



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



**TECNOLÓGICO
NACIONAL DE MÉXICO**

TECNOLÓGICO NACIONAL DE MÉXICO

**CENTRO NACIONAL DE INVESTIGACIÓN
Y DESARROLLO TECNOLÓGICO**

Tesis de Maestría

**Sobresgmentación de Imágenes a Color
Mediante un Grafo Difuso Basado en una Medida
de Distancia Espacial y Color**

Presentada por:

Ing. Diego Carlos Luna Márquez

como requisito para la obtención del grado:
Maestro en Ciencias de la Computación

Director de Tesis:

Dr. Dante Mújica Vargas

Co-director de Tesis:

Dr. Raúl Pinto Elías

Cuernavaca, Morelos, México. Enero de 2021.



“2020, Año de Leona Vicario, Benemérita Madre de la Patria”

Cuernavaca, Morelos **03/diciembre/2020**

OFICIO No. SAC/ 281/2020

Asunto: Autorización de impresión de tesis

DIEGO CARLOS LUNA MÁRQUEZ
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS
DE LA COMPUTACIÓN
P R E S E N T E

Por este conducto tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado *“Sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color”*, ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

A T E N T A M E N T E

Excelencia en Educación Tecnológica®

“Conocimiento y tecnología al servicio de México”

DR. CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO

C.c.p. M.E. Guadalupe Garrido Rivera. Jefa del Departamento de Servicios Escolares

Expediente

CMAZ/CHG



**CENTRO NACIONAL
DE INVESTIGACIÓN
Y DESARROLLO
TECNOLÓGICO
SUBDIRECCIÓN
ACADÉMICA**

Dedicatoria

A mis padres amorosos *Luz María* y *Joel*[†],
quienes han creído siempre en mí.

Agradecimientos

Generalmente los éxitos son merecedores de gran satisfacción. Pero el hecho de haber logrado un paso significa, a veces, haber tenido algunos tropiezos anteriores. Por supuesto supongo que es afable al Universo ser humilde sabiendo que el hecho que no son logros totalmente personales: circunstancias, tiempos, anhelos y personas hacen parte de esta travesía que llamamos vida.

Así que desearía agradecer al tiempo, pero no le hago ningún mérito en ello: es demasiado fugaz y escurridizo para detenerse a leerme.

Desearía agradecer al destino, pero ¿no acaso el destino somos nosotros y sería demasiado ufano agradecerme a mí mismo?

Desearía entonces agradecer a la persistencia: pero todos hemos abandonado alguna vez cosas que creímos bellas y dejaron de estar, de ser o de aparentar.

Así que, sin que sean menos importantes y siendo al mismo tiempo destino y persistencia, me queda agradecer a aquellas bellas personas que han estado (y algunas quienes no ha podido estar) en estos sinuosos trechos. Agradezco a la Dra. Azucena Rendón quien creyó en mí y me incentivó, en contraparte a mis temores, para emprender esta fase de mi vida. Agradezco a quien me apoyó en momentos complicados aún sin conocerme: mil gracias Dr. Gerardo Guerrero. Al ingeniero Nestor Navarrete quien ha sido más que un jefe laboral, un amigo personal. A todo el personal del CENIDET, tanto administrativo como de profesorado y en especial al comité académico que me apoyó a hacer posible este documento de tesis, igualmente mi gratitud por el apoyo y la enseñanza.

No menos responsable de mi aprendizaje durante este trayecto, gracias especiales a quien ha sido mi director de tesis de maestría en el CENIDET, al Dr. Dante Mújica por la paciencia ofrecida en estos dos años y que se remonta a la entrevista misma para mi ingreso al Centro. Deseo que esta travesía no la haya catafixiado por algunas canas no indispensables.

Me siento afortunado de tener siempre a mi lado a cómplices en este camino, para no andarlo solo. A todos y cada uno de mis compañeros y amigos, en especial a Omar, Irving, Viridiana, Vitervo, Edgar, Sergio, Elizabeth³, Cristina, Nancy, Fátima, Alfredo, Francesca y Harri con quienes ha habido momentos de risas, aprendizaje y crecimiento personal. Los vientos favorables les sean por siempre.

Agradezco finalmente el apoyo económico otorgado por el CONACYT a través del CENIDET/TECNM reflejado en la beca recibida oportunamente durante mis estudios de maestría, al igual que el uso de sus instalaciones.

Resumen

En este trabajo de tesis se describe el algoritmo de “Sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color”.

En el tratamiento digital de imágenes, la segmentación es una fase importante de nivel medio que aporta resultados hacia niveles superiores de comprensión de la imagen. Una cantidad importante de técnicas para eficientar el proceso han sido desarrolladas y, entre ellas, los métodos basados en teoría de grafos han dado muy buenos resultados.

Estos métodos proponen una representación de la imagen donde cada pixel mantiene una relación de similaridad para con sus vecinos. Las aristas de un grafo se utilizan para simbolizar dicho vínculo y los vértices se corresponden con los pixeles. Este enfoque permite el empleo de procedimientos pertenecientes a la teoría de grafos para el análisis y procesamiento de las imágenes trasladadas.

La teoría difusa, aplicada al procesamiento de imágenes, permite el modelado y tratamiento de una imagen regular hacia un enfoque trasladado al plano difuso. Bajo este recurso, los componentes de imagen se describen en relaciones de pertenencia a uno o más conjuntos mediante funciones de pertenencia. Estas técnicas son de aplicación en diversas fases del procesamiento de imágenes para el manejo y/o reducción de incertidumbre o imprecisión.

La fusión de ambas teorías da origen al concepto de **grafos difusos**. Las relaciones de vecindad entre pixeles (las aristas) son entonces simbolizadas por funciones de pertenencia con respecto a grupo(s), al igual que los pixeles. El procedimiento permite modelar una imagen como relaciones no binarias de elementos. Se pretende utilizar este enfoque para aprovechar la cercanía entre pixeles y segmentar en valores de cortes difusos.

Palabras clave: Lógica Difusa, Teoría de Grafos, Sobresegmentación de imágenes.

Abstract

This thesis work describes the algorithm “Over-segmentation of color images using a fuzzy graph based on a measure of spatial distance and color”.

In digital image processing, segmentation is an important mid-level phase that brings results towards higher levels of image understanding. A significant number of techniques to streamline the process have been developed and, among them, methods based on graph theory have given very good results.

These methods propose a representation of the image where each pixel maintains a relationship of similarity with its neighbors. The edges of a graph are used to symbolize this link and the vertices correspond to the pixels. This approach allows the use of procedures belonging to the theory of graphs for the analysis and processing of translated images.

Fuzzy theory, applied to image processing, allows the modeling and treatment of a regular image towards an approach translated to the fuzzy space. Using this resource, image components are described in relationships of membership to one or more sets by membership functions and values. These techniques are applicable in various phases of image processing for the management and/or reduction of uncertainty or imprecision.

The fusion of both theories gives rise to the concept of **fuzzy graphs**. The neighborhood relationships between pixels (the edges) are then symbolized by membership functions with respect to group(s), just like the pixels. The procedure allows to model an image as non-binary relationships of elements. It is intended to use this approach to take advantage of the proximity between pixels and segment into fuzzy cutoff values.

Keywords: Fuzzy Logic, Graph Theory, Image oversegmentation.

Índice General

Resumen	VII
Índice de Tablas	XI
Índice de Figuras	XII
1. Introducción	1
1.1. Planteamiento del Problema	2
1.1.1. Hipótesis	2
1.1.2. Delimitación del Problema	2
1.1.3. Complejidad	2
1.2. Objetivos	2
1.2.1. Objetivo general	2
1.2.2. Objetivos específicos	3
1.3. Alcances y limitaciones	3
1.3.1. Alcances	3
1.3.2. Limitaciones	3
1.3.3. Hipótesis	3
1.4. Justificación	4
1.5. Organización del documento	4
2. Marco Teórico	5
2.1. Segmentación de imágenes	5
2.1.1. Clasificación de métodos de segmentación	6
2.2. Sobresegmentación de imágenes	7
2.3. Teoría de grafos	9
2.4. Teoría de Conjuntos Difusos	10
2.4.1. Conjuntos difusos	11
2.4.2. Incertidumbre en imágenes digitales	11
2.4.3. Grafos Difusos	13
2.4.4. Imagen digital como grafo difuso	14
2.5. Métricas de evaluación de sobresegmentación	16
2.5.1. Recuerdo de límites (<i>Bundary Recall</i> , Rec)	16
2.5.2. Error de subsegmentación (<i>Undersegmentation Error</i> , UE)	17
2.5.3. Variación explicada (<i>Explained Variation</i> , EV)	17
2.6. Algoritmos de sobresegmentación examinados	17
2.6.1. Contour Relaxed Superpixels (CRS)	18

2.6.2.	Eikonal Region Growing Clustering (ERGC)	18
2.6.3.	Entropy Rate Superpixels (ERS)	19
2.6.4.	Superpixels Extracted via Energy-Driven Sampling (SEEDS)	20
2.6.5.	Simple Linear Interactive Clustering (SLIC)	20
2.6.6.	Watershed (W)	21
2.6.7.	Intuitionistic Fuzzy C-Means (IFCM)	22
2.7.	Posprocesamiento y segmentación basada en grafos	25
2.7.1.	Grafo de Adyacencia de Regiones (RAG)	26
3.	Estado del Arte	30
3.1.	Antecedentes	30
3.2.	Trabajos relacionados	34
3.2.1.	Discusión del Estado del Arte	47
4.	Metodología	52
4.1.	Algoritmo de sobresegmentación propuesto	52
4.1.1.	Algoritmo de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color	52
5.	Experimentación y resultados	56
5.1.	Descripción general	56
5.2.	Conjuntos de datos	57
5.2.1.	Berkeley Segmentation Dataset (BSDS500)	57
5.2.2.	Weizmann Segmentation Evaluation Database	58
5.3.	Resultados	59
5.3.1.	Resultados del algoritmo propuesto	59
5.3.2.	Posprocesamiento mediante grafo de adyacencia de regiones (RAG)	70
6.	Conclusiones	73
6.1.	Productos	74
6.2.	Aportaciones	75
6.3.	Trabajos futuros	75
	Apéndice 1. Constancias y reconocimientos	76
	Bibliografía	86
	Índice alfabético	90

Índice de Tablas

3.1. Valores de pertenencia de las aristas del grafo difuso planar bipolar obtenida de una imagen (Pramanik <i>et al.</i> , 2018).	41
3.2. Resumen del Estado del Arte	48
5.1. Evaluación final para la métrica <i>Rec</i>	60
5.2. Evaluación final para la métrica $(1-UE)$	62
5.3. Evaluación final para la métrica <i>EV</i>	63
5.4. Resumen final para las métricas <i>Rec</i> , $(1-UE)$ y <i>EV</i>	64
6.1. Recomendaciones de utilización de algoritmos de sobresegmentación . .	74

Índice de Figuras

2.1. Fases del procesamiento de imágenes (Vargas <i>et al.</i> , 2011).	6
2.2. Métodos de segmentación (con información de (Basavaprasad y Hegadi, 2014)).	7
2.3. Usos de superpíxeles en visión artificial (con información de (Stutz <i>et al.</i> , 2018)).	8
2.4. Ejemplo de un multigrafo G	9
2.5. Conocimiento imperfecto en procesamiento de imágenes (Haußecker y Tizhoosh, 1999)	11
2.6. Sistema de inferencia difuso. (Poudel <i>et al.</i> , 2011)	12
2.7. Fuzzificación basada en histograma de niveles de gris (Haußecker y Tizhoosh, 1999)	12
2.8. Comparación de conectividad regular y difusa (Haußecker y Tizhoosh, 1999)	13
2.9. Imagen como un arreglo de conjuntos unitarios (<i>singletons</i>) difusos (Haußecker y Tizhoosh, 1999)	15
2.10. Creación de un grafo difuso $\xi = (V, \sigma, \mu)$ a partir de una imagen digital.	16
2.11. Proceso de segmentación del algoritmo SEEDS (Van den Bergh <i>et al.</i> , 2012).	20
2.12. Zonas de inundación asociadas con los varios tipos de inundación de una superficie topográfica (Meyer, 1992).	22
2.13. IFCM, reducción de espacio de búsqueda (Domínguez, 2019).	24
2.14. Grafo de Adyacencia de Regiones (Trémeau y Colantoni, 2000)	26
2.15. Utilización de un RAG en el procesamiento de imágenes	27
2.16. Simplificación de un RAG después de haber fusionado dos regiones adyacentes (Trémeau y Colantoni, 2000).	28
2.17. Segmentación regular como producto de superpíxeles	29
3.1. Vecindad entre píxeles (Ríos, 2004)	34
3.2. Resultados de la Segmentación del algoritmo NMFCM (Aljebory y Mohammed, 2018).	36
3.3. Resultados de la segmentación de TurboPixel y Cortes de Grafo de Kernel Multiregional. (Zhang <i>et al.</i> , 2018)	40
3.4. Ejemplo de conversión de una imagen a un grafo difuso planar bipolar (Pramanik <i>et al.</i> , 2018)	41
3.5. Resultado final de la reducción basada en grafos difusos polares (Pramanik <i>et al.</i> , 2018).	42

3.6.	Algoritmos de generación de superpíxeles y supervoxeles. (Tamajka y Benešová, 2018)	42
3.7.	Estructura de grafo difuso 4-polar. (Akram <i>et al.</i> , 2016)	45
3.8.	Curva Bezier para la Segmentación del Ventrículo Izquierdo. (Bernier <i>et al.</i> , 2017)	46
3.9.	Construcción del grafo por encima del espacio Bezier de la imagen MRI. (Bernier <i>et al.</i> , 2017)	47
4.1.	Esquema de funcionamiento del algoritmo de sobresegmentación propuesto.	54
5.1.	Ejemplo de imágenes en el Repositorio <i>BSDS500</i>	58
5.2.	Ejemplo de imágenes en el Repositorio Weizmann	59
5.3.	Evaluación final para la métrica Recuerdo del límite (<i>Rec</i>) en imágenes del subconjunto <i>test</i> , <i>BSDS500</i>	61
5.4.	Evaluación final para la métrica Error de subsegmentación (<i>1-UE</i>) en imágenes del subconjunto <i>test</i> , <i>BSDS500</i>	62
5.5.	Evaluación final para la métrica Variación explicada (<i>EV</i>) en imágenes del subconjunto <i>test</i> , <i>BSDS500</i>	63
5.6.	Resultados comparativos de métricas <i>Rec</i> , (<i>1-UE</i>) y <i>EV</i> para los algoritmos analizados	64
5.7.	Comparación cualitativa de los algoritmos evaluados con $k \approx 200$	65
5.8.	Comparación cualitativa de los algoritmos evaluados con $k \approx 400$	66
5.9.	Comparación cualitativa de los algoritmos evaluados con $k \approx 100$	68
5.10.	Comparación de utilización de color para el cálculo de la distancia en la sobresegmentación, $k \approx 400$	69
5.11.	Ejemplo de imagen procesada mediante RAG, repositorio <i>BSDS500</i>	71
5.12.	Ejemplo de imagen procesada mediante RAG, repositorio <i>Weizmann</i>	72
6.1.	Reconocimiento como ponente en el 1er Coloquio “Artificial Intelligence”. Iguala, Guerrero	77
6.2.	Constancia de participación con el artículo “2-D Superpixel Oversegmentation of Images Evaluated by Different Color Spaces”, Cuernavaca, Morelos	78
6.3.	Primer página del artículo “2-D Superpixel Oversegmentation of Images Evaluated by Different Color Spaces”, Cuernavaca, Morelos	79
6.4.	Constancia de participación con el artículo “The Theta* Algorithm for Path Calculation”, Cuernavaca, Morelos	80
6.5.	Primer página del artículo “The Theta* Algorithm for Path Calculation”, Cuernavaca, Morelos	81
6.6.	Reconocimiento como ponente en la Escuela de Inteligencia Artificial y Robótica 2019, Cuernavaca, Morelos	82
6.7.	Primer página del artículo “Segmentación de imágenes basada en teoría de grafos”, Emiliano Zapata, Morelos	83

6.8. Reconocimiento como ponente en la Semana Académica y Cultural de la Licenciatura en Computación 2019 en la UABJO. Oaxaca de Juárez, Oaxaca	84
6.9. Reconocimiento como instructor en la Semana Académica y Cultural de la Licenciatura en Computación 2019 en la UABJO. Oaxaca de Juárez, Oaxaca	85

1

Introducción

Las características de la percepción humana son altamente apropiadas para ciertas tareas aunque no tan eficientes para otras. Una de dichas tareas de alto nivel se refleja en el grado de especialización de la visión humana para tareas de reconocimiento de formas y detección de movimiento. En contraparte, factores de ruido, representación de información, limitaciones de sensores, etc., hacen del procesamiento de imágenes por computadora un problema computacionalmente demandante y complejo.

La segmentación es una tarea importante dentro del procesamiento de imágenes. El objetivo es el de separar una imagen en regiones sin intersección cuya unión la constituyan adecuadamente en completud. El proceso consiste en agrupar coherentemente píxeles considerando propiedades de similitud (o no similitud) de la imagen o entre los mismos. Regularmente se utilizan atributos como intensidad, textura y vecindad, entre otros.

Se han desarrollado múltiples enfoques para realizar la tarea de segmentación de imágenes. Un problema actual es que la gran mayoría de los mismos utilizan sólo características a nivel global, es decir, realizan cálculos para estimar las particiones atendiendo a las propiedades de todos y cada uno de los píxeles de la imagen. Este hecho genera pérdida de información y baja calidad en el aglomerado de píxeles. Una solución es considerar las relaciones entre píxeles en términos de vecindad, similaridad y bordes, se accede con ello a similitudes de vecindad y agrupamiento homogéneo.

Por otro lado, la técnica de agrupamiento basada en grafos difusos, tema de la tesis propuesta, permitirá aumentar la expresividad local de relaciones no binarias (difusas) entre píxeles, esto concede una mejora en la segmentación basándose en información de cercanía espacial. Ello se pretende lograr con la representación y procesamiento en grafos difusos, de los vínculos de similitud local por segmento.

En la presente propuesta de tesis se desarrollan conceptos e ideas de aplicación de grafos difusos a la representación y análisis de grafos difusos con orientación al problema de la sobresegmentación de imágenes.

1.1. Planteamiento del Problema

Los algoritmos de segmentación tradicionales consideran únicamente la información global para definir las regiones a segmentar. En este enfoque se pierde información valiosa de cercanía o similitud entre vecinos, misma que aportaría a crear las relaciones necesarias de agrupamiento en regiones. Muchos de estos algoritmos, por ejemplo los basados en umbrales, separarán píxeles basados en únicamente una propiedad, aún cuando se cumpla que los mismos sean vecinos y, al mismo tiempo, con una alta similaridad.

La utilización de información local, como se propone en el presente tema de tesis, permite añadir criterios de similaridad espacial y/o vecinal. Con esto se pretende realizar una aglomeración de mayor uniformidad entre píxeles cercanos. El enfoque de grafos difusos agrega similaridad y pertenencia difusa a los píxeles con respecto a los segmentos.

1.1.1. Hipótesis

La utilización de grafos difusos en la segmentación de imágenes permitirá una mejora en la calidad del agrupamiento de píxeles en una imagen, derivado del uso de información local como relaciones difusas entre píxeles vecinos.

1.1.2. Delimitación del Problema

En el presente trabajo de investigación se implementó un algoritmo de aplicación basado en teoría de grafos difusa al problema de la sobresegmentación de imágenes. El propósito es eficientar la segmentación de regiones. Se desarrollará una implementación del mismo en lenguaje Python/C.

1.1.3. Complejidad

- No existe mucha documentación relacionada a grafos difusos.
- La segmentación por minimización de energía de un grafo es considerado un problema de alta complejidad.
- La técnica de cortes de grafos es considerada un problema de alta complejidad.

1.2. Objetivos

1.2.1. Objetivo general

- Formular en el dominio difuso una técnica de sobresegmentación de imágenes basada en teoría de grafos.

1.2.2. Objetivos específicos

- Estudiar y comprender los conceptos de Teoría de Grafos.
- Estudiar y comprender los conceptos de la Lógica Difusa enfocada en el planteamiento de la Teoría de Grafos.
- Entender el concepto de la segmentación de imágenes en escala de grises y a color.
- Replicar en el dominio difuso algún algoritmo de teoría de grafos aplicado a la segmentación de imágenes.
- Hacer la programación de la propuesta bajo el paradigma de cómputo paralelo o en el mejor de los casos cómputo heterogéneo.
- Evaluar el algoritmo desarrollado.

1.3. Alcances y limitaciones

1.3.1. Alcances

- Se utilizó la intensidad de la imagen como parámetro para la segmentación.
- Se pretende la implementación en un paradigma de cómputo heterogéneo en el mejor de los casos; inicialmente se realizó una implementación secuencial.

1.3.2. Limitaciones

- Se consideran solamente imágenes en escala de grises.
- Las imágenes no deben tener ruido añadido.
- Dado que la solución óptima global por minimización discreta de energía es considerado un problema NP-completo, no se pretende generar una solución que reduzca la complejidad computacional.

1.3.3. Hipótesis

La utilización de grafos difusos en la segmentación de imágenes permitirá una mejora en la calidad del agrupamiento de píxeles en una imagen, derivado del uso de información local como relaciones difusas entre píxeles vecinos.

1.4. Justificación

Diversos algoritmos orientados a resolver el problema de segmentación en imágenes han sido desarrollados. Muchos de estos algoritmos desestiman información valiosa de relación entre pixeles vecinos, esto reduce su eficiencia en la delimitación de regiones en la imagen. La necesidad de utilizar un grafo difuso obedece a la capacidad de esta estructura en ofrecer una manera precisa de representar elementos y relaciones. Esta representación es, asimismo, de gran conveniencia ya que la teoría de grafos es aplicable para su manejo y proceso.

1.5. Organización del documento

El presente documento de tesis consta de 6 capítulos organizados en acuerdo a la siguiente estructura:

El Capítulo 2 introduce el Marco Teórico que da soporte a la tesis. Definiciones relacionadas al procesamiento digital de imágenes, teoría matemática sobre grafos y teoría de conjuntos difusos son presentadas y, en su caso, desarrolladas como base conceptual.

En el Capítulo 3, el Estado del Arte presenta algunos trabajos utilizados como antecedentes y relacionados al tema en atención. Se describen tanto trabajos realizados en el CENIDET como aquellos desarrollados por investigadores a nivel mundial presentados en congresos y publicados en revistas en forma reciente.

La Metodología que describe el algoritmo propuesto en el presente documento, se encuentra en el Capítulo 4. En éste se presentan y detallan las fases que componen el algoritmo, tanto en forma gráfica así como procedimental mediante un listado de algoritmo.

Descripciones del trabajo de experimentación y resultados son exhibidas en el Capítulo 5. Los dos conjuntos de imágenes (*BSDS 500* y *Weizmann Segmentation Evaluation Database*) utilizados son igualmente presentados con algunas imágenes como ejemplo.

Finalmente, las Conclusiones son desarrolladas en el Capítulo 6. Tanto el aporte realizado así como productos y trabajos posibles futuros son enumerados y comentados para su atención y/o aprovechamiento posterior.

2

Marco Teórico

El presente capítulo tiene por finalidad introducir conceptos y métodos base que confluyen en el tratamiento de imágenes, en particular hacia la aplicación de segmentación con un enfoque de grafos difusos.

2.1. Segmentación de imágenes

Una imagen está normalmente constituida por varios objetos presentes e identificables con relativa facilidad por la percepción humana, no así para los sistemas de cómputo. El objetivo de la segmentación de imágenes (basada en computadora) es el de generar aglomeraciones de píxeles de una imagen tales que constituyan partes de objetos representados y diferenciables (Toennies, 2017). Esta aglomeración se realiza en un espacio de características (utilizando umbrales de intensidad o posición, por ejemplo), aunque en muchas ocasiones este principio no es suficiente para objetos de interés no homogéneos ya que estos presentan ruido, texturas diferentes y desorden, por citar algunas complicaciones.

Formalmente, la segmentación es un problema en el cual, dado R que representa la imagen completa, ésta se divide en N conjuntos no vacíos R_1, R_2, \dots, R_N , tales que se cumplen las siguientes condiciones (Gong *et al.*, 2019):

1. $\bigcup_{i=1}^N R_i = R$;
2. Para $i = 1, 2, \dots, N, P(R_i) = TRUE$;
3. Para $\forall_i, j, i \neq j, R_i \cap R_j = \emptyset$;
4. Para $\forall_i, j, i \neq j, P(R_i \cup R_j) = FALSE$;
5. Para $i = 1, 2, \dots, N, R_i$ es la región conectada.

Las anteriores condiciones establecen que en una segmentación existen N conjuntos como disjuntos, complementarios y finitos. En el tipo de segmentación denominada binaria, $N = 2$.

La segmentación ocurre como una fase dentro de niveles de procesamiento de imágenes. Una posible estructura en fases está representada en la Figura 2.1. Se reconocen 5 diferentes tareas: preprocesamiento, reducción de datos, segmentación, reconocimiento de objetos y comprensión.

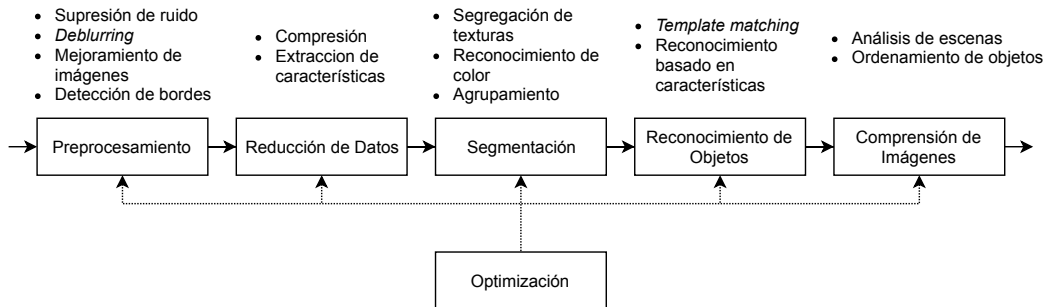


Figura 2.1: Fases del procesamiento de imágenes (Vargas *et al.*, 2011).

2.1.1. Clasificación de métodos de segmentación

Existen múltiples clasificaciones de los métodos de segmentación en la literatura. Desde el punto de vista de los autores cada uno cumple el objetivo de categorizar correctamente dichas técnicas. El problema es la presencia de alta heterogeneidad tanto en el ordenamiento jerárquico como en las consideraciones que cada uno de ellos hace. No obstante la mayoría se basa en dos propiedades de valores de intensidad (Gonzalez y Woods, 2002):

- **Discontinuidades.** Particionando una imagen basado en cambios abruptos en intensidad, tales como bordes.
- **Similaridad.** Seccionando una imagen en regiones que poseen semejanza de acuerdo a un conjunto de criterios definidos.

Métodos como el umbralizado, el crecimiento de regiones, y la división de regiones, utilizan la similaridad para la segmentación.

Una categoría general de segmentación se realiza de acuerdo al enfoque en que los elementos grupales son considerados:

- **Segmentación basada en pixel.** En que se considera únicamente el nivel de gris de cada pixel para su clasificación.
- **Segmentación basada en regiones.** Toma en cuenta los niveles de gris de los pixeles vecinos.

- Segmentación basada en bordes. Orientada a la detección de cambios bruscos de intensidad.

Una clasificación más específica que incluye métodos mixtos se observa en el diagrama de la Figura 2.2. El mismo fue realizado con información de (Basavaprasad y Hegadi, 2014).

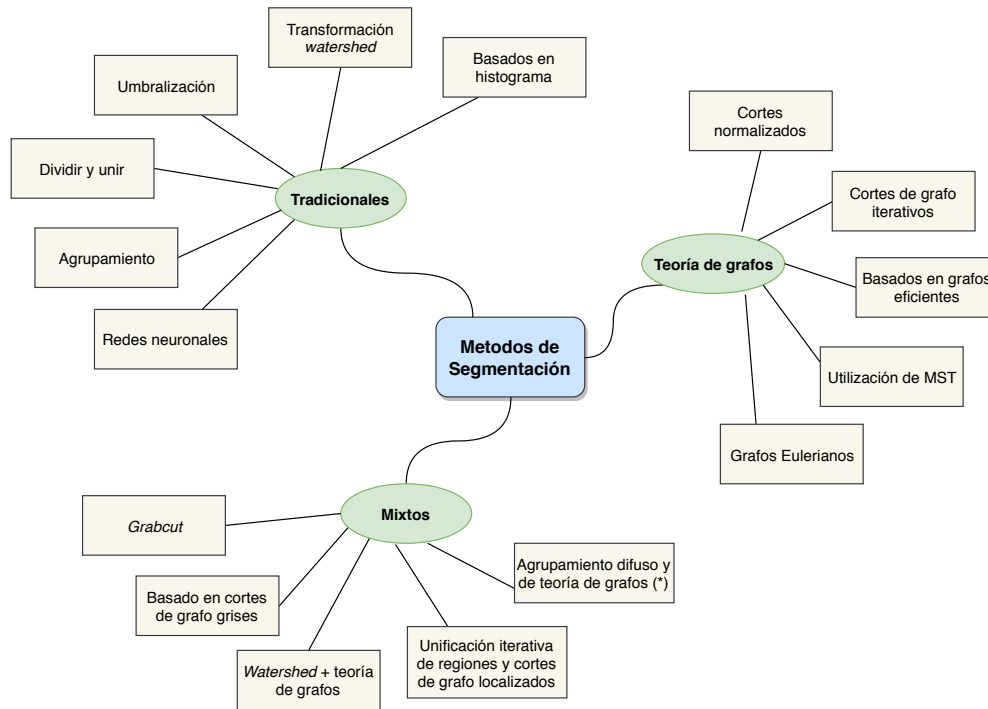


Figura 2.2: Métodos de segmentación (con información de (Basavaprasad y Hegadi, 2014)).

2.2. Sobresegmentación de imágenes

La sobresegmentación es un caso especial de la segmentación de imágenes en la que el número de regiones creadas es de una mayor cantidad que el número de segmentos reales (Tamajka y Benešová, 2018) en que los N conjuntos R_1, R_2, \dots, R_N pueden ser considerados dentro de los principios de agrupamiento desarrolladas en la teoría Gestalt: proximidad, similitud y continuidad. Popularizados por (Ren y Malik, 2003) los superpíxeles utilizados en la sobresegmentación agrupan píxeles similares en color y otras propiedades de bajo nivel.

A saber, la segmentación de imágenes en superpíxeles es una sobresegmentación en muchas subregiones, todas ellas como un apropiado subconjunto de exactamente una región “semántica” (con significado). La segmentación en superpíxeles (o la sobresegmentación) como precursor a tareas de nivel mayor tales como segmentación por instancias, estimación de movimiento y traza, etcétera, donde los cálculos son

llevados en forma eficiente en algunos cuantos cientos de grupos en lugar de varios miles de ellos (Conrad *et al.*, 2013) (Vargas *et al.*, 2011).

Por otro lado, la sobresegmentación posibilita el rescate de información que pudiera ser abandonada en técnicas que generan segmentos de tamaño mayor. Las regiones pequeñas son posteriormente agrupadas, dada alguna métrica de similaridad, formando superpíxeles; se utilizaron estas construcciones en el tema de tesis descrito.

Dada la gran similitud que poseen los algoritmos de superpíxeles y los algoritmos de sobresegmentación, ambos conceptos son utilizados indistintamente por la mayoría de los autores. Aún así se puede observar que los algoritmos de superpíxeles ofrecen control sobre la cantidad de superpíxeles generados, mientras que los algoritmos de segmentación ejecutando en “modo de sobresegmentación” no presentan dicho control (Stutz *et al.*, 2018).

Debido a la conveniencia de procesamiento posterior, los métodos de generación de superpíxeles han ido incrementando en cantidad y uso hacia diversas áreas de la visión artificial. La Figura 2.3 ejemplifica algunas áreas en visión artificial en que los algoritmos de generación de superpíxeles han incursionado.



Figura 2.3: Usos de superpíxeles en visión artificial (con información de (Stutz *et al.*, 2018)).

Para ser considerados como superpíxeles, deben cumplir las siguientes características que han sido aceptadas por la mayoría de los autores (Stutz *et al.*, 2018):

- **Partición.** Los superpíxeles deben definir una partición en la imagen. Esto es, los superpíxeles deben ser disjuntos y asignar una etiqueta a cada píxel.
- **Conectividad.** Se espera que los superpíxeles representen conjuntos conectados de píxeles.
- **Adherencia los límites.** Los superpíxeles deben preservar los límites en la imagen. La definición apropiada de “límite” puede depender del uso respectivo.

- Compacidad, regularidad y suavidad. En la ausencia de límites en la imagen, los superpíxeles deben ser compactos, ubicados en forma regular y mostrar límites suaves.
- Eficiencia. Los superpíxeles deben ser generados en forma eficiente.
- Cantidad controlable del número de superpíxeles. El número de superpíxeles a generar debe ser controlable.

2.3. Teoría de grafos

Una gran cantidad de descripciones de objetos y las relaciones entre los mismos es fácilmente descrita y modelada mediante un objeto matemático versátil conocido como *grafo*. Es posible definir intuitivamente un grafo como una representación de objetos y relaciones que mantienen dichos objetos. Es posible entonces formalizar esta concepto mediante la siguiente definición:

Definición 1. Se denomina *grafo* G , a la pareja ordenada

$$G = (V, E) \quad (2.1)$$

en la que V es un conjunto no vacío de *vértices* (o *nodos*) y E un conjunto de *aristas*. E a su vez, consta de pares no ordenados de vértices $(u, v) \in E, u \neq v$.

A estos dos nodos u y v se les conoce como adyacentes en G , si ambos son puntos extremos de una arista e de G . En forma complementaria, una arista e es incidente con dos nodos u y v cuando los últimos están conectados por la primera. En la Figura 2.4 se muestra un multigrafo donde los nodos 1 y 2 son adyacentes por medio de la arista b y la arista h (en verde) es un lazo del vértice 2.

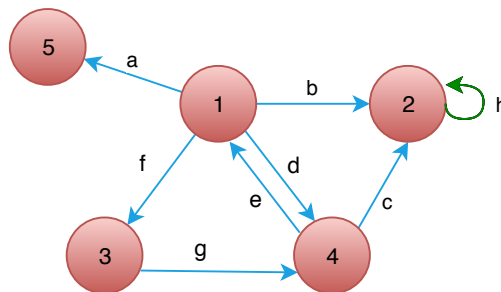


Figura 2.4: Ejemplo de un multigrafo G . Los vértices (en rojo y numerados) se relacionan mediante las aristas (líneas en azul).

Un grafo se denomina simple si no tiene lazos (definición 1) o *loops* (aristas conectando un nodo consigo mismo) o aristas múltiples (conectando únicamente dos

nodos). A estos últimos se les conoce como multigrafos. Un grafo plano es aquel en que sus aristas se intersecan solo en los vértices (no tiene cruces).

Se define el grado (o valencia) de un nodo u_1 como el número de aristas que inciden en él. En el caso de multigrafos, un lazo contribuye en dos veces al grado del nodo. Un vértice no adyacente a ningún otro vértice (de grado cero) es llamado aislado y aquel con un grado igual a uno se le conoce como nodo terminal o vértice hoja.

El problema de la equivalencia (isomorfismo) de subgrafos es de interés particular debido a que muchas propiedades de un grafo son heredadas de subgrafos. Técnicas de segmentación basadas en grafos generan subgrafos para cada región.

Definición 2. Dado un grafo $G = (V, E)$, se forma un *subgrafo* $H = (V', E')$ de G como un subconjunto de vértices y aristas de G , esto es, se cumple que, $V(H) \subseteq V(G)$ y $E(H) \subseteq E(G)$. Un *subgrafo abarcador* (también llamado *recubridor* o *expandido*) es aquél que incluye todos los vértices del grafo original, $V = V'$.

2.4. Teoría de Conjuntos Difusos

En general, la percepción humana del mundo a través de los sentidos se realiza en procesos detectados como continuos. Por otro lado, podemos notar que la discretización de procesos, colores, formas, etc., obedece a un proceso de categorización o delimitación conveniente para su manejo (y acaso comprensión), más que a un proceso aparentemente natural de pensamiento. Ambas circunstancias derivan en la aparición de alternativas de representación y análisis de conocimiento que, mismas que proponen métodos y técnicas de procesamiento continuo intentando modelar formas de razonamiento consideradas más humanas. La teoría difusa es una de las vertientes matemáticas que aportan en este sentido.

Para comprender las ventajas y diferencias de la teoría difusa con respecto al enfoque tradicional de pertenencia, se observa intuitivamente que un conjunto [clásico] es una colección no ordenada de objetos, denominados elementos o miembros del conjunto. Por intuición se expresa que un conjunto contiene a sus elementos. Matemáticamente esto se especifica de acuerdo a:

Definición 3. (Zimmermann, 2001) Un *conjunto clásico* (*crisp set*) se define como una colección A de elementos u objetos $x \in X$ que pueden ser finitos contables.

Cada elemento de dicho conjunto pertenece o no a un subconjunto A , $A \subseteq X$. En el primer caso se dice que la afirmación “ x pertenece a A ” es *verdadera*, en caso contrario es *falsa*.

$$A = \{x_1, x_2, \dots, x_n\} \quad (2.2)$$

Para los conjuntos clásicos la función de pertenencia se puede expresar como:

$$f_A : X \rightarrow \{0, 1\}, \quad f_A = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases} \quad (2.3)$$

2.4.1. Conjuntos difusos

Los conjuntos difusos son una generalización de los conjuntos clásicos. La propuesta formal relacionada a conjuntos difusos se debe a Lofti Zadeh, que define un conjunto difuso como una clase de objetos con un continuo de grados de pertenencia (a diferencia de los conjuntos clásicos).

Definición 4. Si X es una colección de objetos denominados genéricamente por x , entonces un *conjunto difuso* (en inglés *fuzzy set*) \bar{A} en X es un conjunto de pares ordenados tales que:

$$\bar{A} = \{(x, \mu_{\bar{A}}(x)) | x \in X\} \tag{2.4}$$

donde $\mu_{\bar{A}}(x)$ es llamada la *función de pertenencia* o *grado de pertenencia* (también llamada *grado de compatibilidad* o *grado de certeza*) de x en \bar{A} que mapea X al espacio de pertenencia M .

Definición 5. Un *conjunto difuso* \bar{A} en un conjunto X se caracteriza por una transformación $m : X \rightarrow [0, 1]$ llamado *función de pertenencia*.

Una *relación difusa* en X es un conjunto difuso en el espacio del producto $X \times X$. Es decir, $\mu : X \times X \rightarrow [0, 1]$ tal que $0 \leq \mu(x, y) \leq 1$ para todo $(x, y) \in X \times X$. Cuando μ toma únicamente los valores de 0 o 1, entonces es la función característica de una relación en X .

2.4.2. Incertidumbre en imágenes digitales

Los métodos de análisis de la teoría difusa encuentran una aplicación natural en el tratamiento de la incertidumbre e imprecisión intrínseca de una imagen digital. Elementos como ruido (originado por artefactos, por ejemplo) y bordes difusos, son el común de una imagen digital (sobre todo las llamadas “naturales”). Estas características se pueden representar y procesar desde el punto de vista de la teoría difusa en diferentes fases del procesamiento digital de imágenes.

En la Figura 2.5 se observan algunas fases del procesamiento digital de imágenes y el tratamiento respectivo de la incertidumbre y del conocimiento imperfecto de una imagen.

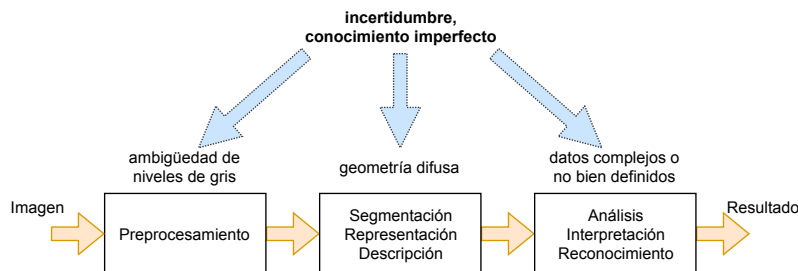


Figura 2.5: Conocimiento imperfecto en procesamiento de imágenes (Haußecker y Tizhoosh, 1999)

Para llevar a cabo tareas de procesamiento de dicha incertidumbre se requiere de un esquema como el mostrado en la Figura 2.6. Se puede observar que, debido a limitantes en la representación final de los resultados (o porque el proceso así lo amerita), la teoría difusa requiere del ingreso de información derivada de conjuntos clásicos y preprocesada por un fuzzificador. Un tratamiento difuso es llevado mediante un sistema de inferencia soportado por una base de conocimiento difusa. En la salida un defuzzificador retorna los valores de pertenencia difusa a correspondientes en conjuntos clásicos para su representación final.

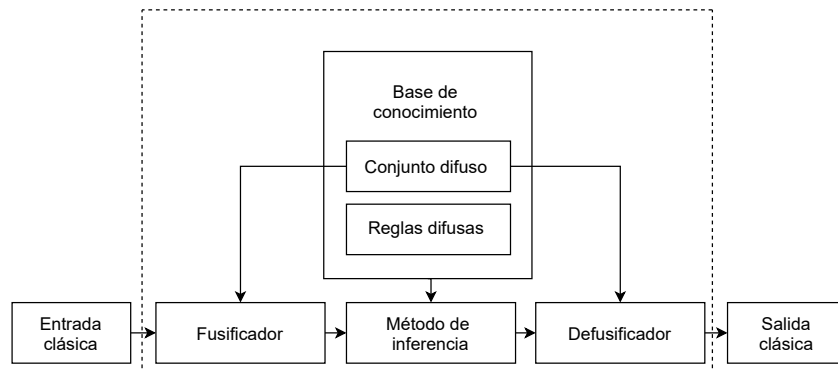


Figura 2.6: Sistema de inferencia difuso (Poudel *et al.*, 2011).

Diferentes técnicas de enfoque difuso han sido aplicadas a problemas de procesamiento de imágenes, algunos ejemplo se dan en (Chaira, 2015). En particular, en el caso de la segmentación, la decisión de pertenencia a un segmento se realiza representando las transiciones entre regiones mediante técnicas difusas. La Figura 2.7 ejemplifica la asignación de píxeles del histograma de la imagen a funciones de pertenencia difusas.

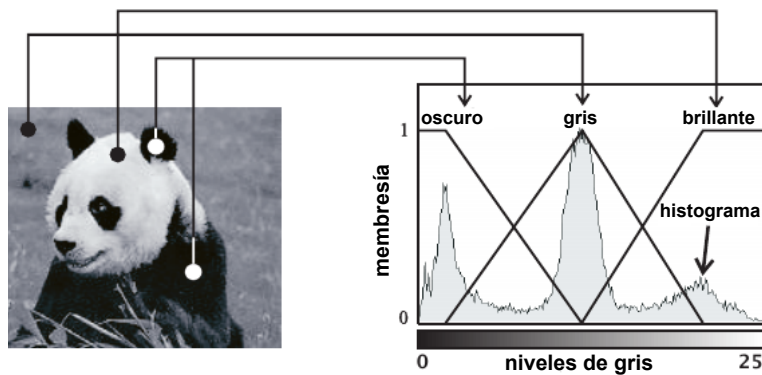


Figura 2.7: Fuzzificación basada en histograma de niveles de gris. La ubicación de funciones de pertenencia está determinado por los puntos específicos del histograma de la imagen (Haußecker y Tizhoosh, 1999).

Se observa que la diferencia con respecto al enfoque tradicional es la posible pertenencia de un pixel a más de un conjunto con diferentes (o iguales) valores de pertenencia. En la Figura 2.8 se muestra otra consideración de representación natural, en conjuntos difusos, de la imagen original y una segmentación básica de umbral para obtener la imagen binaria de la derecha.

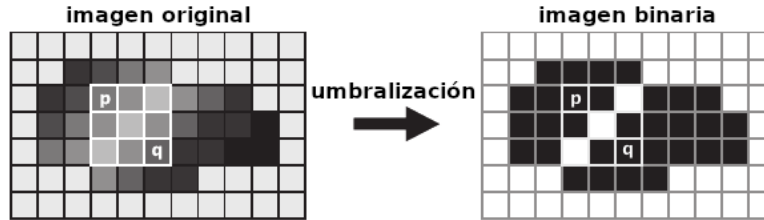


Figura 2.8: Comparación de conectividad regular y difusa. Los píxeles p y q tienen una conexión difusa en la imagen original y no están conectados en la imagen binaria (Haußecker y Tizhoosh, 1999).

2.4.3. Grafos Difusos

Bajo el enfoque difuso, las relaciones entre píxeles son consideradas relaciones difusas y pueden ser representadas por grafos. Esta factibilidad permite un análisis de las relaciones de similitud entre píxeles desde ambas teorías unificadas: de grafos y difusa.

Dada la diversidad, es posible categorizar el concepto de difuso como aplicable en la teoría de grafos en distintas maneras:

- *Tipo I.* Vértices como un conjunto de tipo clásico y aristas como un conjunto difuso.
- *Tipo II.* Vértices de tipo clásico y aristas con conectividad difusa.
- *Tipo III.* Grafo tipo clásico con pesos difusos.
- *Tipo IV.* Conjunto difuso de grafos tipo clásico.

Algunas definiciones relacionadas al uso de la teoría difusa en el tratamiento y representación de grafos son definidas a continuación.

Definición 6. (Mathew *et al.*, 2018) Un *grafo difuso* es una estructura algebraica definida:

$$\xi = (V, \sigma, \mu) \quad (2.5)$$

consistente de un conjunto no vacío V y un par de funciones $\sigma : V \rightarrow [0, 1]$ y $\mu : V \times V \rightarrow [0, 1]$ tal que $\forall x, y \in V, \mu(x, y) \leq \sigma(x) \wedge \sigma(y)$.

El conjunto difuso σ es llamado el conjunto difuso de vértices y μ el conjunto difuso de aristas. Se observa que μ es una relación difusa en σ .

Definición 7. (Mathew *et al.*, 2018) Sea $\xi = (V, \sigma, \mu)$ un grafo difuso. Entonces un grafo difuso $H = (V, \tau, \nu)$ es llamado un *grafo parcial difuso* de ξ si $\tau \subseteq \sigma$ y $\nu \subseteq \mu$. En forma similar el grafo difuso $H = (P, \tau, \nu)$ es denominado un *subgrafo difuso* de ξ inducido por P si $P \subseteq V$, $\tau(x) = \sigma(x)$ para todo $x \in P$ y $\nu(xy) = \sigma(xy)$ para todo $x, y \in P$.

Conectividad difusa

Al igual que en grafos regulares, el concepto de conectividad es aplicable a grafos difusos. En general, se describe la conectividad entre grupos que tienen un componente espacial de modo que la vecindad entre elementos del grupo puede ser definida.

La conectividad difusa entre dos elementos de la escena está dada por *afinidad* local de valores a de una ruta entre dos elementos. La afinidad de la ruta está, a su vez, dada por la afinidad local más pequeña entre elementos de la vecindad a lo largo de dicha ruta. De esto, la conectividad μ entre dos elementos \mathbf{p}_j y \mathbf{p}_k es la afinidad mínima de la ruta más fuerte entre los elementos, como lo describe la ecuación 2.6:

$$\mu(p_j, p_k) = \max_{p(p_j, p_k) \in P(p_j, p_k)} [\min_{(p_{s1}, p_{s2}) \in (p_j, p_k)}] a(p_{s1}, p_{s2}) \quad (2.6)$$

donde $P()$ es el conjunto de todas las rutas entre dos pixeles, $p()$ es la ruta y $(\mathbf{p}_{s1}, \mathbf{p}_{s2})$ son pixeles adyacentes bajo alguna relación de adyacencia.

El valor a de afinidad entre dos pixeles es una propiedad local, la cual depende de la distancia entre dos pixeles, en la homogeneidad de los valores característicos de los dos pixeles y en la desviación de los valores característicos de valores esperados.

2.4.4. Imagen digital como grafo difuso

La incertidumbre intrínseca encontrada en una imagen puede ser utilizada para su análisis y procesamiento. La traslación de características regulares hacia el dominio difuso se lleva a cabo considerando una imagen I de tamaño $M \times N$ con L niveles de gris como un arreglo de conjuntos unitarios (*singletons*) difusos ¹ que indica los valores de pertenencia μ_{mn} de cada punto de la imagen x_{mn} correspondiente a una propiedad predefinida de la imagen (intensidad, homogeneidad, ruido, etc) (Haußecker y Tizhoosh, 1999):

$$I = \bigcup_{m=1}^M \bigcup_{n=1}^N \frac{\mu_{mn}}{x_{mn}} \quad (2.7)$$

donde μ_{mn}/x_{mn} no se debe interpretar como una división numérica sino como un valor de pertenencia.

¹Conjuntos difusos con un único punto soporte

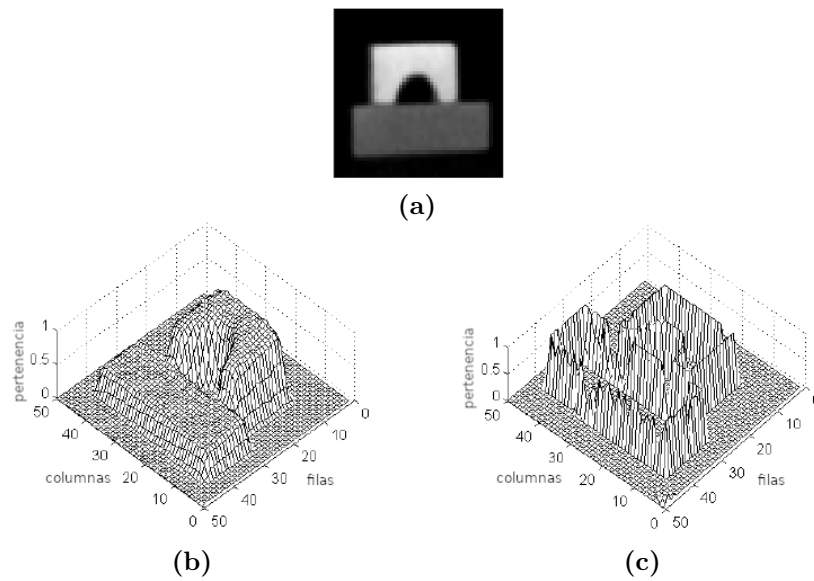


Figura 2.9: Imagen como un arreglo de conjuntos unitarios difusos. (a) Una imagen como conjunto difuso; (a) conjuntos unitarios difusos como valores de intensidad; (a) conjuntos unitarios difusos como valores de cercanía a bordes (Haußecker y Tizhoosh, 1999)

La definición anterior es un ejemplo de conversión de imagen hacia el dominio difuso pero carece del modelado eficiente de relaciones entre los conjuntos unitarios. El presente trabajo propone utilizar una estructura derivada de las matemáticas discretas denominado **grafo** (objeto de estudio de la teoría de grafos y expresado por la Definición 1) que se ha extendido hacia el ámbito difuso incorporando grados de pertenencia, en forma de aristas, entre los pixeles (nodos). Las componentes así configurados, permiten el tratamiendo mediante técnicas y métodos de la teoría de grafos y, en añadidura, de la teoría difusa.

En la Figura 2.10 se muestra resumidamente la construcción de un grafo difuso. El valor de intensidad normalizado de cada pixel se asigna al conjunto de vértices σ y una función μ determina las relaciones difusas entre pixeles como aristas.

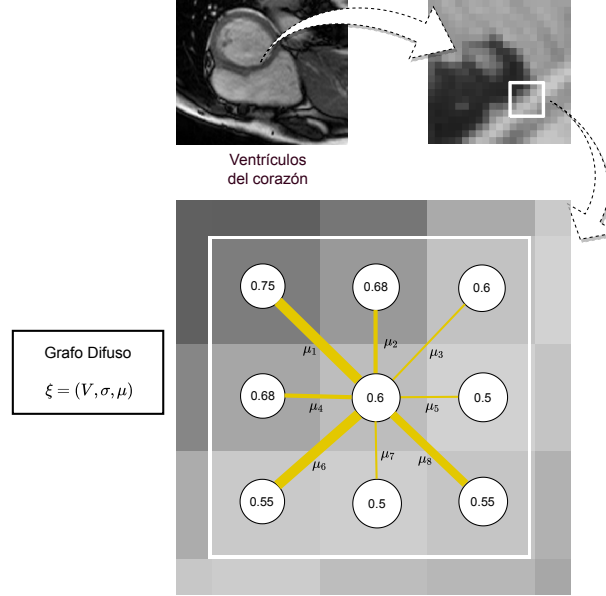


Figura 2.10: Creación de un grafo difuso $\xi = (V, \sigma, \mu)$ a partir de una imagen digital.

2.5. Métricas de evaluación de sobresegmentación

En toda experimentación satisfactoria es requerido evaluar cuantitativamente los resultados. En el caso particular de tareas de sobresegmentación las métricas empleadas pueden describir la adherencia a bordes con respecto al *ground truth* o la compacidad de los superpíxeles generados. El primer enfoque es elegido en este documento debido a que varios de los algoritmos generan píxeles no necesariamente compactos.

Sea $S = \{S_j\}_{j=1}^K$ y $G = \{G_i\}$ particiones de la misma imagen $I : x_n \mapsto I(x_n)$, $1 \leq n \leq N$, donde S representa un superpíxel en la sobresegmentación y G una segmentación parte del *ground truth*, es posible definir métricas que satisfagan los requerimientos que definen a un superpíxel (Stutz *et al.*, 2018):

2.5.1. Recuerdo de límites (*Boundary Recall, Rec*)

El Recuerdo de límites (*Boundary Recall*, (Martin *et al.*, 2004)) es la métrica más utilizada para evaluar la adherencia los bordes con respecto al *ground truth*. Sea $\text{FN}(G, S)$ y $\text{TP}(G, S)$ el número de píxeles como falsos negativos y verdaderos positivos respectivamente en S relativo a G , se define *Rec* como (Stutz *et al.*, 2018):

$$\text{Rec}(G, S) = \frac{\text{TP}(G, S)}{\text{TP}(G, S) + \text{FN}(G, S)} \quad (2.8)$$

Así que, un valor alto de *Rec* significa una mejor adherencia a bordes con respecto a los bordes en el *ground truth*, es decir, un valor alto es mejor.

2.5.2. Error de subsegmentación (*Undersegmentation Error*, **UE**)

El Error de subsegmentación (*Undersegmentation Error*, (Neubert and Protzel, 2012)) mide la “fuga” de superpíxeles con respecto a G y, de la misma manera, también mide la adherencia a bordes. “Fuga” se refiere, en este caso, al traslape de superpíxeles con respecto a múltiples segmentos cercanos del *ground truth* (Stutz *et al.*, 2018):

$$UE = \frac{1}{N} \sum_{G_i} \sum_{S_j \cap G_i \neq \emptyset} \min\{|S_j \cap G_i|, |S_j - G_i|\} \quad (2.9)$$

donde el término “interno” representa la “fuga” del superpíxel S_j con respecto a G

Un valor positivo pequeño (cerca de cero) representa una mejor segmentación pero, es importante mencionar, en este documento se utilizó la diferencia con respecto a la unidad (esto es $1 - UE$), con la finalidad de observar un comportamiento homogéneo (en aumento) con respecto al resto de las gráficas de las restantes métricas consideradas (*Rec* y *EV*).

2.5.3. Variación explicada (*Explained Variation*, **EV**)

La Variación explicada (*Explained Variation*, (Moore *et al.*, 2008)) cuantifica la calidad de una segmentación en superpíxeles sin depender del *ground truth*. Es decir, EV evalúa la adherencia a los bordes sin depender de las anotaciones humanas midiendo qué tan bien los datos en los píxeles originales están representados por los superpíxeles. Está definido como (Stutz *et al.*, 2018):

$$EV(S) = \frac{\sum_{S_j} |S_j| (\mu(S_j) - \mu(I))^2}{\sum_{x_n} (I(x_n) - \mu(I))^2} \quad (2.10)$$

donde $\mu(S_j)$ y $\mu(I)$ se corresponden con la media del color del superpíxel S_j y de la imagen I , respectivamente. Un valor mayor es preferible.

2.6. Algoritmos de sobresegmentación examinados

Como parte de la experimentación se eligieron 7 algoritmos de sobresegmentación o generadores de superpíxeles como elementos de comparación con respecto al algoritmo propuesto: *Contour Relaxed Superpixels* (**CRS**, (Conrad *et al.*, 2013)), *Eikonal Region Growing Clustering* (**ERGC**, (Buyssens *et al.*, 2014)), *Entropy Rate Superpixels* (**ERS**, (Liu *et al.*, 2011)), *Superpixels Extracted via Energy Driven Sampling* (**SEEDS**,

(Van den Bergh *et al.*, 2012)), *Simple Linear Interactive Clustering* (**SLIC**, (Achanta *et al.*, 2010)), *Watershed* (**W**, (Meyer, 1992)) e *Intuitionistic Fuzzy C-Means* (**IFCM**, (Domínguez, 2019)).

A pesar de las diferencias de criterio por algunos autores, es posible clasificar a los anteriores algoritmos como pertenecientes a un enfoque particular: uno de ellos se basa en grafos (ERS), uno está cimentado en evolución de contorno (ERGC), uno fundado en generación de clústers (SLIC), dos soportados por la optimización de una función de energía (CRS y SEEDS), uno basado en cuencas hidrográficas (W) y el último en agrupamiento difuso intuitivo (IFCM).

La elección particular de dichos algoritmos obedece a los siguientes factores:

- Variedad de enfoques. Se optó por seleccionar algoritmos que utilizan técnicas diversas para la generación de superpíxeles, esto con la finalidad de comparar con diferentes métodos.
- Información o implementación disponible. La existencia de una cantidad suficiente de información del algoritmo para su descripción precisa y adecuada.
- Actualidad. La mayoría son algoritmos desarrollados en forma reciente por los autores.

Se describe brevemente a continuación cada uno de los algoritmos seleccionados.

2.6.1. Contour Relaxed Superpixels (CRS)

El algoritmo CRS genera superpíxeles bajo la consideración de obtener máxima homogeneidad de la textura dentro de cada uno y la máxima conformidad de los contornos con respecto al contenido de la imagen y a un modelo de campo aleatorio de *Gibbs-Markov* (Conrad *et al.*, 2013).

El algoritmo tiene las siguientes ventajas:

- a) Control directo del número de píxeles K
- b) Control de la compacidad de los superpíxeles mediante un único parámetro κ
- c) Un modelo explícitamente estadístico de la forma y contenido de los superpíxeles permite realizar la segmentación en un número arbitrario de características como intensidad, color, profundidad, etc. La función de energía a optimizar está derivado de este modelo estadístico de la imagen.

2.6.2. Eikonal Region Growing Clustering (ERGC)

El algoritmo ERGC (Buyssens *et al.*, 2014) se basa en la ecuación Eikonal para generar superpíxeles. El enfoque de este algoritmo es un esquema basada en grafos de crecimiento de regiones donde el mapa potencial utilizado es dinámico y evoluciona durante la difusión. El algoritmo completo consta de 3 pasos (Domínguez, 2019):

1. La difusión de las semillas deseadas se da inicialmente con el 90 % colocándolos en una cuadrícula
2. Se mejora la sobresegmentación agregando el 10 % restante de las semillas
3. Se itera el procedimiento de semillas hasta que se alcance el criterio de paro.

La inicialización consiste del muestreo de K pixeles semilla en una rejilla regular en intervalos S iguales a $S = \sqrt{N/K}$, con N igual al número de pixeles en la imagen. Posteriormente son calculadas la media y la varianza de color de las semillas y sus 4-vecinos; el mismo cálculo es realizado en los pixeles que se encuentran en un vecindario de 3×3 y la semilla es desplazada a un pixel con la menor varianza de color. Después de la inicialización, es generado un frente de propagación que actualiza los superpixeles.

Algoritmo 1 ERGC

Input: Imagen I , grafo G , número de superpixeles K

Output: Segmentación de superpixel R

- 1: $V_0 \leftarrow \arg \min_{v \in V} (\delta_v / N_v)$
 - 2: Resolver $\|(\nabla_w - f)(u)\|_p = P(u) \quad \forall u \in V; f(u) = \phi(u) \quad \forall u \in V_0$
 - 3: Guardar distancias geodésicas U_0 ;
 - 4: $it \leftarrow 1$;
 - 5: **while** $it < n$ **do**
 - 6: $V_0 \leftarrow V_0 \cup \arg \max_{v \in V} (U_{it}(v))$
 - 7: Repetir el paso 2 con V_0 $it \leftarrow it + 1$
 - 8: Guardar distancias geodésicas U_{it}
 - 9: **end while**
-

2.6.3. Entropy Rate Superpixels (ERS)

En ERS la función objetivo es del tipo submodular con incremento monótonico orientado a grafos y consta de dos componentes: la rapidez de la entropía en un proceso de camino aleatorio (*random walk*) y un término de balanceo. La maximización de una función submodular es un problema NP-complejo, es por ello que se utiliza un algoritmo del tipo avaricioso (*greedy*) para la optimización de la misma.

La rapidez de la entropía favorece la formación de clústers compactos y homogéneos, mientras que la función de balanceo promueve clústers de tamaño similar. El algoritmo inicia con un conjunto vacío (un grafo completamente desconectado, $A = \emptyset$) y secuencialmente añade aristas al conjunto. En cada iteración agrega la arista que arroja la mayor ganancia. Las iteraciones son detenidas cuando el número de subgrafos conectados alcanza un valor predeterminado, $N_A = K$ (Liu *et al.*, 2011).

Algoritmo 2 ERS**Input:** Grafo $G = (V, E)$, $w : E \rightarrow \mathbb{R}^+$, número de superpíxeles K , α **Output:** Segmentación de superpíxel A

```

1:  $A \leftarrow \emptyset$ ,  $U \leftarrow E$ 
2: repeat
3:    $\hat{a} \leftarrow \arg \max_{a \in U} \mathcal{F}(A \cup \{a\}) - \mathcal{F}(A)$ 
4:   if  $A \cup \{\hat{a}\} \in \mathcal{I}$  then
5:      $A \leftarrow A \cup \{\hat{a}\}$ 
6:   end if
7: until  $U = \emptyset$ 

```

2.6.4. Superpixels Extracted via Energy-Driven Sampling (SEEDS)

El algoritmo SEEDS está basado en una optimización de escalado de montaña (*hill-climbing*). Inicialmente se genera una rejilla con el número deseado de superpíxeles; el proceso continúa intercambiando píxeles en los límites entre superpíxeles vecinos hasta alcanzar la maximización de la función objetivo.

El algoritmo utiliza una función de energía objetivo robusta y veloz basada en reforzamiento de similaridad por color entre los contornos y el histograma de color del superpíxel (Van den Bergh *et al.*, 2012).



Figura 2.11: Proceso de segmentación del algoritmo SEEDS (Van den Bergh *et al.*, 2012).

2.6.5. Simple Linear Interactive Clustering (SLIC)

Radhakrishna Achanta propuso con SLIC un agrupamiento local de píxeles en el espacio 5-D definido por los valores L , a , b del espacio de color CIELAB y las coordenadas x , y de los píxeles. La distancia descrita permite asegurar compacidad y regularidad en los contornos de los superpíxeles, logrando procesar imágenes en escala de grises y a color (Achanta *et al.*, 2010).

El algoritmo acepta como entrada un número deseado de superpíxeles K que tendrán aproximadamente un tamaño igual, así que se eligen K centroides correspondientes a cada clúster de superpíxel $C_k = [l_k, a_k, b_k, x_k, y_k]^T$ con $k = [1, K]$ en intervalos regulares S . Para una imagen con N píxeles, el tamaño aproximado de cada superpíxel es de N/K , así que, para cada uno de ellos, habrá un centroide en cada intervalo $S = N/K$ en la rejilla. Para disminuir la posibilidad de que el centro del grupo sea

un borde o un punto de ruido, se calcula el gradiente mínimo en un vecindario 3×3 . El gradiente se calcula mediante:

$$G(x, y) = \|I(x + 1, y) - I(x - 1, y)\|^2 + \|I(x, y + 1) - I(x, y - 1)\|^2 \quad (2.11)$$

donde $I(x, y)$ es el vector de características C_k correspondiente al pixel en la posición (x, y) , y $\|\cdot\|$ es la norma L_2 .

Debido a que la extensión espacial de cualquier superpixel es aproximadamente S (el área aproximada de un superpixel), es posible asumir que los pixeles asociados con el centroide de su clúster estarán dentro de una área de $2S \times 2S$ alrededor del centroide del superpixel en el plano xy . Esta, entonces, se convierte en el área de búsqueda de los pixeles cercanos a cada centroide. Por otro lado, las distancias euclídeas en espacio de color CIELAB son perceptualmente significantes para distancias pequeñas (m en ecuación 2.12). En caso que las distancias espaciales en pixeles excedan este límite de distancia de color perceptual se empiezan a sobre exceder las similitudes de color en pixeles. Es por ello que, en lugar de utilizar una norma euclídea en espacio 5D, se utiliza una medida de distancia D_s definida de la siguiente manera:

$$\begin{aligned} d_{lab} &= \sqrt{(l_j - l_i)^2 + (a_j - a_i)^2 + (b_j - b_i)^2} \\ d_{xy} &= \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \\ D_s &= d_{lab} + \frac{m}{S} + d_{xy} \end{aligned} \quad (2.12)$$

donde D_s es la suma de la distancia lab y el plano xy normalizado por el intervalo de la rejilla S . En la ecuación 2.12 se agrega una variable m para permitir el control de la compacidad de un superpixel. Entre mayor sea el valor de m , la proximidad espacial es enfatizada y el clúster se vuelve más compacto. Este valor puede estar en el rango $[1, 20]$.

2.6.6. Watershed (W)

Basado en el algoritmo de cuencas de agua (*watersheds*). En forma general el algoritmo opera la imagen como un mapa topográfico con elevaciones y valles. Cualquier imagen en tonalidades de gris (aunque también es aplicable a imágenes a color) puede ser considerada como una superficie topográfica, con la altitud de cada posición de la superficie siendo igual al valor en tono de gris en la misma ubicación de la imagen. Los puntos más altos de la superficie se corresponden a los puntos de mayor brillo en la imagen. Una “gota de agua” depositada en una superficie se desliza hacia abajo hasta alcanzar un mínimo regional y se vuelve parte de la zona de inundación en ese mínimo. La inundación de una superficie topográfica genera lagos como zonas bajo las cuales las características subyacentes de la superficie quedan completamente desvanecidas (Meyer, 1992).

En la Figura 2.12 se muestra en forma general el proceso de inundación descrito.

Algoritmo 3 SLIC**Input:** Imagen I , número de superpíxeles K **Output:** Imagen sobresegmentada R

- 1: Inicializar $C_k \leftarrow [l_k, a_k, b_k, x_k, y_k]^T$ por muestreo de píxeles en rejillas regulares S
- 2: Desplazar centros de clúster C_k al menor valor de gradiente en un vecindario de 3×3
- 3: **repeat**
- 4: **for** cada centro de clúster C_k **do**
- 5: Asignar el mejor píxel de un cuadrado de $2S \times 2S$ alrededor del centro del clúster acorde a una medida de distancia D_s ▷ Ecuación 2.12
- 6: **end for**
- 7: Calcular nuevos centros de clúster y error residual $E\{\text{L1 distancia entre centros previos y actuales}\}$
- 8: **until** $E \leq \text{umbral}$
- 9: Asegurar conectividad

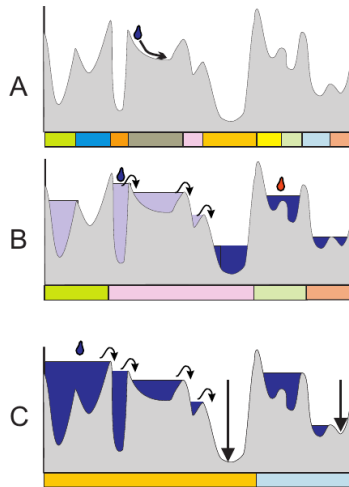


Figura 2.12: Zonas de inundación asociadas con los varios tipos de inundación de una superficie topográfica. En (a) una gota de agua colocada como semilla se desliza hasta un mínimo regional; (b) y (c) muestran la inundación de cuencas hasta formar superficies (lagos) (Meyer, 2019).

2.6.7. Intuitionistic Fuzzy C-Means (IFCM)

El algoritmo IFCM es una adaptación del algoritmo de Agrupamiento Difuso Intuitivo para generar superpíxeles y que considera las propiedades de los superpíxeles antes mencionadas (Domínguez, 2019).

El algoritmo IFCM agrupa los píxeles representados como Conjuntos Difusos Intuitivos (IFS). Para una imagen en escala de grises, con ancho W , altura H y

$N = W \times H$ píxeles, con niveles de intensidad entre 0 y $L - 1$, la representación de la imagen con conjuntos difusos intuitivos se describe como:

$$I = \{x_i, \mu(x_i), \nu(x_i), \pi(x_i)\} \quad 1 < i < N \quad (2.13)$$

Donde $\mu(x_i)$, $\nu(x_i)$ y $\pi(x_i)$ son los grados de pertenencia, no pertenencia e indecisión respectivamente, del píxel i en la imagen, y N es el número total de píxeles en la imagen. Las siguientes ecuaciones definen las transformaciones realizadas para la obtención de cada uno de los valores de μ , ν y π .

$$\mu(x_i) = \frac{x_i - \text{mín } x_i}{\text{máx } x_i - \text{mín } x_i} \quad i = 1, \dots, N; x_i \in 0, \dots, L - 1 \quad (2.14)$$

donde $\text{máx}(x_i)$ y $\text{mín}(x_i)$ denotan el máximo y el mínimo valor de los niveles de gris (o canal de color) en la imagen.

En grado de no pertenencia se utilizó la función de negación de Sugeno:

$$\nu(x_i) = \frac{1 - \mu(x_i)}{1 + \lambda\mu(x_i)} \quad \lambda > 0 \quad (2.15)$$

donde λ es una variable que se deriva de la solución original de negación de Sugeno.

El grado de indecisión (también llamado índice intuicionista) se obtiene mediante:

$$\pi(x_i) = 1 - \mu(x_i) - \nu(x_i) \quad (2.16)$$

con las condiciones $\pi(x_i) \leq 1$ y $\mu(x_i) + \nu(x_i) + \pi(x_i) = 1$. Los conjuntos difusos intuicionistas (IFS) son agrupados posteriormente mediante el Agrupamiento Difuso Intuitivo propuesto por (Xu y Wu, 2010) mediante la función objetivo:

$$\text{mín } J_m(U, Z, V) = \sum_{j=1}^p \sum_{i=1}^c \mu_{ij}^m d^2(Z_j, V_i) \quad (2.17)$$

donde $Z = \{Z_1, Z_2, \dots, Z_p\}$ son p IFS, c es el número de grupos y $V = \{V_1, V_2, \dots, V_c\}$ son los centroides iniciales. El parámetro m es el coeficiente difuso ($m > 0$), también llamado *fuzzifier*, u_{ij} es el grado de pertenencia de i -ésimo dato al i -ésimo grupo, $U = (u_{ij})_{c \times p}$ es la matriz de pertenencia y $d^2(Z_j^{IFS}, V_i^{IFS})$ es la distancia euclidiana difusa intuitiva al cuadrado que se calcula mediante:

$$d^2(Z_j^{IFS}, V_i^{IFS}) = \left((\mu(Z_j) - \mu(V_i))^2 + (\nu(Z_j) - \nu(V_i))^2 + (\pi(Z_j) - \pi(V_i))^2 \right) \quad (2.18)$$

Dado que la descripción anterior obedece al algoritmo IFCM como método de optimización, en (Domínguez, 2019) se adaptó para generación de superpíxeles con la función objetivo dada ahora por:

$$\text{mín } J_m(U, V^{IFS} : X^{IFS}) = \sum_{j=1}^N \sum_{i=1}^K \mu_{ij}^m d^2(x_j, v_i) \quad (2.19)$$

donde X^{IFS} son N datos y V^{IFS} son los centroides de los K superpíxeles iniciales, ambos como conjuntos difusos intuitivos.

La expresión para calcular los centroides iniciales está dada por:

$$\mu_{ij} = \frac{1}{\sum_{r=i}^{i \in K_p} \left(\frac{d^2(x_j^{IFS}, v_i^{IFS})}{d^2(x_j^{IFS}, v_r^{IFS})} \right)^{\frac{1}{m-1}}} \quad (2.20)$$

donde μ_{ij} es la pertenencia, K_p es un vector con tres centroides más próximos ($K_p \subseteq K$) y $d^2(X_j^{IFS}, v_i^{IFS})$ es la distancia euclidiana descrita anteriormente.

La actualización del vector de centroides se lleva a cabo mediante:

$$\begin{aligned} \mu(v_i) &= \frac{\sum_{j=1}^N u_{ij}^m \mu(x_j)}{\sum_{j=1}^N u_{ij}^m} \\ \nu(v_i) &= \frac{\sum_{j=1}^N u_{ij}^m \nu(x_j)}{\sum_{j=1}^N u_{ij}^m} \\ \pi(v_i) &= \frac{\sum_{j=1}^N u_{ij}^m \pi(x_j)}{\sum_{j=1}^N u_{ij}^m} \end{aligned} \quad (2.21)$$

donde $v_i^{IFS} = \{\mu(v_i), \nu(v_j), \pi(v_j)\}$ es el centroide, x_j es un dato, u_{ij} es el grado de pertenencia del j -ésimo grupo y m es el coeficiente difuso. Este proceso se repite hasta que el error sea menor al establecido o se llegue al número máximo de iteraciones.

En esta implementación el espacio de búsqueda se limita en una región de tamaño $2R \times 2R$, esto con el fin de aumentar la velocidad del algoritmo y conservar la homogeneidad de los segmentos. Esta reducción del espacio de búsqueda se muestra en la Figura 2.13.

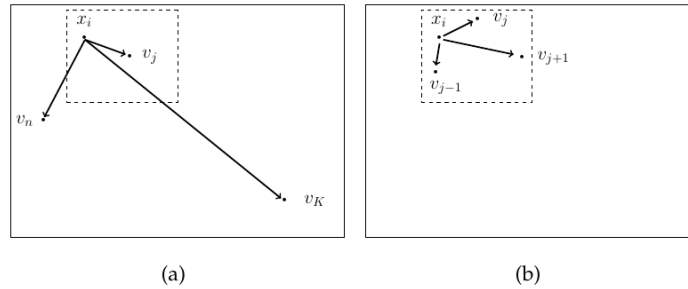


Figura 2.13: Reducción de espacio de búsqueda; (a) proceso de búsqueda del algoritmo IFCM; (b) búsqueda del algoritmo modificado, en una región limitada (Domínguez, 2019).

El listado del Algoritmo 4 describe las fases en que el algoritmo IFCM es desarrollado.

Algoritmo 4 IFCM**Input:** Imagen es escala de grises X , K **Output:** Imagen sobresegmentada Y

```

1:  $X \mapsto \hat{X}$ 
2:  $\hat{X} \mapsto X^{\text{IFS}}$ 
3: Inicializar  $V_i^{\text{IFS}} = [\mu(v_i), \nu(v_i), \pi(v_i)]$ ;  $i = 1, \dots, K$ ; muestreando pixeles en la rejilla regular
4: Calcular  $u_{ij}^{(0)} \leftarrow \frac{1}{\sum_{r=1}^K \left( \frac{d^2(x_j^{\text{IFS}}, v_i^{\text{IFS}})}{d^2(x_j^{\text{IFS}}, v_r^{\text{IFS}})} \right)^{\frac{1}{m-1}}}$ 
5:  $l \leftarrow 0$ 
6: while  $\max ||U^{(l+1)} - U^l|| < \epsilon$  do
7:    $v_i^{\text{IFS}} \leftarrow \left\{ \frac{\sum_{j=1}^N u_{ij}^m \mu(x_j)}{\sum_{j=1}^N u_{ij}^m}, \frac{\sum_{j=1}^N u_{ij}^m \nu(x_j)}{\sum_{j=1}^N u_{ij}^m}, \frac{\sum_{j=1}^N u_{ij}^m \pi(x_j)}{\sum_{j=1}^N u_{ij}^m} \right\}$ 
8:   for  $i \leftarrow 1$  to  $K$  do
9:     for all  $x_j^{\text{IFS}} \in 2R \times 2R$  do
10:      if  $\forall i, r, d^2(x_j^{\text{IFS}} - v_r^{\text{IFS}}) > 0$  then
11:         $u_{ij}^{(l)} \leftarrow \frac{1}{\sum_{r=1}^K \left( \frac{d^2(x_j^{\text{IFS}}, v_i^{\text{IFS}})}{d^2(x_j^{\text{IFS}}, v_r^{\text{IFS}})} \right)^{\frac{1}{m-1}}}$ 
12:      end if
13:    end for
14:  end for
15:   $l \leftarrow l + 1$ 
16: end while
17: for  $i \leftarrow 1$  to  $N$  do
18:   for  $j \leftarrow 1$  to  $K$  do
19:     $Y \leftarrow \arg \max_j p(u_{ij} \in U : u_{ij} > Y_i)$ 
20:   end for
21: end for
22: return  $Y$ 

```

2.7. Posprocesamiento y segmentación basada en grafos

Durante el desarrollo del trabajo fueron expresadas, por integrantes del comité y compañeros alumnos, algunas interrogantes relacionados a la forma en que los superpíxeles son aprovechados como entrada a procesos de nivel superior. Dado ello se implementó una etapa que no había sido considerada en los objetivos originales de tesis y se realizó como complemento al trabajo de sobresegmentación utilizando grafos difusos: la utilización de los superpíxeles generados por el algoritmo en una etapa posterior de segmentación regular. Se plantearon dos objetivos principales:

1. Obtener una segmentación final por objeto. Como consecuencia de la eficiencia del algoritmo propuesto se utilizaron resultados de la sobresegmentación para evidenciar su uso en una tarea de nivel superior: segmentación por objetos (segmentación simple o agrupamiento).
2. Demostrar la viabilidad y eficiencia de los algoritmos basados en grafos.

Los productos derivados de estos objetivos se describen al final de la sección 5.3.2, como parte de los resultados .

Esta etapa utiliza una construcción de la teoría de grafos denominada *grafo de adyacencia de regiones* (*Region Adjacency Graph* o RAG), que se describe a continuación.

2.7.1. Grafo de Adyacencia de Regiones (RAG)

Como su nombre lo indica, un Grafo de Adyacencia de Regiones (*Region Adjacency Graph* o RAG) representa, mediante un grafo adireccional, las relaciones de adyacencia (vecindad) entre regiones. Cada región (en este caso cada superpixel) es un vértice en un grafo. Entre cada par de regiones adyacentes existe una arista cuyo peso está determinado mediante alguna función, ya sea de similitud o de distancia entre las mismas.

En su forma general este concepto se muestra en la Figura 2.14. Se ilustran las regiones (vértices, v_i) como enumeradas y las aristas como líneas punteadas.

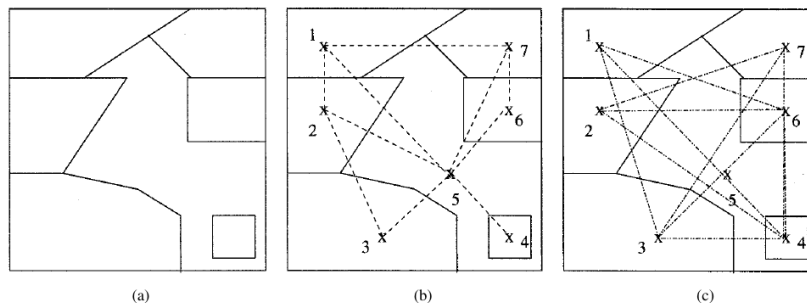


Figura 2.14: Dos niveles en un Grafo de Adyacencia de Regiones. a) Imagen original; b) relaciones de conectividad simple; c) relaciones de conectividad doble (Trémeau y Colantoni, 2000).

Un ejemplo aplicado al análisis de imágenes se observa en la Figura 2.15. El grafo RAG (en 2.15c) corresponde cada vértice a una región (superpixel) derivadas de una sobresegmentación y las aristas a la diferencia de color entre nodos. La barra de color a la derecha de (c) ilustra el peso de las aristas del grafo calculado como distancias entre regiones.

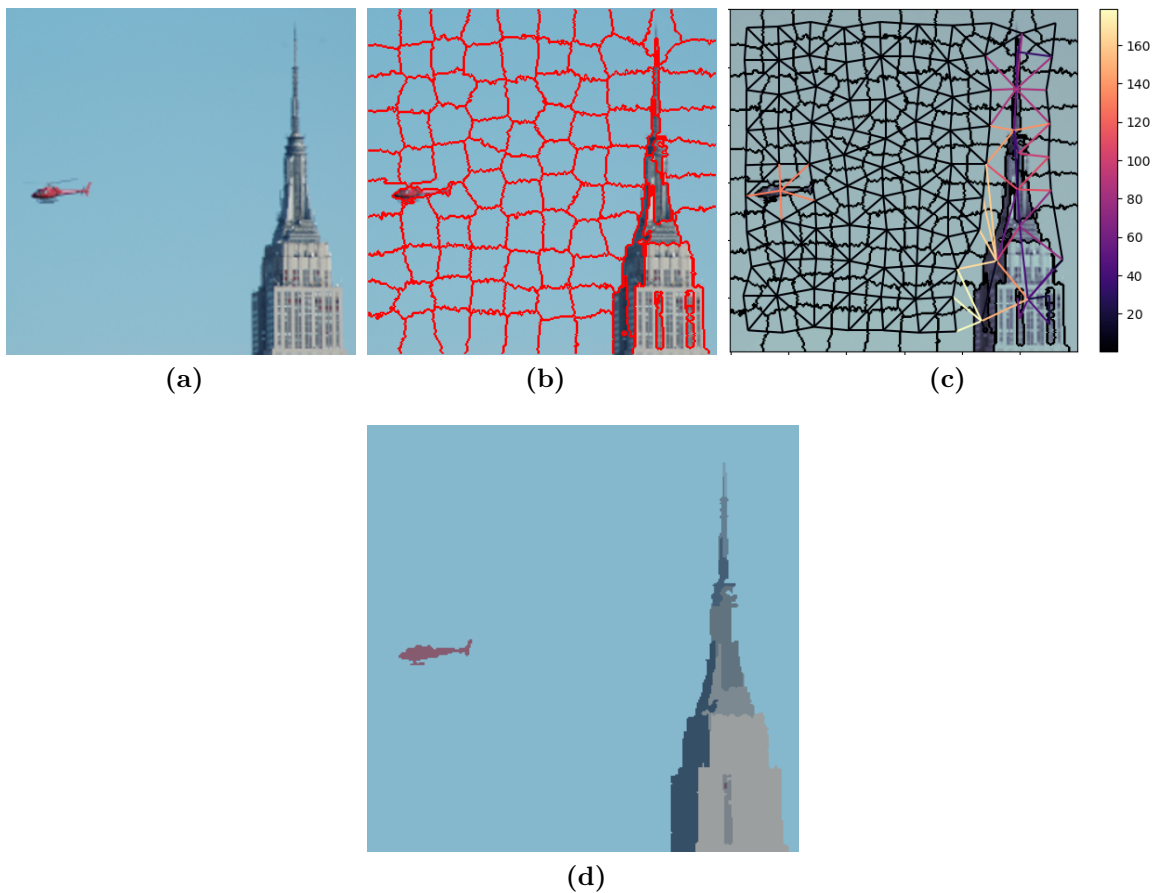


Figura 2.15: Utilización de un RAG en el procesamiento de imágenes. Imagen *dsc_7742* del repositorio *Weizmann*. (a) Imagen original; (b) sobresegmentación: $k \approx 100$; (c) grafo de adyacencia de regiones (RAG) generado. (d) resultado de la segmentación mediante RAG.

Lo que la anterior imagen representa (en particular (c)), es la “cercanía” y posibilidad de agrupamiento (o, lo que es equivalente, la reducción del grafo de adyacencia) de las regiones donde las aristas tengan un peso mayor o menor a un umbral dado. Las aristas en colores claros alrededor del edificio y del helicóptero en la Figura 2.15 (c), indican una diferencia (distancia) mayor entre superpíxeles; las aristas oscuras son más adecuadas para ser fusionadas.

Formalmente se observa que, para cada vértice v_i , se corresponde una región R_i y un valor de color c_i representativo de la distribución en dicha región. Para cada arista e_{ij} existe un par de regiones adyacentes $\{R_i, R_j\}$ y una distancia de color $d^2(R_i, R_j)$ la cual puede ser utilizada para comparar la distribución de color de ambas regiones de acuerdo a (Trémeau y Colantoni, 2000):

$$d^2(R_i, R_j) = d^2(c_i, c_j) \quad (2.22)$$

Para $d^2(c_i, c_j)$, en (Trémeau y Colantoni, 2000) se utilizó la distancia de *Fisher*. En el trabajo descrito en esta tesis se utilizó una distancia euclídea entre las medias del color de cada superpixel.

El grafo de adyacencia de regiones puede ser usado para fusionar regiones adyacentes siempre y cuando estas regiones tengan una distribución de color lo suficientemente cercana. Esta unificación introduce una reorganización del grafo de adyacencia de regiones que puede ser utilizada para fusionar los vértices v_i y v_j en un único vértice v_k uniendo sus correspondientes aristas y reuniendo dichos vértices en un solo conjunto $\{v_i, v_j\}$ con la misma etiqueta v_k (Figura 2.16).

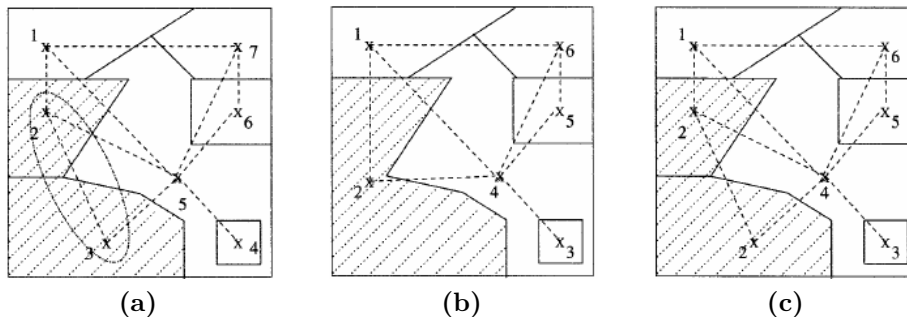


Figura 2.16: Simplificación de un RAG después de haber fusionado dos regiones adyacentes. (a) Vértice a ser fusionados (2 y 3); (a) Fusión de dos vértices; (a) Obtención de dos vértices con la misma etiqueta. (Trémeau y Colantoni, 2000)

El proceso llevado a cabo en el presente trabajo como aplicación posterior del algoritmo motivo de tesis se ilustra en la Figura 2.17.

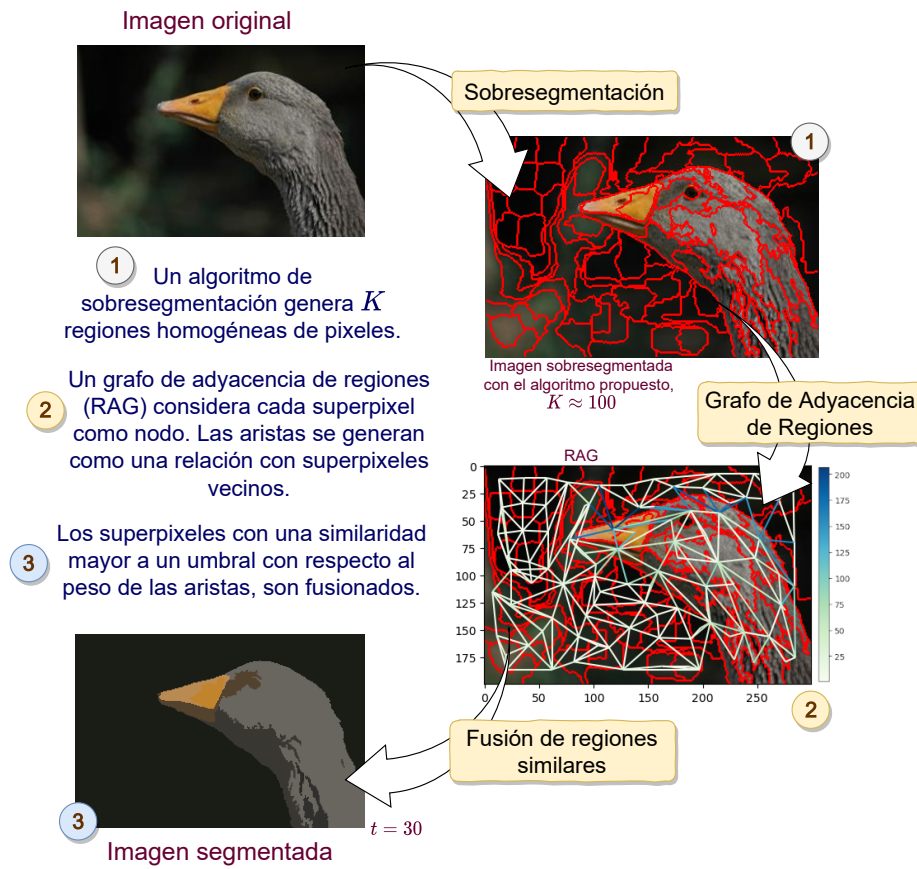


Figura 2.17: Segmentación regular como producto de superpíxeles. La barra de color indica la “fortaleza” de medida de distancia (euclídea, en este caso).

3

Estado del Arte

Como base de conocimiento y comparación se llevó a cabo una revisión de diversas publicaciones en revistas y congresos que describen técnicas y algoritmos semejantes al propuesto o que se consideraron como apoyo en el desarrollo del mismo. Se enlistan igualmente trabajos realizados anteriormente en el CENIDET.

3.1. Antecedentes

El siguiente resumen describe trabajos anteriormente realizados en el CENIDET. Estas tesis de maestría y doctorado han sido consideradas de utilidad en el desarrollo del algoritmo presentado.

Extracción de superpíxeles utilizando agrupamiento intuicionista local (Domínguez, 2019)

En este documento se describió la implementación de 3 algoritmos (SLIC, FCM y IFCM) y la implementación y pruebas con dos versiones difusas de SLIC: SLIC difuso y SLIC difuso intuitivo. El algoritmo descrito en el documento como de superpíxeles que utiliza agrupamiento intuicionista local es evaluado con respecto a tres métricas: recuerdo de límites, error de subsegmentación y variación explicada. Otros 5 algoritmos fueron contrastados con respecto al resultado del algoritmo propuesto: W (Watershed), SLIC (Simple Linear Iterative Clustering), LSC (Linear Spectral Clustering), TP (TurboPixel) y PB (Pseudo Boolean Superpixel); para dos conjuntos de datos: BSDS500 y BrainWeb.

En los resultados se demostró un buen rendimiento en las pruebas con los conjuntos de datos de BrainWeb y BSDS500. Obtuvo los mejores resultados en la mayoría de las métricas de recuperación de límites. El algoritmo propuesto generó superpíxeles irregulares con alta adherencia a los bordes, esta característica le permite preservar detalles pequeños en las imágenes, en los conjuntos de datos considerados.

Este algoritmo fue de apoyo importante en la especificación del algoritmo motivo de la presente tesis, por ello se describe en mayor detalle en la subsección 2.6.7.

Metodología para la Estructuración y Uso de Conocimiento en Segmentación de Imágenes Digitales (Cervantes, 2006)

Se presentó un entorno de segmentación de imágenes basado en *scripts* (con un respectivo intérprete) y una interfaz gráfica para el despliegue y manipulación de las operaciones. Los guiones describen las operaciones posibles de realizar en las imágenes. El sistema permite procesar imágenes mediante segmentación basada en identificación por bordes utilizando los algoritmos *Canny* y *Frei-Chen*.

La arquitectura propuesta mostró que es posible realizar una separación de algoritmos de segmentación (conocimiento procedimental) y su conocimiento declarativo relacionado; a la vez que permite agregar nuevos operadores y algoritmos a su biblioteca.

Caracterización de Texturas Naturales (Avellaneda, 2009)

Esta tesis de maestría propuso un proceso de segmentación y reconocimiento de texturas basado en reglas generadas por el software *See5*, segmentación con el algoritmo *Fuzzy C-Means* y 5 clases de texturas. Se utilizaron imágenes de escenas naturales, en modelo de color HSI, caracterizando sus texturas mediante transformada *wavelet* discreta.

La textura es una característica, factible de ser caracterizada, que permite cuantificar propiedades como suavidad, rugosidad y regularidad. Se utilizó la transformada *Wavelet* Discreta (la familia *wavelet Haar*) para dicha caracterización debido a que permite analizar una imagen a mayor detalle y es posible encontrar información que a simple vista es imperceptible.

Para la segmentación se seleccionaron para pruebas las técnicas: *Fuzzy C-Means*, crecimiento de regiones y *K-Nearest Neighbour* debido a su eficiencia y popularidad. En particular se utilizó el algoritmo *Fuzzy C-Means* por ser el que se aproxima al eficiente para segmentar imágenes de escenas naturales y consistente en:

Dado un conjunto de clases c_1, c_2, \dots, c_c y un conjunto de datos $X = \{x_1, x_2, \dots, x_n\}$ de n muestras, se considera que x_j puede pertenecer a más de una clase mediante el grado de pertenencia $P(c_i/x_j)$ indicando en qué medida la j -ésima muestra x_j pertenece a la clase c_i . Los centros de clase son m_1, m_2, \dots, m_c . El algoritmo:

1. Extraer el número de muestras n a utilizar y el número de clases c .
2. Inicializar los centros de las clases m_i y las probabilidades $P(c_i/x_j), i = 1 \dots c; j = 1 \dots n$.
3. Normalizar las probabilidades por medio de la ecuación:

$$\sum_{i=1}^c P(c_i/x_j) = 1; i = 1 \dots c \quad (3.1)$$

4. Recalcular $P(c_i/x_j)$ mediante:

$$P(c_i/x_j) = \frac{\left(\frac{1}{d_j}\right)^{\frac{1}{(b-1)}}}{\sum_{r=1}^c \left(\frac{1}{d_{ij}}\right)^{\frac{1}{(b-1)}}} \quad y \quad d_{ij} = \|x_j - m_i\|^2 \quad (3.2)$$

5. Obtener m_i de acuerdo a:

$$m_i = \frac{\sum_{j=1}^n [P(c_i/x_j)]^b x_j}{\sum_{j=1}^n [P(c_i/x_j)]^b} \quad (3.3)$$

6. Repetir los pasos del 3 al 5 hasta que m_i y $P(c_i/x_j)$ no cambien o el cambio sea pequeño.

En la caracterización de texturas se describen estadísticos de primero (píxeles simples) y segundo orden (toman en cuenta la relación espacial). La media, la varianza, y la desviación estándar pertenecen al primer grupo; la matriz de co-ocurrencia de niveles de gris al segundo.

Se describe la transformada *wavelet* como un método que convierte una función o señal en otra forma mediante la descomposición de la misma en ondas sinusoidales de diferentes frecuencias. La versión discreta permite reducir la cantidad de escalas y posiciones que utiliza la versión continua.

La metodología propuesta fue probada con diversas imágenes de escenas naturales. Los resultados obtenidos muestran que dicha metodología permite que la segmentación y el reconocimiento de las texturas sea invariante a traslación y ante algunos cambios de rotación y escala.

Sintonización de una Red Totalmente Conectada para Segmentación de Dos Clases de Objetos en Imágenes (Suárez, 2018)

Enfocado a la segmentación de imágenes por píxeles. Este trabajo implementó, en una GPU, una Red Neuronal Convolutiva con los parámetros adecuados para segmentar dos clases de objetos (persona y ave). El objetivo del trabajo fue encontrar los parámetros de sintonización de la red que le permitan una mayor eficiencia en la segmentación. Utiliza una capa de *deconvolución* para la operación de segmentación. El método de gradiente descendente estocástico fue el de mejores resultados para la optimización.

Los resultados mostraron que se logró una mejora en el rendimiento de la segmentación, utilizando la media del repositorio BSD500, en un 3.43 % con respecto al modelo FCN-Alexnet y 1.23 % con respecto al modelo FCN-8s, al promediar las tres métricas de precisión de la segmentación (VI, PRI y GCE).

Segmentación no Paramétrica de Tejidos Cerebrales Mediante una Arquitectura Paralela de Redes Neuronales Convolucionales (Morales, 2018)

En este trabajo se propuso, desarrolló e implementó un método de segmentación no paramétrica de tejidos cerebrales, basada en una arquitectura paralela de Redes Neuronales Convolucionales. La segmentación fue evaluada con las métricas Jaccard, Coeficiente de Similitud Dice (DSC) y Área Bajo la Curva (AUC) en imágenes de dos diferentes repositorios: *BrainWeb* y *BraTS 2017*.

El interés de esta tesis es hacia los conceptos y tratamiento de la segmentación de imágenes. Se destaca la ausencia de parámetros para realizar la segmentación respectiva. Conceptos como *segmentación*, *homogeneidad*, *similitud* y *discontinuidad* son presentados.

Los resultados cuantitativos del método propuesto con respecto a FCM muestran mejores valores con imágenes del repositorio *BrainWeb*. De la misma manera con respecto a la métrica ISMM-SC del estudio completo de (Fan *et al.*, 2018) se observan mejores resultados. La nula necesidad de parametrizar el proceso de segmentación es una aportación importante del método descrito.

Seguimiento y Predicción del Movimiento de Objetos Rígidos en un Ambiente Complejo (Ríos, 2004)

En este documento se reporta la realización e implementación de etapas de adquisición de imágenes y segmentación además de otras tres etapas de detección, seguimiento y predicción del movimiento de objetos. El objetivo, como lo menciona el título, es el diseño de métodos, mecanismos y algoritmos que permitan realizar el seguimiento y predicción del movimiento de objetos rígidos.

Se procesaron secuencias de imágenes sin atender a la frecuencia de cuadros por segundo a partir de archivos con formato *Audio Video Interleave* (AVI) capturados con una única cámara.

- La fase de *detección* se realiza mediante “resta de imágenes”.
- La etapa de *segmentación*, basada en segmentación por regiones y detección y extracción de rasgos (velocidad de movimiento, dirección de trayectoria, etc.). Se desarrolla recorriendo la matriz de ocurrencias y se utiliza un algoritmo recursivo de conectividad 8-píxel con la finalidad de agrupar las regiones en movimiento.
- La etapa de *seguimiento* usa una retícula cuadrada donde se dibuja una cruz en el centro.
- En la fase de *predicción* se utiliza la regresión lineal y mínimos cuadrados.

Dos relaciones básicas entre píxeles son introducidos:

- *Vecindad*. Un píxel tiene 4 vecinos verticales y horizontales, y cuatro vecinos diagonales, siempre y cuando se cumpla que éste no se encuentre al borde de la imagen.

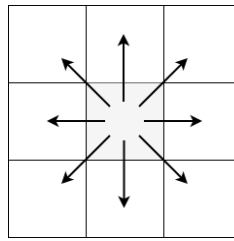


Figura 3.1: Vecindad entre píxeles. Se utilizó la vecindad 8-píxel

Las coordenadas que representan dicha vecindad se expresan mediante la siguiente ecuación:

$$N_8(p) = \{(x + 1, y), (x - 1, y), \\ (x, y + 1), (x, y - 1), \\ (x + 1, y + 1), (x + 1, y - 1), \\ (x - 1, y + 1), (x - 1, y - 1)\}$$

- *Conectividad.* Dos píxeles se encuentran conectados si cumplen algún criterio de similitud (por ejemplo intensidad o textura) y si se puede establecer un camino de adyacencia entre ellos.

Se describe el algoritmo de agrupamiento de regiones *8-píxel*: se establece un umbral en la variable de tono y se realiza la conectividad con los píxeles semejantes; la tonalidad, el tamaño y la posición del objeto son guardados en estructuras de datos destinados para ello. Otro algoritmo verifica que no existan regiones traslapadas.

El sistema completo fue realizado con la herramienta de Borland Software Corporation: C++ Builder.

Discusión

En el conjunto de tesis realizadas en el CENIDET se localizaron múltiples documentos de tesis con diversos enfoques a la segmentación de imágenes; de entre ellas dos son de interés particular para la presente tesis ya que utilizan un acercamiento desde la teoría difusa, (Avellaneda, 2009) y (Domínguez, 2019). El primero de ellos emplea como base el algoritmo *Fuzzy C-Means* y el segundo un enfoque de agrupamiento difuso intuitivo, *Intuitionistic Fuzzy C-Means* o IFCM.

No se encontraron trabajos finalizados relacionados a la segmentación de imágenes utilizando algún enfoque desde la teoría de grafos difusos.

3.2. Trabajos relacionados

Se presenta una introducción a diversos artículos recientes que presentan técnicas, métodos y/o algoritmos representativos y útiles para el desarrollo del tema de tesis

correspondiente. El orden en que se describen es descendente por fecha de publicación.

Superpíxeles: Una Evaluación del Estado del Arte (Stutz *et al.*, 2018)

En este artículo se presenta una evaluación exhaustiva de 28 algoritmos de superpíxeles del estado del arte. Para ello, los algoritmos fueron categorizados en de acuerdo al siguiente listado:

- Basadas en cuencas hidrográficas (W). Usualmente difieren en el preprocesamiento de la imagen y en la forma en que los marcadores son elegidos.
- Basados en densidad. Estos algoritmos buscan densidades en la imagen. *Edge-Augmented Mean Shift* (EAMS) y *Quick Shift* (QS) son ejemplos de este tipo.
- Basadas en grafos. Difieren en el algoritmo de particionado. FH, ERS y POISE implementan una fusión ascendente de píxeles hacia superpíxeles. NC y CIS utilizan cortes, mientras PB usa eliminación.
- Evolución del contorno. Estos algoritmos representan superpíxeles como contornos en evolución partiendo desde las semillas iniciales.
- Basados en rutas. Particionan una imagen en superpíxeles conectando puntos semilla mediante rutas de píxeles que siguen criterios específicos.
- Basados en agrupamiento. Inspirados por algoritmos de agrupamiento como *k-means* ya que son inicializados por píxeles semilla y utilizan información de color, espacial y adicionales como la profundidad.
- Optimización de energía. Estos algoritmos optimizan iterativamente una función formulada de energía. La imagen es particionada en una rejilla regular como segmentación inicial de superpíxel y posteriormente los píxeles son intercambiados entre superpíxeles vecinos en relación a la energía.
- Basados en *wavelets*. El algoritmo de *Superpixels from Edge-Avoiding Wavelets* (SEAW) no es comparable a la algoritmos anteriores.

Para el comparativo se eligieron 5 conjuntos de datos de prueba: BSDS500, SBD, NYUV2, SUNRGBD y Fash. Con respecto a ellos, los 28 algoritmos fueron sometidos a cálculos de métricas para verificar particularmente la adherencia bordes y compacidad. Las métricas de desempeño utilizadas para la evaluación fueron: recuperación de límites, error de subsegmentación, variación explicada y compacidad.

Los resultados visuales señalan una buena adherencia a los límites e inclusive algunos con capacidad de captar detalles pequeños. Con respecto a la velocidad de ejecución, se encontraron algoritmos con capacidad de ejecución en tiempo real y se encontró que los algoritmos iterativos permiten reducir el tiempo de ejecución sin afectar en gran

medida al desempeño. Los resultados de las métricas de desempeño empleadas para la evaluación no permitieron la discriminación confiable para dictaminar jerarquías.

Aplicación de un Algoritmo Modificado de Agrupamiento *Fuzzy C-Means* en Segmentación de Imágenes Médicas (Aljebory y Mohammed, 2018)

En este documento se presenta una modificación al algoritmo *Fuzzy C-Means* espacial (MFCM) que considera un efecto de pixeles vecinos con la finalidad de producir regiones más homogéneas que el algoritmo FCM regular y un mejor manejo del ruido. Esto se logra tomando en cuenta la topología de la imagen. Básicamente esto se logra en cuatro fases:

1. Inicialización de los centros basados en el algoritmo esperanza-maximización para localizar los centros óptimos
2. Consideración de la topología de la imagen, encontrando un vector de longitud variable de pixeles vecinos
3. Incorporación de la variación de distancia entre x_i y los centroides del grupo
4. Modificación de la función objetivo $F_p = \sum_{i=1}^c \sum_{k=1}^n u_{ik}^p |x_k - v_i|^2$ El resultado final de esta modificación se muestra en la siguiente ecuación:

$$J_p = \sum_{i=1}^c \sum_{k=1}^n u_{ik}^p \left[\sum_{XIZ_k} W(x_k, x_{k,1}) |x_{k,1} - \beta_k^* - v_i|^2 \right] + \dots + \sum_{i=1}^c \sum_{k=1}^n [u_{ik}^p |x_k - \beta_k^* - v_i|^2] + G_{ki} \quad (3.4)$$

donde $x_{k,1}$ representa la vecindad del pixel x_k , β_k es el campo de ganancia en el pixel k y $W(x_k, x_{k,1})$.

Se hizo una implementación del método en el *software* Matlab. Se realizaron pruebas de sensibilidad al ruido en imágenes de resonancia magética y rayos-X, contrastando resultados con los algoritmos FCM y MFCM. La Figura 3.2 presenta los resultados visuales de la comparativa.

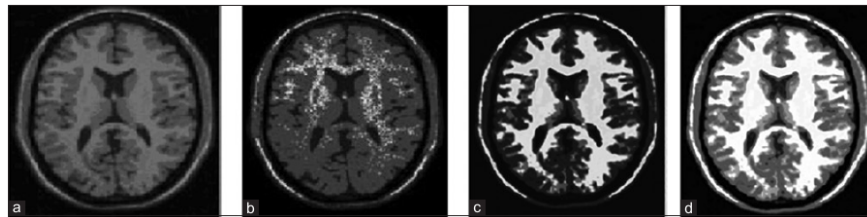


Figura 3.2: Resultados de la Segmentación del algoritmo NMFCM. a) Imagen T1 original. b) *Fuzzy C-Means* (FCM). c) FCM modificado (MFCM). d) El algoritmos propuesto (NMFCM) (Aljebory y Mohammed, 2018).

Modelo de Calidad de Información Teórica para Segmentación de Imágenes (Murashov, 2017)

En este artículo se trata el problema de la selección de la mejor segmentación de un conjunto de imágenes generadas por un algoritmo con diferentes parámetros. Se propone un modelo basado en redundancia de información como un criterio de calidad de la segmentación. Se hicieron pruebas con los algoritmos SLIC y *grabcut*.

El modelo de segmentación propuesto se describe como:

$$V = F(U, t) \quad (3.5)$$

donde U es una imagen de entrada, V es una imagen segmentada, F es un operador que describe el algoritmo de segmentación y t es un parámetro. Sea un conjunto de Q imágenes segmentadas de U , $V = \{V_1, V_2, \dots, V_q, \dots, V_Q\}$ en que se varía t , es necesario elegir la imagen V_q como el mínimo por el criterio:

$$q_{min} = \arg \min_{\mathbf{q}} (M(U, V_q)), \quad q = 1, 2, \dots, Q \quad (3.6)$$

La función de probabilidad de masa de la luminosidad de la imagen de entrada está representada por un modelo de mezcla gaussiana de K número de componentes que se traslapan parcialmente. La medida de la redundancia se da por:

$$R(K) = \frac{\log e + \log(2\pi\sigma_\psi^2)}{2\log K} \quad (3.7)$$

esto es, la medida de la redundancia depende linealmente del logaritmo de la varianza del ruido del sistema y es inversamente proporcional al número de segmentos producidos. La imagen de Q con menor R es seleccionada.

Los resultados, con imágenes de *Berkeley Segmentation Dataset*, muestran que el método propuesto selecciona correctamente la mejor imagen dentro del conjunto siendo la que de mayor similitud para con el *ground-truth*.

Completitud y Regularidad en Grafos Difusos Generalizados (Samanta et al., 2016)

El documento presenta múltiples conceptos relacionados a grafos difusos. El principal de ellos es el de *grafo difuso generalizado*.

Existen dos enfoques para asignar los valores de vértices y aristas a un grafo. En una de ellas los valores de pertenencia de los vértices son considerados y posteriormente, dependiendo de dichos valores, se asignan los valores de pertenencia de las aristas. El segundo enfoque realiza la operación en forma invertida. Respectivamente se les conoce como grafo difuso generalizado de tipo uno (GFG1) y grafo difuso generalizado de tipo 2 (GFG2).

Definición 8. (Samanta et al., 2016) Sea V un conjunto no vacío. Considerando dos funciones de la siguiente manera: $\rho : V \rightarrow [0, 1]$ y $\omega : V \times V \rightarrow [0, 1]$. Haciendo la

suposición $A = \{(\rho(x), \rho(y)) | \omega(x, y) > 0\}$. La triada (V, ρ, ω) está definida como el *grafo difuso generalizado de primer tipo* (GFG1) si existe una función $\phi : A \rightarrow (0, 1]$ tal que $\omega(x, y) = \phi(\rho(x), \rho(y))$ donde $x, y \in V$.

Definición 9. (Samanta *et al.*, 2016) Sea V un conjunto no vacío. Consideremos dos funciones de la siguiente manera: $\rho : V \rightarrow [0, 1]$ y $\omega : V \times V \rightarrow [0, 1]$ y considérese a B perteneciente al rango de ω . La triada (V, ρ, ω) está definida como el *grafo difuso generalizado de segundo tipo* (GFG2) si existe una función $\psi : B \rightarrow (0, 1]$ tal que para cada $x \in V$, $\rho(x) = \psi(\omega(e_x))$, donde $e_x = (x, y)$ tal que $y \in V$. Acá, $\rho(x)$, $x \in V$ son los valores de pertenencia de los vértices x y $\omega(x, y)$ es el valor de pertenencia generalizado de la arista (x, y) .

Procesamiento de Imágenes basado en Grafos Espectrales (Cheung *et al.*, 2018)

En este trabajo se describen aplicaciones del concepto de *procesamiento de señales basado en grafos* (GSP) en áreas como la compresión, restauración, filtrado y segmentación de imágenes digitales.

Una representación válida de la estructura intrínseca de la imagen en términos de un grafo de elementos conectados puede ser interpretada como una *señal en un grafo* con lo que es posible recurrir a la base práctica y matemática del procesamiento digital de señales en dominio espectral. En particular, en la segmentación por cortes de grafos la función de energía puede ser optimizada mediante algoritmos basados en grafos y teoría espectral de grafos. La eigendescomposición de la matriz laplaciana de un grafo permite expresar el espectro de las frecuencias de un grafo que represente la estructura intrínseca de una imagen, de esta manera es posible también derivar conceptos como transformaciones, *wavelets*, suavidad, etc., del mismo.

En el documento, la sección de segmentación se enfoca en el modelo Mumford-Shah de cortes de grafos, representado por el siguiente funcional:

$$J(u, \phi) = \int_{\Omega} (u - f)^2 dx + v \int_{\Omega} |\nabla_{\phi}| dx \quad (3.8)$$

Un estudio de grafos con etiquetado difuso (Borzooei *et al.*, 2016)

En este documento se definen algunos conceptos sobre conectividad de grafos con etiquetado difuso. Se describen conceptos como arco fuerte, corte de nodo parcial, puente y bloque.

Definición 10. (A. Nagoor Gan *et al.*, 2014) Se dice que un grafo $G = (\sigma, \mu)$ es un grafo con etiquetado difuso si $\sigma : V \rightarrow [0, 1]$ y $\mu : V \times V \rightarrow [0, 1]$ es inyectivo tal que el valor de pertenencia de los vértices y las aristas son distintos y $\mu(u, v) < \sigma(u) \wedge \sigma(v)$, para todo $u, v \in V$.

La conectividad difusa de un par de nodos en un grafo con etiquetado difuso de un par de nodos $u, v \in V(G)$, se define como $CONN_G(u, v) = \max\{S(P) : P \text{ es una ruta } u - v \text{ en } G\}$

Lo que resulta importante para el caso de la presente tesis, es la posibilidad de realizar cortes en este tipo de grafos. Para ello se define: un nodo w es llamado un nodo de corte parcial ($p - cutnode$) de G , si existe un par de nodos en u, v en G tal que $u \neq v \neq w$ y $CONN_{G-w}(u, v) < CONN_G(u, v)$.

Incorporación de Información Adaptativa Local en Agrupamiento Difuso para Segmentación de Imágenes (Liu *et al.*, 2015)

En este trabajo se muestran mejoras al algoritmo propuesto por S.P. Chatzis y T.A. Varvarigou, llamado HMRF-FCM debido a los métodos que utiliza: *hidden Markov random field* y *Fuzzy C-Means*. El objetivo es la incorporación de información local a nivel de región al procedimiento difuso de agrupamiento, esto añade un control adaptativo del rango e intensidad de pixeles relacionados. En la primera fase del método propuesto se utiliza el método *Mean Shift* para sobresegmentar la imagen manteniendo discontinuidades.

En el modelo se define una función objetivo como:

$$Q_\lambda = \sum_{i=1}^N \sum_{k=1}^K r_{i,k} D_{i,k} + \lambda \sum_{i=1}^N \sum_{k=1}^K r_{i,k} \log\left(\frac{r_{i,k}}{\pi_{i,k}}\right), \quad (3.9)$$

donde la función de distancia:

$$D_{i,k} = \frac{d_{i,k} + d_{v_i,k}}{2}, \quad (3.10)$$

donde $d_{i,k}$ es la función de disimilaridad a nivel de pixel, v_i es la región que contiene el pixel i y $d_{v_i,k}$ es la función de disimilaridad a nivel de región de la clase k con respecto a la región v_i y está dada por:

$$d_{v_i,k} = \frac{1}{a_{v_i}} \sum_{j \in v_i} d_{j,k}, \quad (3.11)$$

donde a_{v_i} se refiere al número de pixeles en la región v_i .

Se observa que la función objetivo (ecuación 3.9), es una combinación de la función de disimilaridad, dada respectivamente por la ecuación 3.11, y definida en un único pixel que a su vez define la región inicial correspondiente.

En la función objetivo se utiliza una función de probabilidad a priori definida por la siguiente ecuación

$$\pi_{i,k} = \frac{(\sum_{j \in \partial_i} w_j \eta_j r_{j,k})^\gamma}{\sum_{k=1}^K (\sum_{j \in \partial_i} w_j \eta_j r_{j,k})^\gamma}, \quad (3.12)$$

donde el parámetro γ controla la intensidad global entre pixeles vecino y η_j se refiere a un elemento importante en el modelo y que representa la intensidad entre las regiones v_i y v_j :

$$\eta_j = \frac{1}{Z}(E_{v_i, v_j} + 1)^{-1}, \quad (3.13)$$

y donde $E_{v_i, v_j} = \|\mu_{v_i} - \mu_{v_j}\|$, es la distancia euclidea entre los valores promedio de las regiones v_i y v_j , y $Z = \sum_{j \in \partial_i} (E_{v_i, j_i} + 1)^{-1}$ es una constante normalizada.

TurboPixel con Detección Difusa de Bordes y cortes de Grafo de Kernel Multiregional para la Segmentación de Imágenes (Zhang *et al.*, 2018)

Se presenta una integración de dos algoritmos principales: TurboPixel y Cortes de Grafo de Kernel Multiregional. El algoritmo resultante tiene por objetivo principal la segmentación eficiente de imágenes que tienen bajo contraste y un ruido elevado. El algoritmo TurboPixel, con agregada detección difusa de ejes, es utilizado inicialmente para pre-segmentar la imagen en regiones homogéneas. Posterior a un tratamiento con estadísticos de tipo Gaussiano, el método de cortes de grafo de kernel multiregional mapea los datos de la imagen en un espacio de mayor dimensión mediante una función kernel. Este mapeo se utiliza como el modelo de corte de grafo no supervisado para segmentar la imagen.

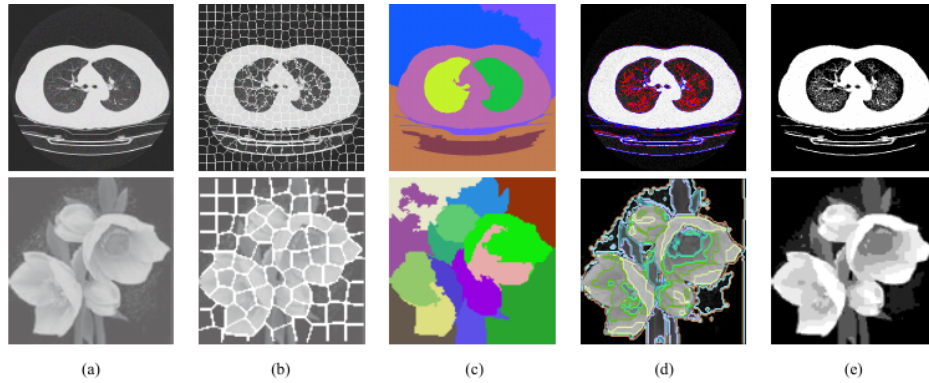


Figura 3.3: Resultados de la segmentación de TurboPixel y Cortes de Grafo de Kernel Multiregional. (a) Imagen original. (b) Resultado de la sobresegmentación. (c) Resultados del agrupamiento. (d) Resultados de la segmentación. (e) Etiquetas finales (Zhang *et al.*, 2018).

Se realizaron experimentos con respecto a otros dos algoritmos, TurboPixel original y Cortes Normalizados con respecto a las medidas de error de segmentación reducida, recuperación de límites y precisión de segmentación alcanzable *Achievable Segmentation Accuracy* (ASA), dando como resultado mejores valores en el algoritmo propuesto.

Con respecto a los resultados de la segmentación, se contrastó con tres algoritmos definidos en Salah *et al.* (2011), Kaba *et al.* (2015) y Hanson y Lundervold (2013)

teniendo mejores resultados en la precisión y la supresión de interferencia de bajo contraste.

Un estudio de Grafos Bipolares Difusos Planos y su Aplicación en la Contracción de Imágenes (Pramanik *et al.*, 2018)

La utilización de grafos polares difusos en el procesamiento de imágenes permite representar los pixeles como vértices y la intensidad y la no intensidad de pixeles como valores de pertenencia positivos y negativos respectivamente. Intuitivamente esto facilita la segmentación.

En el documento se definen los grafos difusos bipolares (BFG), los multi-conjuntos difusos bipolares (BFMS), los multigrafos difusos bipolares (BFMG), los grafos duales difusos bipolares (BFDG) y los grafos planares difusos bipolares (BFPG).

Se presenta un ejemplo de segmentación basada en el uso de grafos planares difusos en la Figura 3.4, donde la imagen entera en a) es convertida a un BFG, en b), y en la que cada vértice coloreado se corresponde con la porción del mismo color en la imagen.

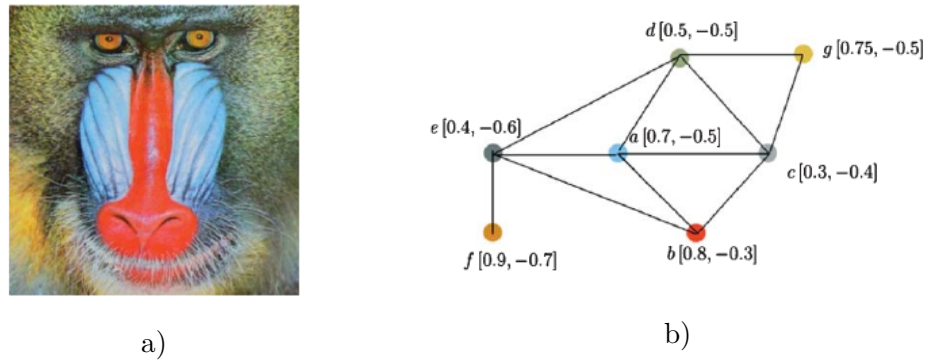


Figura 3.4: Ejemplo de conversión de una imagen a un grafo difuso planar bipolar. a) Ejemplo de imagen a color. b) Grafo difuso planar bipolar (BFPG) de a) (Pramanik *et al.*, 2018).

Los valores de pertenencia de las aristas del grafo en b) se muestran en la tabla 3.1.

Tabla 3.1: Valores de pertenencia de las aristas del grafo mostrado en la Figura 3.4 (Pramanik *et al.*, 2018).

Edge	(a, b)	(a, c)	(a, d)	(a, e)	(b, c)
Membership Value	[0.7, -0.3]	[0.3, -0.4]	[0.5, -0.5]	[0.4, -0.5]	[0.3, -0.3]
Edge	(b, e)	(c, d)	(c, g)	(d, e)	(d, g)
Membership Value	[0.4, -0.3]	[0.3, -0.4]	[0.3, -0.4]	[0.4, -0.5]	[0.5, -0.5]
Edge	(e, f)				
Membership Value	[0.4, -0.6]				

Este resultado se logra mediante el procesamiento del BFG, disminuyendo la cantidad de aristas mediante un proceso de contracción de las mismas.

Los resultados de la contracción se muestran, como pirámide de imágenes, en la Figura 3.5 (seccionada por motivos de espacio).

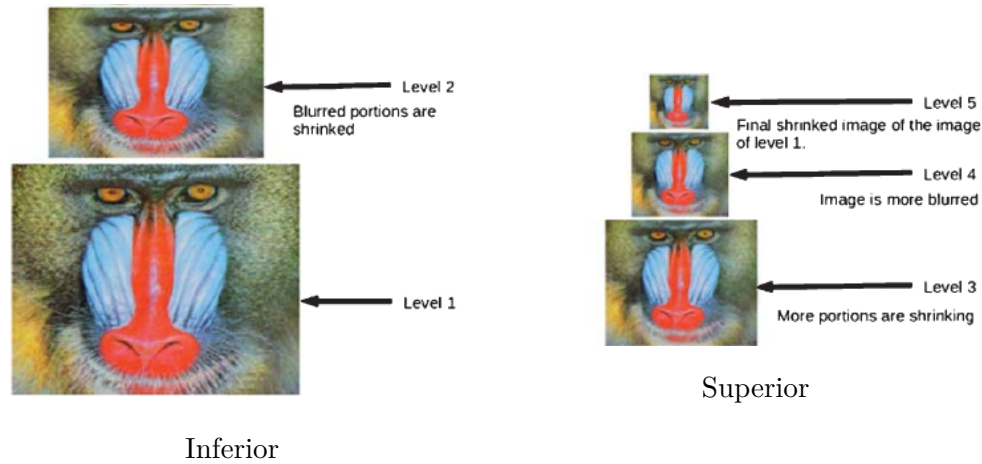


Figura 3.5: Pirámide de imágenes, resultado final de la reducción basada en grafos difusos polares (Pramanik *et al.*, 2018).

Algoritmo de Supervoxel para Procesamiento de Imágenes Médicas (Tamajka y Benešová, 2018)

El documento describe un método de sobresegmentación en supervoxeles para procesamiento de datos volumétricos médicos y que incorpora distancias físicas entre voxeles. La función de costo utilizada posee propiedades de preservación de bordes, homogeneidad y términos regulares.

$$\text{Generación de } \left\{ \begin{array}{l} \text{Basados en grafos} \\ \text{superpíxeles y supervoxeles} \end{array} \right. \left\{ \begin{array}{l} \text{Basados en gradiente ascendente} \end{array} \right.$$

Figura 3.6: Algoritmos de generación de superpíxeles y supervoxeles

Los supervoxeles realizan una reducción de la complejidad en datos volumétricos. Estos son agrupados en grandes cantidades para ser utilizados como una unidad atómica en procesamientos posteriores. Uno de los algoritmos generadores de supervoxeles (y superpíxeles) más utilizados es SLIC, basado en una modificación al algoritmo K-Means.

Un enfoque novedoso de segmentación pulmonar en imágenes de CT del pecho utilizando cortes de grafos

Este trabajo muestra un algoritmo basado en un anterior algoritmo mejorado de corte de grafos utilizando *modelos de mezcla Gaussiana* (GMMs) y donde el peso o probabilidad que cada píxel pertenezca al *foreground* se calcula mediante el *algoritmo*

esperanza-maximización (EM). Un grafo es generado a partir de los pesos calculados y con este se realiza la segmentación aplicando cortes mínimos.

Operaciones en Estructuras de Grafos Difusos Intuicionistas (Akram y Akmal, 2017).

Se introduce el concepto de *estructura de grafo difuso intuicionista* y algunas operaciones con dichas estructuras.

Definición 11. (Akram y Akmal, 2017) Un *conjunto difuso intuicionista* (IFS) en un universo X es un objeto de la forma

$$A = \{ \langle x, \mu_A(x), \nu_A(x) \rangle \mid x \in X \}, \quad (3.14)$$

donde $\mu_A(x) \in [0, 1]$ es llamado el grado de pertenencia de $x \in A$, $\nu_A(x) \in [0, 1]$ es llamado el grado de no-pertenencia de $x \in A$, y μ_A y ν_A satisfacen la siguiente condición: para todo $x \in X$, $\mu_A(x) + \nu_A(x) \leq 1$.

Definición 12. (Akram y Akmal, 2017) Una *relación difusa intuicionista* $R = (\mu_R(x, y), \nu_R(x, y))$ en un universo $X \times Y$ ($R(X \rightarrow Y)$) es un conjunto difuso intuicionista de la forma

$$R = \{ \langle (x, y), \mu_R(x, y), \nu_R(x, y) \rangle \mid (x, y) \in X \times Y \} \quad (3.15)$$

donde $\mu_R : X \times Y \rightarrow [0, 1]$ y $\nu_R : X \times Y \rightarrow [0, 1]$. La relación intuicionista R satisface $\mu_R(x, y) + \nu_R(x, y) \leq 1$ para todo $x, y \in X$.

Definición 13. (Akram y Akmal, 2017) Sea $G^* = (U, E_1, E_2, \dots, E_k)$ un grafo estructurado y $\nu, \rho_1, \rho_2, \dots, \rho_k$ los subconjuntos difusos de U, E_1, E_2, \dots, E_k , respectivamente tal que

$$0 \leq \rho_i(xy) \leq \rho(x) \wedge \nu(y), \quad \text{para todo } x, y \in U \text{ y } i = 1, 2, \dots, k. \quad (3.16)$$

Entonces $G = (\nu, \rho_1, \rho_2, \dots, \rho_k)$ es una *estructura de grafo difuso* de G^* .

Definición 14. (Akram y Akmal, 2017) Sea $G = (\nu, \rho_1, \rho_2, \dots, \rho_k)$ una estructura de grafo difuso de una estructura de grafo $G^* = (U, E_1, E_2, \dots, E_k)$. Entonces $F = (\nu, \tau_1, \tau_2, \dots, \tau_k)$ es una *estructura de subgrafo extendido parcial difuso* de G si, $\tau_i \subseteq \rho_i$ para $i = 1, 2, \dots, k$.

Definición 15. (Akram y Akmal, 2017) Sea $\{E_i : i = 1, 2, \dots, n\}$ un conjunto de relaciones irreflexivas, simétricas y mutuamente disjuntas en un conjunto no vacío U . Una *estructura de grafo difuso intuicionista* (IFGS) con el conjunto de vértices subyacentes U está determinado por $\check{G}_s = (A, B_1, B_2, \dots, B_n)$ donde

- (i) A es un conjunto difuso intuicionista de U con $\mu_A : U \rightarrow [0, 1]$ y $\nu_A : U \rightarrow [0, 1]$, nombrados el grado de pertenencia y el grado de no-pertenencia de $x \in U$, respectivamente, tal que

$$0 \leq \mu_A(x) + \nu_A(x) \leq 1, \quad \text{para todo } x \in U.$$

- (II) Cada B_i es un conjunto difuso intuicionista de E_i tal que las funciones $\mu_{B_i} : E_i \rightarrow$ y $\nu_{B_i} : E_i \rightarrow [0, 1]$ están definidas por

$$\mu_{B_i}(xy) \leq \mu_A(x) \wedge \mu_A(y), \quad \nu_{B_i}(xy) \leq \nu_A(x) \vee \nu_A(y)$$

y

$$0 \leq \mu_{B_i}(xy) + \nu_{B_i}(xy) \leq 1, \quad \text{para todo } xy \in \subset U \times U, i = 1, 2, \dots, n.$$

Sobre estructuras m -polares basadas en grafos difusos. (Akram *et al.*, 2016).

Se introduce la noción de *estructura m -polar basada en grafos difusos* y diversas operaciones sobre dicha estructura. La aproximación de estructura de grafo (introducida a su vez por Sampathkumar (2006)) permite el análisis de relaciones en forma simultánea a la vez que las aristas correspondientes. El enfoque m -polar extiende esta idea a problemas que involucran multi-agentes, multi-atributos, multi-objetos, multi-índices e información multipolar.

El enfoque m -polar se deriva a partir del trabajo de Zhang (2008) en la extensión de los conjuntos difusos hacia el de *conjuntos difusos bipolares* como una transformación que asocia cada elemento de X a un número real en el intervalo $[-1, 1]$. En 2014, Chen *et al.* (2014) introduce la idea de conjuntos difusos m -polares.

Definición 16. (Akram *et al.*, 2016) $\check{G}_b = (M, N_1, N_2, \dots, N_n)$ es llamada una *estructura de grafo difuso bipolar* (BFGS) de una estructura de grafo (GS) $G^* = (U, E_1, E_2, \dots, E_n)$ si $M = (\mu_M^P, \mu_M^N)$ es un conjunto difuso en U y para cada $i = 1, 2, \dots, n$, $N_i = (\mu_{N_i}^P, \mu_{N_i}^N)$ es un conjunto bipolar difuso en E_i tal que

$$\mu_{N_i}^P(xy) \leq \mu_M^P(x) \wedge \mu_M^P(y), \quad \mu_{N_i}^N(xy) \geq \mu_M^N(x) \vee \mu_M^N(y) \quad \forall xy \in E_i \subset U \times U \quad (3.17)$$

Es de notar que $\mu_{N_i}^P(xy) = 0 = \mu_{N_i}^N(xy)$ para todo $xy \in U \times U - E_i$ y $0 < \mu_{N_i}^P(xy) \leq 1$, $-1 \leq \mu_{N_i}^N(xy) < 0 \forall xy \in E_i$, donde U e $E_i (i = 1, 2, \dots, n)$ son llamados *conjunto de vértices subyacentes* y *conjunto de i -aristas subyacentes* de \check{G}_b , respectivamente.

Definición 17. (Chen *et al.*, 2014) Un *conjunto difuso m -polar* (o un conjunto $[0, 1]^m$) en X es exactamente una transformación $A : X \rightarrow [0, 1]^m$.

Definición 18. (Akram y Waseem, 2016), (Chen *et al.*, 2014) Un *grafo difuso m -polar* es un par $G = (C, D)$, donde $C : V \rightarrow [0, 1]^m$ es un conjunto difuso polar en V y $D : V \times V \rightarrow [0, 1]^m$ es una relación difusa m -polar en V tal que:

$$p_i \circ D(xy) \leq \inf p_i \circ C(x), p_i \circ C(y) \quad (3.18)$$

para todo $x, y \in V$.

Definición 19. (Akram *et al.*, 2016) Sea $G^* = (U, E_1, E_2, \dots, E_n)$ una estructura de grafo (GS). Sea C un conjunto difuso m -polar en U y D_i un conjunto difuso m -polar en E_i tal que

$$p_j \circ D_i(xy) \leq \inf\{p_j \circ C(x), p_j \circ C(y)\} \tag{3.19}$$

para todo $x, y \in U, i \in n, j \in m$ y $p_j \circ D_i(xy) = 0$ para $xy \in U \times U \setminus E_i, \forall j$. Entonces $G_{(m)} = (C, D_1, D_2, \dots, D_n)$ es llamada una *estructura de grafo difuso m -polar* (m -PFGS) en G^* donde C es el conjunto de vértices m -polar de $G_{(m)}$ y D_i es el conjunto de i -aristas m -polar de $G_{(m)}$.

En la Figura 3.7 se muestra una estructura de grafo difuso m -polar $G_{(m)} = (C, D_1, D_2)$ de la estructura de grafo $G^* = (U, E_1, E_2)$, tal que $U = \{a_1, a_2, a_3, a_4\}$, $E_1 = \{a_1a_2\}$ y $E_2 = \{a_3a_2, a_2a_4\}$.

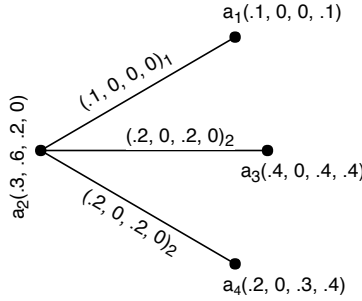


Figura 3.7: Un ejemplo de estructura de grafo difuso 4-polar.

Se concluye que la estructura de grafo es una útil herramienta de apoyo en la solución de problemas de tipo combinatorios. Se caracteriza por la posibilidad de análisis de las relaciones/aristas y los correspondientes vértices en forma simultánea.

Método basado en Cortes de Grafos para la Segmentación del Ventrículo Izquierdo en Imágenes MRI o ecocardiográficas (Bernier *et al.*, 2017).

Se describe un método de segmentación 3D de la pared endocardial del ventrículo izquierdo del corazón basado en un espacio Bezier en grafo 3D (basado a su vez en gradientes y mapa de probabilidad).

Uno de los principales aportes es que el método propuesto trabaja indistintamente con MRI o imágenes ecocardiográficas. Es de notar que ambos esquemas de adquisición de imágenes poseen características muy distintas lo cual refleja la efectividad del método propuesto.

El método consta de 4 fases:

1. La deformación de la imagen en espacio Euclidiano al espacio Bezier (ver Figura 3.8)
2. El cálculo de los valores de gradiente y probabilidad de la energía en un grafo.

3. La localización de la pared endocárdica mediante un algoritmo min-cut/max-flow. Cabe mencionar que se consiguió una rapidez de localización del orden de 2–5 s.
4. El posprocesamiento de la malla resultante.

Como se muestra en la Figura 3.8, se utilizó una curva de revolución Bezier de tres grados alrededor del eje del ventrículo izquierdo. Los puntos A, B, C (y opcionalmente D y E) son marcados por el usuario formando una T (*T-bar*). La curva Bezier es revolucionada a lo largo del eje A–C (Figura 3.8(b))

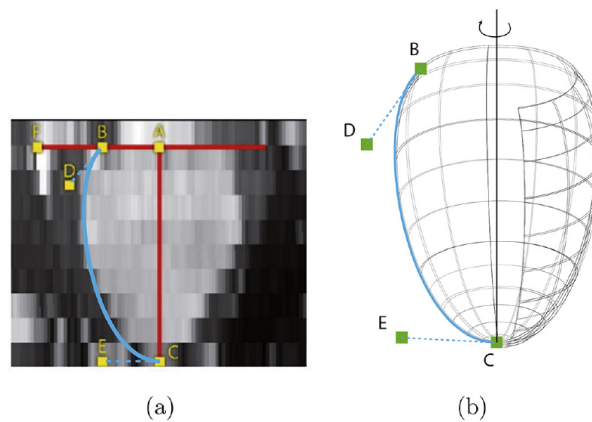


Figura 3.8: (a) Curva Bezier 2D (en azul), hecha por los 4 puntos de control B, D, E y C. La curva en rotación a lo largo del objetivo es ilustrada en el volumen 3D mostrado en negro en (B) (Bernier *et al.*, 2017)

De especial interés para el tema de tesis es la utilización en este método, de un algoritmo de corte de grafos descrito en (Boykov y Kolmogorov, 2004) y que se implementa en la función de optimización para encontrar el corte mínimo.

El hecho que la función de energía esté basado mayormente en un gradiente espacial implica limitaciones. Por ello el resultado final agrega un mapa de probabilidad del LV que expresa qué tan probable se encuentra un voxel de estar dentro del LV con respecto a su distancia de la *T-bar*.

El método muestra mejoras con respecto a algoritmos anteriores y fue probado en las bases de datos de imágenes ecocardiográficas CETUS y un conjunto de datos generados en MRI. Por ejemplo, para distancia de superficie media, el volumen diastólico final observa mejores resultados con respecto los participantes del CETUS MICCAI 2014 challenge.

En la Figura 3.9 se observa el resultado del corte de grafo marcado por la línea roja en b).

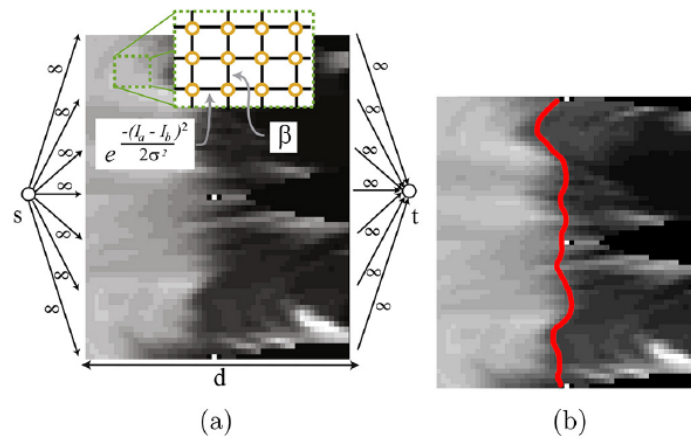


Figura 3.9: (a) Construcción del grafo por encima del espacio Bezier de la imagen MRI en la cual cada voxel es un nodo conectado a sus 6 vecinos. (b) El resultado del corte, la cavidad del ventrículo izquierdo se observa en la fracción izquierda del corte. (Bernier *et al.*, 2017)

3.2.1. Discusión del Estado del Arte

No obstante el desarrollo temprano de la base matemática de grafos difusos, la aplicación práctica en el procesamiento de imágenes ha sido reciente. A modo de introducción se citan trabajos anteriores con propósitos de contextualizar el desarrollo reciente.

La noción básica de grafo difuso fue propuesta por Kauffman (1973). Rosenfeld (1975) y Yeh y Bang (1975) desarrollaron sobre ello la teoría de grafos difusos. Zhang (1998) generalizó los conjuntos difusos desarrollando los conjuntos difusos bipolares. Como una extensión importante, los grafos difusos m-polares, fueron abordados en (Akram *et al.*, 2016). Los grafos planares difusos bipolares fueron presentados en (Pramanik *et al.*, 2018) donde se utilizaron para la contracción de imágenes.

El procesamiento imágenes basado en grafos espectrales fue especificado en (Cheung *et al.*, 2018) como el tratamiento de un grafo como señal en el dominio espectral (valores y vectores propios, o *eigenvalores* y *eigenvectores*).

En la presente propuesta se revisan dos artículos relacionados a la generación de superpíxeles. En (Zhang *et al.*, 2018) se integraron dos algoritmos: TurboPixel y Cortes de Grafo de Kernel Multiregional para lograr una sobresegmentación eficiente. En (Tamajka y Benešová, 2018) fue especificado un método de sobresegmentación en supervoxeles que incorpora distancias físicas entre voxeles.

Tabla 3.2: Resumen del Estado del Arte

Artículo	Objetivo	Método/Algoritmo	Resultado	Utilidad
Superpíxeles: Una Evaluación del Estado del Arte (Stutz <i>et al.</i> , 2018)	Evaluación de 28 algoritmos de generación de superpíxeles	Se utilizaron las métricas de desempeño: recuperación de límites, error de subsegmentación, variación explicada y compacidad.	A nivel visual todos los algoritmos mostraron buena eficiencia. No fue posible discriminar basado en desempeño.	Proporciona referencias valiosas a tomar en cuenta para la evaluación de algoritmos de superpíxeles
Aplicación de un Algoritmo Modificado de Agrupamiento <i>Fuzzy C-Means</i> en Imágenes Médicas (Aljebory y Mohammed, 2018)	Segmentación de imágenes médicas	Modificación a un algoritmo <i>Fuzzy C-Means</i> espacial (MFCM)	Mejor tratamiento del ruido. Regiones más homogéneas con respecto a FCM y MFCM	El tratamiento de la información de vecindad espacial
Modelo de Calidad de Información Teórica para Segmentación de Imágenes (Murashov, 2017)	Seleccionar los mejores parámetros de un algoritmo	Selección mediante una medida de redundancia propuesta	El método selecciona correctamente la mejor imagen/parámetros.	Validación de resultados

Artículo	Objetivo	Método/Algoritmo	Resultado	Utilidad
Complejidad y Regularidad en Grafos Difusos Generalizados (Samanta <i>et al.</i> , 2016)	Presenta los grafos difusos generalizados	Grafos difusos	Se logra una representación eficiente de grafos con un enfoque difuso	Representación difusa de grafos
Procesamiento de Imágenes basado en Grafos Espectrales (Cheung <i>et al.</i> , 2018)	Grafo como señal	Procesamiento de señales basado en grafos (GSP)	Tratamiento de un grafo como señal para su procesamiento	Procesamiento del grafo como señal
Un estudio de grafos con etiquetado difuso (Borzooei <i>et al.</i> , 2016)	Conectividad en grafos mediante etiquetado difuso	Grafos difusos con etiquetado difuso	Se definen múltiples operaciones, incluyendo algunos tipos de cortes	Una representación alternativa de grafos difusos
Incorporación de Información Adaptativa Local en Agrupamiento Difuso para la Segmentación de Imágenes (Liu <i>et al.</i> , 2015)	Incorporación de información local al procedimiento difuso de segmentación	<i>Hidden Markov Random Field – Fuzzy C-Means</i>		El análisis que realiza para agregar la información local

Artículo	Objetivo	Método/Algoritmo	Resultado	Utilidad
Un enfoque novedoso de segmentación pulmonar en imágenes de CT del pecho utilizando cortes de grafos.	Segmentación pulmonar en imágenes.	Corte de grafos con <i>modelos de mezcla Gaussiana</i> (GMMs). La probabilidad de pertenencia al <i>foreground</i> se calcula mediante <i>esperanza-maximización</i> .		
Grafos Planares Difusos	Representación de imágenes mediante grafos planares difusos	Grafo planar difuso	Representación eficiente y difusa de la imagen	Conveniencia del método
Sobre Estructuras <i>m</i> -polares Basadas en Grafos Difusos (Akram <i>et al.</i> , 2016)	Se introduce la noción de estructura <i>m</i> -polar basada en grafos difusos y operaciones	Estructura <i>m</i> -polar basada en grafos difusos	La estructura <i>m</i> -polar descrita es de alta utilidad en problemas combinatorios	Una estructura útil de representación de grafos difusos y operaciones
TurboPixel con Detección Difusa de Bordes y Cortes de Grafo de Kernel Multiregional para la Segmentación de Imágenes (Zhang et al. 2017).	Mejorar la segmentación de imágenes con bajo contraste y ruido elevado.	Extensión difusa de TurboPixel y <i>kernel graph cuts</i> . Kernel Fuzzy C-Means. Estadísticos Gaussianos.	Rendimiento mejorado con respecto a cortes normalizados y TurboPixel original.	Métodos de supresión de ruido y bajo contraste.

Artículo	Objetivo	Método/Algoritmo	Resultado	Utilidad
Método basado de Cortes de Grafos para la Segmentación del Ventrículo izquierdo en Imágenes MRI o ecocardiográficas (2017).	Segmentación 3D de la pared endocardial del ventrículo izquierdo del corazón.(LV)	Espacio Bezier en grafo 3D (basado en gradiente y mapa de probabilidad). Max-flow/min-cut.	Identificación de la pared endocárdica en 3D en 2-5 s. Sin agujeros. Efectivo en imágenes ecocardiográficas y MRI	Segmentación del LV basada en corte de grafos.

4

Metodología

Los métodos de segmentación y (sobresegmentación) basados en grafos se apoyan en agrupamiento perceptual de imágenes e información espacial (Khokher *et al.*, 2012). Aunado a ello, la teoría de conjuntos difusos permite capturar la incertidumbre en la pertenencia de un pixel a una región o superpixel dado. El algoritmo propuesto aprovecha las ventajas de ambas para buscar la eficiencia en el proceso de segmentación de imágenes.

4.1. Algoritmo de sobresegmentación propuesto

En la presente subsección se describen los detalles del algoritmo propuesto de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color. El algoritmo se apoya en conceptos y desarrollo de los trabajos en (Domínguez, 2019) y (Achanta y Susstrunk, 2017).

4.1.1. Algoritmo de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color

En forma inicial el algoritmo propuesto realiza una conversión de la imagen desde el espacio de color RGB hacia CIELAB en los tres canales. La razón de dicha elección es que esta última representación es ampliamente considerada como perceptualmente uniforme para distancias pequeñas de color (Achanta *et al.*, 2010). El resultado de esta conversión es utilizado como característica de la imagen para la construcción de un grafo difuso $\xi = (V, \sigma, \mu)$. Donde σ es la función que define el conjunto difuso de vértices y μ es la función que define el conjunto difuso de aristas.

El conjunto de vértices difusos σ , queda entonces determinado por una función monótonica que transforma el valor entero de intensidad de color de cada pixel, en

un valor de tipo real en el rango $[0, 1]$, $\sigma : \mathbb{N} \rightarrow \mathbb{R}$:

$$\sigma_i = \frac{x_i - \text{mín}(X)}{\text{máx}(X) - \text{mín}(X)} \quad (4.1)$$

Cada σ en la definición anterior es un conjunto unitario (*singleton*) difuso cuyo valor, en el trabajo realizado, se especificó como la conversión al dominio difuso de la intensidad de color para cada canal.

Se utilizó el mismo esquema de inicialización de semillas, con $C[k] = \{\sigma_k, c_k\}$, presentado en (Achanta *et al.*, 2010). En este las mismas se generan en los centros de una cuadrícula a intervalos de $S = \sqrt{N/K}$. Donde N es el número total de píxeles de la imagen y K el número deseado de superpíxeles.

Utilizando dichas semillas, se generan K elementos $e_i = \{\sigma_i, \mathbf{c}_i, k, d_{i,k}\}$ donde a cada k se le asigna una etiqueta identificadora del superpixel entre 1 y K y cada $\mu_{j,k}$ representando la distancia del pixel hacia el k -centroide e inicialmente en cero. Un árbol de prioridad Q es inicializado con estos K elementos. Para cada operación *Pop*, Q siempre retorna el elemento e_i cuya distancia $\mu_{j,k}$ al k -centroide es la menor.

Para la función μ , que define la relación en σ y por tanto las aristas del grafo difuso, se utilizó una distancia definida por:

$$d_{\mu_{i,k}} = \sqrt{\frac{\|\mathbf{x}_i - \mathbf{x}_k\|_2^2}{s} + \frac{\|\mathbf{c}_i - \mathbf{c}_k\|_2^2}{m}} \quad (4.2)$$

donde $\mathbf{x} = [xy]^T$ es la posición espacial y $\mathbf{c} = [l a b]^T$ el color CIELAB de cada pixel y los valores s y m (compacidad, *compactness*) son factores de normalización espacial y de distancia de color respectivamente. El algoritmo puede prescindir del valor de m o definirlo como muy pequeño.

Mientras $Q \neq \emptyset$, el elemento de mayor orden es extraído. Si la posición del pixel en el mapa L de etiquetas a que apunta el elemento no está etiquetado, se le asigna la etiqueta del centroide. El valor del centroide, que es el promedio de todos los píxeles en el superpixel, es actualizado con este pixel. Para cada uno de sus 4 vecinos que no han sido aún etiquetados se genera un nuevo elemento asignándole la distancia desde el centroide conectado y la etiqueta del centroide. Estos nuevos elementos son agregados a la cola Q .

Conforme la ejecución avanza los elementos son extraídos y Q es vaciada. El algoritmo termina hasta que no existan píxeles restantes sin etiquetas y Q se encuentre vacía.

El proceso descrito está ilustrado en la Figura 4.1. Algunas ecuaciones y fases menores han sido omitidas por claridad.

Formalmente, el listado mostrado en el Algoritmo 5 detalla el procedimiento explicado con anterioridad.

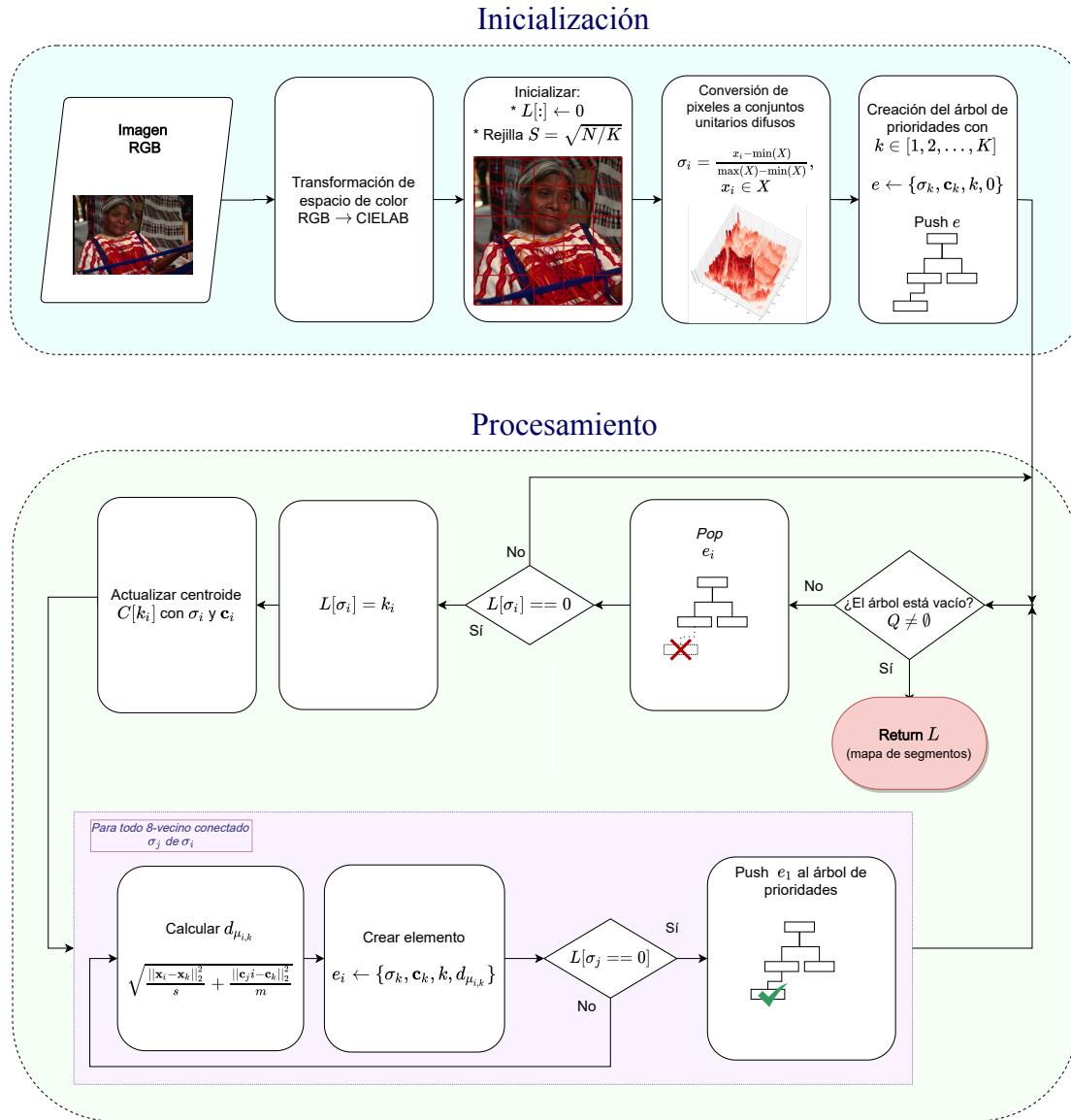


Figura 4.1: Esquema de funcionamiento del *algoritmo de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color.*

Algoritmo 5 Algoritmo de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color.

Input: Imagen I , K centroides iniciales $C[k] = \{\mathbf{x}_k, \mathbf{c}_k\}$,

Output: Mapa L de segmentación en superpíxeles

```

1: RGB  $\rightarrow$  CIELab
2: Inicializar  $L[:] \leftarrow 0$ 
3:  $\sigma_i = \frac{x_i - \min(X)}{\max(X) - \min(X)}$ ,  $x_i \in X$  ▷ Función monótonica difusa
4: for  $k \in [1, 2, \dots, K]$  do
5:   Elemento  $e \leftarrow \{\sigma_k, \mathbf{c}_k, k, 0\}$ 
6:   Push  $e$  en grafo de prioridad  $Q$ 
7: end for
8: while  $Q \neq \emptyset$  do
9:   Pop  $Q$  para  $e_i$ 
10:  if  $L[\sigma_i] == 0$  then
11:     $L[\sigma_i] = k_i$ 
12:    Actualizar centroide  $C[k_i]$  con  $\sigma_i$  y  $\mathbf{c}_i$ 
13:    for all Vecino conectado  $\sigma_j$  de  $\sigma_i$  do
14:      Calcular la distancia  $\mu_{j,i} = \sqrt{\frac{\|\sigma_j - \sigma_i\|_2^2}{s} + \frac{\|\mathbf{c}_j - \mathbf{c}_i\|_2^2}{m}}$  ▷ Distancia difusa euclídea
15:      Crear elemento  $e_j = \{\sigma_j, \mathbf{c}_j, k_i, \mu_{j,i}\}$ 
16:      if  $L[\sigma_j] == 0$  then
17:        Push  $e_j$  en  $Q$ 
18:      end if
19:    end for
20:  end if
21: end while

```

5

Experimentación y resultados

Se describen los detalles relacionados a la experimentación realizada con el algoritmo propuesto con respecto a las métricas Recuerdo de límites (*Rec*), Error de subsegmentación (*UE*) y Variación explicada (*EV*). Los resultados obtenidos de las mediciones son mostrados en ambos formatos de tablas y figuras con la finalidad de visualización y contraste de métricas y algoritmos a mas de verificación.

5.1. Descripción general

Cada una de las 200 imágenes del subconjunto *test* en el repositorio *BSDS500* fue procesada por los 7 algoritmos a comparar y para cada uno de 16 valores de superpíxeles deseados (con K igual a 100, 200, 300, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000, 2400, 2800, 3200, 3600). En total se sobresegmentaron $200 \times 7 \times 16 \approx 25600$ imágenes. Cada una de estas imágenes fue contrastada con las 5 segmentaciones del *ground truth* dando un total de $25600 \times 5 = 128000$ comparaciones totales.

Es necesario clarificar que el número de superpíxeles a generar para la evaluación fue elegido a partir del análisis de los resultados de (Stutz *et al.*, 2018) y (Domínguez, 2019), en que se observa un rápido cambio en valores de las métricas *Rec*, *UE* y *EV* para cantidades de superpíxeles menores a 1000 o 1500 y un cambio menor hasta aproximadamente 3600, siendo después casi imperceptible. Para el presente trabajo, las Figuras 5.3 a 5.5 ilustran este hecho y resultado.

El objetivo primario de la presente experimentación es el de comparar los valores de las métricas obtenidas para cada resultado, por imagen y algoritmo, con las 5 segmentaciones del *ground truth* disponibles, obteniendo un gran promedio que describe la relación métrica-número de superpíxeles por cada algoritmo y que permite contrastar la eficiencia del mismo con respecto a otros en el estado del arte. Este resultado se describe en los resultados cuantitativos.

Un objetivo secundario es el de contrastar la calidad visual de la sobresegmentación

generada por los algoritmos para cada imagen. Se busca en ello una mayor adherencia a los bordes de la imagen. El comparativo se discute como resultados cualitativos.

Se llevó a cabo una segunda etapa de experimentación con imágenes de los repositorios *BSDS500* y *Weizmann* en que se utilizaron los resultados de la etapa de sobresegmentación con el algoritmo propuesto. Es importante señalar que esta parte de la experimentación no fue considerada en los objetivos del trabajo de tesis y se ha realizado únicamente para ejemplificar la eficiencia de los métodos basados en grafos y la obtención de una segmentación regular a partir de los resultados de una sobresegmentación.

El procesamiento de las 200 imágenes del subconjunto *test* del repositorio *BSDS500* y del posprocesamiento mediante grafos de adyacencia de regiones (RAG) fue realizado en una computadora convencional (laptop) con 16G de memoria RAM y 4 núcleos, en un sistema Debian GNU/Linux. En dicho sistema se crearon algunas utilerías en *shell* (bash) para la ejecución de los programas finales sobre el conjunto de imágenes y para el renombrado por lotes de archivos; se utilizó el compilador GCC (versión 9.3.0), para la compilación de los programas creados en C; se crearon *scripts* en Matlab 2018b para el procesamiento y extracción de los *ground truth* del repositorio *BSDS500* y para el comparativo de la subsección de resultados cualitativos; y se crearon *scripts* en lenguaje Python (versión 3.8) para tareas finales de procesamiento del algoritmo propuesto y del tratamiento mediante RAG. Se han utilizado poco más de 111 Gb de espacio para el almacenamiento de las imágenes sobresegmentadas resultantes (en formato PNG) y para los datos obtenidos durante la ejecución de algoritmos y comparaciones (en formato CSV).

5.2. Conjuntos de datos

Para las presente experimentación fueron utilizados dos repositorios de imágenes: *Berkeley Segmentation Dataset* (BSDS500, (Arbelaez *et al.*, 2011)) y *Weizmann Segmentation Evaluation Database*, (Weizmann, (Alpert *et al.*, 2007)).

El repositorio *BSDS500* fue utilizado en la evaluación de la sobresegmentación del algoritmo propuesto y con respecto a los otros 6 algoritmos. En particular se utilizaron para evaluar tres métricas relacionadas a la adherencia a bordes.

El repositorio *Weizmann* ha sido usado para demostrar una etapa posterior a la generación de superpíxeles (o sobresegmentación) y que consiste en una segmentación por objetos basada en grafos de adyacencia de regiones.

5.2.1. Berkeley Segmentation Dataset (BSDS500)

El objetivo de este conjunto es el de proveer una base empírica para la investigación en segmentación de imágenes y detección de bordes (Arbeláez *et al.*, 2011). Está comprendida por 500 imágenes a color en 3 conjuntos utilizados en etapas específicas

durante un proceso de segmentación de imágenes: *train*, *test* y *eval*. Consta de diversas escenas al aire libre, paisajes, edificios, personas y animales en las que es distinguible (humanamente) el objeto del fondo. Se anexan al menos 5 segmentaciones *ground truth* en promedio por imagen.

En el repositorio original, las (al menos 5 y hasta 8) segmentaciones del *ground truth* están contenidas en un único archivo Matlab (.mat) por imagen; fue necesario convertir cada segmentación del *ground truth* a un archivo individual de datos separados por comas (.csv) para su procesamiento respectivo, comprendiendo un total de 1063 archivos finales con etiquetas del *ground truth*.

Se incluye, además, código para la evaluación del desempeño de 3 algoritmos disponibles en el mismo repositorio y descritos en (Arbelaez *et al.*, 2011).

La Figura 5.1 ejemplifica imágenes de dicho repositorio del subconjunto *train* con 5 respectivas segmentaciones proporcionadas.

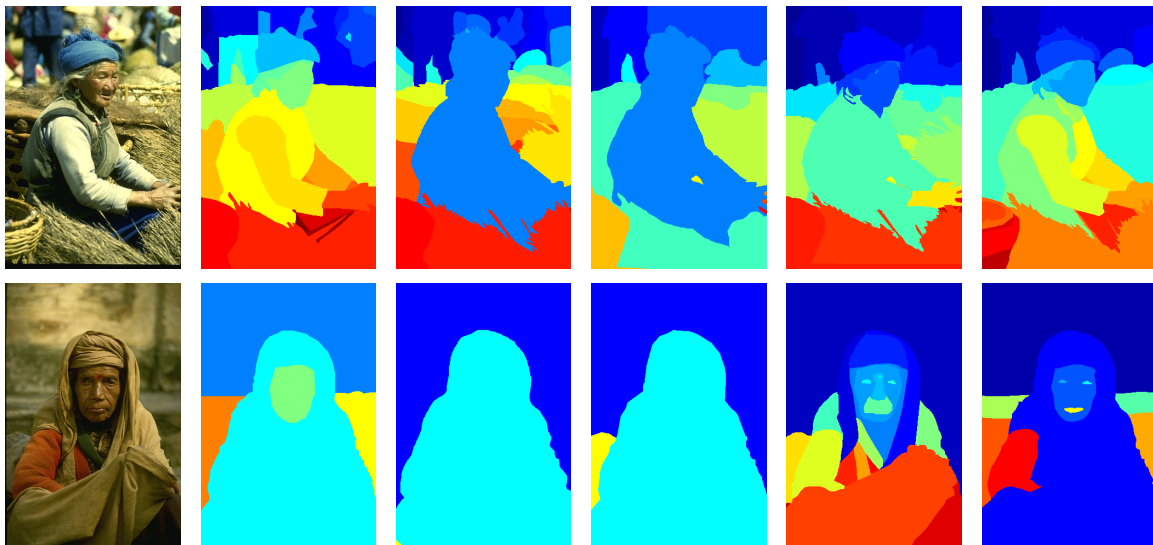


Figura 5.1: Ejemplo de imágenes en el Repositorio *BSDS500*. La primera imagen es la original, le siguen las 5 imágenes que (al menos) se incluyen como *ground truth*.

5.2.2. Weizmann Segmentation Evaluation Database

La Base de datos para evaluación de Segmentación *Weizmann* (o repositorio *Weizmann*) tiene por propósito proveer una base tanto empírica como científica para la investigación en segmentación de imágenes. Contiene 100 imágenes en que aparece un objeto y 100 en las que se deben distinguir dos objetos (segmentación por instancias). Cada imagen por categoría dispone de dos a tres segmentaciones producidas por humanos, la imagen en escala de grises y una imagen a color. La base de datos *Weizmann* se encuentra a disposición pública por el *Weizmann Institute of Science* únicamente con fines de investigación (Alpert, 2007).

La segmentación es evaluada mediante la determinación de su consistencia en contraste con la segmentación de *ground truth* y la cantidades de fragmentación. Se provee igualmente código para la evaluación del algoritmo descrito en (Alpert *et al.*, 2007).

La Figura 5.2 ejemplifica imágenes de dicho repositorio en ambos subconjuntos.

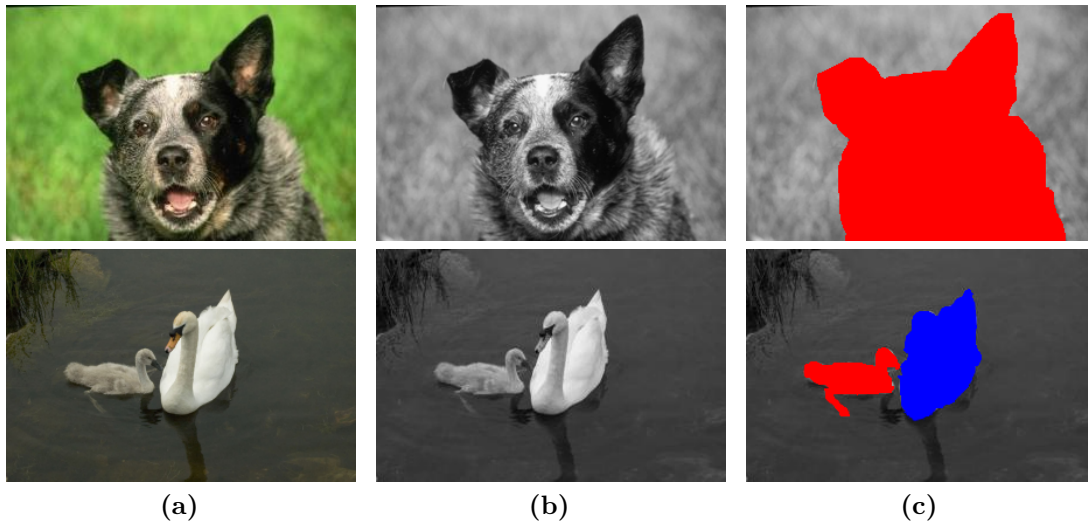


Figura 5.2: Ejemplo de imágenes en el Repositorio Weizmann. La columna (a) muestra la imagen a color; (b) imagen en niveles de gris; (c) una imagen (de tres incluidas en el repositorio) del *ground truth*.

5.3. Resultados

Se presentan los productos de la experimentación comparativa entre los 7 algoritmos analizados en forma de tablas, gráficas y conclusiones.

5.3.1. Resultados del algoritmo propuesto

Resultados cuantitativos

Para las métricas *Rec*, *UE* y *EV*, las tablas 5.1–5.3, respectivamente, presentan los resultados numéricos (con K aproximado) con que se generaron las gráficas presentadas¹. En todas las gráficas se utiliza el siguiente esquema de líneas / colores para identificarlos: —|— CRS, —x— ERGC, —*— ERS, —□— Propuesto, —■— SEEDS, —○— SLIC, —●— W e

¹Por diseño de los mismos, la cantidad de superpíxeles generados por cada algoritmo no es exactamente igual al valor requerido K , por ello las tablas se muestran como $k \approx$ y podrían aparentar divergencia con lo mostrado en las gráficas, situación inexistente.

IFCM $-\Delta-$. A su vez, las Figuras 5.3–5.5 muestran los resultados comparativos de los 6 algoritmos seleccionados con respecto al propuesto.

Métrica Recuerdo de límites (*Rec*)

Como se describió en la subsección 2.5, la adherencia a bordes descrita en este documento por la métrica Recuerdo de límites (*Rec*) mide la capacidad del algoritmo de ajustarse a los bordes definidos en el *ground truth*. La Tabla 5.1 y la Figura 5.3 presentan los resultados obtenidos de la aplicación de los 6 algoritmos descritos en la sección 2.6, así como del algoritmo propuesto.

Se observa en la Tabla 5.1 que el algoritmo SEEDS encabeza los mejores resultados en promedio; el algoritmo propuesto queda en cuarto lugar, superando a ERGC, W y SLIC. Es de notar que SLIC y W no cambiaron de valor en la métrica *Rec* para las últimas 3 evaluaciones (con $K = [2800, 3200, 3600]$). Esto puede significar que, para esta métrica, ya no les es posible mejorar más allá de ese límite.

Tabla 5.1: Evaluación final para la métrica *Rec*

$K \approx$	CRS	ERGC	ERS	Propuesto	SEEDS	SLIC	W	IFCM
100	0.652887	0.549891	0.627257	0.621731	0.834684	0.544622	0.519043	0.472840
200	0.738185	0.654925	0.702249	0.710774	0.882484	0.610907	0.602725	0.560390
300	0.794519	0.719687	0.747916	0.751149	0.906473	0.643648	0.658145	0.624905
400	0.833188	0.756590	0.778705	0.782773	0.922132	0.687424	0.692999	0.647981
600	0.870487	0.808300	0.823573	0.825701	0.941889	0.732231	0.749924	0.704036
800	0.896689	0.842004	0.854497	0.853190	0.952388	0.753051	0.783432	0.753741
1000	0.918087	0.871526	0.877913	0.870928	0.966503	0.789132	0.822392	0.785504
1200	0.929991	0.889978	0.895901	0.887987	0.967167	0.809190	0.843054	0.814278
1400	0.941932	0.906633	0.911274	0.903483	0.975478	0.809190	0.843054	0.828674
1600	0.949005	0.916659	0.923697	0.914796	0.974282	0.833441	0.865465	0.842352
1800	0.953460	0.924095	0.934235	0.923829	0.981553	0.852979	0.890398	0.866662
2000	0.960689	0.937167	0.943055	0.932409	0.986825	0.852979	0.890398	0.873480
2400	0.971708	0.951546	0.956364	0.945595	0.983410	0.885066	0.917414	0.900199
2800	0.977718	0.960317	0.966710	0.954774	0.989911	0.911323	0.945248	0.923313
3200	0.981349	0.967762	0.974053	0.962983	0.989969	0.911323	0.945248	0.927612
3600	0.987859	0.978271	0.980283	0.969507	0.993214	0.911323	0.945248	0.935783
Prom.	0.897359	0.852209	0.868605	0.863225	0.953022	0.783614	0.807136	0.7788593

En el detalle de la Figura 5.3 se evidencia que ERS y el algoritmo propuesto tienen un mejor desempeño con respecto al resto para números menores de superpíxeles. Es decir, se ajustan con mayor rapidez a la demarcación de las regiones en el *ground truth*. En cantidades de K mayores casi todos observan un mismo comportamiento, sobresaliendo SEEDS que es el mejor cuantificado en esta métrica.

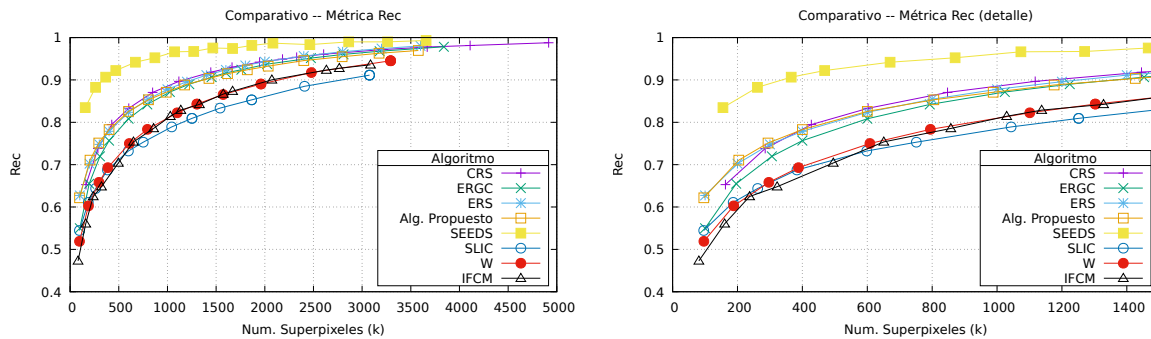


Figura 5.3: Evaluación final para la métrica Recuerdo del límite (Rec) en imágenes del repositorio *BSDS500*, subconjunto *test*.

Métrica Error de subsegmentación (UE)

Como se definió, la métrica Error de subsegmentación mide el traslape de superpíxeles con respecto a múltiples segmentos cercanos del *ground truth*.

La Tabla 5.2 y la Figura 5.4 presentan los resultados para la métrica UE . Como se mencionó anteriormente se opta por presentar resultados y gráfica de $(1-UE)$ (y no de UE) para equipararlos a los resultados de Rec y EV , donde un valor mayor es deseable.

Como se muestra en la Tabla 5.2, el algoritmo CRS obtiene el mejor promedio, seguido por ERGC y por el algoritmo propuesto.

Se observa en la Tabla 5.2 que SLIC y W no cambian de valor en la métrica Rec para las últimas 3 evaluaciones (con $K = [2800, 3200, 3600]$). Esto puede significar que, para la métrica UE , ya no les es posible mejorar más allá de ese límite.

Tabla 5.2: Evaluación final para la métrica ($1-UE$)

$K \approx$	CRS	ERGC	ERS	Propuesto	SEEDS	SLIC	W	IFCM
100	0.816160	0.786170	0.782677	0.777989	0.719872	0.776358	0.698740	0.674099
200	0.856217	0.840228	0.834914	0.838812	0.776924	0.828603	0.772888	0.753831
300	0.876007	0.866060	0.856741	0.860483	0.806330	0.850011	0.817905	0.788208
400	0.890498	0.878205	0.871449	0.875062	0.833256	0.871297	0.837564	0.811978
600	0.903185	0.895077	0.887820	0.893380	0.855365	0.890625	0.866199	0.841805
800	0.911770	0.904562	0.898035	0.903362	0.872376	0.899214	0.879694	0.860576
1000	0.919297	0.912194	0.904767	0.909354	0.882476	0.909849	0.894220	0.873354
1200	0.922701	0.916815	0.910301	0.914736	0.889488	0.914716	0.900908	0.881076
1400	0.926888	0.921514	0.914463	0.919741	0.899842	0.914716	0.900908	0.887923
1600	0.929728	0.923887	0.918322	0.922748	0.904194	0.920416	0.907163	0.894222
1800	0.931298	0.926260	0.921243	0.925313	0.909153	0.924937	0.913937	0.900647
2000	0.933801	0.929270	0.923770	0.928177	0.902017	0.924937	0.913937	0.903692
2400	0.938444	0.933364	0.928120	0.931961	0.917406	0.930849	0.921283	0.910662
2800	0.941272	0.935827	0.931665	0.934532	0.922503	0.935575	0.928409	0.918117
3200	0.943445	0.938710	0.934565	0.937388	0.926157	0.935575	0.928409	0.920277
3600	0.947010	0.941886	0.937049	0.939406	0.930967	0.935575	0.928409	0.924426
Prom.	0.911732	0.903126	0.8972438	0.900777	0.871770	0.897703	0.875660	0.859055

La Figura 5.4 exhibe a SEEDS y W como los peores valorados en la métrica UE . El resto de algoritmos, incluyendo el propuesto, tienen un comportamiento bastante similar.

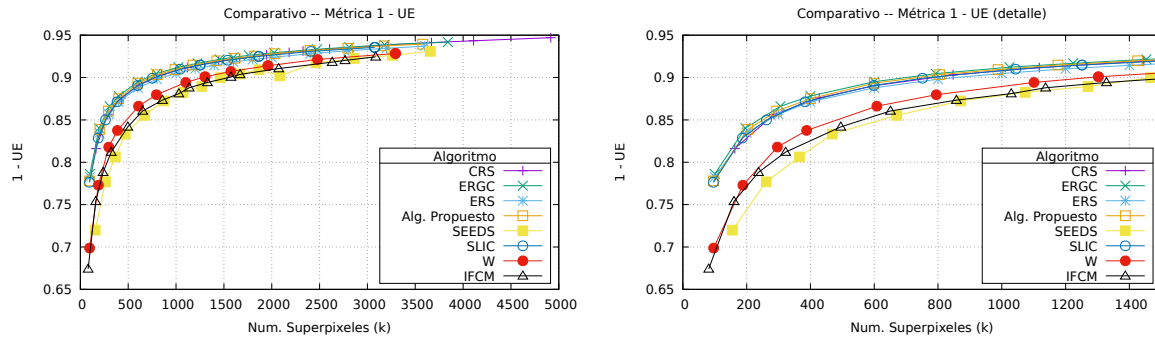


Figura 5.4: Evaluación final para la métrica Error de subsegmentación ($1-UE$) en imágenes del repositorio *BSDS500*, subconjunto *test*.

Métrica Varación explicada (EV)

La métrica EV mide qué tan bien los datos en los pixeles originales están representados por los superpixeles.

De acuerdo a la Tabla 5.3, SEEDS obtiene el mejor promedio seguido por el algoritmo propuesto por un margen muy pequeño de 0.008807. Esto es notable ya que

el siguiente algoritmo, ERGC, se distancia del propuesto por 0.011439. Siendo que esta métrica no depende del *ground truth*, es posible considerar que el algoritmo propuesto obtiene una buena calidad de representación de píxeles en los superpíxeles generados.

Tabla 5.3: Evaluación final para la métrica EV

$K \approx$	CRS	ERGC	ERS	Propuesto	SEEDS	SLIC	W	IFCM
100	0.781892	0.764092	0.719187	0.791923	0.840048	0.769038	0.660470	0.675079
200	0.815277	0.810594	0.764528	0.835434	0.864326	0.808453	0.717307	0.726378
300	0.834185	0.836518	0.787101	0.853476	0.878133	0.828624	0.754624	0.755192
400	0.850525	0.850884	0.801482	0.867869	0.889903	0.844389	0.773384	0.765326
600	0.863107	0.871088	0.821036	0.884930	0.901613	0.864344	0.802638	0.790485
800	0.872172	0.883382	0.833156	0.895715	0.907270	0.875610	0.816675	0.813163
1000	0.882352	0.893619	0.842544	0.903173	0.902653	0.887919	0.833498	0.822016
1200	0.886017	0.901016	0.849669	0.909207	0.917012	0.894361	0.841242	0.830227
1400	0.891087	0.907521	0.855313	0.915881	0.913621	0.894361	0.841242	0.839529
1600	0.894837	0.910854	0.860322	0.919454	0.925402	0.901517	0.849824	0.843061
1800	0.896325	0.914305	0.864611	0.923156	0.919484	0.908336	0.858543	0.851747
2000	0.899183	0.919184	0.868318	0.926291	0.912326	0.908336	0.858543	0.856725
2400	0.905304	0.925413	0.874669	0.931052	0.935122	0.915874	0.867261	0.861976
2800	0.908927	0.929317	0.879867	0.935071	0.930575	0.923182	0.876992	0.873621
3200	0.911956	0.933205	0.884831	0.938661	0.938591	0.923182	0.876992	0.877320
3600	0.916861	0.938679	0.888955	0.941401	0.937521	0.923182	0.876992	0.884978
Prom.	0.875625	0.886854	0.837224	0.898293	0.907100	0.879419	0.819139	0.816676

La Figura 5.5 exhibe el comportamiento particular para la métrica *EV* en que W, SLIC y CRS son los peores comportados y destacan SEEDS y el algoritmo propuesto en prácticamente todos los valores de superpíxeles en esta parte de la experimentación.

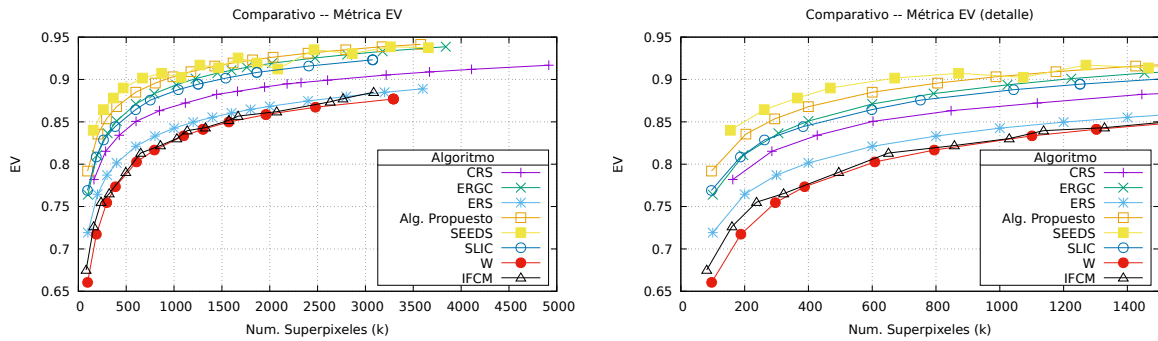


Figura 5.5: Evaluación final para la métrica Variación explicada (*EV*) en imágenes del repositorio *BSDS500*, subconjunto *test*.

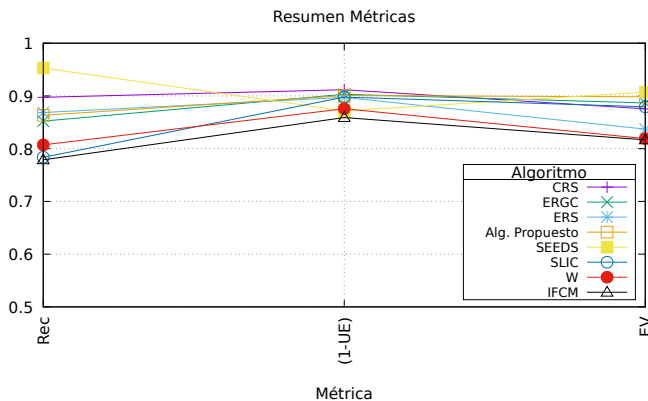


Figura 5.6: Resultados comparativos de métricas Rec , $(1-UE)$ y EV para los algoritmos analizados. Se observa que, en promedio, el algoritmo propuesto tiene un buen desempeño.

Resumen de resultados

La Tabla 5.4 resume los promedios por métrica-algoritmo derivados de la experimentación realizada. Se observa en la misma que el algoritmo propuesto supera a SLIC, W y ERGC en la métrica Recuerdo de límites (Rec). Para la métrica Error de subsegmentación (UE) la propuesta se sitúa en tercer lugar únicamente por debajo de CRS y ERGC. Se destaca que en la evaluación de la Variación explicada el algoritmo propuesto supera a todos exceptuando a SEEDS siendo esto por un pequeño margen (0.008807).

Tabla 5.4: Resumen final para las métricas Rec , $(1-UE)$ y EV

Métrica	CRS	ERGC	ERS	Propuesto	SEEDS	SLIC	W	IFCM
Rec	0.897359	0.852209	0.868605	0.863225	0.953022	0.783614	0.807136	0.778859
$(1-UE)$	0.911732	0.903126	0.8972438	0.900777	0.871770	0.897703	0.875660	0.859055
EV	0.875625	0.886854	0.837224	0.898293	0.907100	0.879419	0.819139	0.816676

La Figura 5.6 refleja gráficamente los resultados mostrados.

Resultados cualitativos

Se realizó un comparativo de la calidad visual de la sobsegmentación del algoritmo propuesto con respecto a cada uno de los 6 algoritmos seleccionados. Las Figuras 5.7 y 5.8 muestran el resultado para $k \approx 200$ y $k \approx 400$ respectivamente. Notese que el comportamiento visual del algoritmo propuesto es consistente hacia una buena adherencia a bordes.

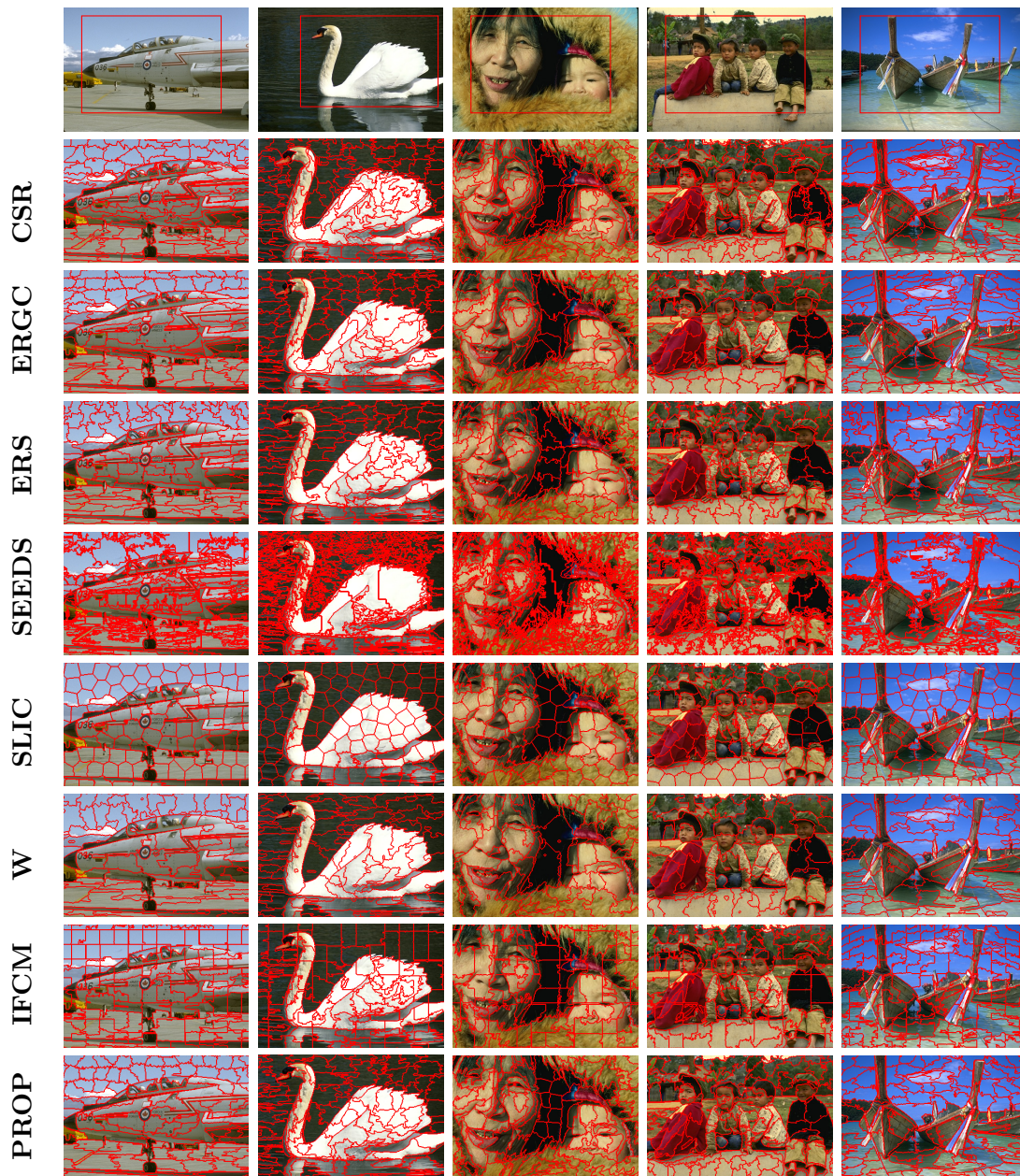


Figura 5.7: Comparación cualitativa de los algoritmos evaluados con $k \approx 200$. Los renglones con resultados por algoritmo son detalles, en rojo, de la imagen original en el primer renglón. Imágenes del repositorio *BSDS500*, subconjunto *test*.



Figura 5.8: Comparación cualitativa de los algoritmos evaluados con $k \approx 400$. Los renglones con resultados por algoritmo son detalles, en rojo, de la imagen original en el primer renglón. Imágenes del repositorio *BSDS500*, subconjunto *test*.

Comparativo visual de las métricas *Rec* y *UE*

Dada la definición de *Rec* y *UE*, es deseable una representación visual de las mismas. Esta forma gráfica permite la comparación con respecto al *ground truth* de las coincidencias (en el caso de *Rec*) y de “fugas” (en el caso de *UE*).

La Figura 5.9 ejemplifica, con una imagen del repositorio *BSDS500*, ambas métricas. Es observable que, para el algoritmo propuesto, el objeto en la columna Media está mejor definido que en el resto de los algoritmos y que, para la columna *Rec*, se encuentran menos (y más pequeñas) zonas en rojo (que determinan errores de “fuga” de píxeles).

Resultados cualitativos en imágenes en tonos de gris

La distancia utilizada en el presente trabajo se expresa mediante diferencias espaciales y diferencias en color. Esta utilización de color permite una mejor adaptación a los bordes de los objetos en la imagen. Los superpíxeles están mejor descritos y homogéneos. Se muestra en la Figura 5.10 el resultado de tres sobresegmentaciones con el algoritmo propuesto. Es notoriamente visible el hecho que la información de color utilizada en el algoritmo propuesto manifiesta bordes mejor definidos en los límites de cambio de color, esto es, 5.10-(e) es mejor que 5.10-(c).

Dado lo anterior, es recomendable la utilización del algoritmo propuesto en imágenes a color para el aprovechamiento del mejor desempeño, por sobre las imágenes en tonos de gris.

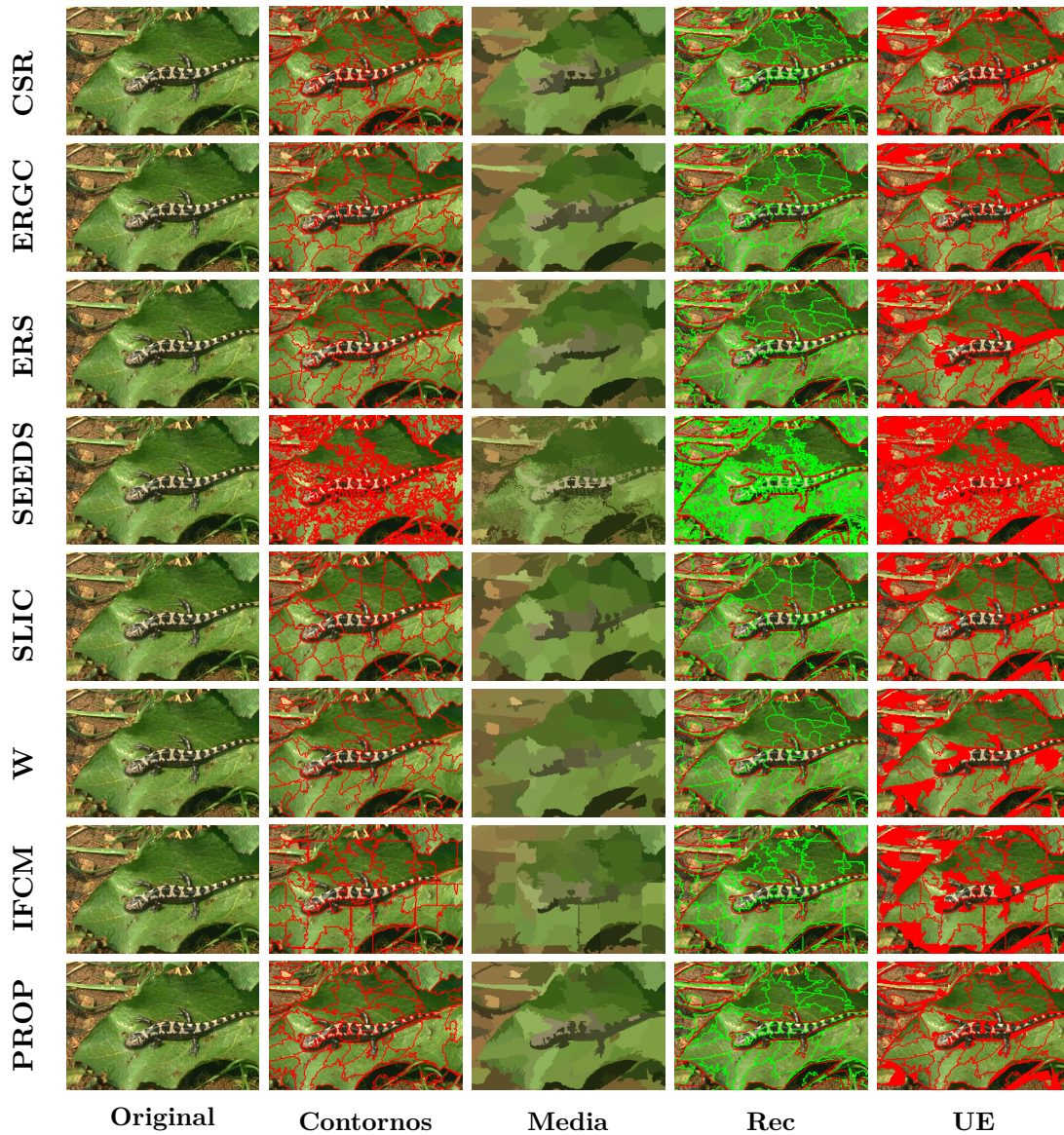


Figura 5.9: Comparación cualitativa de los algoritmos evaluados con $k \approx 100$. Representaciones mediante contornos e intensidad media por superpixel. Comparaciones con el segmento *ground truth* con respecto a la métrica *Rec* y a la métrica *UE* para cada algoritmo. Imagen 75083 del repositorio *BSDS500*, subconjunto *test*.

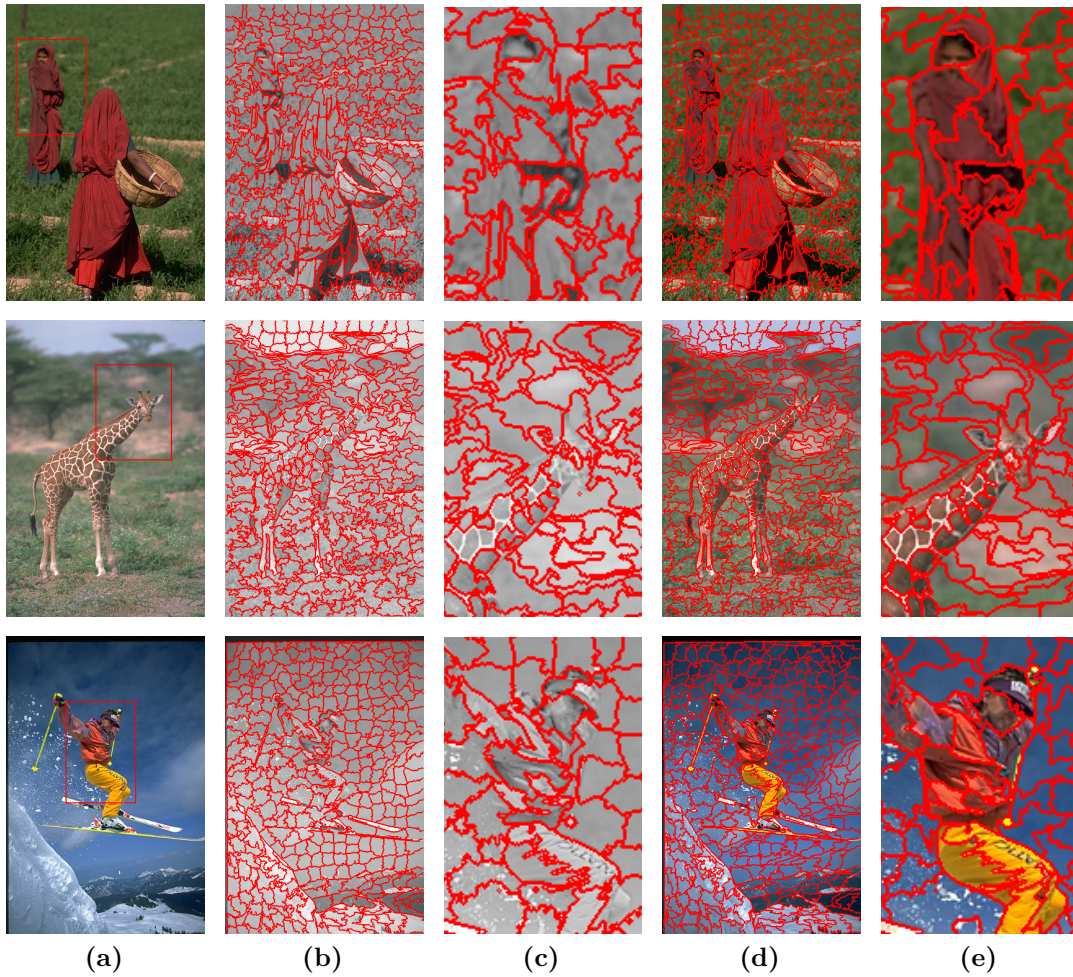


Figura 5.10: Comparación de utilización de color para el cálculo de la distancia en la sobresegmentación, $k \approx 400$. Imágenes del repositorio *BSDS500*, subconjunto *test*. (a) Imagen original; (b) imagen en niveles de gris sobresegmentada; (c) detalle de (b); (d) imagen en color sobresegmentada (algoritmo propuesto); (e) detalle de (d).

5.3.2. Posprocesamiento mediante grafo de adyacencia de regiones (RAG)

Una etapa no considerada en los objetivos originales de tesis se realizó como complemento al trabajo de sobresegmentación utilizando grafos difusos: segmentación regular con información de la sobresegmentación propuesta.

Debido a que no fue parte de los objetivos del trabajo de tesis, únicamente se realizaron pruebas cualitativas (visuales) para demostrar tanto el funcionamiento correcto del algoritmo de sobresegmentación, como un ejemplo práctico de alguna de las diversas aplicaciones que tienen los superpíxeles y de algoritmos basados en grafos.

Resultados cualitativos

En la Figura 5.11 se evidencia el producto del procesamiento de una imagen previamente sobresegmentada con diferentes cantidades de superpíxeles K .

A partir de la figura se observa que un incremento en la cantidad de superpíxeles no necesariamente mejora el resultado final de la segmentación para imágenes con texturas complejas como las encontradas en el repositorio *BSDS500*. El nivel ideal de umbral para el grafo RAG también es una cantidad dependiente del tipo de imagen de que se trate.

La Figura 5.12 presenta imágenes del repositorio *Weizmann* igualmente posprocesadas utilizando grafos de adyacencia de regiones.

De acuerdo a la Figura se observa que la existencia de regiones con texturas o colores homogéneos mejora la calidad en la agrupación de superpíxeles, teniendo, a diferencia de las pruebas con el repositorio *BSDS500*, un objeto casi segmentado en la última columna.

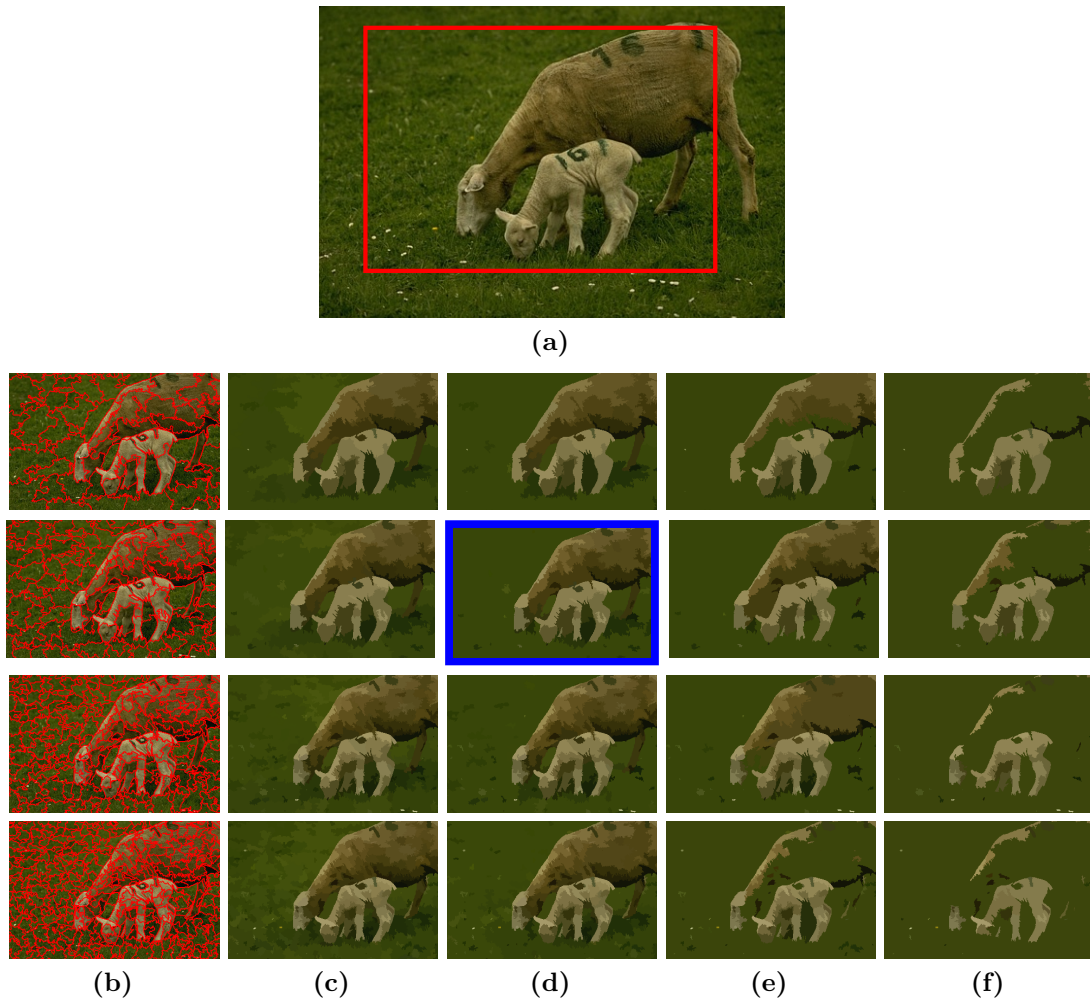


Figura 5.11: Ejemplo de imagen procesada con diferentes valores de umbral en un grafo de adyacencia de regiones (RAG); repositorio *BSDS500*. Los valores en superpixels aumenta en renglones de arriba hacia abajo con $k = 100$, $k = 200$, $k = 400$ y $k = 600$, respectivamente. (a) Imagen 207056 original; columna (b) imagen sobresegmentada; (c) umbral t con valor 5; (d) $t = 10$; (e) $t = 20$; (f) $t = 30$.



Figura 5.12: Ejemplo de imágenes pertenecientes al repositorio *Weizmann*; posprocesadas con RAG para diferentes cantidades de superpíxeles K e igual nivel de umbral $t = 30$. En la columna (a) se presenta la imagen original; las siguientes cuatro columnas se corresponden con (b) 100, (c) 200, (d) 400 y (e) 600 superpíxeles.

6

Conclusiones

En el presente trabajo se ha descrito la propuesta de algoritmo de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color. Este algoritmo aprovecha la información de vecinos (local) para generar un grafo difuso que relaciona píxeles.

La eficiencia del algoritmo ha sido comparada con otros 6 algoritmos dentro del estado del arte: uno de ellos basados en grafos (ERS), uno cimentado en evolución de contorno (ERGC), uno fundado en generación de clústers (SLIC), dos soportados por la optimización de una función de energía (CRS y SEEDS) y uno basado en cuencas hidrográficas (W). Los ensayos se realizaron con las 200 imágenes del repositorio BSDS500-*test*, examinando la calidad de la sobresegmentación mediante tres métricas que, en términos generales, miden la adherencia a bordes relativo al *ground truth* los dos primeros, y sin requerir *ground truth* en el caso del tercero: Recuerdo de límites (*Rec*), Error de subsegmentación (*UE*) y Variación explicada (*EV*).

La experimentación realizada demuestra que el algoritmo es un buen candidato con respecto a las tres métricas. En la métrica *Rec*, el algoritmo propuesto supera en el promedio a ERGC, W y SLIC, y es muy eficiente con pocos superpíxeles. Con respecto a *UE* el algoritmo propuesto se ubica por promedio debajo sólo de CRS y ERGC. Notablemente, para la métrica *EV*, el algoritmo propuesto obtiene una buena calidad de representación de píxeles en los superpíxeles generados, siendo únicamente superado por SEEDS.

Es notable la rápida adherencia a los bordes que posee el algoritmo propuesto. Esto se observa en las gráficas de *Rec* y *EV* (Figuras 5.3 y 5.5), donde los valores obtenidos para $K = 100$ son mayores que el resto de los algoritmos exceptuando tal vez a SEEDS ¹.

¹Ver nota al pie en la página 59.

Recomendaciones de utilización de los algoritmos analizados

Como parte de las conclusiones se ofrece, en la Tabla 6.1, un resumen de posibles usos y/o recomendaciones para los algoritmos analizados durante el desarrollo del proyecto de tesis.

Tabla 6.1: Recomendaciones de utilización de algoritmos de sobresegmentación

Algoritmo	Recomendaciones de uso
CRS	Notable adherencia a los bordes. Dependiendo de los parámetros puede obtener superpíxeles irregulares (Stutz <i>et al.</i> , 2018).
ERGC	Notable adherencia a los bordes. La compacidad es muy dependiente de los parámetros empleados (Stutz <i>et al.</i> , 2018).
ERS	Notable adherencia a los bordes. (Stutz <i>et al.</i> , 2018)
SEEDS	A pesar de que este algoritmo no ofrece visualmente una buena compacidad, su ajuste a bordes es superior, incluso al del algoritmo propuesto.
SLIC	Algoritmo veloz que permite un control de la compacidad (posproceso) y de la cantidad aproximada de superpíxeles. Parte de la inicialización ha sido usada en otros algoritmos incluyendo el presentado y el desarrollado en (Domínguez, 2019).
W	La ubicación y cantidad de los marcadores es importante dado que se basa en algoritmo de inundación. No ofrece control sobre la compacidad.
IFCM	Ofrece buena compacidad.
Propuesto	Al igual que SEED, el seguimiento a los bordes del <i>ground truth</i> para pocos superpíxeles es significativo. Entornos donde se requiera una rápida aproximación al <i>ground truth</i> son los ideales.

6.1. Productos

Durante el desarrollo de la investigación e implementación se desarrollaron los siguientes productos:

- Una implementación del algoritmo propuesto, en lenguajes C y Python
- Un conjunto de 1063 archivos en formato CSV obtenidos a partir de los archivos de Matlab (.mat) incluidos como *ground truth* del repositorio *BSDS500* subconjunto *test*. Incluido el código de extracción
- Un *script* en Matlab para presentación y comparación visual de imágenes

- Una batería de *scripts* en *bash* para la manipulación de archivos y ejecución en lotes del procesamiento con algoritmos
- Un *script* para *gnuplot* destinado a la generación de gráficas a partir de archivos CSV
- Artículo “Segmentación de imágenes basada en teoría de grafos” para la *Escuela de Inteligencia Computacional y Robótica 2019*. Octubre de 2019, Universidad Tecnológica Emiliano Zapata del Estado de Morelos, Cuernavaca, Morelos.

6.2. Aportaciones

- Un algoritmo de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color
- Un ejemplo práctico de segmentación regular, basado igualmente en grafos, de utilización de los superpíxeles generados por una sobresegmentación de imágenes.

6.3. Trabajos futuros

- La versatilidad de los grafos difusos permite establecer relaciones de pertenencia mediante múltiples variables medibles. En el presente trabajo fue utilizada una distancia en espacio 5-D que mezcla diferencias de color y espacial. Es deseable y posible, mejorar la eficiencia agregando características adicionales, tales como textura, topográficas y otras, a las aristas del grafo difuso estableciéndolas como relaciones difusas.
- Un posible trabajo futuro es el análisis de la complejidad algorítmica de los algoritmos presentados, incluyendo el propuesto.

Apéndice 1. Constancias y reconocimientos



SEP
SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

**EL TECNOLÓGICO NACIONAL DE MÉXICO
A TRAVÉS DEL INSTITUTO TECNOLÓGICO DE IGUALA**

OTORGA EL PRESENTE

RECONOCIMIENTO

A

DIEGO CARLOS LUNA MÁRQUEZ

POR SU DESTACADA PARTICIPACIÓN COMO **PONENTE** DE LA CONFERENCIA DENOMINADA: **"SOBRESEGMENTACIÓN DE IMÁGENES UTILIZANDO GRAFOS DIFUSOS"** EN EL MARCO DEL **1er COLOQUIO "ARTIFICIAL INTELLIGENCE"**, REALIZADO EN EL INSTITUTO TECNOLÓGICO DE IGUALA.

IGUALA, GRO., A 9 DE MAYO DE 2019.

"TECNOLOGIA COMO SINÓNIMO DE INDEPENDENCIA"



SECRETARÍA DE
EDUCACIÓN PÚBLICA
INSTITUTO
TECNOLÓGICO DE
IGUALA

M.D.I.S. ARELI BARCENAS NAVA DIRECCIÓN
DIRECTORA



Figura 6.1: Reconocimiento como ponente en el 1er Coloquio "Artificial Intelligence". 9 de mayo de 2019, Instituto Tecnológico de Iguala, Guerrero



SEP
SECRETARÍA DE
EDUCACIÓN PÚBLICA



**EL TECNOLÓGICO NACIONAL DE MÉXICO
A TRAVÉS DEL CENTRO NACIONAL DE INVESTIGACIÓN
Y DESARROLLO TECNOLÓGICO**

OTORGAN LA PRESENTE

CONSTANCIA

A

**David Domínguez López, Dante Mújica-Vargas, Diego C.
Luna Márquez**

POR SU PARTICIPACIÓN CON EL ARTÍCULO: **“2-D SUPERPIXEL
OVERSEGMENTATION OF IMAGES EVALUATED BY DIFFERENT
COLOR SPACES”**, PRESENTADO EN LA SEGUNDA JORNADA DE
CIENCIA Y TECNOLOGÍA APLICADA, QUE SE LLEVO A CABO LOS
DÍAS 4 Y 5 DE ABRIL DE 2019, EN LA CIUDAD DE CUERNAVACA,
MORELOS, MÉXICO.

CUERNAVACA, MOR., A 5 DE ABRIL DE 2019.

DR. VÍCTOR HUGO OLIVARES PEREGRINO
DIRECTOR



S. E. P.

CENTRO NACIONAL DE INVESTIGACION
Y DESARROLLO TECNOLÓGICO



Figura 6.2: Constancia de participación con el artículo “2-D Superpixel Oversegmentation of Images Evaluated by Different Color Spaces”, en la Segunda Jornada de Ciencia y Tecnología Aplicada. 4 y 5 de abril de 2019, CENIDET, Cuernavaca, Morelos

2-D Superpixel Oversegmentation of Images Evaluated by Different Color Spaces

David Domínguez López* Dante Mújica-Vargas**
Diego C. Luna-Márquez***

* CENIDET-TecNM, Cuernavaca, México (e-mail:
david.dominguez17ce@cenidet.edu.mx.)

** CENIDET-TecNM, Cuernavaca, México (e-mail:
dantemv@cenidet.edu.mx.)

*** CENIDET-TecNM, Cuernavaca, México (e-mail:
diego.luna18ca@cenidet.edu.mx.)

Abstract: The image oversegmentation by using superpixels is an efficient way to simplify it through pixel regions perceptually similar; in other words, it reduces their complexity in order to develop a posterior global segmentation. This feature has allowed superpixels to be used in areas such as image processing and computer vision. SLIC (Simple Linear Iterative Clustering) is one of the most efficient superpixel algorithms in the literature, it has a low runtime and produces superpixels with dynamical adjust to objects in size and shape, in addition to its low computational cost. In this paper an evaluation of the SLIC algorithm considering the CIE L*a*b* (original one used), HSV, YCbCr and the orthonormal IJK color spaces. The performance of the algorithm and the quality of the superpixel segmentation are evaluated using the metrics proposed in the literature.

Keywords: Oversegmentation, superpixel, SLIC, color spaces, image processing.

1. INTRODUCTION

The oversegmentation by means of super pixels is considered a subproblem of image segmentation in order to provide a simpler, more compact and efficient representation of an image without losing the detail between pixels. This allows to represent an image with only a couple of hundred segments instead of tens of thousands of pixels Boemer et al. (2018), Schick et al. (2014). Superpixels are homogeneous regions of pixels grouped by low-level characteristics. They were introduced by Ren and Malik (2003), as more natural and coherent entities of an image while at the same time reducing the number of primitives.

The image redundancy is downscale when the graphical nodes are built with superpixels, this causes that both complexity and calculation time decrease substantially. The reduction of complexity allows superpixels to be very useful in various applications of computer vision and image processing Tasli et al. (2015). For example for biomedical image segmentation Ensafi et al. (2016), medical image analysis Verma et al. (2013), to mention just a few.

In recent years, several superpixel algorithms have been introduced in the state-of-art. In relation to the approach used to generate superpixels, the following categorization can be done:

Based on watershed. Some super-pixel algorithms based on this approach control compactness, for example *Compact Watershed* controls compactness with a single parameter by using a distance metric that combines the distance based on the conventional aspect and the Euclidean distance from the pixel to the seed of the segment Neubert and Protzel (2014).

Based on clustering. These algorithms are based on clustering algorithms to generate superpixels. SLIC Achanta et al. (2010) is inspired by the *k-means* algorithm for grouping pixels. The number of superpixels and the compactness are controllable through the parameters k and m , respectively. To group pixels, it uses a distance metric that combines the spatial and color information of each pixel. This algorithm requires further processing to reinforce the connectivity between the superpixels.

Based on the energy optimization. These algorithms iteratively optimize a formulated energy. For example in Yao et al. (2015), the image is divided into a regular grid such as initial segmentation of superpixels, and the pixels are exchanged between neighboring superpixels with respect to energy. The number of superpixels and the compactness can be controlled. The algorithm employs a single iteration to converge on a better minimum of energy, this allows it to be fast.



SEP
SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

**EL TECNOLÓGICO NACIONAL DE MÉXICO
A TRAVÉS DEL CENTRO NACIONAL DE INVESTIGACIÓN
Y DESARROLLO TECNOLÓGICO**

OTORGAN LA PRESENTE

CONSTANCIA

A

**Diego Carlos Luna Márquez, Dante Mújica Vargas, Ana
Montserrat Rojas Fernández**

POR SU PARTICIPACIÓN CON EL ARTÍCULO: **“THE THETA*
ALGORITHM FOR PATH CALCULATION”**, PRESENTADO EN LA
SEGUNDA JORNADA DE CIENCIA Y TECNOLOGÍA APLICADA, QUE
SE LLEVO A CABO LOS DÍAS 4 Y 5 DE ABRIL DE 2019, EN LA
CIUDAD DE CUERNAVACA, MORELOS, MÉXICO.

CUERNAVACA, MOR., A 5 DE ABRIL DE 2019.



S. E. P.
CENTRO NACIONAL DE INVESTIGACION
Y DESARROLLO TECNOLÓGICO

DR. VÍCTOR HUGO OLIVARES PEREGRINO
DIRECTOR



Figura 6.4: Constancia de participación con el artículo “The Theta* Algorithm for Path Calculation”, en la Segunda Jornada de Ciencia y Tecnología Aplicada. 4 y 5 de abril de 2019, CENIDET, Cuernavaca, Morelos

The Theta* Algorithm for Path Calculation

Diego Carlos Luna Márquez* Dante Mújica-Vargas**
Ana Monserrat Rojas Fernández***

* CENIDET-TecNM, Cuernavaca, México (e-mail: diego.luna18ca@cenidet.edu.mx.)

** CENIDET-TecNM, Cuernavaca, México (e-mail: dantemv@cenidet.edu.mx.)

*** CENIDET-TecNM, Cuernavaca, México (e-mail: monserojas@cenidet.edu.mx.)

Abstract: Robot displacement ability is provisioned by its autonomous navigation system. The aim is to move towards a goal point by avoiding obstacles, knowing previously its environment. To achieve this, the robot needs to locate, build, or know the navigation environment through a map, implement a path planning algorithm, and to be able to follow the path previously calculated by the algorithm. This document presents research work results obtained from the implementation and experimentation of the Theta* path planning algorithm. Testing was deployed in the Robotic Operating System (ROS) and Justina, an omnidirectional robot developed in the National Autonomous University of Mexico Bio-robotics Laboratory.

Keywords: Autonomous navigation, path planning, Robotic Operating System (ROS), service robot.

1. INTRODUCTION

Mobile robots commonly interact in environments with non-flat surfaces. This requires the robot includes ability to solve displacement or path problems. A common method to approach the challenge of trajectories planning and tracking in robotics and video games is the A* algorithm and its variants. However, the trajectories generated by A* algorithm are conditioned to the edges of a mesh and to generate rectilinear routes. Those restrictions cause the robot to make an extra effort to follow a path.

The aim of this paper is to report the implementation and testing of accuratest algorithm named Theta*, which is a variant of A* algorithm. Theta* searches in eight directions, without restrictions, the trajectories to the edges of the grid to find diagonal paths. Additionally, it is simple to implement and capable to build short realistic looking paths Nash et al. (2007).

Some concepts and methods for planning trajectories are introduced in Section 2, including Theta* algorithm.

2. PATH PLANNING ALGORITHMS

The main goal of any path planning algorithm is to generate a free-collision path, and subsequently, follow the calculated route. In the present paper, three path planning algorithms are compared: Dijkstra, A* and Theta*.

2.1 Dijkstra Algorithm

The Dijkstra algorithm, created by Edsger Dijkstra in 1959, finds the shortest path from the origin node to all

different nodes in the network. Given a vertex (s) in a weighted directed graph $G = (V, E)$ where all weights must be positive, Dijkstra finds the path with the lowest cost (shortest path) between (s) and another vertex in G . First step is to initialize the distances to zero, since they are unknown values at the beginning, $dist[s] = 0$, and $dist = [x]$. It is the shortest known path length from (s) to $x \in Q$. In each cycle, Dijkstra extracts the vertex $u \in Q$, with the minimum $dist[]$ in Q . Then, for each neighbor v of u establishes the relaxation operation:

$$dist[v] = \min(dist[v], dist[u] + w(u, v)) \quad (1)$$

When the shortest path is obtained from the origin vertex, the algorithm stops. Because all the nodes are visited in this algorithm, the calculation of a solution for large graphs becomes slow, which increases the computational cost value Dijkstra and Scholten (1980).

2.2 A* Algorithm

The A* algorithm, proposed by Peter E. Hart, Nils J. Nilsson and Bertram Raphael in 1968, is a most used planning algorithm Flether et al. (2008). If there exists a solution, A* finds it and guarantees that it is optimal. Notably the computational cost is not as high as the Dijkstra algorithm. According to Nash (2012), it has the properties of simplicity, efficiency and generality. To focus search, the A* algorithm uses the following values:

$h(s) \rightarrow$ Is the estimated distance from the initial position to the final destination position. A heuristic function to calculate the estimated value is used.

UNIVERSIDAD TECNOLÓGICA
EMILIANO ZAPATA DEL ESTADO DE MORELOS
ORGANISMO PÚBLICO DESCENTRALIZADO DEL GOBIERNO DEL ESTADO DE MORELOS

otorga el presente

RECONOCIMIENTO

A: Diego Carlos Luna Márquez

Por haber impartido la ponencia:
"Segmentación de imágenes basada en teoría de grafos"
llevada a cabo en las instalaciones de esta Universidad Tecnológica,
en el marco de la "Escuela de inteligencia computacional y robótica 2019"
los días del 22 al 26 de octubre del presente año.

Emiliano Zapata, Mor., octubre de 2019



M. en C. Jaime Vázquez Colín
Director de la División Académica
de Mecánica Industrial



Figura 6.6: Reconocimiento como ponente en la Escuela de Inteligencia Artificial y Robótica 2019. Universidad Tecnológica "Emiliano Zapata" del Estado de Morelos, Emiliano Zapata, Morelos.

Segmentación de imágenes basada en teoría de grafos

Diego Carlos Luna Márquez, Dante Mújica Vargas, Raúl Pinto Elías, Andrea Magadán Salazar y Manuel Mejía Lavall
 Departamento de Ciencias Computacionales
 Tecnológico Nacional de México - CENIDET
 Cuernavaca, México
 {diego.luna18ca, dantemv, rpinto, magadan, mlavalle}@cenidet.edu.mx

Resumen—Múltiples métodos y algoritmos de procesamiento para análisis de imágenes han utilizado conceptos de la teoría de grafos. La naturaleza discreta de las imágenes estructuradas como rejillas regulares de píxeles, es apropiada para su representación en grafos. En este documento se describe el método de corte de grafos, utilizado para segmentar imágenes. La imagen es trasladada a una representación como red de flujo, en que el corte mínimo que define la segmentación es encontrado mediante un algoritmo de aumento de trayectoria.

Index Terms—Segmentación, procesamiento de imágenes, teoría de grafos.

I. INTRODUCCIÓN

La segmentación de imágenes es una fase del procesamiento digital de señales cuyo objetivo es la etiquetación de píxeles para generar conjuntos finitos y disjuntos a partir de una o más propiedades de la imagen o los elementos que la constituyen. Siendo que es comúnmente usada como una fase intermedia, se considera una de las etapas más importantes del procesamiento digital de señales y en particular del procesamiento digital de imágenes. El resultado de la segmentación es utilizado en la etapa siguiente, la de reconocimiento de objetos, empleando diversos métodos (Mújica-Vargas, Gallegos-Funes, y Cruz-Santiago, 2011)

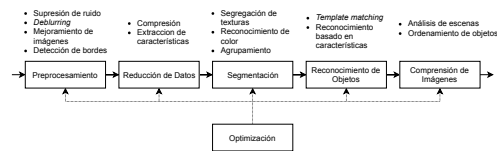


Figura 1: Fases del procesamiento de imágenes.

Varios métodos en procesamiento digital de imágenes que hacen uso de la teoría de grafos han sido propuestos. Algunos de ellos se utilizan, por ejemplo, en estimación de profundidad estéreo (Boykov y Kolmogorov, 2004), fotomontaje interactivo (Agarwala y cols., 2004), remoción de sombras en imágenes (Miyazaki, Matsushita, y Ikeuchi, 2009) (Nielsen y Madsen, 2007), entre otras aplicaciones.

El presente documento está organizado de la siguiente manera. La sección II establece el marco conceptual para la

segmentación de imágenes y el tratamiento de las mismas como grafos. La sección III describe la experimentación realizada, finalizando con la sección IV y conclusiones descritas en la sección V.

II. SEGMENTACIÓN DE IMÁGENES

Formalmente, la segmentación es un problema en el cual, dado R que representa la imagen completa, ésta se divide en N conjuntos no vacíos R_1, R_2, \dots, R_N , tales que se cumplen las siguientes condiciones (Gong y cols., 2019):

1. $\bigcup_{i=1}^N R_i = R$;
2. Para $i = 1, 2, \dots, N, P(R_i) = TRUE$;
3. Para $\forall i, j, i \neq j, R_i \cap R_j = \emptyset$;
4. Para $\forall i, j, i \neq j, P(R_i \cup R_j) = FALSE$;
5. Para $i = 1, 2, \dots, N, R_i$ es la región conectada.

Las anteriores condiciones establecen que, en una segmentación existen N conjuntos como disjuntos, complementarios y finitos. En el tipo de segmentación denominada binaria, $N = 2$.

II-A. Teoría de grafos

Un grafo se define como un conjunto no vacío de vértices o nodos (V) que pueden estar conectados mediante enlaces denominados aristas (E):

$$G = (V, E) \quad (1)$$

Dada la anterior definición, es posible realizar un mapeo de correspondencia de una imagen con uno o más grafos, equivalentes o complementarios, donde los elementos del conjunto V se corresponden generalmente con los píxeles de la imagen (aunque pudieran ser regiones, bordes, etc.) y los valores de peso en E mediante una o más medidas de similitud (o divergencia) entre dichos nodos.

Un ejemplo representativo en que el tratamiento como grafo a partir de una imagen es apropiado, se muestra en la Figura 2 (Boykov y Jolly, 2001), mediante la segmentación en dos regiones de una imagen simple de 3×3 píxeles.

Figura 6.7: Primer página del artículo “Segmentación de imágenes basada en teoría de grafos”. Escuela de Inteligencia Artificial y Robótica 2019. Universidad Tecnológica “Emiliano Zapata” del Estado de Morelos, Emiliano Zapata, Morelos.



Figura 6.8: Reconocimiento como ponente en la Semana Académica y Cultural de la Licenciatura en Computación 2019 en la UABJO. Noviembre 2019. Oaxaca de Juárez, Oaxaca



Figura 6.9: Reconocimiento como instructor del taller “Inteligencia Artificial” en la Semana Académica y Cultural de la Licenciatura en Computación 2019 en la UABJO. Noviembre 2019. Oaxaca de Juárez, Oaxaca

Bibliografía

- Achanta, R. y Susstrunk, S. Superpixels and polygons using simple non-iterative clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., y Sússtrunk, S. Slic superpixels. *EPFL Technical Report*, page 15, 2010. URL <http://infoscience.epfl.ch/record/149300>.
- Akram, M. y Akmal, R. Intuitionistic fuzzy graph structures. *Kragujevac Journal of Mathematics*, 2017. ISSN 14509628. doi: 10.5937/KgJMath1702219A.
- Akram, M., Akmal, R., y Alshehri, N. On m-polar fuzzy graph structures. *SpringerPlus*, 5(1):1448, aug 2016. ISSN 2193-1801. doi: 10.1186/s40064-016-3066-8. URL <https://doi.org/10.1186/s40064-016-3066-8>.
- Aljebory, K. M. y Mohammed, T. S. Modified fuzzy c-means clustering algorithm application in medical image segmentation. *JEA - Journal of Electrical Engineering*, 2(1), 2018. ISSN 2220-234X.
- Alpert, S. Segmentation evaluation database, 2007. URL http://www.wisdom.weizmann.ac.il/~vision/Seg_Evaluation_DB/.
- Alpert, S., Galun, M., Basri, R., y Brandt, A. Image segmentation by probabilistic bottom-up aggregation and cue integration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2007.
- Arbelaez, P., Maire, M., Fowlkes, C., y Malik, J. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(5):898–916, May 2011. ISSN 0162-8828. doi: 10.1109/TPAMI.2010.161. URL <http://dx.doi.org/10.1109/TPAMI.2010.161>.
- Arbeláez, P., Michael, M., Charless, F., y Jitendra, M. Contour detection and hierarchical image segmentation, 2011. URL <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html>.
- Avellaneda, D. A. Caracterización de texturas naturales. Tesis de maestría, Departamento de Ciencias de la Computación, Centro Nacional de Investigación y Desarrollo Tecnológico, 2009.

- Basavaprasad, B. y Hegadi, R. S. A survey on traditional and graph theoretical techniques for image segmentation. *International Journal of Computer Applications*, 2014.
- Bernier, M., Jodoin, P.-M., Humbert, O., y Lalande, A. Graph cut-based method for segmenting the left ventricle from mri or echocardiographic images. *Computerized Medical Imaging and Graphics*, pages 1–12, 2017.
- Borzooei, R. A., Rashmanlou, H., Samanta, S., y Pal, M. A study on fuzzy labeling graphs. *Journal of Intelligent & Fuzzy Systems*, pages 3349–3355, 2016. doi: 10.3233/IFS-152082.
- Boykov, Y. y Kolmogorov, V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE transactions on pattern analysis and machine intelligence*, 26(9):1124–1137, 2004.
- Buysens, P., Gardin, I., Ruan, S., y Elmoataz, A. Eikonal-based region growing for efficient clustering. *Image and Vision Computing*, 32, 12 2014. doi: 10.1016/j.imavis.2014.10.002.
- Cervantes, J. F. A. Metodología para la estructuración y uso de conocimiento en segmentación de imágenes digitales. Tesis de maestría, Departamento de Ciencias de la Computación, Centro Nacional de Investigación y Desarrollo Tecnológico, 2006.
- Chaira, T. *Medical Image Processing*. CRC Press, 2015. ISBN 978-1-4987-0047-4.
- Cheung, G., Magli, E., Tanaka, Y., y Ng, M. K. Graph spectral image processing. *Proceedings of the IEEE*, 106(5):907–930, May 2018. doi: 10.1109/JPROC.2018.2799702.
- Conrad, C., Mertz, M., y Mester, R. Contour-relaxed superpixels. In Heidelberg, S.-V. B., editor, *booktitle*, pages 80–293, 2013.
- Domínguez, L. D. Extracción de superpíxeles utilizando agrupamiento intuicionista local. Tesis de maestría, Departamento de Ciencias de la Computación, Centro Nacional de Investigación y Desarrollo Tecnológico, 2019.
- Gong, S., Liu, C., Ji, Y., Zhong, B., Li, Y., y Dong, H. *Advanced Image and Video Processing Using MATLAB*. Springer International Publishing AG, 2019. ISBN 978-3-319-77223-3. doi: 10.1007/978-3-319-77223-3.
- Gonzalez, R. C. y Woods, R. E. *Digital Image Processing Second Edition*. Prentice-Hall, 2002. ISBN 0-201-18075-8.
- Haußecker, H. y Tizhoosh, H. R. *Fuzzy Image Processing*. Academic Press, 1999. ISBN 0–12–379772-1.

- Haußecker, H. y Tizhoosh, H. R. *Fuzzy Image Processing*, chapter 22, pages 683–728. Academic Press, 1999. ISBN 0–12–379772-1. doi: 10.1016/B978-012379777-3/50017-0.
- Khokher, M. R., Ghafoor, A., y Siddiqui, A. M. Graph cuts image segmentation using fuzzy rule based system. *Radioengineering*, 21(4):1236–1245, 2012. ISSN 1805-9600.
- Liu, G., Zhang, Y., y Wang, A. Incorporating adaptive local information into fuzzy clustering for image segmentation. *IEEE Transactions on Image Processing*, 24(11): 3990–4000, November 2015.
- Liu, M.-Y., Tuzel, O., Ramalingam, S., y Chellappa, R. Entropy rate superpixel segmentation. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR '11*, page 2097–2104, USA, 2011. IEEE Computer Society. ISBN 9781457703942. doi: 10.1109/CVPR.2011.5995323. URL <https://doi.org/10.1109/CVPR.2011.5995323>.
- Mathew, S., Mordeson, J. N., y Malik, D. S. *Fuzzy Graph Theory*, volume 363 of *Studies in Fuzziness and Soft Computing*. Springer International Publishing, 1 edition, 2018. ISBN 978-3-319-71406-6. doi: 10.1007/978-3-319-71407-3.
- Meyer, F. Color image segmentation. In IET, editor, *1992 International Conference on Image Processing and its Applications*, pages 303–306. IET, 1992.
- Meyer, F. *Topographical Tools for Filtering and Segmentation*, volume 1. Wiley, 2019.
- Morales, M. X. Segmentación no paramétrica de tejidos cerebrales mediante una arquitectura paralela de redes neuronales convolucionales. Tesis de maestría, Departamento de Ciencias de la Computación, Centro Nacional de Investigación y Desarrollo Tecnológico, 2018.
- Murashov, D. Theoretical-information quality model for image segmentation. In *Procedia Engineering*, 2017. doi: 10.1016/j.proeng.2017.09.603.
- Poudel, A., Li, S., Chu, T., Palmer, D., y Engelbart, R. An intelligent systems approach for detecting delamination defects due to impact damage in cfrp panel by using ultrasonic testing. In *Proceedings of the 2011 ASNT Fall Conference, Palm Springs, CA*, October 2011.
- Pramanik, T., Pal, M., Mondal, S., y Samanta, S. A study on bipolar fuzzy planar graph and its application in image shrinking. In *Journal of Intelligent and Fuzzy Systems*, 2018. doi: 10.3233/JIFS-171209.
- Ren, X. y Malik, J. Learning a classification model for segmentation. In *Proc. 9th Int. Conf. Computer Vision*, pages 10–17, 2003.

- Ríos, E. A. Seguimiento y predicción del movimiento de objetos rígidos en un ambiente complejo. Tesis de maestría, Departamento de Ciencias de la Computación, Centro Nacional de Investigación y Desarrollo Tecnológico, 2004.
- Samanta, S., Sarkar, B., Shin, D., y Pal, M. Completeness and regularity of generalized fuzzy graphs. *SpringerPlus*, 5(1):1979, nov 2016. ISSN 2193-1801. doi: 10.1186/s40064-016-3558-6. URL <https://doi.org/10.1186/s40064-016-3558-6>.
- Stutz, D., Hermans, A., y Leibe, B. Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding*, 2018. ISSN 1090235X. doi: 10.1016/j.cviu.2017.03.007.
- Suárez, D. G. S. Sintonización de una red totalmente conectada para segmentación de dos clases de objetos en imágenes. Tesis de maestría, Departamento de Ciencias de la Computación, Centro Nacional de Investigación y Desarrollo Tecnológico, 2018.
- Tamajka, M. y Benešová, W. Supervoxel algorithm for medical image processing. In *IEEE International Conference on Power, Control, Signals and Instrumentation Engineering, ICPCSI 2017*, 2018. ISBN 9781538608135. doi: 10.1109/ICPCSI.2017.8392300.
- Toennies, K. D. *Guide to Medical Image Analysis*. Advances in Computer Vision and Pattern Recognition. Springer, second edition edition, 2017. ISBN 978-1-4471-7320-5.
- Trémeau, A. y Colantoni, P. Region adjacency graph applied to color image segmentation. *IEEE transactions on image processing*, 9(4):735–744, 2000.
- Van den Bergh, M., Boix, X., Roig, G., de Capitani, B., y Van Gool, L. Seeds: Superpixels extracted via energy-driven sampling. In Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., y Schmid, C., editors, *Computer Vision – ECCV 2012*, pages 13–26, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33786-4.
- Vargas, D. M., Gallegos-Funes, F. J., y Cruz-Santiago, R. Segmentation of noisy images using the rank m-type l-filter and the fuzzy c-means clustering algorithm. In *Mexican Conference on Pattern Recognition*, pages 184–193. Springer, 2011.
- Zhang, Q., Xue, M. G., y Zhou, P. C. Turbopixel with fuzzy edge detection and multiregional kernel graph cuts for image segmentation. In *Proceedings - 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2017*, 2018. ISBN 9781538619377. doi: 10.1109/CISP-BMEI.2017.8301984.
- Zimmermann, H.-J. *Fuzzy Set Theory – and Its Applications*, volume 2001. Kluwer Academic Publishers, 01 2001. ISBN 978-94-010-3870-6. doi: 10.1007/978-94-010-0646-0.

Índice alfabético

- Algoritmo de sobresegmentación de imágenes a color mediante un grafo difuso basado en una medida de distancia espacial y color, 52
- Arista, 9
- BSDS500 (repositorio), 57
- Conjunto difuso, 11
- Contour Relaxed Superpixels (CRS), 18
- Eikonal Region Growing Clustering (ERGC), 19
- Entropy Rate Superpixels (ERS), 19
- Error de subsegmentación (UE), 17
- Grafo, 9
- Grafo de adyacencia de regiones, 26
- Grafo difuso, 13
- Incertidumbre en imágenes, 11
- Intuitionistic Fuzzy C-Means (IFCM), 23
- Métricas sobresegmentación, 16
- Nodo, 9
- Recuerdo de límites (Rec), 16
- Region Adjacency Graph (RAG), 26
- Segmentación de imágenes, 5
- SLIC, 20
- Sobresegmentación de imágenes, 7
- Subgrafo, 10
- superpixel, 7
- Superpixels Extracted via Energy-Driven Sampling (SEEDS), 20
- Teoría de conjuntos difusos, 10
- Teoría de grafos, 9
- Variación explicada (EV), 17
- Watershed (W), 21
- Weizmann (repositorio), 58