vértice sea mucho más grande o mucho más pequeño respecto a la similaridad calculada entre el vértice en cuestión y el vértice más parecido en la secuencia contra la cual se está comparando (la media y la desviación estándar son índices sensibles a valores extremos), lo anterior puede provocar valores de similaridad incorrectos para pares de vértices en G y G' , lo cual a su vez, puede afectar el valor de similaridad total calculado entre dos formas articuladas y por lo tanto influir en la clasificación.

Con el objetivo de obtener mejores resultados de clasificación y al mismo tiempo eliminar los problemas antes descritos con el uso de la penalización constante en el trabajo de Bai & Latecki (2008), en esta tesis se propone un nuevo esquema de penalización variable para el método *OSB* (Pinilla-Buitrago et al., 2013):

$$Q((i, j), (k, l)) = \begin{cases} dc(v_i, v'_j) & \text{Si } i + 1 = k \text{ y } j + 1 \leq l \\ \text{penalizacionVariable} & \text{Si } i + 1 < k \text{ y } j + 1 \leq l \\ \infty & \text{De otro modo} \end{cases} \quad (4.8)$$

$$\text{penalizacionVariable} = \sum_{j=j+1}^{l-1} \min_j \gamma(i, j)$$

Donde la penalización sobre los píxeles finales para los cuales no fue encontrada una correspondencia depende del píxel final que está siendo saltado. Al aplicar una penalización variable, la distancia entre nodos que no se corresponden en G y G' se hace más grande, permitiendo que valores mayores de similaridad sean obtenidos mientras menor sea el número de nodos que no se corresponden en G y G' .

Sin embargo, determinar la distancia entre los pares de píxeles finales en G y G' teniendo en cuenta todos los posibles caminos geodésicos que emanan desde v_i y v'_j , resulta ser un proceso muy costoso computacionalmente, puesto que el método *OSB* debe ser aplicado $K \times N$ veces.

Finalmente, el método propuesto en esta tesis para la clasificación de formas articuladas *2D* genera una sola matriz, la cual contiene los valores de disimilaridad entre pares de vértices finales en G y G' , donde la información asociada a cada vértice en G y G' es obtenida a partir del centroide ajustado a los esqueletos. El valor total de similaridad entre dos formas se obtiene haciendo uso de *OSB* con penalización constante, pero realizando antes un ajuste de la matriz γ , en función de los dos vértices más parecidos en G y G' .

El método *OSB* a diferencia de otros métodos tales como el método Húngaro (Kuhn, 1955), preserva el orden de las secuencias para las cuales se busca una correspondencia con costo mínimo. En nuestro caso, las secuencias son los vértices de los grafos G y G' , los cuales tienen etiquetas asignadas en sentido de las manecillas del reloj (la línea verde discontinua en la *figura 4.14*).

Sin embargo, por efecto de posición en las articulaciones y/o rotación en las formas articuladas y debido a que el etiquetamiento se realiza de forma automática, no es posible garantizar la alineación entre los vértices v_i en G y v'_j en G' , lo que podría provocar el salto de varios vértices en G y G' para los cuales puede ser encontrada una correspondencia.

En la *figura 4.14* podemos ver dos imágenes iguales A y B (con diferentes ángulos de rotación) y sus grafos G y G' respectivamente, donde las etiqueta l_1 y l'_1 no están asignadas al mismo vértice (alineadas) en G y G' , el vértice correspondiente de l_1 en G esta

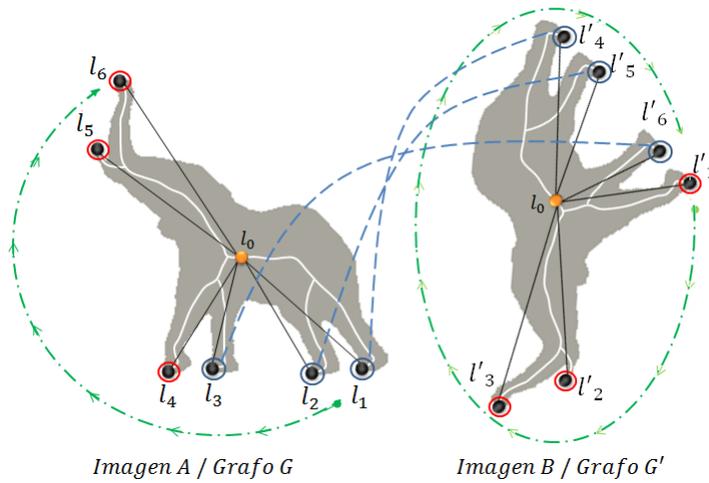


Figura 4.14: Grafos para dos imágenes iguales con diferente rotación

etiquetado como l'_4 en G' . Por lo tanto al aplicar *OSB* sobre la matriz $\gamma(G, G')$, los vértices l_1, l_2 y l_3 en G encuentran sus vértices correspondientes l'_4, l'_5 y l'_6 en G' respectivamente. En consecuencia, para los vértices l'_1, l'_2 y l'_3 en G' y l_4, l_5 y l_6 en G no se encuentran correspondencias y por lo tanto son saltados, efectuando sobre ellos una penalización. Lo anterior implica que el valor de similitud entre dos imágenes iguales (el cual debería ser igual a cero) se aleje de cero.

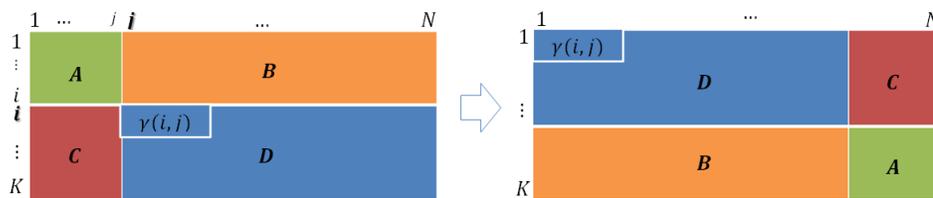


Figura 4.15: Ordenamiento de la matriz $\gamma(G, G')$

Por lo tanto se propone encontrar los dos vértices más parecidos en G y G' , obtenidos a partir del valor mínimo en $\gamma(G, G')$ en la posición (i, j) . Posteriormente consideramos las filas $(1 : i - 1)$ y las columnas $(1 : j - 1)$ como elementos saltados en ambas secuencias, re-ordenando la matriz $\gamma(G, G')$ como se muestra en la *figura 4.15*. Este es un procedimiento equivalente a re-etiquetar los vértices del grafo, asignando la etiqueta l_1 a

los vértices más parecidos en G y G' , y a partir de éstos etiquetar los demás vértices en el sentido de las manecillas del reloj. Finalmente el método *OSB* es aplicado sobre la matriz $\gamma(G, G')$ ordenada, para obtener el valor que determina la similitud entre dos formas A y B representadas a través de los grafos G y G' respectivamente.

4.3.2. Clasificación de formas articuladas

Como ya lo mencionamos, en este trabajo utilizaremos un clasificador de tipo *vecino más cercano*. Partiendo de un conjunto de N formas articuladas, donde para cada forma articulada se extrae el esqueleto y su grafo pesado correspondiente.

Posteriormente, para clasificar una nueva forma articulada se extrae su esqueleto y su grafo pesado correspondiente y se calcula el valor de similitud usando la modificación propuesta al *OSB* (sección 4.3.1) con cada uno de los grafos pesados en el conjunto de entrenamiento. Finalmente, el grafo pesado de la nueva forma articulada es clasificado según la clase a la que pertenezca el grafo con quien obtuvo el mayor valor de similitud (valores cercanos a cero corresponden a mayor similitud); es decir, la clasificación se realiza de forma supervisada según su vecino más cercano.

Capítulo 5

Resultados Experimentales

En este capítulo se presentan los resultados experimentales que permiten mostrar el comportamiento del método propuesto para la obtención de esqueletos presentado en la *sección 4.1*. Los esqueletos son obtenidos haciendo uso de un mismo valor en el parámetro de entrada para diferentes conjuntos de formas articuladas *2D*. Se realiza una comparación entre métodos reportados en la literatura y el método propuesto respecto a la representación de la forma a través del esqueleto, al tiempo de ejecución y la calidad del esqueleto.

Adicionalmente, se presenta una comparación entre el método de clasificación propuesto en la *sección 4.3* y el método presentado por Bai & Latecki (2008). Ambos métodos utilizan el algoritmo *OSB* (del inglés *Optimal Subsequence Bijection*) para encontrar la similitud entre pares de grafos (cada forma articulada es representada a través de un grafo pesado, obtenido a partir del esqueleto), con el objetivo de reconocer formas articuladas.

Ambos métodos realizan la clasificación en función de la similaridad encontrada entre pares de formas articuladas. El algoritmo presentado por Bai & Latecki (2008) encuentra la similitud entre grafos de esqueleto mediante la comparación de caminos geodésicos que emanan desde pixeles finales de los esqueletos. En contraste, nuestro método utiliza grafos pesados obtenidos mediante el método presentado en la *sección 4.2*, los cuales contienen

información de la forma desde el centroide ajustado al esqueleto, hacia cada pixel final del esqueleto.

Para evaluar los métodos propuestos en este trabajo se usaron las siguientes bases de datos, las cuales son frecuentemente utilizadas en la literatura, para clasificación de formas articuladas haciendo uso de los esqueletos que las representan:

- Mio et al. (2007) que contiene 50 formas articuladas que representan 8 gestos de manos diferentes (*figura 5.1*).
- Aslan & Tari (2005) formada por 14 clases, 4 formas articuladas por clase que representan caballos, tortugas, estrellas, entre otros; para un total de 56 formas (*figura 5.2*).
- Aslan et al. (2008), contiene 180 formas articuladas divididas en 30 clases, cada clase contiene 6 formas (*figura 5.3*). Es una base de datos extendida de Aslan & Tari (2005) que presenta más ejemplos por clase y además contiene formas de perros, dinosaurios, peces raya y ranas, entre otros.

Se utilizaron dos bases de datos adicionales, las cuales contienen formas articuladas y no articuladas, tomando únicamente las clases que contienen imágenes de nuestro interés (formas articuladas):

- Baseski et al. (2009), la base de datos está compuesta por 50 clases, cada clase contiene 20 formas para un total de 1000 formas. (*figura 5.4*). Es una base de datos extendida de Aslan et al. (2008) que presenta un conjunto más amplio de ejemplos para cada clase y además contiene elementos tales como camellos, vacas, jirafas y mariposas, entre otros. El subconjunto tomado de esta base de datos contiene 34 clases, para un total de 680 formas.
- Latecki & Lakämper (2000), contiene 1400 formas articuladas y no articuladas, 20 formas por clase (*figura 5.5*), tomando un subconjunto compuesto por 30 clases,

para un total de 600 formas. Esta base de datos es comúnmente conocida como *MPEG-7*.

En las *figuras 5.1, 5.2, 5.3, 5.4 y 5.5* se muestran algunas (o todas) formas que pertenecen a cada clase, para cada una de las bases de datos mencionadas anteriormente. Además, una etiqueta *Id* es asignada a cada clase para su identificación en la base de datos.

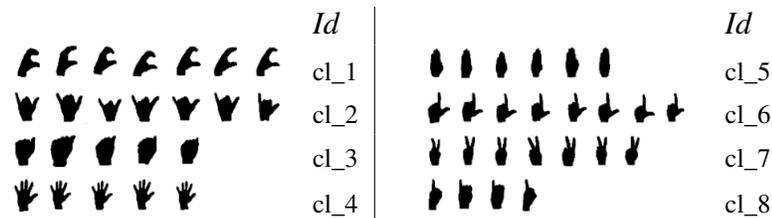


Figura 5.1: 50 gestos con las manos Mio et al. (2007)

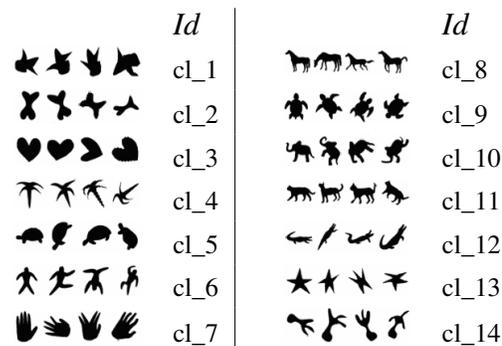


Figura 5.2: 56 formas articuladas Aslan & Tari (2005).

5.1. Construcción de esqueletos

Con el objetivo de comparar los resultados obtenidos con el método propuesto para la construcción de esqueletos a partir de formas articuladas 2D, respecto a trabajos reportados en la literatura, se implementaron los métodos propuestos por: Zhang & Suen (1984), Bai et al. (2007) y Montero & Lang (2012).

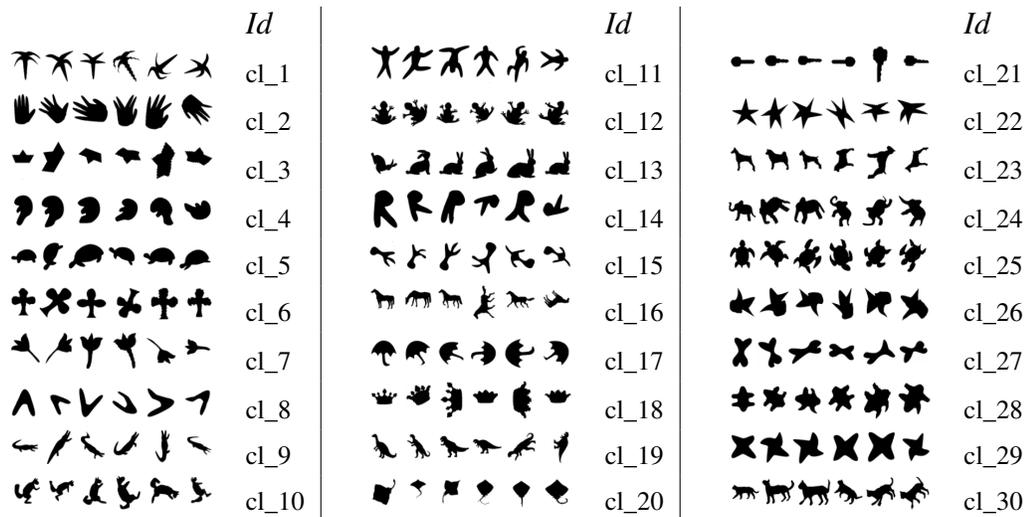


Figura 5.3: 180 formas articuladas Aslan et al. (2008)

El método de Zhang & Suen (1984) pertenece a la categoría de algoritmos de adelgazamiento; además es considerado como uno de los métodos más populares y reconocidos para la obtención de esqueletos.

Los métodos propuestos por Bai et al. (2007) y Montero & Lang (2012) pertenecen a la categoría de algoritmos basados en transformada de distancia. En ambos métodos el esqueleto es construido en función de la segmentación a trozos del contorno, definida por un conjunto de puntos dominantes en el contorno de la forma. El método propuesto por Bai et al. (2007) obtiene puntos dominantes mediante *DCE* (del inglés *discrete curve evolution*), mientras que Montero & Lang (2012) utiliza aproximación al contorno mediante curvas cúbicas de Bezier.

En nuestro primer experimento queremos mostrar la consistencia de los resultados obtenidos por el método propuesto para la construcción de esqueletos de formas articuladas presentado en la *sección 4.1* respecto a factores de rotación y escalamiento. En la *figura 5.6* podemos ver los esqueletos obtenidos para dos formas que representan la silueta de un camello y una jirafa, obtenidos de Baseski et al. (2009). La *figura 5.6a* y *figura 5.6e* corresponden a las formas originales, mientras que la *figura 5.6b* y *figura 5.6f*; *figura*

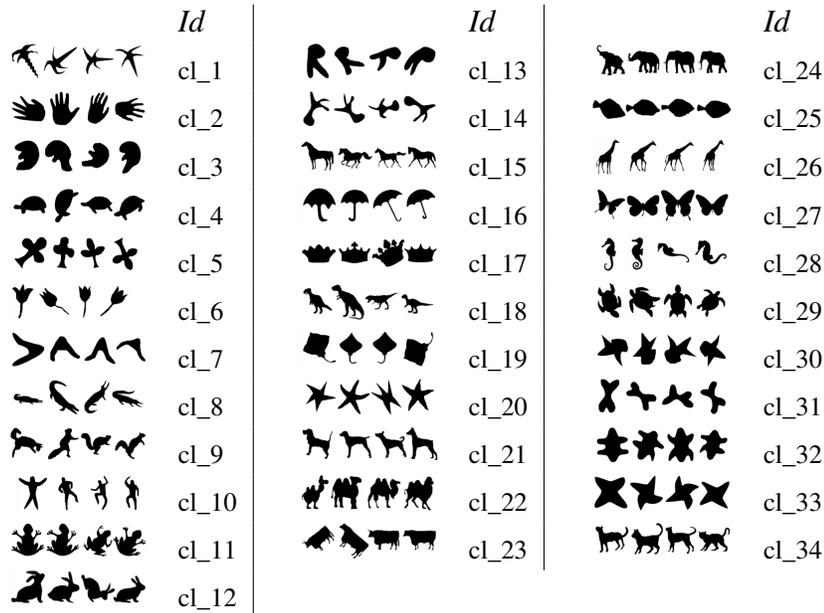


Figura 5.4: Ejemplos de formas articuladas de la colección Baseski et al. (2009), cuatro formas por clase.

5.6c y figura 5.6g; y figura 5.6d y figura 5.6h corresponden a las formas y sus respectivos esqueletos obtenidos con $s = 1.27$ bajo un escalamiento de 0.5, rotación de 45° y escalamiento de 2.0, respectivamente. Como se puede apreciar, en este experimento el método propuesto genera esqueletos conectados, libres de ramas no deseadas, ramas faltantes o ramas recortadas, bajo las diferentes transformaciones geométricas.

En los experimentos realizados con el método propuesto para la construcción de esqueletos, el parámetro de entrada s fue obtenido mediante prueba y error, donde para $s = 1.27$ el método obtuvo los mejores resultados, es decir con el menor número de ramas no-deseadas y sin ramas faltantes.

En la figura 5.7 se muestran algunos resultados para diferentes formas articuladas obtenidas de MPEG-7 (Latecki & Lakämper, 2000), utilizando el mismo parámetro de entrada $s = 1.27$. Se puede observar que cada una de las partes importantes y/o partes articuladas de los objetos están representadas mediante una rama final en el esqueleto, descartando ramas no deseadas, producto de las deformaciones y ruido presente en las formas.

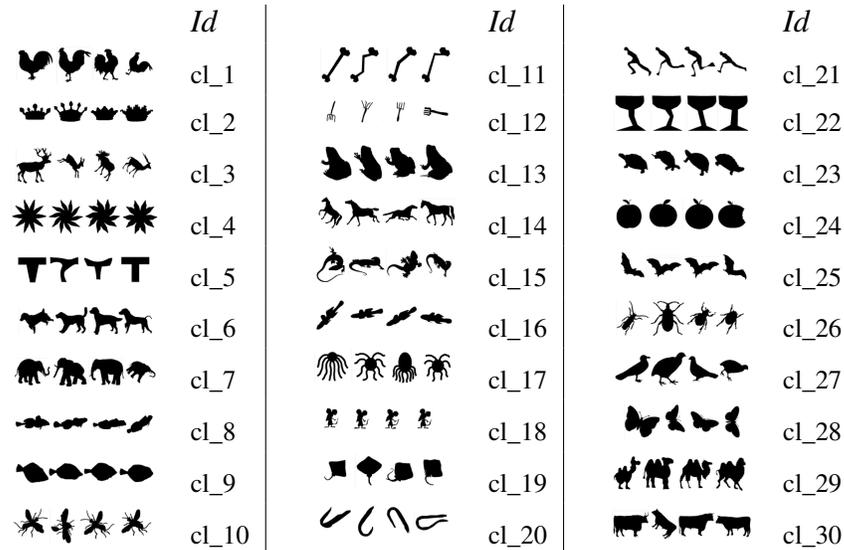


Figura 5.5: Ejemplos de formas articuladas de la colección Latecki & Lakämper (2000) *MPGE-7*, cuatro formas por clase.

En la *figura 5.8a* se pueden apreciar algunas formas articuladas con sus respectivos esqueletos obtenidos mediante el método *IMA* (del inglés *Integer Medial Axis*) (Hesslink & Roerdink, 2008), donde el número de ramas no deseadas es mucho mayor a la representación de las formas en la *figura 5.8b*, debido a que pequeñas deformaciones o ruido presente en el contorno de la forma produce ramas adicionales. En la *figura 5.8b* se presentan los esqueletos calculados mediante *IMA* con ramas finales incidentes en los puntos convexos dominantes calculados mediante el método no-paramétrico presentado por Prasad et al. (2012). Por último en la *figura 5.8c* se presentan los esqueletos resultantes después de llevar a cabo la etapa de poda. Visualmente se puede apreciar que el número de ramas finales en los esqueletos presentados en la *figura 5.8b* es mucho menor a los esqueletos en la *figura 5.8a*. Es necesario resaltar que aunque los esqueletos presentados en la *figura 5.8b* aún contienen ramas no deseadas, éstos no presentan pérdida de información estructural, además esta primera aproximación al esqueleto se obtiene de forma no paramétrica, a diferencia de métodos como Bai et al. (2007) y Montero & Lang (2012), entre otros.

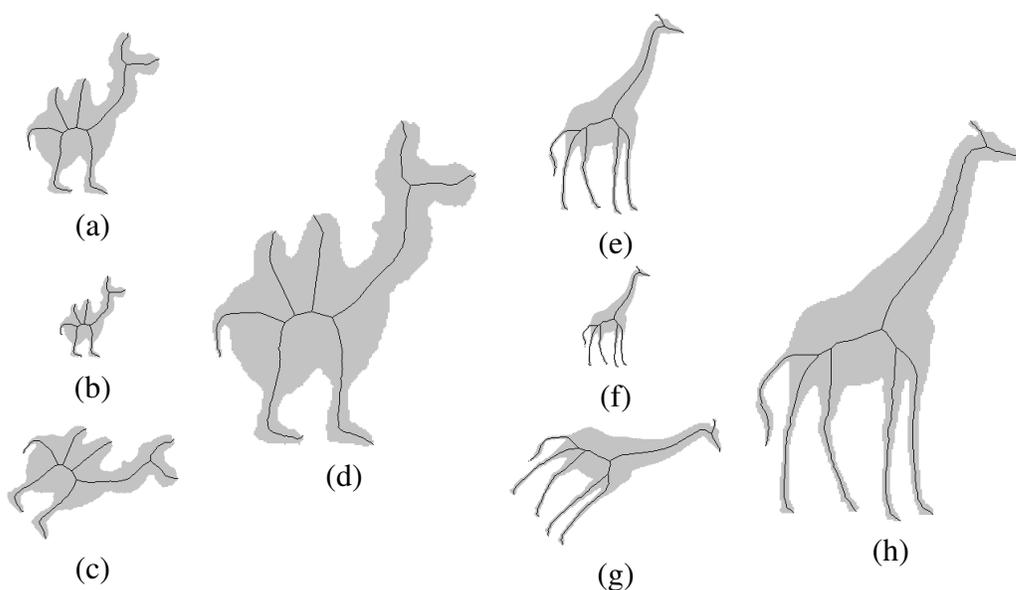


Figura 5.6: Esqueletos generados por el método propuesto para $s = 1.27$ sobre formas articuladas con transformaciones geométricas: (a) y (e) forma original; (b) y (f) factor de escalamiento 0.5; (c) y (g) 45° de rotación; (d) y (h) factor de escalamiento 2.0

En la *figura 5.9* podemos ver los esqueletos generados para cinco formas articuladas. El método por Zhang & Suen (1984) produce esqueletos libres de ramas no deseadas o faltantes (ver *figura 5.9a*), sin embargo es común que este método produzca esqueletos con ramas finales recortadas, este efecto es considerado en algunos trabajos como pérdida de información estructural. Respecto al método de Bai et al. (2007)(ver *figura 5.9b*) se puede observar que para un mismo parámetro, el método no genera buenos resultados sobre colecciones de imágenes que poseen características diferentes; podemos ver que algunos esqueletos contienen ramas faltantes y ramas no deseadas, por lo que definir el parámetro N adecuado para cada forma es un proceso tedioso y que no garantiza una buena representación de la forma mediante el esqueleto. Para los esqueletos obtenidos por Montero & Lang (2012)(ver *figura 5.9c*), el número de errores es menor en comparación a los obtenidos por Bai et al. (2007); no obstante, se puede observar que el método de Montero & Lang (2012) elimina algunas ramas importantes. Finalmente, podemos observar que el

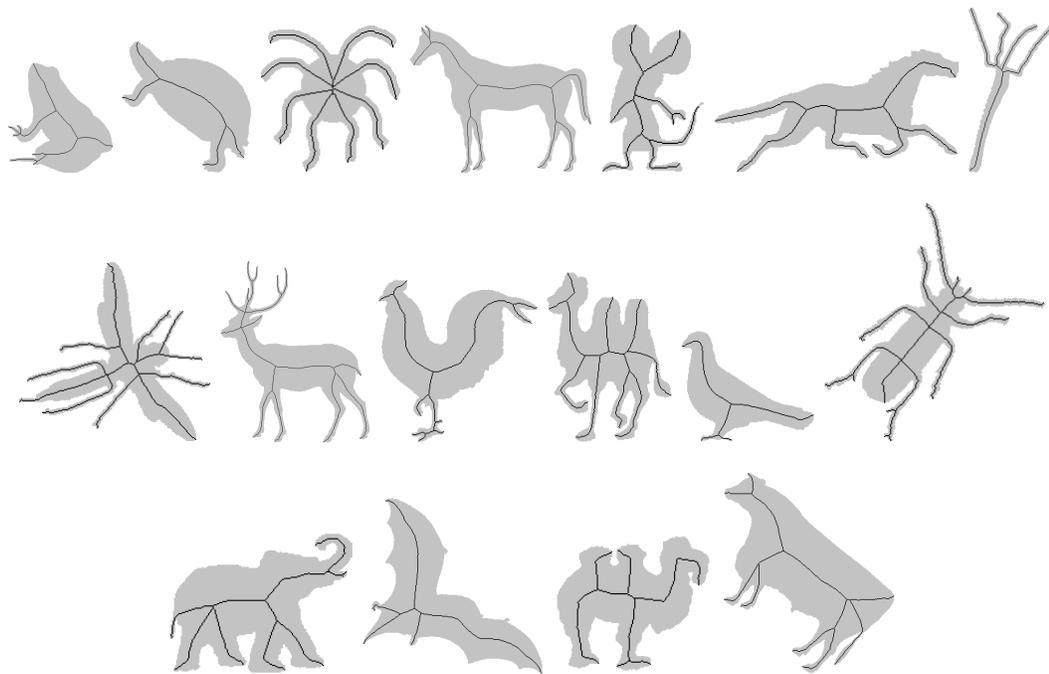


Figura 5.7: Esqueletos de diferentes formas articuladas extraídos con el método propuesto, haciendo uso del mismo parámetro de entrada $s = 1.27$ para diferentes formas articuladas obtenidas de *MPEG-7* (Latecki & Lakämper, 2000)

método propuesto (ver *figura 5.9d*) genera una buena representación, libre de ramas no deseadas y sin pérdida de ramas que representan información importante para formas con características diferentes, haciendo uso de un mismo valor ($s = 1.27$) para el parámetro de entrada.

Una de las técnicas más comunes para evaluar la calidad de los esqueletos es la razón de error de reconstrucción *RER* (del inglés *Reconstruction Error Ratio*). *RER* evalúa la diferencia entre la forma reconstruida a partir del esqueleto y la forma original. La forma es reconstruida a partir de las distancias asociadas en el mapa de distancia a cada pixel del esqueleto, las cuales corresponden a los radios de los discos máximos inscritos en la forma, cuyos centros de disco máximo corresponden a los puntos del esqueleto (ver *ecuación 5.1*).

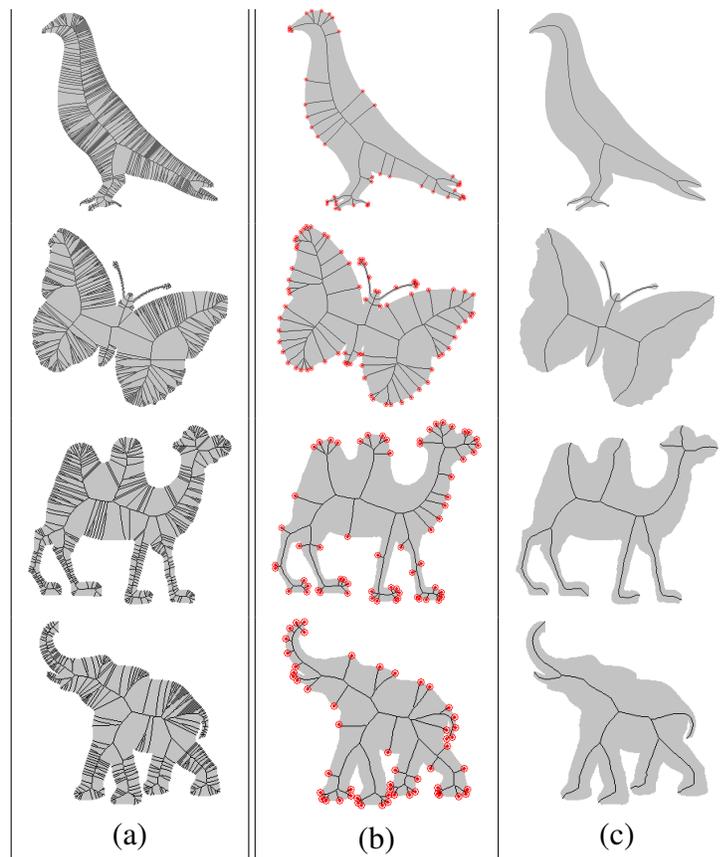


Figura 5.8: Esqueletos generados por el método propuesto para diferentes formas articuladas que pertenecen a la base de datos *MPEG-7*: (a) esqueletos generados por *IMA*; (b) esqueletos generados con ramas finales incidentes en los puntos convexos dominantes obtenidos mediante el método no-paramétrico, (c) esqueleto final podado

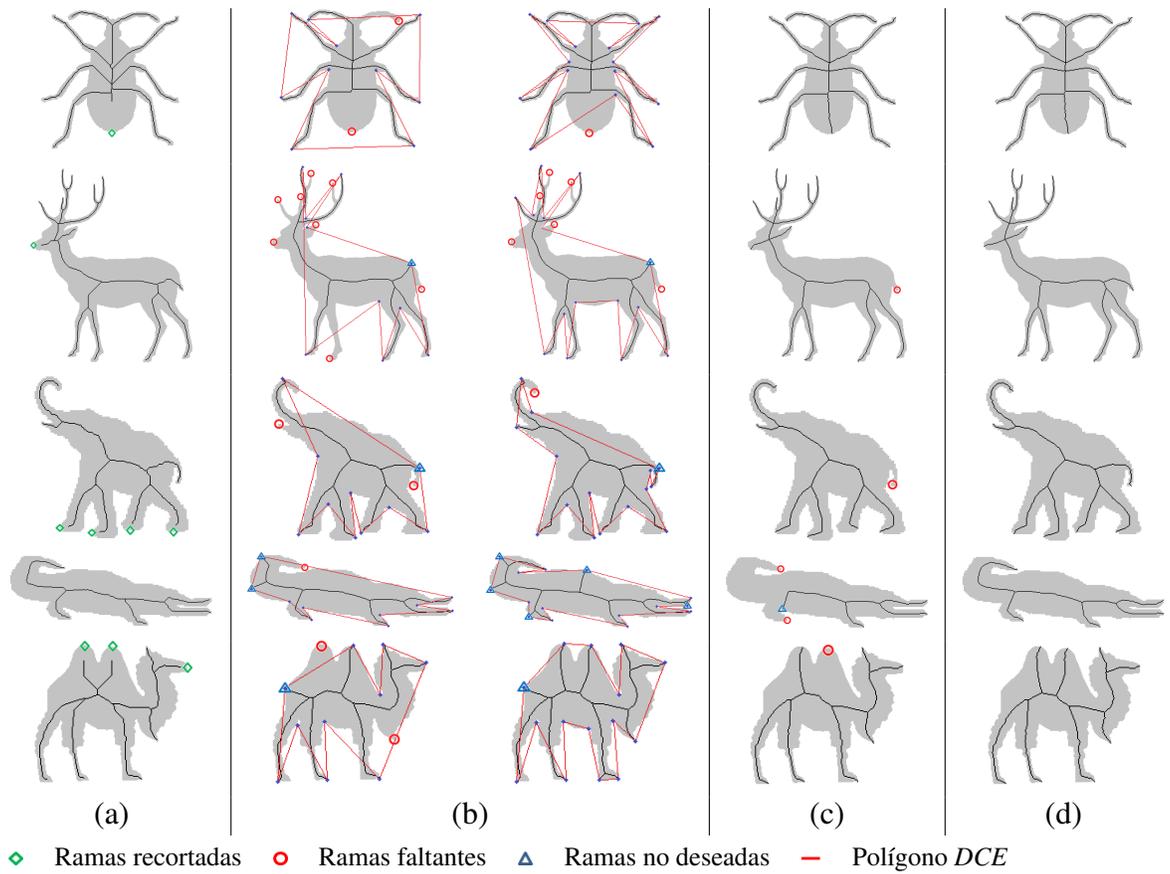


Figura 5.9: (a) Esqueletos obtenidos por Zhang & Suen (1984). (b) Esqueletos calculados por Bai et al. (2007) para $N = 10$ y $N = 15$ en la segunda y tercera columna respectivamente. (c) Esqueletos generados por Montero & Lang (2012) para $d = 15$ y $s = 1.25$. (d) Esqueletos obtenidos mediante el método propuesto para $s = 1.27$

$$RER(S, A) = \frac{|area(A) - area(R(S))|}{area(A)} \quad (5.1)$$

Donde S es el esqueleto de la forma A , y $R(S)$ es la forma reconstruida a partir de S , $area(.)$ denota el área de la forma medida en píxeles.

En la *tabla 5.1* se muestran resultados del método propuesto y los métodos propuestos por Zhang & Suen (1984); Bai et al. (2007) y Montero & Lang (2012) en función del tiempo de ejecución y el error de reconstrucción (RER) para una forma que representa a un camello.

Como se puede observar en la *tabla 5.1*, nuestro método obtiene el mejor RER y ocupa el segundo lugar respecto al tiempo de ejecución requerido para la construcción del esqueleto, mientras que el método más rápido fue el que obtuvo el peor resultado en RER . Aunque el método de adelgazamiento propuesto por Zhang & Suen (1984) genera una buena representación de la forma es el que consume mayor tiempo de ejecución y el error de reconstrucción es mucho mayor que el obtenido mediante el método propuesto en este trabajo.

Por el contrario, los resultados generados por Bai et al. (2007) y Montero & Lang (2012) no son buenos respecto a RER a causa de las ramas faltantes en el esqueleto, evitando la reconstrucción de algunas regiones de la forma. En las figuras de la *tabla 5.1* el área reconstruida a partir del esqueleto está marcada en gris y el error de reconstrucción en negro.

Para validar la calidad de los esqueletos obtenidos mediante el método propuesto, se realizó el experimento sobre cada conjunto de datos (Mio et al., 2007; Aslan & Tari, 2005; Aslan et al., 2008; Baseski et al., 2009; Latecki & Lakämper, 2000). Los resultados obtenidos para cada clase en cada una de las bases de datos y para cada conjunto de datos, depende en gran medida del tamaño de las imágenes, por lo tanto, en la *tabla 5.2* se muestra información respecto al tamaño de las imágenes que hacen parte de cada una de las bases de datos. En la *tabla 5.2* la *columna 1* muestra el nombre de la colección de imá-

Tabla 5.1: Comparación de los resultados obtenidos mediante el método propuesto y los métodos Zhang & Suen (1984), Bai et al. (2007) y Montero & Lang (2012) en función del tiempo de ejecución y la calidad del esqueleto (*RER*) para una sola forma articulada

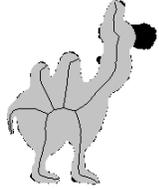
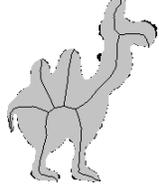
	Parámetros	Tiempo de ejecución (Segundos)	Calidad del esqueleto (<i>RER</i>)	Forma reconstruida
Zhang & Suen (1984)	–	2.6399	0.0560	
Bai et al. (2007)	$N = 10$	2.2068	0.0663	
Montero & Lang (2012)	$d = 15$ $s = 1.25$	1.2534	0.0709	
Método propuesto	$s = 1.27$	1.6842	0.0281	

Tabla 5.2: Información sobre el tamaño de las imágenes en las bases de datos utilizadas en la construcción de esqueletos

Colección	<i>MIN</i>	<i>MAX</i>	Promedio
Mio et al. (2007)	76 × 117	121 × 333	81 × 185
Aslan & Tari (2005)	152 × 162	381 × 356	229 × 244
Aslan et al. (2008)	96 × 86	280 × 828	208 × 250
Baseski et al. (2009)	73 × 106	452 × 708	199 × 253
Latecki & Lakämper (2000)	50 × 83	993 × 1010	363 × 407

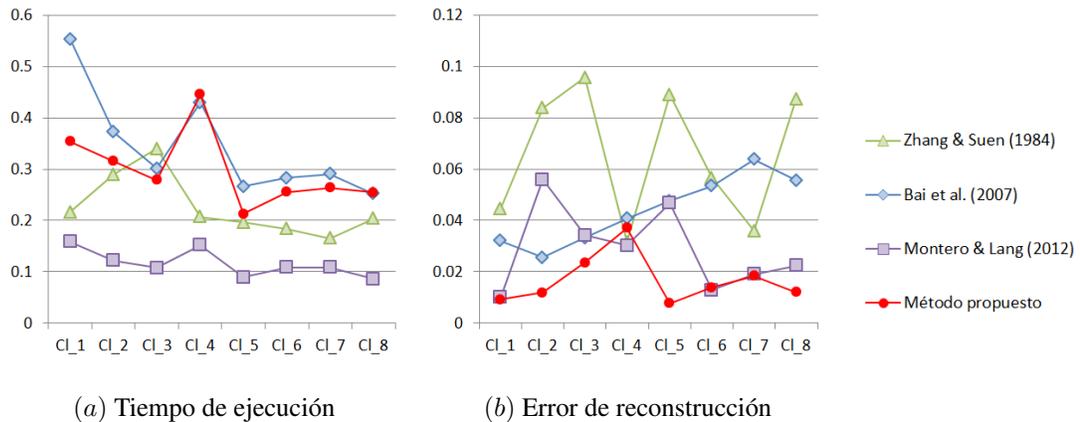
genes, la *columna 2* y la *columna 3* muestra el tamaño de la imagen más pequeña y la más grande en la colección de imágenes respectivamente, la *columna 4* muestra el tamaño promedio de las imágenes para cada base de datos. El tamaño esta dado en *alto × ancho*.

En la *tabla 5.3*, *tabla 5.4*, *tabla 5.5*, *tabla 5.6* y *tabla 5.7* podemos ver el promedio en *RER* y tiempo de ejecución para las bases de datos Mio et al. (2007); Aslan & Tari (2005); Aslan et al. (2008); Baseski et al. (2009) y Latecki & Lakämper (2000), respectivamente. Como se puede observar en cada una de las tablas, para cada uno de los conjuntos de datos, nuestro método obtiene el mejor resultado en *RER* y es competitivo en tiempo de ejecución.

Para la base de datos Mio et al. (2007) en la *tabla 5.3*, el mejor tiempo de ejecución lo obtiene el método de Montero & Lang (2012), mientras que nuestro método obtiene el tercer lugar. Para las bases de datos de Aslan et al. (2008) y Latecki & Lakämper (2000) en la *tabla 5.5* y la *tabla 5.7* respectivamente, nuestro método ocupa el primer lugar en tiempo de ejecución, mientras que para Aslan & Tari (2005) y Baseski et al. (2009) en la *tabla 5.4* y *tabla 5.6* nuestro métodos se mantiene como el segundo más rápido para procesar cada una de las bases de datos.

Tabla 5.3: Promedio en tiempo de ejecución y error de reconstrucción (*RER*) para la base de datos Mio et al. (2007)

Método	Parámetros	Promedio tiempo de ejecución (segundos)	Promedio calidad del esqueleto (<i>RER</i>)
Zhang & Suen (1984)	–	0.2251	0.0655
Bai et al. (2007)	$N = 10$	0.3436	0.0439
Montero & Lang (2012)	$d = 15$ $s = 1.25$	0.1166	0.0289
Método propuesto	$s = 1.27$	0.2975	0.0167

Figura 5.10: Tiempo de ejecución y calidad del esqueleto (*RER*) para cada clase en la base de datos Mio et al. (2007)

La variación en los tiempos de ejecución para cada una de las bases de datos depende de diferentes factores, los cuales están principalmente relacionados con el tamaño de la imagen (ver figura 5.2) o el ancho en el *foreground* de la imagen. Para una imagen con pocos píxeles de ancho en el *foreground*, el método de Zhang & Suen (1984) puede requerir menor tiempo de ejecución respecto a los otros métodos, puesto que pocas iteraciones son necesarias para adelgazar la imagen. En el método de Bai et al. (2007) el tiempo de ejecución está relacionado con el número de puntos convexos obtenidos en el contorno de

la imagen. El tiempo de ejecución requerido por el método de Montero & Lang (2012), tiene una relación entre el tamaño de la imagen y el valor d , puesto que si d es pequeño respecto a la imagen, el método puede requerir varias iteraciones para que la condición de ajuste (distancia máxima d permitida entre la curva original y la curva ajustada) en la aproximación al contorno se cumpla para toda la imagen, lo anterior influye en el número de puntos obtenidos desde el contorno y por ende en el número de ramas finales, las cuales deben ser procesadas en la etapa final de poda. El tiempo de ejecución requerido por nuestro método, está relacionado con la cantidad de ruido y la curvatura presente en la imagen, estos dos aspectos influyen en la cantidad de puntos dominantes obtenidos y por lo tanto en la construcción del esqueleto podado.

En la *figura 5.10*, *figura 5.11*, *figura 5.12*, *figura 5.13* y *figura 5.14* se muestran gráficas que reflejan el promedio en el tiempo de ejecución y el error de reconstrucción para cada clase de imágenes en cada una de las bases de datos. Cada clase tiene un identificador, el cual ha sido previamente definido durante la presentación de cada una de las bases de datos.

El método de Zhang & Suen (1984) en la *figura 5.10a* a diferencia de las demás figuras (*figura 5.11a*, *figura 5.12a*, *figura 5.13a* y *figura 5.14*) donde se gráfica el tiempo de ejecución requerido por cada uno de los métodos sobre cada clase de imágenes, obtiene un tiempo promedio de ejecución competitivo con el obtenido por los demás métodos, esto se debe al tamaño de las imágenes en la base de datos. Para la mayoría de clases, el método de Zhang & Suen (1984) produce el más alto error de reconstrucción. Para algunas clases, con método de Bai et al. (2007) se puede observar que se obtiene un error de reconstrucción muy cercano a cero como en el caso de *Cl_14* en la *figura 5.11b*; *Cl_4*, *Cl_14* y *Cl_20* en la *figura 5.12b*; *Cl_13*, *Cl_25* y *Cl_30* en la *figura 5.13b*; y *Cl_9* y *Cl_24* en la *figura 5.14b*, lo anterior se debe a que el número de vértices obtenidos en la aproximación poligonal *DCE* es mayor al número de píxeles finales requeridos para representar a la forma a través del esqueleto. Para otras clases como *Cl_12* en la *figura 5.12b*, *Cl_4*, *Cl_10*,

Cl_{17} y Cl_{14} en la figura 5.14b se obtienen los peores errores de reconstrucción debido a que el conjunto de vértices no es suficiente para construir un esqueleto que represente apropiadamente a las formas que pertenecen a esas clases.

En cada una de las figuras (figura 5.10, figura 5.11, figura 5.12, figura 5.13 y figura 5.14) podemos observar que los resultados promedio obtenido por nuestro método para cada una de las clases es al menos competitivo tanto en tiempo de ejecución como en error de reconstrucción, obteniendo en muchos casos los mejores resultados.

Tabla 5.4: Promedio en tiempo de ejecución y error de reconstrucción (RER) para la base de datos Aslan & Tari (2005)

Método	Parámetros	Promedio tiempo de ejecución (segundos)	Promedio calidad del esqueleto (RER)
Zhang & Suen (1984)	–	3.8776	0.0442
Bai et al. (2007)	$N = 10$	2.4822	0.0120
Montero & Lang (2012)	$d = 15$ $s = 1.25$	1.3951	0.0186
Método propuesto	$s = 1.27$	1.8157	0.0115

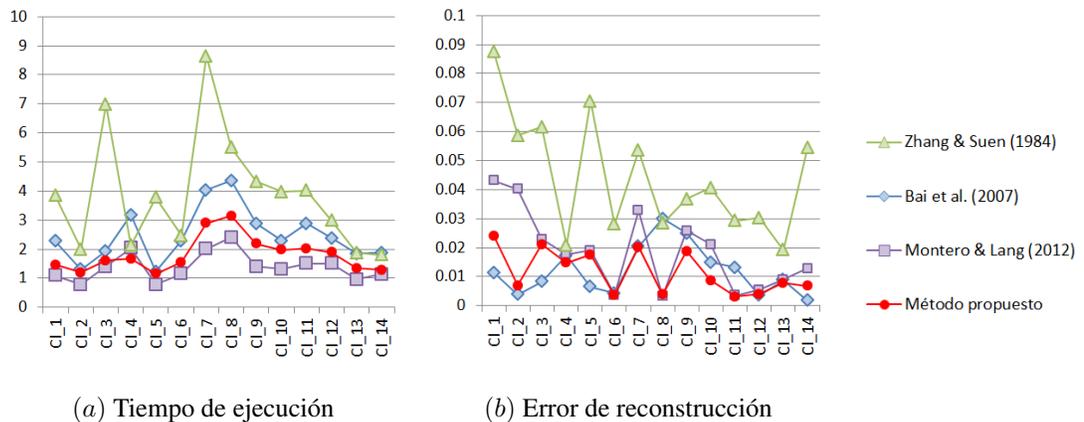


Figura 5.11: Tiempo de ejecución y calidad del esqueleto (RER) para cada clase en la base de datos Aslan & Tari (2005)

Tabla 5.5: Promedio en tiempo de ejecución y error de reconstrucción (*RER*) para la base de datos Aslan et al. (2008)

Método	Parámetros	Promedio tiempo de ejecución (segundos)	Promedio calidad del esqueleto (<i>RER</i>)
Zhang & Suen (1984)	–	3.8289	0.0512
Bai et al. (2007)	$N = 10$	2.0127	0.0129
Montero & Lang (2012)	$d = 15$ $s = 1.25$	1.5021	0.0188
Método propuesto	$s = 1.27$	1.4545	0.0106

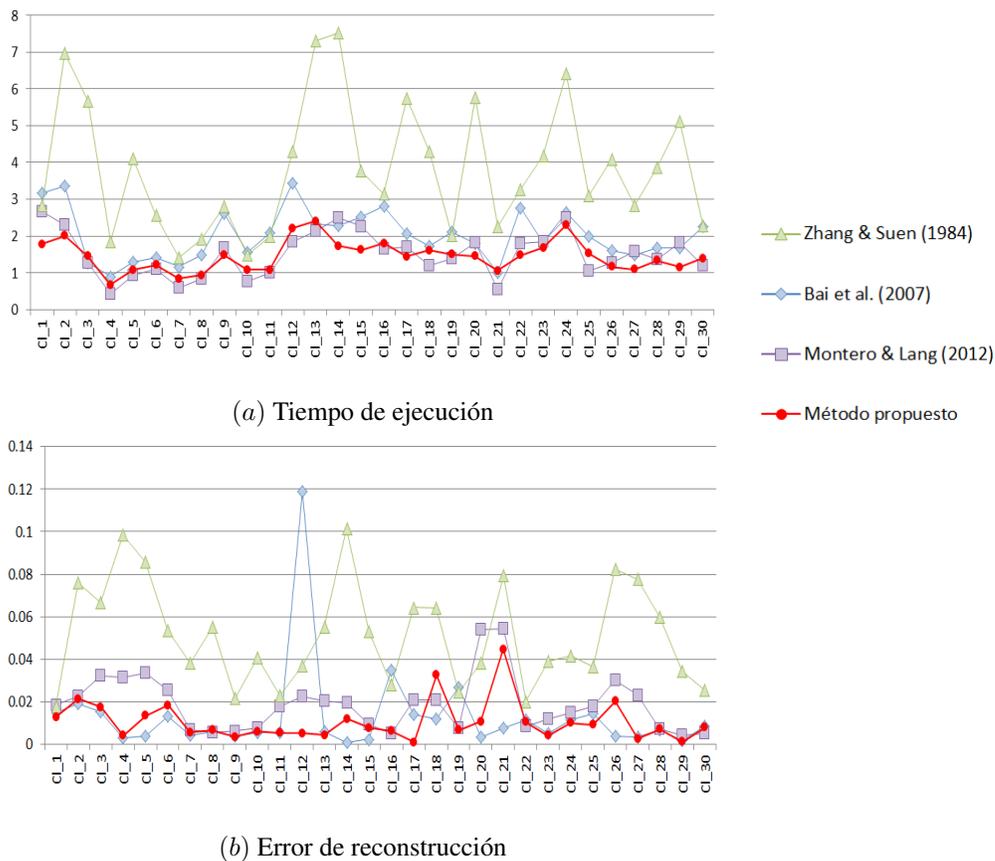
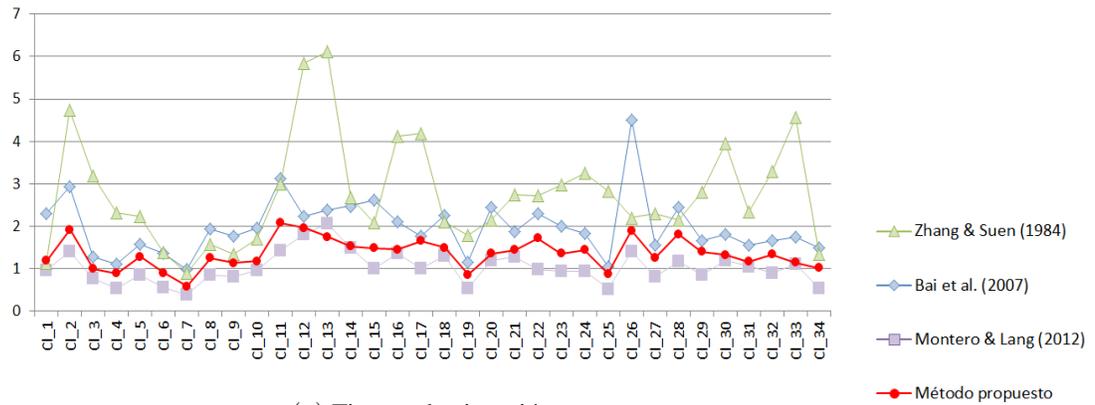


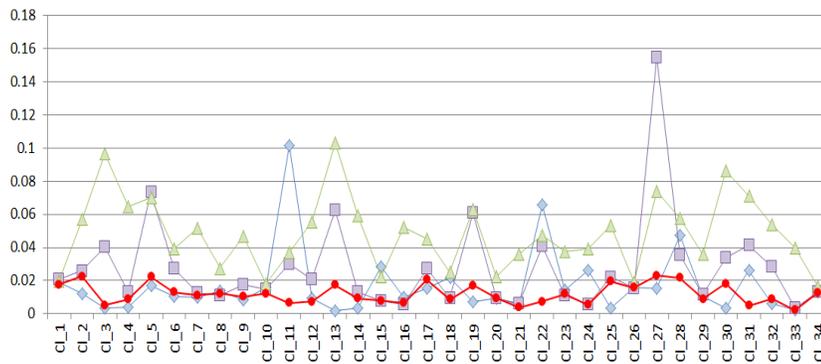
Figura 5.12: Tiempo de ejecución y calidad del esqueleto (*RER*) para cada clase en la base de datos Aslan et al. (2008)

Tabla 5.6: Tiempo promedio de ejecución y error de reconstrucción (*RER*) para la base de datos Baseski et al. (2009)

Método	Parámetros	Promedio tiempo de ejecución (segundos)	Promedio calidad del esqueleto (<i>RER</i>)
Zhang & Suen (1984)	–	2.7650	0.0483
Bai et al. (2007)	$N = 10$	1.9786	0.0170
Montero & Lang (2012)	$d = 15$	1.0308	0.0272
	$s = 1.25$		
Método propuesto	$s = 1.27$	1.3592	0.0129



(a) Tiempo de ejecución

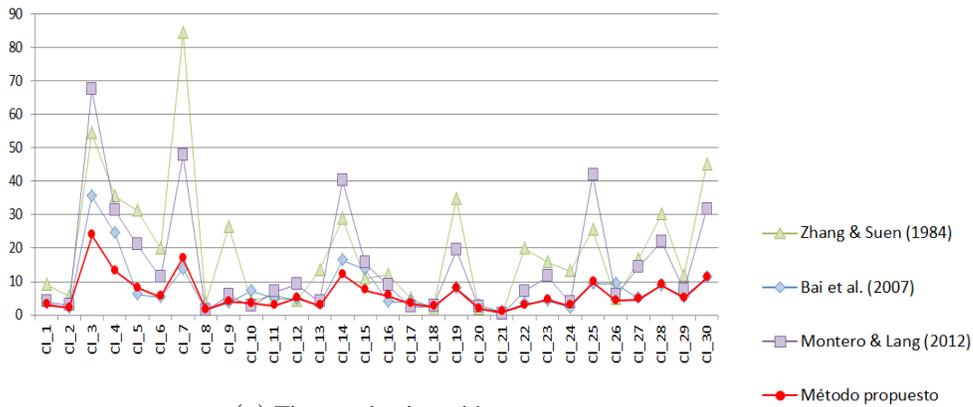


(b) Error de reconstrucción

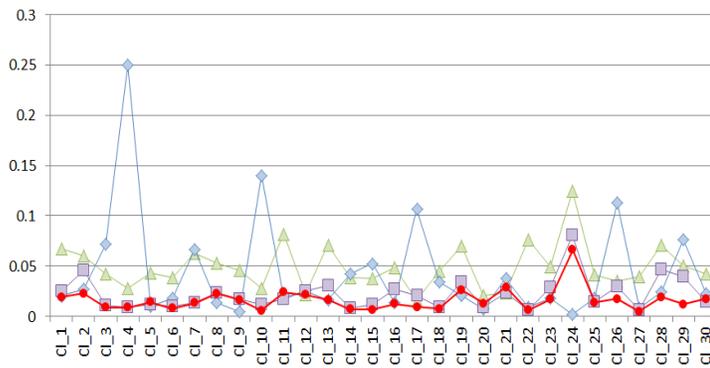
Figura 5.13: Tiempo de ejecución y calidad del esqueleto (*RER*) para cada clase en la base de datos Baseski et al. (2009)

Tabla 5.7: Tiempo promedio de ejecución y el error de reconstrucción (*RER*) para la base de datos Latecki & Lakämper (2000)

Método	Parámetros	Promedio tiempo de ejecución (segundos)	Promedio calidad del esqueleto (<i>RER</i>)
Zhang & Suen (1984)	–	19.2083	0.0490
Bai et al. (2007)	$N = 10$	7.56667	0.0430
Montero & Lang (2012)	$d = 15$	15.0156	0.0221
	$s = 1.25$		
Método propuesto	$s = 1.27$	6.2864	0.0162



(a) Tiempo de ejecución



(b) Error de reconstrucción

Figura 5.14: Tiempo de ejecución y calidad del esqueleto (*RER*) para cada clase en la base de datos Latecki & Lakämper (2000)

Es necesario resaltar que no existe una medida que permita calcular la calidad del esqueleto en función de la simplicidad y el error de reconstrucción de manera combinada, puesto que para un esqueleto con todas las ramas no deseadas que representan el ruido y las deformaciones presentes en la forma, el *RER* es cercano o igual a cero.

5.2. Clasificación

Para evaluar los resultados de clasificación se utilizó el método de clasificación propuesto en la sección 4.3, en nuestros experimentos se usó validación cruzada con un tamaño de ventana igual a uno (*leave-one-out cross-validation*) sobre el conjunto de grafos pesados obtenidos a partir de los esqueletos, ya que la cantidad de ejemplos por clase en las diferentes bases de datos es pequeña; la base de datos que contiene un mayor número de ejemplos por clase es la presentada por Baseski et al. (2009), con 20 ejemplos por clase.

En la *tabla 5.8* se muestra la información de las bases de datos utilizadas para las pruebas de clasificación. En la primera columna se presenta la información sobre los autores de la base de datos. La segunda columna corresponde al identificador asignado a cada conjunto de datos para hacer referencia en el resto del documento. La tercera y cuarta columna proporcionan respectivamente información acerca del número de clases y el número de imágenes por clase que contiene cada base de datos.

Tabla 5.8: Bases de datos utilizadas en el proceso de clasificación

Colección	Etiqueta	N° Clases	N° Formas por clase
Mio et al. (2007)	50H	8	6 ¹
Aslan & Tari (2005)	56F	14	4
Aslan et al. (2008)	180F	30	6
Baseski et al. (2009)	680F	34	20

En esta sección se muestra una comparación experimental entre el método propuesto

¹Promedio de formas por clase: cada clase contiene diferente número de formas articuladas.

en esta tesis y el método de clasificación propuesto en (Bai & Latecki, 2008). Los parámetros α y N son utilizados en Bai & Latecki (2008); α corresponde a un factor de peso que permite darle mayor importancia al valor de similaridad obtenido entre las distancias de los caminos geodésicos que emanan desde los píxeles finales de los esqueletos. N corresponde al número de radios equidistantes tomados sobre cada camino geodésico, para el cálculo de similaridad entre pares de píxeles finales.

Debido a que el método propuesto para el cálculo de similaridad entre los vértices que representan píxeles finales en el esqueleto es similar al presentado por (Bai & Latecki, 2008), en nuestro método utilizamos el mismo valor $\beta = 70$.

Los experimentos son llevados a cabo haciendo uso de los dos métodos de clasificación sobre las diferentes bases de datos (presentadas en las columnas en la tabla 5.9) haciendo uso de los diferentes métodos para la construcción del esqueleto (corresponde a las filas en la tabla 5.9). El objetivo es validar la utilidad del método propuesto para la construcción de esqueletos, en combinación con el esquema de clasificación presentado en este trabajo, haciendo uso de los grafos pesados obtenidos a partir de los esqueletos.

En los experimentos también mostramos el tiempo de ejecución promedio requerido por cada método para llevar a cabo el proceso de clasificación sobre cada base de datos.

Como se puede observar en la *tabla 5.9*, el método propuesto de clasificación junto con el método propuesto para la construcción y poda de esqueletos obtiene los mejores resultados de clasificación para las bases de datos $56F$ y $680F$, con una exactitud de 98.21 % y 94.41 % respectivamente. Sin embargo, aunque el resultado de clasificación obtenido para las bases de datos $50F$ y $180F$ es el mismo para ambos métodos (100 % y 97.8 %, respectivamente), el tiempo de ejecución del método propuesto en esta tesis es mucho menor al requerido por Bai & Latecki (2008).

En general, los resultados alcanzados con el método propuesto, el cual incluye la construcción de esqueletos podados, la construcción de grafos pesados basados en esqueleto y la clasificación, genera los mejores resultados respecto a los otros métodos. Además, es

Tabla 5.9: Resultados de la clasificación, en términos del porcentaje de aciertos (*exactitud*), utilizando el método propuesto y el método de Bai & Latecki (2008) en varias colecciones de imágenes

Método para la construcción de esqueletos	Método de clasificación								
	Bai & Latecki (2008) $\alpha = 70$ y $m = 50$			Propuesto $\alpha = 70$					
	50F	56F	180F	680F	50F	56F	180F	680F	
Zhang & Suen (1984)	100 %	89.29 %	90.00 %	89.11 %	98.00 %	94.64 %	86.67 %	87.50 %	
Bai et al. (2007)	100 %	96.43 %	94.44 %	93.68 %	100 %	94.64 %	95.00 %	92.50 %	
Montero & Lang (2012)	100 %	96.43 %	95.56 %	93.68 %	100 %	94.64 %	95.56 %	88.97 %	
Método Propuesto	100 %	96.43 %	97.78 %	93.68 %	100 %	98.21 %	97.78 %	94.41 %	
Tiempo de ejecución	7.8min	9.5min	1.67h	23.5h	25seg	38seg	6.6min	1.6h	

de gran importancia resaltar el tiempo de ejecución y el número de parámetros de entrada requerido por nuestro método para llevar a cabo todo el proceso, el cual es en total de únicamente dos parámetros (s y α); donde s corresponde al parámetro para la poda del esqueleto y α es el factor de peso para el cálculo de la distancia que permite encontrar la similitud entre los grafos.

5.3. Discusión

La construcción de un esqueleto descriptivo, donde cada parte importante esté representada por una rama final en el esqueleto y que provea un error de reconstrucción lo más cercano a cero es una tarea no trivial, especialmente si deseamos que el método genere buenos resultados haciendo uso de los mismos parámetros de entrada para un conjunto de imágenes con diferentes características. El método debe considerar factores tales como el ruido, altas deformaciones, rotación, escalamiento, entre otras.

Teniendo en cuenta los resultados obtenidos por los diferentes métodos en la construcción de esqueletos, se puede observar que tanto el tiempo de ejecución como el error de reconstrucción dependen del tamaño, de las características de cada imagen (curvatura, ruido, rotación, entre otras), de los parámetros de entrada asignados a cada método y de la manera en que cada método evalúa la información del objeto para la construcción del esqueleto. Por ejemplo, el método de Zhang & Suen (1984) a pesar de ser no-paramétrico es poco útil para representar formas articuladas, puesto que a menudo produce ramas recortadas o ramas no deseadas (en dependencia del ruido presente en la imagen), además el esqueleto resultante no es consistente bajo factores de rotación o escalamiento, puesto que es dependiente del orden en que los píxeles del contorno van siendo procesados. En el método de Bai et al. (2007), asignar el mismo parámetro de entrada para todo el conjunto de datos puede provocar ramas no deseadas o ramas faltantes en el esqueleto, sin embargo la asignación de un parámetro para cada imagen mediante prueba y error es un proceso

tedioso y poco práctico y que además, no garantiza una buena representación, puesto para algunas imágenes, un esqueleto sin ramas faltantes implica la presencia de ramas no deseadas. Para el método de Montero & Lang (2012) la cantidad de puntos dominantes obtenidos a partir del contorno, es dependiente del tamaño de la imagen, del valor asignado a d y del punto de partida en el contorno para la aproximación por curvas cúbicas de Bezier. Por otro lado, la condición de poda no garantiza que la importancia de la rama sea considerada en su totalidad, especialmente para formas con partes articuladas, donde la deformación provoca que el pixel final de la rama que representa a la parte articulada, caiga muy cerca al contorno adyacente del disco máximo del pixel de unión de dicha rama. Otro aspecto que afecta los resultados del método en cuanto a *RER* es la evaluación sobre todas las ramas finales del esqueleto en cada iteración, puesto que puede provocar la poda de ramas importantes para la representación de la forma a través del esqueleto.

El método para la construcción de esqueletos propuesto en este trabajo, produce buenos resultados independiente del tamaño, rotación y curvatura en el contorno que describe a la forma. El método no-paramétrico para la obtención de puntos dominantes, garantiza que el conjunto de puntos dominantes dependa de la información del contorno en cada forma articulada, y por ende el primer esqueleto este libre de pérdida de ramas importantes. La etapa de poda introducida en este trabajo, evita la pérdida de información estructural, preservando ramas finales que representan información importante y eliminando ramas que representan pequeñas variaciones o ruido presente en la forma articulada.

Las características obtenidas a partir del esqueleto para representar a las formas articuladas por medio de grafos pesados, logran abstraer la información que describe a la forma y por lo tanto muestran ser útiles en tareas de clasificación, además la topología del grafo y la forma en que las correspondencias para dos grafos G y G' son encontradas permiten que el tiempo de ejecución consumido por nuestro método sea mucho menor, al requerido por el método de Bai & Latecki (2008). Lo anterior se debe al número de veces que el método *OSB* es aplicado para obtener el valor de similaridad. Por ejemplo, en (Bai

& Latecki, 2008) para dos grafos G y G' con N y M píxeles finales respectivamente, el método *OSB* es aplicado $M \times N$ veces para calcular la similaridad entre pares de píxeles finales; mientras que en nuestro método se aplica una sola vez para calcular la similaridad total, una vez que la matriz ha sido ordenada según el par de vértices más parecidos en G y G' . Lo anterior provoca que nuestro método sea aproximadamente 16 veces más rápido que el método propuesto por Bai & Latecki (2008), lo cual cobra importancia a medida que la cantidad de información a ser procesada aumenta, y por ende el número de comparaciones.

Capítulo 6

Conclusiones

6.1. Conclusiones

En esta tesis se propone un método para la construcción de esqueletos a partir de formas articuladas *2D*. El método propuesto en esta tesis para la construcción de esqueletos es comparado con tres métodos propuestos en la literatura, dos de los cuales (Bai et al., 2007; Montero & Lang, 2012) trabajan de forma similar a nuestra propuesta, puesto que utilizan segmentación del contorno a trozos para la construcción de una primera aproximación al esqueleto final, además están ubicados en la categoría de algoritmos basados en transformada de distancia. El otro método con el que comparamos nuestra propuesta es el método presentado por Zhang & Suen (1984), éste es uno de los métodos más populares y reconocidos en la literatura, pertenece a la categoría de algoritmos de adelgazamiento y, aunque este tipo de algoritmos requiere de varias condiciones para el adelgazamiento del contorno, es un método no-paramétrico. Teniendo en cuenta los experimentos reportados se pudo llegar a las siguientes conclusiones:

- El método propuesto en esta tesis para la construcción de esqueletos a partir de formas articuladas *2D* produce los mejores resultados en *RER* y es competitivo en

tiempo de ejecución respecto a los resultados obtenidos con los métodos presentados por Zhang & Suen (1984), Bai et al. (2007) y Montero & Lang (2012).

- El método propuesto para la construcción de esqueletos hace uso de un único parámetro de entrada y genera buenos resultados para colecciones de imágenes que contienen diferentes características, a diferencia de otros métodos recientes reportados en la literatura; donde los parámetros de entrada deben ser ajustados para cada forma o formas con características similares, además, por lo general, dichos métodos requieren de más de un parámetro de entrada.

Los esqueletos obtenidos a partir de formas articuladas *2D* fueron empleados para la construcción de grafos pesados y estos a su vez utilizados en tareas de clasificación. El método para la construcción de grafos pesados y la estrategia de clasificación propuesta se comparó contra otros métodos de construcción de esqueletos y con el método de clasificación propuesto por Bai & Latecki (2008). Teniendo en cuenta los experimentos reportados podemos concluir:

- Las características obtenidas a partir del esqueleto para la construcción de los grafos pesados son útiles para la clasificación de formas articuladas *2D*, puesto que permiten mejorar los resultados de clasificación.
- El método de clasificación propuesto, combinado con los grafos pesados obtenidos a partir de los esqueletos generados por el método propuesto para la obtención de esqueletos podados a partir de formas articuladas, obtiene los mejores resultados de clasificación (*accuracy*) respecto a los resultados de clasificación obtenidos mediante el método propuesto por Bai & Latecki (2008) junto con los diferentes métodos para la construcción de esqueletos y el método de clasificación propuesto en combinación con los métodos para la construcción de esqueletos presentados por Zhang & Suen (1984), Bai et al. (2007) y Montero & Lang (2012).

- El tiempo de ejecución requerido por la estrategia de clasificación propuesta, junto con los grafos pesados obtenidos a partir de los esqueletos, es mucho menor al requerido por el método propuesto por Bai & Latecki (2008).

6.2. Aportaciones del trabajo de investigación

Las aportaciones de este trabajo de investigación son las siguientes:

- Un método para la construcción de esqueletos podados a partir de formas articuladas 2D, obteniendo mejores resultados en *RER* que con los métodos propuestos por Zhang & Suen (1984), Bai et al. (2007) y Montero & Lang (2012), y manteniendo un compromiso entre número de parámetros de entrada y tiempo de ejecución.
- Un método para la construcción de grafos pesados a partir de esqueletos obtenidos desde formas articuladas 2D.
- Un método para la clasificación de formas articuladas, basado en la similaridad obtenida entre pares de grafos que representan formas articuladas, mejorando el *accuracy* en la clasificación.

6.2.1. Publicaciones derivadas de este trabajo de tesis

- Laura Alejandra Pinilla-Buitrago, José Francisco Martínez-Trinidad, Jesús Ariel Carrasco-Ochoa: *New Penalty Scheme for Optimal Subsequence Bijection*. CIARP (1) 2013: LNCS 8258 206-213.
- Laura Alejandra Pinilla-Buitrago, José Francisco Martínez-Trinidad, Jesús Ariel Carrasco-Ochoa: *A New Method for Skeleton Pruning*. MCPR 2014: LNCS 8495 301-310

6.3. Trabajo futuro

Los resultados obtenidos en esta tesis nos motivan a realizar como trabajo futuro:

- Un estudio de otras técnicas de podado que incluyan medidas de importancia globales y locales para cada rama, con las cuales se podrían obtener buenas representaciones para las formas *2D*, manteniendo un compromiso entre el tiempo de ejecución y el número de parámetros de entrada, además que supere los resultados obtenidos en este trabajo.
- Estudiar qué otro tipo de características se pueden obtener a partir de los esqueletos, para la construcción de grafos pesados útiles en tareas de clasificación de formas articuladas.
- Extender el método propuesto en este trabajo para la construcción y poda de esqueletos sobre imágenes *3D*.
- Extender el método de construcción de grafos pesados y clasificación a formas articuladas *3D*.

Notacion

Abreviatura	Significado
O	Forma articulada
S	Forma binaria obtenida a partir de O
$N_8(p)$	Vecindad 8 de p
DT	Transformada de distancia
FT	Transformada de características
min	mínimo
$d(\cdot, \cdot)$	Distancia entre dos puntos
G_k	Grafo estrella con k hojas
$dc(\cdot, \cdot)$	Distancia entre dos vértices
$\gamma(G, G')$	Distancia entre pares de vértices de dos Grafos G y G'
std	Desviación estándar

Bibliografía

- Carlo Arcelli & Gabriella S. Di Baja. A width-independent fast thinning algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-7(4):463–474, Jul 1985.
- Carlo Arcelli & Gabriella S. Di Baja. Euclidean skeleton via centre-of-maximal-disc extraction. *Image and Vision Computing*, 11(3):163 – 173, 1993.
- Cagri Aslan & Sibel Tari. An axis-based representation for recognition. *ICCV*, 2005.
- Cagri Aslan, Aykut Erdem, Erkut Erdem, & Sibel Tari. Disconnected skeleton: Shape at its absolute scale. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(12):2188–2203, Dic 2008.
- Dominique Attali, Jean-Daniel Boissonnat, & Herbert Edelsbrunner. Stability and computation of medial axes - a state-of-the-art report. *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*, páginas 109–125, 2009.
- Soumen Bag & Gaurav Harit. An improved contour-based thinning method for character images. *Pattern Recognition Letters*, 32(14):1836 – 1842, 2011.
- Soumen Bag, Gaurav Harit, & Partha Bhowmick. Recognition of bangla compound characters using structural decomposition. *Pattern Recognition*, 47(3):1187 – 1201, 2014.
- Xiang Bai & Longin Jan Latecki. Discrete skeleton evolution. In *Energy Minimization Methods in Computer Vision and Pattern Recognition*, volumen 4679 of *Lecture Notes in Computer Science*, páginas 362–374. 2007.
- Xiang Bai & Longin Jan Latecki. Path similarity skeleton graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(7):1282–1292, Jul 2008.

- Xiang Bai, Longin J. Latecki, & Wen-yu Liu Liu. Skeleton pruning by contour partitioning with discrete curve evolution. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(3):449–462, 2007.
- Xiang Bai, Wenyu Liu, & Zhuowen Tu. Integrating contour and skeleton for shape classification. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, páginas 360–367, Sep 2009.
- Emre Baseski, Aykut Erdem, & Sibel Tari. Dissimilarity between two skeletal trees in a context. *Pattern Recognition*, 42(3):370–385, 2009.
- S. Belongie, J. Malik, & J. Puzicha. Shape matching and object recognition using shape contexts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(4):509–522, Abr 2002.
- Harry Blum. Biological shape and visual science. i. *Journal of theoretical biology*, (28): 205–287, 1973.
- Gunilla Borgefors. Distance transformations in digital images. *Computer Vision, Graphics, and Image Processing*, 34(3):344–371, 1986.
- Michael Brady & Haruo Asada. Smoothed local symmetries and their implementation. *International Journal of Robotics*, 3:36–61, 1984.
- Jonathan Worthen Brandt & V. Ralph Algazi. Continuous skeleton computation by voronoi diagram. *CVGIP: Image Understanding*, 55(3):329–338, 1992.
- Ángel Carmona-Poyato, Nicolás Luis Fernández-García, Rafael Medina-Carnicer, & Francisco José Madrid-Cuevas. Dominant point detection: A new proposal. *Image and Vision Computing*, 23(13):1226–1236, 2005.
- Wai-Pak Choi, Kin-Man Lam, & Wan-Chi Siu. Extraction of the euclidean skeleton based on a connectivity criterion. *Pattern Recognition*, 36(3):721–729, 2003.
- Philippe Cornic. Another look at the dominant point detection of digital curves. *Pattern Recognition Letters*, 18(1):13–25, 1997.
- Mary Demuth, Franz Aurenhammer, & Axel Pinz. Straight skeletons for binary shapes. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, páginas 9–16, Jun 2010.
- Weian Deng, Sitharama Sitharama Iyengar, & Nathan E. Brener. A fast parallel thinning algorithm for the binary image skeletonization. *Int. J. High Perform. Comput. Appl.*, 14 (1):65–81, Feb 2000.

- Pavel Dimitrov, James N. Damon, & Kaleem Siddiqi. Flux invariants for shape. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volumen 1, páginas I-835–I-841, Jun 2003.
- David H. Douglas & Thomas K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, 10(2):112–122, Oct 1973.
- Huichuan Duan, Jinling Wang, Xiyu Liu, & Hong Liu. A skeleton pruning approach using contour length as the significance measure. 1:360–364, Oct 2008.
- Aykut Erdem & Sibel Tari. A similarity-based approach for shape classification using aslan skeletons. *Pattern Recogn. Lett.*, 31(13):2024–2032, Oct 2010.
- Jacob Feldman & Manish Singh. Bayesian estimation of the shape skeleton. *Proceedings of the National Academy of Sciences*, 103(47):18014–18019, 2006.
- Yaorong Ge & J. Michael Fitzpatrick. On the generation of skeletons from discrete euclidean distance maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(11):1055–1066, Nov 1996.
- Polina Golland & William Eric Leifur Grimson. Fixed topology skeletons. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volumen 1, páginas 10–17, 2000.
- Lei He, Chia Y. Han, & William G. Wee. Object recognition and recovery by skeleton graph matching. In *Multimedia and Expo, 2006 IEEE International Conference on*, páginas 993–996, Jul 2006.
- Wim H. Hesselink & Jos B. T. M. Roerdink. Euclidean skeletons of digital image and volume data in linear time by the integer medial axis transform. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(12):2204–2217, 2008.
- Wim H. Hesselink, Menno Visser, & Jos B. T. M. Roerdink. Euclidean skeletons of 3d data sets in linear time by the integer medial axis transform. In *Computational Imaging and Vision*, páginas 259–268. Springer-Verlag, 2005.
- Shubhalaxmi P. Hingway & Kishor M. Bhurchandi. A comparative study of shock graph methods for shape recognition. In *Emerging Trends in Engineering and Technology (ICETET), 2009 2nd International Conference on*, páginas 288–293, Dic 2009.
- Shubhalaxmi P. Hingway & Kishor M. Bhurchandi. A simple graph theoretic approach for object recognition. In *Emerging Trends in Engineering and Technology (ICETET), 2011 4th International Conference on*, páginas 200–205, Nov 2011.

- John A. Horst & Isabel Beichel. A simple algorithm for efficient piecewise linear approximation of space curves. In *Image Processing, 1997. Proceedings., International Conference on*, volumen 2, páginas 744–747, Oct 1997.
- Lei Huang, Genxun Wan, & Changping Liu. An improved parallel thinning algorithm. In *Document Analysis and Recognition, 2003. Proceedings. Seventh International Conference on*, páginas 780–783, Ago 2003.
- Bo Jiang, Jin Tang, Bin Luo, Zhazhan Chen, & Zinian Chen. Skeleton graph matching based on a novel shape tree. In *Computing, Communication, Control, and Management, 2009. CCCM 2009. ISECS International Colloquium on*, volumen 4, páginas 636–639, Ago 2009.
- Stelios Krinidis & Michail Krinidis. Empirical mode decomposition on skeletonization pruning. *Image and Vision Computing*, 31(8):533 – 541, 2013.
- Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955.
- Longin Jan Latecki & Rolf Lakämper. Shape similarity measure based on correspondence of visual parts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(10):1185–1190, Oct 2000.
- Longin Jan Latecki & Rolf Lakämper. Convexity rule for shape decomposition based on discrete contour evolution. *Computer Vision and Image Understanding*, 73:441–454, 1999.
- Longin Jan Latecki, Qiang Wang, Suzan Koknar-Tezel, & Vasileios Megalooikonomou. Optimal subsequence bijection. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, páginas 565–570, Oct 2007.
- Wing-Nin Leung, C. M. Ng, & P. C. Yu. Contour following parallel thinning for simple binary images. In *Systems, Man, and Cybernetics, 2000 IEEE International Conference on*, volumen 3, páginas 1650–1655, 2000.
- Hongzhi Liu, Zhonghai Wu, Frank Hsu, Bradley S. Peterson, & Dongrong Xu. On the generation and pruning of skeletons using generalized voronoi diagrams. *Pattern Recognition Letters*, 33(16):2113–2119, 2012.
- Hongzhi Liu, Zhong-Hai Wu, Xing Zhang, & D. Frank Hsu. A skeleton pruning algorithm based on information fusion. *Pattern Recogn. Lett.*, 34(10):1138–1145, Jul 2013.
- Diego Macrini, Sven Dickinson, David Fleet, & Kaleem Siddiqi. Bone graphs: Medial shape parsing and abstraction. *Computer Vision and Image Understanding*, 115(7): 1044–1061, 2011.

- Petros Maragos & Ronald W. Schafer. Morphological skeleton representation and coding of binary images. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 34(5):1228–1244, Oct 1986.
- M. Pilar Martínez-Pérez, Javier Jiménez, & José L. Navalón. A thinning algorithm based on contours. *Computer Vision, Graphics, and Image Processing*, 39(2):186 – 201, 1987.
- Niranjan R. Mayya & V. T. Rajan. Voronoi diagrams of polygons: A framework for shape representation. In *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on*, páginas 638–643, Jun 1994.
- Washington Mio, Anuj Srivastava, & Shantanu Joshi. On shape of plane elastic curves. *International Journal of Computer Vision*, 73(3):307–324, 2007.
- Andrés Solís Montero & Jochen Lang. Skeleton pruning by contour approximation and the integer medial axis transform. *Computers and Graphics*, 36(5):477–487, 2012.
- Thanh Phuong Nguyen & Isabelle Debled-Rennesson. A discrete geometry approach for dominant point detection. *Pattern Recognition*, 44(1):32 – 44, 2011.
- Robert L. Ogniewicz & O. Kübler. Hierarchic voronoi skeletons. *Pattern Recognition*, 28(3):343–359, 1995.
- Laura A. Pinilla-Buitrago, José F. Martínez-Trinidad, & Jesús A. Carrasco-Ochoa. New penalty scheme for optimal subsequence bijection. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, volumen 8258, páginas 206–213. 2013.
- Dilip K. Prasad, Chai Quek, Maylor K. H. Leung, & Siu-Yeung Cho. A parameter independent line fitting method. In *Pattern Recognition (ACPR), 2011 First Asian Conference on*, páginas 441–445, Nov 2011.
- Dilip K. Prasad, Maylor K.H. Leung, Chai Quek, & Siu-Yeung Cho. A novel framework for making dominant point detection methods non-parametric. *Image and Vision Computing*, 30(11):843–859, 2012.
- Lakshman Prasad. Rectification of the chordal axis transform skeleton and criteria for shape decomposition. *Image and Vision Computing*, 25(10):1557–1571, 2007.
- Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3):244–256, 1972.
- Di Ruberto Rodriguez, G. Rodriguez, & L. Casta. Recognition of shapes by morphological attributed relational graphs, 2002.

- Thomas B. Sebastian & Benjamin B. Kimia. Curves vs. skeletons in object recognition. *Signal Processing*, 85(2):247 – 263, 2005.
- Doron Shaked & Alfred M. Bruckstein. Pruning medial axes. *Computer Vision and Image Understanding*, 69(2):156 – 169, 1998.
- Lejun Shao & Hao Zhou. Curve fitting with bézier cubics. *Graphical Models and Image Processing*, 58(3):223–232, 1996.
- Wei Shen, Xiang Bai, Rong Hu, Hongyuan Wang, & Longin Jan Latecki. Skeleton growing and pruning with bending potential ratio. *Pattern Recogn.*, 44(2):196–209, Feb 2011.
- Wei Shen, Xiang Bai, XingWei Yang, & LonginJan Latecki. Skeleton pruning as trade-off between skeleton simplicity and reconstruction error. *Science China Information Sciences*, 56(4):1–14, 2013a.
- Wei Shen, Yan Wang, Xiang Bai, Hongyuan Wang, & Longin Jan Latecki. Shape clustering: Common structure discovery. *Pattern Recognition*, 46(2):539 – 550, 2013b.
- Zhijun Song, Jun Yu, Changle Zhou, Dapeng Tao, & Yi Xie. Skeleton correspondence construction and its applications in animation style reusing. *Neurocomputing*, 120:461–468, 2013.
- Patrick Sprote & Roland W. Fleming. Concavities, negative parts, and the perception that shapes are complete. *Journal of Vision*, 13(14):1–23, 2013.
- Hari Sundar, Deborah E. Silver, Nikhil P. Gagvani, & Sven J. Dickinson. Skeleton based shape matching and retrieval. In *Shape Modeling International, 2003*, páginas 130–139, May 2003.
- Gábor Székely. Voronoi skeletons. In *Medial Representations*, volumen 37 of *Computational Imaging and Vision*, páginas 191–221. 2008.
- Sibel Goktepe Tari, Jayant Shah, & Homer Pien. Extraction of shape skeletons from grayscale images. *Computer Vision and Image Understanding*, 66(2):133 – 146, 1997.
- Cho H. Teh & Roland T. Chin. On the detection of dominant points on digital curves. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 11(8):859–872, Ago 1989.
- Sen Wang, Jianhuang Wu, Mingqiang Wei, & Xin Ma. Robust curve skeleton extraction for vascular structures. *Graphical Models*, 74(4):109 – 120, 2012.

- Aaron D. Ward & Ghassan Hamarneh. The groupwise medial axis transform for fuzzy skeletonization and pruning. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(6):1084–1096, Jun 2010.
- Kan Wu, Li Chen, Jing Li, & Yanheng Zhou. Tooth segmentation on dental meshes using morphologic skeleton. *Computers and Graphics*, 38(0):199 – 211, 2014.
- Zhaohua Wu & Norden E. Huang. Ensemble empirical mode decomposition: A noise-assisted data analysis method. *Advances in Adaptive Data Analysis*, 1(1):1–41, 2009.
- Wenjie Xie, Robert P. Thompson, & Renato Perucchio. A topology-preserving parallel 3d thinning algorithm for extracting the curve skeleton. *Pattern Recognition*, 36(7):1529 – 1544, 2003.
- Yao Xu, Bo Wang, Wenyu Liu, & Xiang Bai. Skeleton graph matching based on critical points using path similarity. In *Computer Vision – ACCV 2009*, volumen 5996, páginas 456–465. 2010.
- Xiaojun Yang, Xiang Bai, Xingwei Yang, & Luan Zeng. An efficient quick algorithm for computing stable skeletons. In *Image and Signal Processing, 2009. CISP '09. 2nd International Congress on*, páginas 1–5, Oct 2009.
- T. Y. Zhang & C. Y. Suen. A fast parallel algorithm for thinning digital patterns. *Commun. ACM*, 27(3):236–239, Mar 1984.
- Ruwei Zhou, Chai Quek, & Geoksee S. Ng. A novel single-pass thinning algorithm and an effective set of performance criteria. *Pattern Recognition Letters*, 16(12):1267–1275, 1995.
- Ziheng Zhou & Anastasios N. Venetsanopoulos. Analysis and implementation of morphological skeleton transforms. *Circuits, Systems and Signal Processing*, 11(1):253–280, 1992.
- Xuefang Zhu & Shuyi Zhang. A shape-adaptive thinning method for binary images. In *Cyberworlds, 2008 International Conference on*, páginas 721–724, Sep 2008.