

UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

Detección de células Cancerosas en imágenes de Neoplasia Intraepitelial Cervical (NIC)

María Martínez Samboal

Junio 2012

Detección de células Cancerosas en imágenes de Neoplasia Intraepitelial Cervical (NIC)

AUTOR: María Martínez Samboal

TUTOR: Jesús Bescós Cano



Video Processing and Understanding Lab
Dpto. de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio de 2012

Palabras Clave

PAP, segmentación, Grey Level Method, Energy Method, características: textura, intensidad, extracción de núcleos de células, Chang, Plisitti.

Resumen

Actualmente el cáncer cervical uterino o NIC (Neoplasia intraepitelial Cervical) es el segundo cáncer más común en la mujer y cada año se reportan 500.000 nuevos casos; está sólo por debajo del cáncer de mama [1]. Este trabajo se enmarca en un proyecto para mejorar el sistema de diagnóstico precoz de este tipo de cáncer en el medio rural de México, haciendo aportaciones a un sistema portátil que intenta automatizar el proceso de captura de imágenes de muestras de tejido y realizar una clasificación inicial de las imágenes más adecuadas para ser enviadas a un experto de un hospital central.

La metodología aplicada a lo largo del proyecto, por uso en sistemas similares, está basada en un conjunto de algoritmos de reconocimiento de objetos, detección de bordes, operaciones morfológicas, etc.

Teniendo en cuenta el tipo de imágenes que poseemos en nuestro dataset, imágenes PAP (extraídas de la prueba de Papanicolaou), hemos realizado un estudio de las distintas técnicas actuales. Una vez realizado el estudio seleccionamos los algoritmos más adecuados para nuestro tipo de imagen.

El primer trabajo seleccionado es el descrito en el artículo de [1], que está dividido en dos técnicas que se aplican a las imágenes por separado extrayendo unos resultados parciales que posteriormente se complementan para obtener el resultado final del algoritmo. Estos métodos utilizados siguen un esquema básico: preprocesado de la imagen (para eliminar los efectos indeseados que poseen las imágenes, como por ejemplo el ruido o el bajo contraste), segmentación (extrae las regiones deseadas, en este caso los núcleos) y por último clasificación de las regiones en núcleos anormales o normales. La primera técnica “Grey Level Method” trabaja sobre la información o escala de grises de la imagen, y la segunda “Energy Method”, por el contrario, aprovecha las tres bandas de color, aportando distintos resultados que se complementan para obtener los resultados finales (clasificación de núcleos en normales o anormales). A este artículo hemos tenido que añadir una etapa previa de adaptación de las imágenes debido a que las imágenes que utiliza el artículo son imágenes uniformes y nuestro dataset no lo era.

El informe que describe este primer trabajo es poco detallado en su explicación y esto nos lleva a resultados poco precisos; por tanto realizamos la implementación de un segundo artículo más detallado que el anterior.

El segundo trabajo es un algoritmo de extracción de características descrito en el artículo de [2]. Está basado en combinar la forma, la textura e intensidad de las imágenes para extraer los núcleos de las células PAP sin confundirlos con otras zonas de la imagen que pueden considerarse como núcleos; para detectarlas usa: segmentación, aplicación operadores morfológicos para extraer los núcleos, y además utiliza la textura e intensidad para clasificar las regiones extraídas en núcleos / no núcleos y así evitar resultados erróneos.

Tras realizar algunas modificaciones sobre los algoritmos propuestos, éstos han sido implementados y evaluados con el fin de encontrar un modelo que permita reconocer las células PAP, con independencia de las imágenes capturadas. La evaluación de estos algoritmos se ha hecho sobre una colección de imágenes capturadas por dicho microscopio: un dataset de 88 imágenes PAP, las cuales contenían un total de 857 núcleos reales.

Conforme al estudio hecho en este proyecto, se incluye un repaso detallado sobre todos los algoritmos implementados, acompañado por resultados cuantificados tras las simulaciones. De esta manera, cada algoritmo ha sido evaluado sobre una colección de imágenes reales, no sintéticas. Este es un marco de evaluación muy exigente y ambicioso que consigue replicar los resultados del estado del arte, que rondan el 85% de acierto.

Keywords

PAP, segmentation, Grey Level Method, Energy Method, features: texture, intensity, extract cell nuclei, Chang, Plisitti

Abstract

Currently uterine cervical cancer or CIN (cervical intraepithelial neoplasia) is the second most common cancer in women and each year 500,000 new cases are reported, is just below the breast cancer [5559114]. This work is part of a project to improve the system for early diagnosis of this cancer in rural Mexico, contributing to a portable system that attempts to automate the process of capturing images of tissue samples and perform an initial classification more adequate images to be sent to an expert from a central hospital.

The methodology used throughout the project, use similar systems, is based on a set of algorithms for object recognition, edge detection, morphological operations, etc..

Taking into account the type of images that we have in our dataset, images PAP (extracted Papanicolaou test), we studied the different techniques. Once the study selected the most appropriate algorithms for our type of image.

The first selected work is described in the article [5559114], which is divided into two techniques used to separate images by extracting some further partial results are combined to obtain the

final result of the algorithm. These methods are a basic scheme: preprocessing of the image (to eliminate the undesirable effects which have images, such as noise or low contrast), segmentation (extract the desired regions, in this case the nuclei) and finally classification of regions in nuclei abnormal or normal. The first technique "Grey Level Method" works on grayscale image and the second "Energy Method", by contrast, takes the three bands of color, bringing different results which are combined to obtain the final results (classification of nuclei in normal and abnormal). In this article we have had to add a previous stage adaptation of the images because the images used by the article are consistent images and our dataset was not.

The first report describing this work is very detailed in his explanation and this leads to inaccurate results and therefore we make the implementation of a second more detailed article than the last.

The second work is a feature extraction algorithm described in the article [Plissiti2011838]. It is based on combining shape, texture and intensity of the images to extract the nuclei of cells without confusing PAP with other areas of the image can be considered as nuclei, to detect used: segmentation, morphological operators application to extract the nuclei, and also uses the texture and intensity to classify the extracted regions in core / non core to avoid erroneous results.

After some modifications to the proposed algorithms, these have been implemented and evaluated in order to find a model to recognize PAP cells, regardless of the captured images. The evaluation of these algorithms has been done on a collection of images captured by the microscope: a dataset of 88 images PAP, which contained a total of 857 real cores.

Under study in this project includes a detailed review of all the algorithms implemented, accompanied by quantified results after the simulations. Thus, each algorithm has been evaluated on a collection of real images, not synthetic. This is a very demanding evaluation framework that achieves ambitious replicate the performance of state of art, of around 85% accuracy.

Agradecimientos

En primer lugar, quiero agradecer a las cuatro personas más importantes de mi vida y sin las cuales probablemente no hubiera sido capaz de llegar hasta aquí: mis padres, Ángel y M^a del Carmen, y mis hermanas, Gloria y Ana. Gracias por apoyarme día a día durante toda mi vida y en estos casi 7 años de carrera. Gracias a los cuatro por darme siempre esa suerte antes de salir de casa en todos y cada uno de los exámenes.

Gracias Mamá, porque cuando empecé la carrera todos me decían que era algo muy difícil y que me tenía que acostumbrar a suspender, pero siempre creíste en mí y eso solo fue el principio de todos los apoyos que me has ido dando, interesándote por asignaturas que ni siquiera comprendías, pero que en ese momento, en el que necesitaba que estuvieras parecía que lo entendías todo.

Gracias Papá, porque siempre me apoyaste y me ayudaste a elegir este camino. Siempre creíste que podía ser ingeniera; aunque todo el resto del mundo dudara, nunca ví que tú lo hicieras, eso me hizo no tambalearme y seguir adelante. Preguntándome por cada examen y transmitiéndome siempre la tranquilidad de que había más oportunidades y que no había prisa, esta carrera es una carrera de fondo, no es llegar el primero, lo importante es llegar.

Gracias a mi hermana Gloria, por ayudarme y transmitirme esa fuerza siempre. Porque siempre has creído en mí; y por el orgullo con que dices a los demás lo que estudio, siempre me has dado mucha fuerza para todo, pero en particular por prestarme a los muñequitos sin los cuales no iba a ningún examen, ya que si a ti te dieron suerte una vez a mí me han ayudado durante toda la carrera, además de acompañarme a mirar notas al tablón casi con más nervios de los que yo tenía, siempre siendo tan positiva, porque yo iba a aprobar seguro, y si no lo hacía, “Ves a la revisión, que seguro que se han equivocado”. Muchísimas gracias.

Gracias a mi hermana Ana, porque aun costándole me acompañaba a la biblioteca y me apoyaba, me hablaba de los circuitos y de los sistemas lineales como si supiera de ellos, aunque realmente no tuviera ni idea, pero siempre para sacarme fuerzas en los momentos donde las fuerzas ya empezaban a faltar y el ser positiva ya no era mi fuerte. Gracias por tus mensajes que recibía dos minutitos antes de empezar el examen, era el poquito de fuerza que me faltaba para hacerlo muy bien. Y por creer en mí de manera incondicional, porque cuando hablabas de mí me hacías sentir como que yo podía con todo, y ya se sabe que los buenos siempre ganan. Muchísimas gracias.

Gracias a mis abuelos, Isaac, Pablo, Gloria y Agustina y a mi tío Jose, porque no han podido estar para darme un abrazo y decirme lo orgullosos que están en persona, pero sé que lo están y me han acompañado en cada paso, se que esa es la fuerza más grande de todas.

Gracias a Sergio, por acompañarme todos estos años y por apoyarme en todo, entendiendo el tiempo que no he podido pasar contigo porque tenía que dedicarme a estudiar, y por comprender los nervios que me han supuesto todos estos de carrera, que quizás en muchos momentos has tenido que aguantar, pero siempre estando a mi lado.

Gracias a Javi, mi amigo y gran compañía durante toda la carrera, por ayudarme a entender todas las cosas que no era capaz, por aguantar mis momentos bajos, porque en los que se está bien todo es más fácil, por recordarme siempre que borraré unas prácticas pero siempre con una sonrisa, y por todos los momentos buenos que hemos vivido juntos, ya que para mí esta carrera no tendría el mismo sentido si todo este camino no lo hubiéramos andado juntos. Porque eres el primero que conocí al llegar aquí y tuve mucha suerte de que fueras tú.

Gracias a todos mis compañeros a los primeros que llegaron y supieron hacerme un hueco en sus vidas, los que compartieron muchas horas de prácticas, los que me dejaron apuntes, los que repitieron conmigo asignaturas, los que me deseaban suerte antes de los exámenes, los que te daban la enhorabuena por los aprobados y te animaban cuando eso no ocurría, los que festejaban conmigo el fin de las épocas de exámenes, y a todos ellos se unieran desde el principio o los que se han unido al final gracias por haber estado conmigo en esta carrera, en particular a Miguel por siempre estar ahí.

Gracias a Luis, porque llegaste muy pronto y te marchaste muy pronto, pero para mí nunca lo has hecho, siempre has estado junto a mi, durante todos estos años, sintiendo que esto también lo hacía por ti.

Gracias a mis amigas desde pequeñas: Elena, Naiara, Laura y las dos Cristinas (Jauset y Plaza), y a mi amigo Marco, que siempre confiaron en mi y me apoyaron en todos los momentos que he ido viviendo cuando todo era fácil y aun más difícil cuando no lo era.

Gracias a mi tutor Jesús, por la ayuda y dedicación que me ha prestado durante estos meses, y a mi compañera de laboratorio Isa.

Gracias a mis correctoras (Claudia, Ana y Gloria), que estos últimos momentos de la memoria han estado para leerse todo, sacándome un hueco aunque para ellas no fuera fácil.

Gracias a mis tíos y primos. En particular, a mi madrina y a mi tía Margari, por toda su preocupación.

Gracias a todos los que en algún momento se han preocupado por mí durante todo este camino, porque todos han ayudado a que llegue a la meta. En definitiva, GRACIAS A TODOS.

Índice general

1. Introducción	1
1.1. Motivación.	1
1.2. Objetivos.	4
1.3. Estructura del documento	5
2. Antecedentes y selección de trabajos	7
2.1. Introducción	7
2.1.1. Técnicas de Tinción	7
2.1.2. Técnicas de Segmentación	9
2.1.2.1. GVF- <i>Snake</i> (<i>Gradient Vector Flow</i>)	9
2.1.2.2. Contornos Activos (ACM)	13
2.1.2.3. Segmentación por <i>Watershed</i> - Transformada de <i>watershed</i>	15
2.1.2.4. Segmentación por umbralización	19
2.1.2.5. Segmentación en imágenes a color	20
2.1.2.6. Segmentación con técnicas de clusterización	22
2.1.3. Selección de algoritmos	28
2.1.4. Introducción a los algoritmos seleccionados	30
2.1.4.1. Estudio de la segmentación para la extracción de núcleos en imágenes médicas	30
2.1.4.2. Introducción a los métodos <i>Grey Level Method</i> y <i>Energy Method</i>	32
2.1.4.3. Introducción al algoritmo de extracción de características	33

3. Alg. basado en niveles de gris y energía de las bandas de color	35
3.1. Introducción	35
3.2. Adaptación de las imágenes del dataset	37
3.3. Desarrollo del algoritmo	40
3.3.1. <i>Grey Level Method</i>	40
3.3.1.1. Preprocesado	41
3.3.1.2. Segmentación	45
3.3.1.3. Clasificación de regiones segmentadas	47
3.3.1.4. Resultados de <i>Grey Level Method</i>	49
3.3.2. <i>Energy Method</i>	51
3.3.2.1. Preprocesado	52
3.3.2.2. Resultados de <i>Energy Method</i>	55
3.4. Resultados finales del algoritmo	56
3.5. Análisis Critico del Algoritmo	58
3.6. Propuestas de Mejora del algoritmo	59
4. Alg. basado en la extracción de características de los núcleos	61
4.1. Introducción	61
4.2. Detección de Marcadores de Núcleo	63
4.2.1. Preprocesado	64
4.2.1.1. Ecualización	64
4.2.1.2. Umbralización	65
4.2.1.3. Dilatación.	67
4.2.2. Estimación de los centroides de los núcleos candidatos	68
4.2.2.1. Transformación h-minima	68
4.2.2.2. Reconstrucción Morfológica	70
4.2.2.3. Detección de regiones mínimas, a través del Método “Non Máxima Suppression”	71
4.2.2.4. Extracción de Centroides	73
4.3. Segmentación	74

4.3.1.	Gradiente de la imagen	74
4.3.1.1.	Gradiente morfológico en color de la imagen	74
4.3.1.2.	Gradiente Morfológico de una imagen en escala de grises	75
4.3.2.	<i>Transformada de Watershed</i>	76
4.4.	Extracción y selección de características	77
4.4.1.	Extracción de características	77
4.4.1.1.	Características de Forma	80
4.4.1.2.	Características de Textura	81
4.4.1.3.	Características de Intensidad	83
4.4.2.	Selección de características	84
4.5.	Agrupamiento de características y clasificación de regiones	85
4.5.1.	Agrupamiento de características	86
4.5.2.	Clasificación de regiones	87
4.6.	Resultados finales del algoritmo	87
4.7.	Análisis Critico del Algoritmo	89
4.8.	Propuestas de Mejora del algoritmo	90
5.	Conclusiones	91
5.1.	Conclusiones	91
5.1.1.	Conclusiones del algoritmo Grey Level Method y Energy Method	91
5.1.2.	Conclusiones del Algoritmo de extracción de características	92
5.1.3.	Conclusiones finales	93
5.2.	Lineas de Trabajo Futuras	94
	Appendix.	95
A.	Herramientas de procesado de imágenes	95
A.1.	Ecualización del histograma	95
A.2.	Algoritmo de Canny	96
A.3.	CLAHE (<i>Contrast Limited Adaptive Histogram Equalization</i>)	98
A.4.	Método de Otsu	100

A.5. Operadores morfológicos: dilatación y erosión	103
A.5.1. Dilatación y Erosión	103
A.5.1.1. Dilatación	103
A.6. Gradiente morfológico de color	105
A.7. Local Binary Patterns (LPB)	107
A.7.0.2. Algoritmo LBP [3]	108
B. Herramientas de selección de variables	115
B.1. mRMR(mínima Redundancia-Máxima Relevancia)	115
I Presupuesto	121
II Pliego de condiciones	125

Índice de figuras

1.1. Tipos de cáncer en países subdesarrollados.	2
1.2. Anatomía Femenina	2
1.3. Tipos de células del epitelio escamoso del cérvix.	3
2.1. Diagrama de flujo del proceso de extracción de células	7
2.2. Método tradicional de Snake. (Figura extraída de [4])	11
2.3. GVF Snake. (Figura extraída de [4]).	12
2.4. Ejemplo 1 GVF Snake. (Figura extraída de [4])	12
2.5. Ejemplo 2 GVF-Snake en células. (Figura extraída de [4]).	13
2.6. Ejemplo de ACM. (Figura extraída de [5]).	14
2.7. Ejemplo ACM con células. (Figura extraída de [5]).	14
2.8. Problemas de ACM. (Figura extraída de [5]).	15
2.9. Línea de Cresta.	16
2.10. Fases de <i>Watershed</i> . Figura extraída de [6].	17
2.11. Eliminación de contornos irrelevantes. (Figura extraída de [6]).	18
2.12. Segmentación por umbral.	19
2.13. Segmentación de imágenes a color. Imagen autoexplicativa.	20
2.14. Segmentación de imágenes a color.	22
2.15. <i>Clusters</i>	23
2.16. Matriz de datos y <i>clusters</i>	23
2.17. Centroides de la matriz de entrenamiento.	24
2.18. División en regiones de <i>Kmeans</i>	25

2.19. <i>Parametric Fitting</i>	26
2.20. Comparación de imágenes con células solapadas de los algoritmos de “ <i>Kmeans y Parametric Fitting</i> ”.	26
2.21. SVM.	27
2.22. SVM con combinación no lineal de RBF.	28
2.23. Tipos de Imágenes del estudio extraídas de nuestro dataset.	28
2.24. Proceso Global de la Segmentación de Imágenes.	30
2.25. Simplificación: Eliminación de ruido	31
2.26. Diagrama de flujo del primer artículo implementado. (Figura extraída de[1]).	32
3.1. Diagrama de flujo.	36
3.2. Extracción de Imagen Uniforme.	38
3.3. Diagrama del Algoritmo de baricentro	39
3.4. Diagrama de flujo “ <i>Grey Level Method</i> ” (Figura extraída de [1]).	41
3.5. Ejemplos de Imagen Clara e Imagen oscura del dataset.	42
3.6. División de la imagen en sus componentes YUV.	42
3.7. Ecualización sobre la imagen clara.	43
3.8. Imagen Umbralizada.	44
3.9. <i>Line-Scanning</i> Imagen Clara.	45
3.10. Imagen Clara con contornos.	46
3.11. Imagen obtenida tras <i>bwconncomp</i>	47
3.12. Características de núcleo.	49
3.13. Proceso Imagen Oscura.	49
3.14. Contornos en Imagen Oscura.	50
3.15. Selección de Núcleos Anormales	50
3.16. Diagrama de flujo de <i>Energy Method</i> . (Figura extraída de [1]).	51
3.17. Preprocesado de <i>Energy Method</i> : Filtros medianas.	52
3.18. <i>Line-Scanning</i>	54
3.19. Preprocesado + Segmentación de <i>Energy Method</i>	55
3.20. Resultados de <i>Energy Method</i>	56

3.21. Resultados finales de cada método por separado	57
3.22. Resultado final algoritmo <i>Grey Level Method + Energy Method</i>	57
4.1. Diagrama de Flujo del algoritmo de extracción de características.	62
4.2. CLAHE aplicado a la imagen.	65
4.3. Umbralizada con Método Otsu.	66
4.4. Operador morfológico de dilatación.	67
4.5. Unión de las tres componentes dilatadas.	68
4.6. Función “Imhmin”.	69
4.7. Aplicación de Transformada h-minima.	69
4.8. Reconstrucción morfológica.	70
4.9. Reconstrucción Morfológica de la imagen.	71
4.10. <i>Non Maxima Suppresion</i>	72
4.11. Método <i>Non Maxima Supresion</i> aplicada a la Imagen.	72
4.12. Imagen original con el resultado de la extracción de Centroides.	73
4.13. .Detector de Canny.	75
4.14. Gradiente Morfológico en escala de grises.	76
4.15. <i>Watershed</i>	76
4.16. Relación de las intensidades de los núcleos verdaderos con otras zonas.	78
4.17. A, B y Ngh.	79
4.18. Extracción de área Ngh.	80
4.19. Características de forma . Figura extraída de [2]	81
4.20. Topología de los vecinos: a)Circular b)Hipérbola. Figura extraída de[7][3]	82
4.21. Tabla de características de texturas. Figura extraída de [2].	82
4.22. Tabla de características de intensidad. Figura extraída de [2]	83
4.23. Detección de Centroides erróneos de las regiones mínimas.	87
4.24. Resultados artículo.	89
A.1. Ejemplo de ecualización de histograma.	96
A.2. Direcciones de Búsqueda para cerrar los contornos abiertos.	98

A.3. Algoritmo de Canny.	98
A.4. Ejemplo de imagen mejorada mediante el algoritmo CLAHE.	99
A.5. Método de Otsu.	102
A.6. Composición de dilatación.	103
A.7. Composición de Erosión.	104
A.8. Propiedades de Erosión: distribución y composición.	105
A.9. Ejemplos de resultados de la aplicación de: dilatación y erosión.	105
A.10. Gradiente Morfológico.	106
A.11. Procedimiento del cálculo del LBP original. Figura extraída de [3].	107
A.12. Topologías de Vecindarios. Figura extraída de [7].	108
A.13. Simetría circular para Conjuntos de Vecinos. Figura extraída de [7].	108
A.14. Simetría circular $LBP_{8,R}^{ri}$. Figura extraída de [7].	110
A.15. Tipos de células del artículo. Figura extraída de [7].	113
A.16. Resultados de imágenes PAP distintas topologías. Figura extraída de [7].	114

Índice de cuadros

4.1. 57 Características seleccionadas.	84
4.2. 16 características más discriminantes extraídas con mRMR	85
4.3. Porcentajes Medios de regiones extraídos como núcleos de más en la imagen	88
4.4. Resultados en función del número de clusters empleados.	89
5.1. Resultados del algoritmo de extracción de características	93

Capítulo 1

Introducción

1.1. Motivación.

El cáncer cervicouterino (CaCu) representa el segundo puesto en el ranking de mortalidad en el mundo [1]. El cáncer cervicouterino, una clase común de cáncer en la mujer, es una enfermedad en la cual se encuentra células cancerosas (malignas) en los tejidos del cuello uterino. El cuello uterino es la abertura del útero.

El cáncer cervicouterino suele crecer lentamente por un período de tiempo. Antes de que se encuentre células cancerosas en el cuello uterino, sus tejidos experimentan cambios y empiezan a aparecer células anormales (proceso conocido como displasia). La prueba de Papanicolaou generalmente encuentra estas células; por ello se utiliza esta técnica. Posteriormente, las células cancerosas comienzan a crecer y se diseminan con mayor profundidad en el cuello uterino y en las áreas circundantes.

Ya que en general no hay síntomas asociados con el cáncer cervicouterino, el médico debe hacer una serie de pruebas para buscar el cáncer. La primera prueba es la de Papanicolaou, que explicaremos más abajo.

El pronóstico (posibilidades de recuperación) y la selección del tratamiento dependen de la etapa en que se encuentra el cáncer (si se encuentra en el cuello uterino o si se ha diseminado a otros lugares) y el estado de salud en general de la paciente.

Las muertes por CaCu han disminuido en un 70 % en las últimas 5 décadas como resultado de la introducción de la prueba de citología cervical en la década de los 40 (la cual extrae células del cuello uterino para analizar), pasando de ser una de las principales causas de muerte a un lugar 13, pero siendo aún a primera o segunda causa de muerte de mujeres en países en desarrollo (véase la figura 1.1 para observar los porcentajes de mortalidad de cada tipo de cáncer el tipo CaCu es el que hace referencia la matriz). En México, en particular, cada 2 horas muere una mujer; son 4380 casos que no son detectados a tiempo. Para evitar el Cáncer Cervicouterino el

gobierno ha realizado programas para buscar métodos de actuación efectivos para la prevención de esta enfermedad.

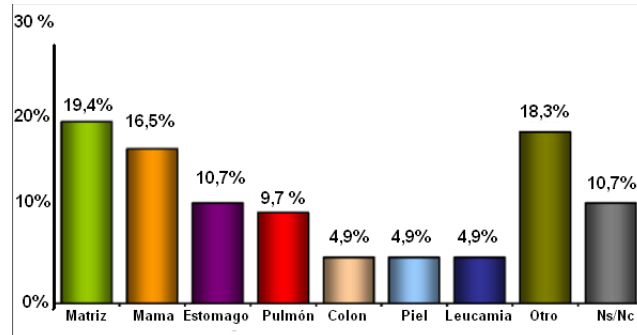


Figura 1.1: Tipos de cáncer en países subdesarrollados.

En la figura observamos la distribución de mortalidad según los tipos de cáncer. En ella el cáncer cervicouterino viene dado como cáncer de matriz (matriz del útero). Y como podemos ver es el que mayor porcentaje de mortalidad posee.

Para extraer y analizar estas células se utiliza el examen de Papanicolaou [8, 9] (como mencionamos anteriormente), presentado en 1941 como un método de detección de carcinomas en los genitales femeninos, el cual ha demostrado su efectividad desde hace más de medio siglo. Los resultados de los estudios estadísticos que realizaron al examen de Papanicolaou [8, 9], fueron contundentes, porque el uso de esta técnica permite diagnosticar un buen número de casos de neoplasia asintomática que no es visible por el ojo humano y necesitaba forzosamente de biopsia para ser detectada cuando ya se había extendido la enfermedad. Esta técnica se lleva a cabo usando un pedazo de algodón, un cepillo o una espátula de madera pequeña para raspar suavemente el exterior del cuello uterino con el fin de recoger células, estas muestras de células son recogidas de los conductos cervicales del útero (tanto de la parte interna, endocérvix, como de la externa, exocérvix) (véase la figura 1.2 que muestra la parte exacta de la anatomía femenina de dónde se extraen las células). Además de ser un método sencillo, económico, puede utilizarse de manera masiva y permite detectar el cáncer en su etapa preinvasiva .



Figura 1.2: Anatomía Femenina

Mediante este método es posible observar características en las células de forma clara en el mi-

croscopio. El propósito de la clasificación del examen de Papanicolaou es diagnosticar los cambios pre-malignos de las células antes de que se transformen a carcinoma invasivo. Las muestras contienen dos tipos de células: de epitelio columna y de epitelio escamoso. Estas a su vez se dividen en cuatro capas: basal, parabasal, intermedia y superficial, cuando las células son normales; y cuando son anormales se subdividen en suave, moderada y displasia severa (véase la figura 1.3 para observar los tipos de células anteriormente citados).

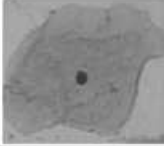
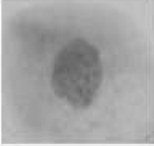





Normal cells		Abnormal cells	
<p>Superficial squamous 1</p> <ul style="list-style-type: none"> ● Shape: Flat/oval ● Nucleus very small ● N/C very small 			<p>4 Mild dysplasia</p> <ul style="list-style-type: none"> ● Nucleus light/large ● N/C medium
<p>Intermediate squamous 2</p> <ul style="list-style-type: none"> ● Shape: Round ● Nucleus large ● N/C small 			<p>5 Moderate dysplasia</p> <ul style="list-style-type: none"> ● Nucleus large/dark ● Cytoplasm dark ● N/C large
<p>Columnar 3</p> <ul style="list-style-type: none"> ● Shape: Column-like ● Nucleus large ● N/C medium 			<p>6 Severe dysplasia</p> <ul style="list-style-type: none"> ● Nucleus large/dark/deform ● Cytoplasm dark ● N/C very large
			<p>7 Carcinoma in situ</p> <ul style="list-style-type: none"> ● Nucleus large/dark/deform ● N/C very large

Figura 1.3: Tipos de células del epitelio escamoso del cérvix.

Vemos siete tipos, de los cuales los que se presentan en la columna de la izquierda son células normales, mientras que por el contrario los de la derecha son los que hace referencia a las células cancerígenas. Viendo que las células que poseen una relación núcleo/ citoplasma menor son las malignas. Figura extraída de [10].

Cuando el sistema ha terminado de escanear el portaobjetos, de las millones de imágenes que toma, identifica las células más significativas que pueden ser anormales, para una posterior evaluación visual por el especialista. Cuando analiza células normales las imágenes son archivadas y sometidas a un procedimiento de control de calidad como revisión rápida, pero cuando detecta células anormales las imágenes son pasadas para la revisión manual.

La motivación principal es que este tipo de enfermedad depende de la etapa donde sea detectado.

Este trabajo se plantea para mejorar el diseño y realizar aportaciones científicas a un “Sistema semiautomático de captura y análisis de imágenes de NIC (Neoplasia intraepitelial cervical)”. Dentro de este sistema, el cual se haya dentro del campo de la visión artificial (VA), se distingue el subsistema de procesamiento de imágenes, en este caso aplicado al dominio de las imágenes biomédicas.

Más específicamente en este proyecto nos centraremos en el análisis comparativo de algoritmos de segmentación y de preprocesado o procesado previo de las imágenes.

1.2. Objetivos.

El objetivo principal de este trabajo es el análisis comparativo de algoritmos de segmentación y preprocesado de las imágenes para mejorar la calidad de las mismas, de cara a localizar las imágenes más adecuadas y sospechosas de contener tejido canceroso. Este estudio consistirá en evaluar los diferentes algoritmos implementados en términos de reconocimiento de los núcleos existentes en las imágenes. Este reconocimiento se llevará a cabo en una colección de 90 imágenes reales .

Los algoritmos serán seleccionados entre el estado del arte, teniendo en cuenta el tipo de imágenes que poseemos; los algoritmos elegidos serán implementados y evaluados. Más concretamente:

1. Crear base de datos: Recopilación de imágenes, de distintos tipos, de células cancerígenas.
2. Estudio del estado del arte: Analizaremos los tipos de imágenes y seleccionaremos las técnicas más relevantes para proceder a su implementación y comparación de resultados proporcionados por el autor.
3. Preprocesamiento: Se investigará en la etapa de procesamiento cuál es la mejor opción para que la imagen sea mejorada.
4. Segmentación: Lograr una segmentación adecuada de la imagen de la célula. Todo esto, se investigará por medio de un método o combinación de estos, y, con ello, se logrará obtener mejores resultados para la representación y descripción.
 - a) Método de Segmentación en escala de grises [1].
 - b) Método de Segmentación de Energía [1].
 - c) Método de Segmentación por división de regiones [2].
5. Identificación.- Una vez segmentada la imagen se realizarán pruebas para comprobar la identificación del núcleo de la célula anormal y normal, además, de acuerdo al estudio del estado del arte, evaluaremos los resultados y parámetros obtenidos realizando un análisis comparativo de las técnicas implementadas.
6. Aplicación. Se integrarán los algoritmos de acuerdo a los mejores resultados para la identificación de las células NIC al sistema.

Las tareas previas serán seguidas por un estudio exhaustivo de cada algoritmo puesto en práctica, conduciendo a la introducción de posibles mejoras y futuras líneas de investigación.

1.3. Estructura del documento

El contenido de este documento está organizado para atender los objetivos que se describen en la sección anterior.

En el capítulo 1, dónde nos encontramos, hemos visto la motivación, objetivos y la estructura del documento.

En el capítulo 2, estudiaremos el estado del arte de las técnicas de preprocesado y segmentación de las mismas relacionadas con el proyecto. Mostrando una selección de los algoritmos en función de los tipos de imágenes que poseemos, las cuales mostramos en este capítulo. Una vez conocemos los tipos de imágenes que poseemos seleccionaremos los algoritmos y realizaremos una introducción de los mismos.

Por otra parte, en el capítulo 3, se describe el algoritmo que está compuesto por los siguientes métodos: método de escala de grises “*Grey Level Method*”, y método de energía “*Energy Method*” [1] .

A continuación en el capítulo 4, desarrollaremos el algoritmo de extracción de características, [2].

En los capítulos 3 y 4 realizaremos, además de la explicación de los algoritmos, un estudio con los respectivos resultados de los algoritmos implementados y la evaluación de los resultados de las pruebas realizadas a los mismos.

Finalmente, capítulo 5 y último, que contiene las conclusiones y las líneas de trabajo futuras.

Capítulo 2

Antecedentes y selección de trabajos

2.1. Introducción

En primer lugar mostraremos los pasos realizados para la extracción de células de imagen médica:



Figura 2.1: Diagrama de flujo del proceso de extracción de células

En este diagrama de flujo podemos apreciar las distintas etapas que seguirá nuestro proyecto, centrándonos en el bloque de segmentación principalmente.

2.1.1. Técnicas de Tinción

Según se ha comentado anteriormente, existen diferentes sistemas comerciales de ayuda para los citopatólogos como PAPNET, Autopap, Thinprep, CPap, Apap, Champ, etc.; dichos sistemas son formas de hacer que las tinciones de Papanicolaou sean más exactas para reconocer las células anormales mediante segmentación. Esta etapa también la realizarán los métodos PAPNET, AutoPap y Champ.

Así, PAPNET [11], teniendo como iniciativa crear una base de datos de células anormales a partir de la prueba de Papanicolaou; es un sistema que combina la velocidad de un programa de bajo nivel con la toma de decisión mediante inteligencia artificial usando redes neuronales para seleccionar las regiones de interés. Cuando el sistema ha terminado de escanear el portaobjetos, de las 128 imágenes que toma, identifica las células más significativas que pueden ser anormales, para una posterior evaluación visual por el especialista.

Este sistema se diseñó inicialmente para volver a seleccionar imágenes extraídas de pruebas de Papanicolaou que resultaron negativas, para eliminar los casos de falsos negativos, pero también puede utilizarse para una selección primaria eficaz.

A raíz de la exploración PAPNET, las imágenes digitalizadas se sometieron y se clasificaron como anormal o negativo. Todas las anormales tenían una selección manual completa, mientras que las negativas tenían una selección parcial. Los resultados de cada método de detección se compararon y los casos discordantes fueron revisados.

La conclusión es que PAPNET es prácticamente igual a la selección manual del experto en la detección de una amplia gama de anomalías en las células y es más eficaz en la detección de anomalías en el extremo inferior del espectro anormal. Teniendo en cuenta que utiliza una técnica de selección por umbral, la decisión del mismo influye de una manera importante en el resultado final.

Otro método utilizado es AutoPap [12], un sistema computerizado, el cual analiza y guarda de forma automática todas las imágenes de Papanicolaou. Cuando analiza células normales las imágenes son archivadas y sometidas a un procedimiento de control de calidad como revisión rápida, pero cuando detecta células anormales las imágenes son pasadas para la revisión manual.

El método de AutoPap fue diseñado para buscar las anomalías lámina a lámina, y para excluir el 25% de las láminas con el riesgo más bajo. Estas láminas eran automáticamente excluidas de la lista, que requería revisión microscópica manual y así reducir el trabajo de los examinadores un 25%. Siendo capaz de detectar falsos-negativos, es decir, detectar células que han sido consideradas anormales, cuando realmente no lo son. El resultado es una reducción significativa en el laboratorio de falsos negativos si este dispositivo se utiliza en la práctica habitual.

El método Thinprep es una forma de preparar la muestra de células y hace más fácil reconocer las anomalías.

Esto me lleva a citar el sistema Champ, referenciado en [13], el cual es un sistema muy completo, ya que se dedica exclusivamente al análisis de imágenes de Papanicolaou. A diferencia de otros sistemas, se basa en un algoritmo de reconocimiento de objetos en base al color, detección de bordes y morfología. En base a estas tres características realiza una segmentación por clusterización, esto es, divide en diferentes clases las distintas áreas extraídas y las clasifica en un tipo de célula u otro.

Todos los sistemas comerciales anteriormente citados, además de ser técnicas de tinción, trabajan con diferentes técnicas de segmentación para el reconocimiento de células benignas o malignas. Esta etapa la vimos en el diagrama de bloques anterior (véase la figura 2.1 para observar las distintas etapas).

2.1.2. Técnicas de Segmentación

La segmentación de núcleos es una tarea clásica en el análisis de imágenes de citologías. Se basa en la partición de la imagen en regiones u objetos con características similares, y la posterior identificación de los núcleos, que son diferenciados del resto y etiquetados.

Las técnicas de segmentación en general son muy dependientes del propósito de la aplicación y del tipo de imágenes a analizar. Antes de segmentar es preciso definir qué objetos interesa determinar. Tras la segmentación es posible realizar operaciones de filtrado (a nivel de objetos), así como determinar características que permitan clasificar los objetos.

Una buena segmentación es difícil de evaluar, pero fundamentalmente, lo que se busca es que objetos diferentes tengan valores claramente diferentes de las características seleccionadas, es decir, que éstas sean lo más discriminantes posible.

Esta etapa tiene un peso especial dentro del proceso global de diagnóstico: una buena segmentación de los núcleos permite una óptima extracción de características, y por consiguiente, una clasificación más eficiente de los núcleos. La principal dificultad que existe en este proceso es la presencia en las imágenes de núcleos solapados, imágenes a color, imágenes más oscuras o más claras, que dificultan este proceso. Por ejemplo, la separación de núcleos solapados se realiza normalmente de modo manual.

En la actualidad hay diferentes técnicas de segmentación, como se cita en los artículos [1, 14].

Las técnicas de segmentación más utilizadas son:

- GVF-Snake (*Gradient Vector Flow*), segmentación basada en transiciones.
- ACM (*Active Contours*), segmentación basada en transiciones.
- *Watershed*, segmentación de homogeneidad, crecimiento de regiones.
- Segmentación por Umbralización.
- Segmentación por clusterización: *K-means*, *Parametric Fitting*, *Spectral clustering*, *Support Vector Machine (SVM) con kernel Radial basis function (RBF)*.

Dichas técnicas las explicaremos a continuación.

2.1.2.1. GVF-Snake (*Gradient Vector Flow*)

Dicha técnica [4] está basada en las transiciones que se producen a lo largo de la imagen, entre sus niveles de intensidad, ya que estos referencian los diferentes objetos de la imagen.

Los métodos de segmentación basada en transiciones aplican un detector de bordes sobre la imagen o sobre un plano de características.

Las regiones se definen a partir de las fronteras delimitadas, que presentan energía mínima. La técnica general para detectar los bordes, y por tanto reducir dicha energía al mínimo, es medir el gradiente de la imagen 'f' en una dirección determinada 'θ' a lo largo de una recta 'r', la cual extrae los vectores gradiente de un mapa de nivel de grises o binarios de una imagen.

Para extraer el gradiente aplicaremos las siguientes ecuaciones:

$$\frac{\partial f}{\partial r} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial r} = f_x \cos\theta + f_y \sin\theta \quad (2.1)$$

$$\left(\frac{\partial f}{\partial r}\right)_{max} \Rightarrow \frac{\partial}{\partial \theta} \frac{\partial f}{\partial r} = 0 = -f_x \sin\theta^* + f_y \cos\theta^* \Rightarrow \theta^* = \text{atan}\left(\frac{\partial f_y}{\partial f_x}\right) \rightarrow \text{dirección de borde} \quad (2.2)$$

$$\Rightarrow \left(\frac{\partial f}{\partial r}\right)_{max} = \sqrt{f_x^2 + f_y^2} \rightarrow \text{magnitud del vector Gradiente} \quad (2.3)$$

Además a esta técnica le añadimos la técnica de Snake, "serpientes" que rodean al objeto para delimitar su contorno. Este método, *GVF snake*, presenta una gran ventaja, en particular, sobre la técnica tradicional de Snake, ya que este método proporciona insensibilidad a la inicialización, permitiendo extraer los objetos de toda las partes de la imagen.

En primer lugar veremos la "Snake" tradicional, sin incluir, el vector gradiente. dicho algoritmo presenta la siguiente estructura:

1- Encontramos una curva inicial, $x(s) = [x(s), y(s)]$, $s \in [0, 1]$, que se mueve a través del dominio espacial de una imagen para minimizar la energía funcional:

$$E = \int_0^1 \frac{1}{2} [\alpha |x'(s)|^2 + \beta |x''(s)|^2 + E_{ext}(x(s))] ds \quad (2.4)$$

donde α y β son parámetros de ponderación que controlan la tensión de la serpiente y su rigidez, respectivamente, x' y x'' denotan las primera y segunda derivada de con respecto a $x(s)$.

La función de la energía externa, E_{ext} , se deriva de la imagen por lo que se trata de los límites. Dicha función es extraída de la siguiente manera, asumiendo que tenemos una imagen en escala de grises, tal como $I(x, y)$:

$$E_{ext}^{(1)}(x, y) = -|\nabla I(x, y)|^2 \quad (2.5)$$

$$E_{ext}^{(2)}(x, y) = -|\nabla [G_\sigma(x, y) * I(x, y)]|^2 \quad (2.6)$$

donde G es una función gaussiana bidimensional con varianza σ y ∇ es el operador gradiente.

Como queremos obtener la energía mínima, debemos saber que para que esta cumpla dicha condición, debe cumplir la siguiente ecuación de Euler:

$$\alpha x''(s) - \beta x'''(s) - \nabla E_{ext} = 0 \quad (2.7)$$

Para encontrar la solución a la ecuación anterior, la serpiente debe ser dinámica, y tiene que depender de una variable temporal:

$$X_t(s, t) = \alpha x''(s, t) - \beta x'''(s, t) - \nabla E_{ext} \quad (2.8)$$

Y cuando la ecuación se estabiliza, encontramos la solución a $x(s, t)$.

Un ejemplo del resultado obtenido es el siguiente:

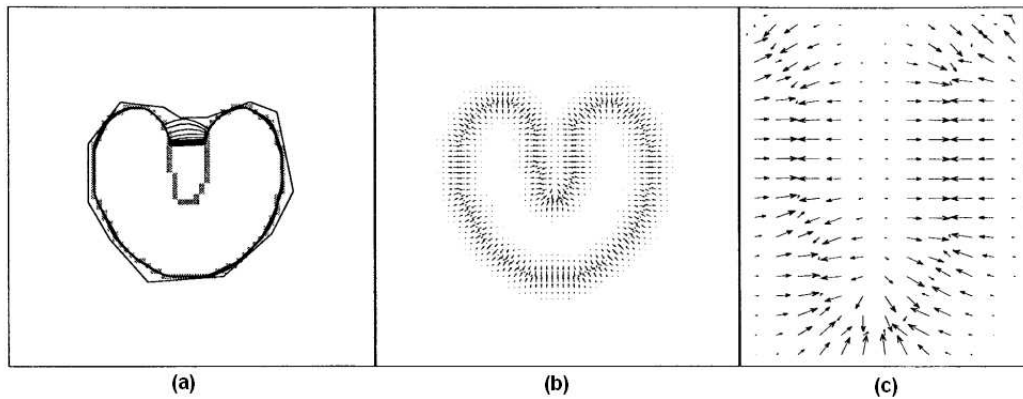


Figura 2.2: **Método tradicional de Snake.** (Figura extraída de [4])

(a) Curva de convergencia de la serpiente, (b) Fuerzas potenciales (c) Límite de la concavidad.

Si aplicamos el algoritmo de *GVF Snake*, añadiremos a la ecuación (5) una nueva fuerza externa : $F_{ext}^{(g)} = v(x, y)$, que llamaremos *Gradient Vector Flow*, que reemplaza a $-\nabla E_{ext}$, quedando la ecuación como:

$$X_t(s, t) = \alpha x''(s, t) - \beta x'''(s, t) - v \quad (2.9)$$

Definimos a GVF, como el vector que minimiza la energía funcional de v , tras aplicar el algoritmo completo, que encontramos en dicho artículo, obtenemos el siguiente resultado:

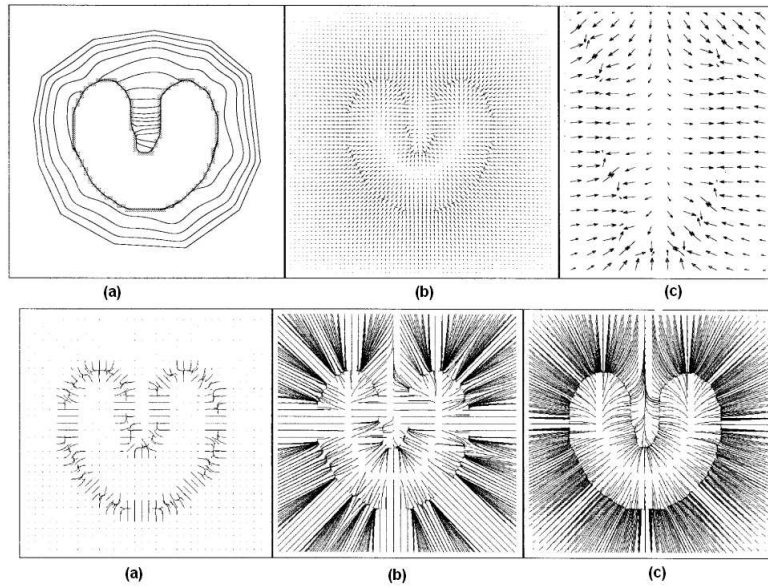


Figura 2.3: **GVF Snake**. (Figura extraída de [4]).

La superior tiene: (a) curva de convergencia extraída con (b) fuerzas externas y (c) muestra la concavidad de curvas cerradas. La inferior en (a) muestra el potencial de fuerzas, (b) el potencial de distancia y (c) las fuerzas del GVF.

Viéndolo más claro en los ejemplos que vemos a continuación:

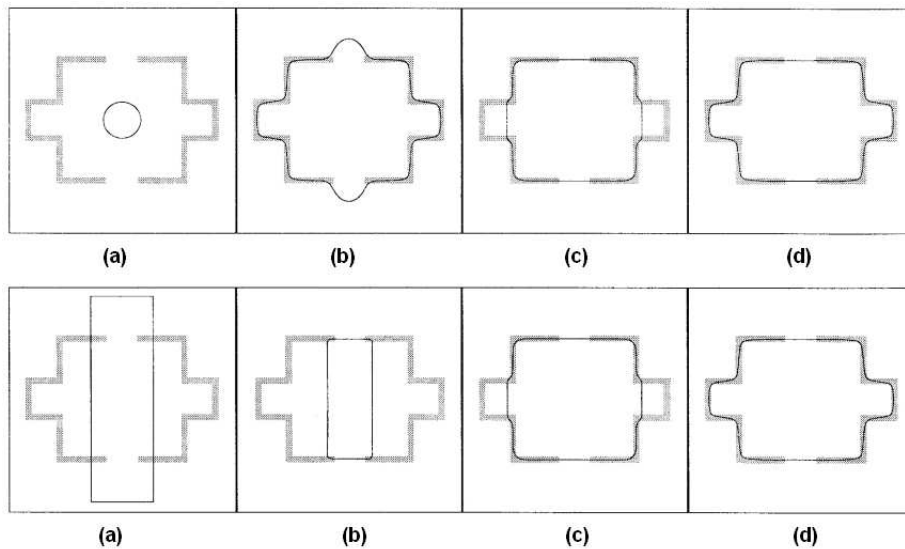


Figura 2.4: **Ejemplo 1 GVF Snake**. (Figura extraída de [4])

La superior: (a) Curva inicial (b) curva que se hincha hasta llegar al límite del objeto (c) eliminación de los bordes que se encuentran por fuera del objeto, calculando las fuerzas potenciales de distancia (d) Resultado final. La inferior: ejemplo con los mismos pasos, pero la curva inicial es mayor que los límites del contorno del objeto.

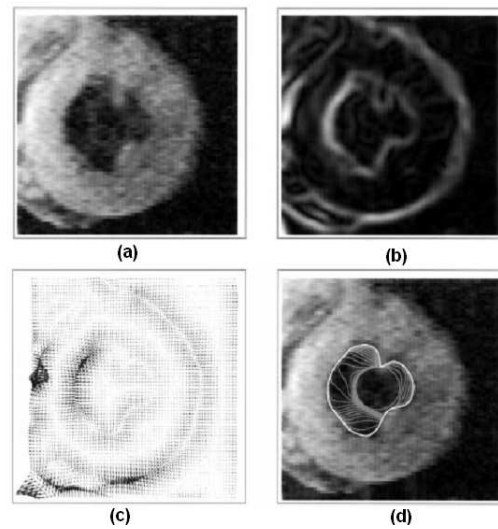


Figura 2.5: **Ejemplo 2 GVF-Snake en células.** (Figura extraída de [4]).

Muestra el proceso de GVF en una imagen médica: (a) Imagen de una resonancia del ventrículo derecho del corazón (b) Curva de convergencia (c) Fuerzas potenciales de distancia (d) resultado de GVF snake.

2.1.2.2. Contornos Activos (ACM)

El modelo basado en detección de regiones llamado modelo de contorno activo (ACM) [5] se propone en el documento anteriormente citado. Se lleva a cabo con un proceso especial llamado “*Selective Binary and Gaussian Filtering Regularized Level Set*” (SBGFRLS), que por primera vez realiza una función de ajuste para obtener una imagen binaria, y luego usa un kernel Gaussiano suavizado para regularizar la misma.

Las ventajas de este método son las siguientes: en primer lugar, se extrae una nueva región base de la obtención de las fuerzas externas de la misma “*signed pressure force*” (SPF); la función que se propone detecta los contornos de manera eficiente. En segundo lugar, los límites exterior e interior se pueden detectar automáticamente con el contorno inicial pudiendo estar el objeto en cualquier parte de la imagen. En tercer lugar, la propuesta tiene la propiedad de la segmentación selectiva local o global: se puede segmentar no sólo el objeto deseado, sino también los otros objetos. En cuarto lugar, la función de ajuste de nivel puede ser fácilmente inicializada con una función binaria, que es más eficiente de construir que el comúnmente utilizado por distancias. El coste computacional de reinicialización también se puede reducir. Por último, el algoritmo propuesto puede ser implementado de manera eficiente por el sistema de diferencias finitas.

Viendo más claro el método en los siguientes ejemplos. En primer lugar, para la detección de multiobjetos:

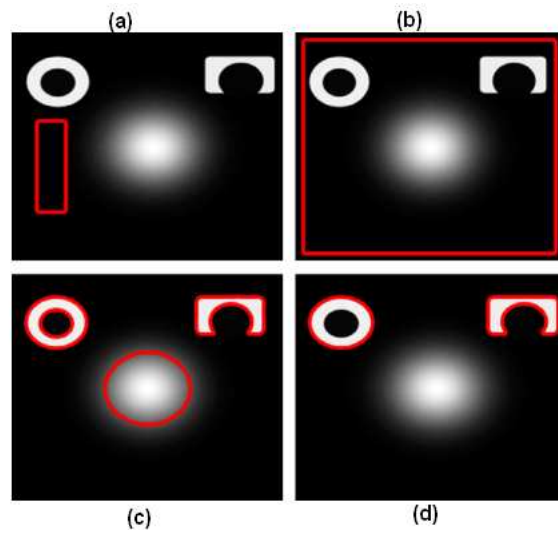


Figura 2.6: **Ejemplo de ACM.** (Figura extraída de [5]).

Muestra el proceso de extracción por contornos activos. (a) la curva inicial (b) la curva inicial se pone en los bordes de la imagen total (c) detecta los objetos (d) selección de contornos definidos.

Aplicado a una imagen de células:

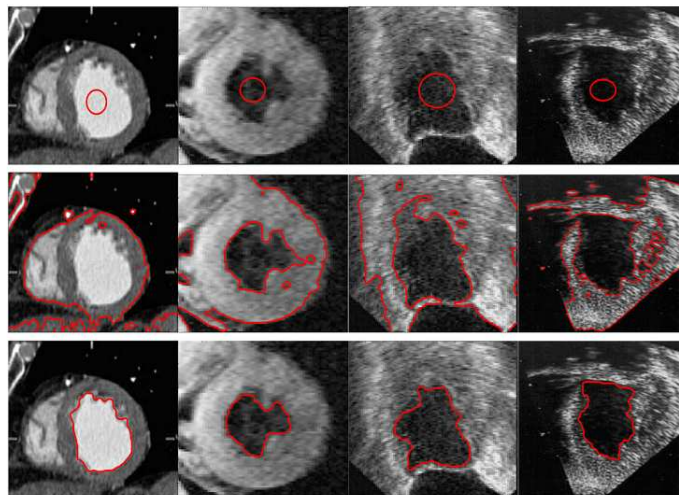


Figura 2.7: **Ejemplo ACM con células.** (Figura extraída de [5]).

Vemos el proceso de ACM aplicado a una imagen con células para observar el comportamiento del algoritmo en nuestro tipo de imágenes.

Pero dicho método presenta limitaciones para detectar objetos en una imagen, dónde existen muchas regiones a detectar y además se encuentran unidas. Esto lo vemos a continuación:

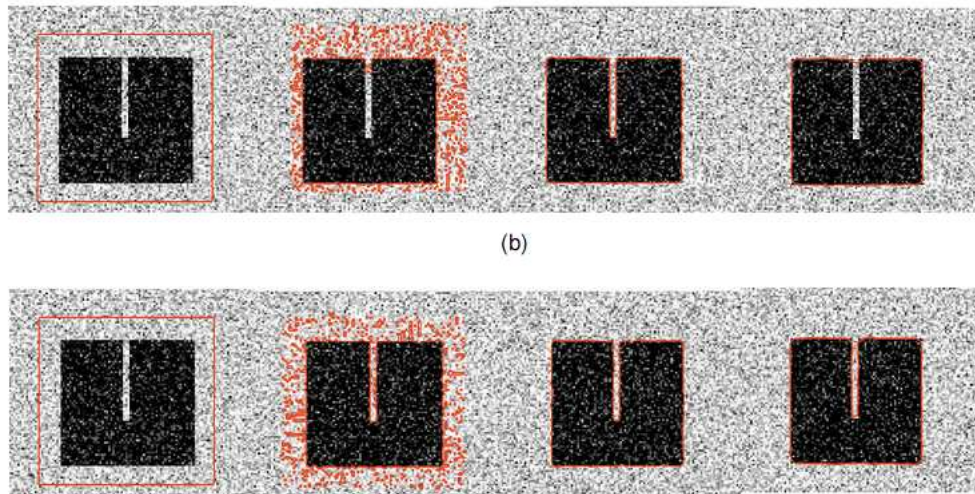


Figura 2.8: **Problemas de ACM.** (Figura extraída de [5]).

Aplicación en imagen con ruido. El efecto es: “objetos no detectados”.

2.1.2.3. Segmentación por *Watershed*- Transformada de *watershed*

En el campo del procesado de imagen, y más concretamente en el de la morfología matemática, las imágenes de niveles de gris son habitualmente consideradas como relieves topográficos. En la representación topográfica de dichas imágenes la altura de cada punto corresponde con el nivel de gris del píxel correspondiente. Esta representación es muy adecuada para poder percibir mejor el efecto de una determinada transformación sobre una imagen. Por ejemplo, que una apertura elimine algunos picos y líneas de cresta, mientras que el cierre tiende a rellenar valles o pequeñas llanuras.

La técnica de *Watershed* [6] constituye una de las más poderosas herramienta de segmentación aportada por la morfología matemática. Análogamente a otras técnicas de segmentación, el objetivo de la técnica es dividir en regiones la imagen de nivel de grises analizada. Generalmente una de ellas se corresponde con el fondo de la imagen y el resto con los objetos o regiones que se pretende extraer. El objetivo último de esta técnica es determinar los contornos que definen dichos objetos. En este punto el problema es definir qué es contorno y que no lo es.

El punto de partida es considerar que los contornos de una imagen se corresponden con las líneas donde el nivel de gris varía más rápidamente que en un determinado entorno vecino. Se define la imagen del gradiente, extraída en el punto anterior, la cual está formada por píxeles cuyo valor es el módulo del gradiente en dicho punto (conocida como imagen gradiente). De esta manera, se considera que los contornos de la imagen original se corresponden con las líneas de cresta de la imagen gradiente.

Algoritmo de *Watershed*

El concepto de watershed se basa en visualizar una imagen en 3 dimensiones (3D): dos coordenadas espaciales frente a la coordenada de niveles de gris.

En esta interpretación topográfico, consideramos tres tipos de puntos, o categorías en que se dividen los píxeles, con las que conseguiremos separar en la distintas partes características de la imagen:

1. puntos que corresponden a mínimos locales.
2. puntos en los que, si se coloca una gota de agua, esta cae con certeza en un único mínimo.
3. puntos en los que el agua caería con igual probabilidad hacia más de uno de estos mínimos.

Para un mínimo local particular, el conjunto de puntos que satisfacen la condición (2) se llama *catchment basin o watershed o cuenca* de este mínimo. Los que satisfacen la condición (3) forman líneas de cresta en la superficie topográfica y son llamadas **líneas divisorias o líneas de watershed o líneas de cresta**. El objetivo principal de los algoritmos de segmentación es alcanzar estas **líneas divisorias o de cresta** 2.9.

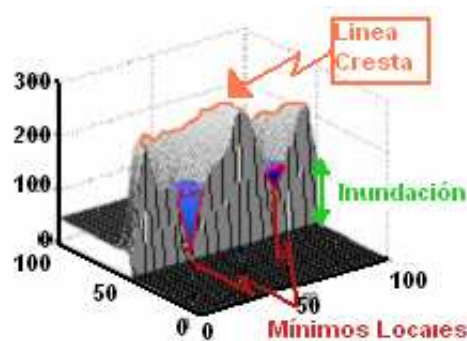


Figura 2.9: **Línea de Cresta.**

Ejemplo de nivel topográfico que podría darse en cualquier imagen. Este nivel está compuesto por diferentes zonas: las cuencas o mínimos locales, la líneas de cresta (valores de intensidad pueden caer con igual probabilidad hacia más de un mínimo local) y la inundación.

La idea básica es la siguiente: supongamos que se hace un pequeño agujero en cada mínimo local, y que todo el relieve topográfico es inundado desde abajo, dejando que el agua entre a velocidad constante.

El agua va subiendo e inundando las cuencas. Cuando el agua de dos cuencas está a punto de juntarse, se construye un dique (*dam*) para evitar la fusión.

La inundación continúa, y llega a un punto en que sólo se ve la parte de arriba de los diques por arriba de la línea de agua. Las líneas de watershed forman un camino conexo, dando por lo tanto bordes continuos entre las regiones (Véase la figura 2.10 para observar la fase I de Watershed) .

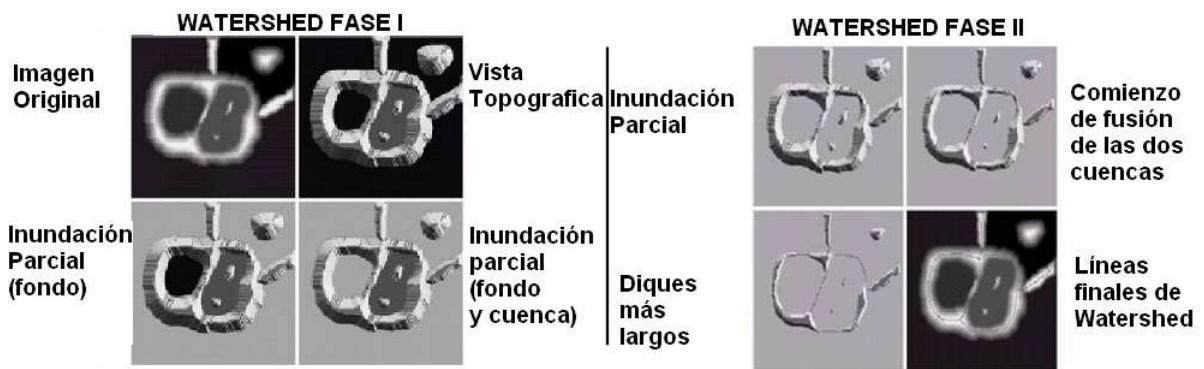


Figura 2.10: Fases de *Watershed*. Figura extraída de [6].

Una de las grandes ventajas del algoritmo de *Watershed* es que extrae de manera eficaz las distintas regiones de las imágenes.

Por otro lado, este método no es bueno para aplicar directamente sobre la imagen gradiente. Pues, habitualmente, el uso directo sobre la misma para detectar las líneas de contorno, produce el efecto conocido con el nombre de sobresegmentación, aunque también se da al aplicar la técnica sobre imágenes originales. Este efecto provoca que los verdaderos contornos queden enmascarados por una infinidad de contornos falsos o irrelevantes.

Habitualmente, la sobresegmentación es debida a la presencia de ruido. Pero, también los efectos de las texturas o de objetos presentes poco relevantes, que no se desean segmentar, provocan la aparición de falsos contornos asociados a multitud de valles o zonas de depresión que no se corresponden con objetos o regiones. Debemos solucionar dicho problema para la aplicación óptima de dicho método.

Para solucionar esta sobresegmentación existen dos posibilidades:

1. Eliminar los contornos irrelevantes una vez realizado el *watershed*.
2. Modificar la imagen gradiente, de tal forma que las regiones de depresión o valles se correspondan únicamente con los objetos deseados.

1.-Eliminar los contornos irrelevantes una vez realizado el *watershed*:

Para implementarla, debemos considerar la imagen obtenida como un conjunto de contornos de los cuales se debe eliminar los que no sean representativos de regiones u objetos de la imagen. O considerar la imagen como un conjunto de regiones adyacente que se pueden unir siguiendo un determinado criterio, para que finalmente sólo se tengan las regiones que se correspondan con los objetos que se pretendía detectar. Esta segunda posibilidad se englobaría en el conjunto de algoritmos de Crecimiento de Regiones, analizado anteriormente.

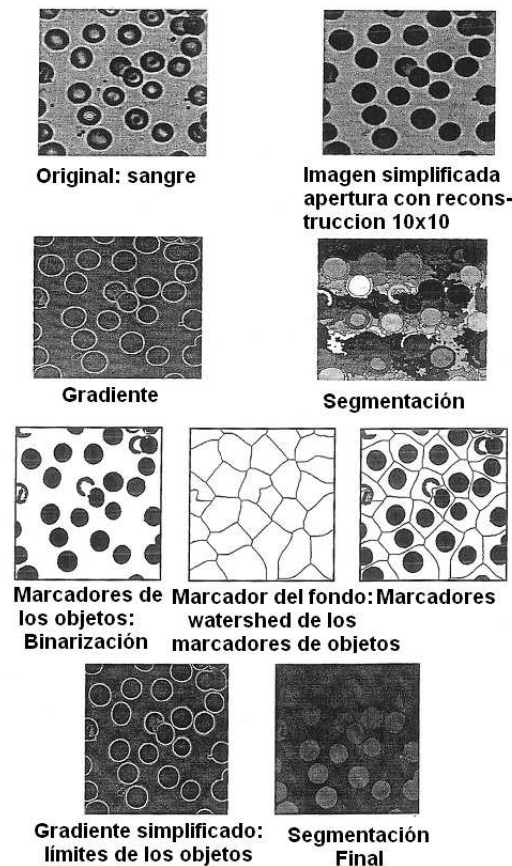


Figura 2.11: **Eliminación de contornos irrelevantes.** (Figura extraída de[6]).

Una vez extraídos los marcadores, se realizara una transformación morfológica basada en operaciones geodésicas, que permiten:

1. Imponer marcadores como mínimos de la imagen gradiente.
2. Suprimir todos los demás mínimos del gradiente, los irrelevantes, rellenando los correspondientes valles.
3. Preservar las líneas de cresta más importantes de la imagen gradiente localizadas entre los marcadores.

Esta transformación es denominada variación de la homotopía del gradiente, o simplemente, modificación del gradiente. El cálculo de las líneas de cresta o watershed de la imagen gradiente modificada aporta los resultados esperados: el marcador del fondo se recompone el mismo; mientras que el resto de los contornos obtenidos se corresponden con los límites de los objetos que se deseaba detectar.

Pero en la mayoría de los casos no ocurre de esta manera, y, por tanto, debemos acudir al histograma de la imagen para poder establecer un umbral.

2.1.2.4. Segmentación por umbralización

Utiliza como característica solamente la luminancia de cada píxel. Esto es útil si los distintos objetos se corresponden con niveles de gris diferentes.

El nivel de gris determina alguna propiedad física en una imagen específica de una aplicación (Rayos X, RMN, bandas en satélite, etc) .

En el caso ideal en que el objeto posea un rango estrecho de niveles de gris frente a un fondo uniforme, podemos establecer un nivel de gris intermedio (umbral) para separar objeto y fondo. Pero en la mayoría de los casos no ocurre de esta manera y por tanto debemos acudir al histograma de la imagen para poder establecer un umbral.

En los histogramas encontramos dos tipos de tramos: Picos (a menudo indican la presencia de zonas homogéneas) o valles (establecen los umbrales de separación)

Problemas:

Pero encontramos varios problemas en este tipo de segmentación:

- Objetos con un amplio rango de niveles de gris
- Fondo no uniforme
- Imagen con ruido

Dicha segmentación quedaría de la siguiente manera:

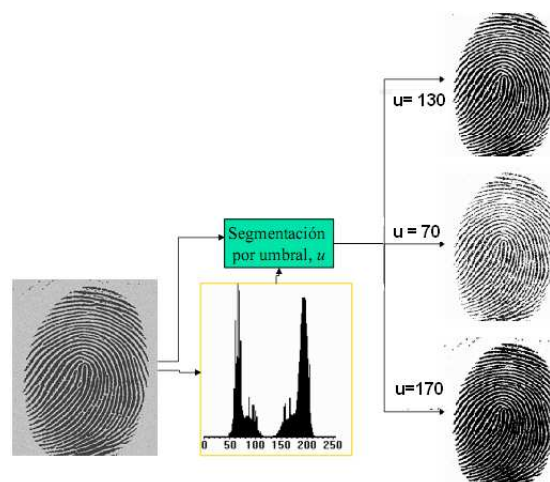


Figura 2.12: Segmentación por umbral.

En esta figura encontramos la aplicación del método a una imagen de huella dactilar (imagen de la izquierda). La gráfica central es el histograma de la misma (repartición de los píxeles de la imagen a lo largo de los valores posibles de intensidad [0-255]) y las imágenes de la derecha son los resultados obtenidos tras umbralizar, con tres posibles umbrales {130,70,170}.

De esta manera al aplicar un umbral u , se genera una partición de la imagen:

$$L(x, y) = \begin{cases} 1 & I(x, y) \leq u \\ 0 & I(x, y) > u \end{cases}$$

Esto equivale a definir el conjunto :

$$L_u = \{(x, y) \in \Omega / I(x, y) < u\}$$

Este conjunto produce una división del espacio. La cantidad de componentes conexas de L_u determinan el número de regiones.

La determinación del umbral adecuado para una imagen dada es un factor crítico de la segmentación. Conociendo las distribuciones de luminancia propias de cada objeto en la imagen, el umbral puede estimarse mediante consideraciones estadísticas. Sin embargo, puede haber limitaciones: si las distribuciones están solapadas, ningún umbral aplicado directamente sobre la luminancia de la imagen puede obtener una segmentación libre de errores. Esto puede mejorarse aplicando un preprocesado a la imagen mediante un filtro mediana.

2.1.2.5. Segmentación en imágenes a color

La segmentación en imágenes a color se realiza mediante umbralización. Esta segmentación se puede realizar de dos maneras:

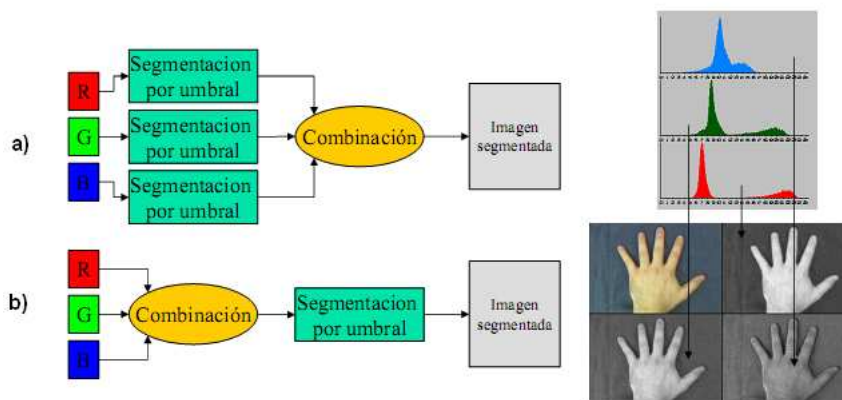


Figura 2.13: Segmentación de imágenes a color. Imagen autoexplicativa.

a) Combinación de segmentaciones:

La combinación de los resultados de la segmentación de cada componente por separado suele ser subóptima, porque limita severamente la partición del espacio de color P.

b) Combinaciones lineales

En algunas imágenes se pueden combinar linealmente las componentes de color y crear una única característica a partir de la cual segmentar. Pero el problema es cómo obtener la combinación lineal óptima a partir de la descomposición en componentes principales.

Las combinaciones lineales se hacen de la siguiente manera:

1. Transforman las componentes R, G y B en una sola componente : $C = w_R R + w_G G + w_B B$
2. Aplicada a todos los puntos de la imagen $f(x, y)$ de componentes $f_R(x, y)$, $f_G(x, y)$ y $f_B(x, y)$, obtenemos

$$c(x, y) = w_R f_R(x, y) + w_B f_B(x, y) + c(x, y) = w_R f_R(x, y) + w_G f_G(x, y)$$

3. En forma vectorial $c(x, y) = w^T f(x, y)$, donde $w = (W_R, W_G, W_B)^T$. Además si w es vector unitario, c es la proyección ortogonal de f sobre la recta en dirección W .

Así para calcular la componente principal:

1. Se calcula la media de toda la imagen

$$\bar{f} = \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M f(x, y)$$

2. Se calcula la matriz de varianzas de color

$$\bar{f} = \frac{1}{MN} \sum_{x=1}^N \sum_{y=1}^M (f(x, y) - \bar{f})(f(x, y) - \bar{f})^T$$

3. Se calculan los autovalores y autovectores de dicha matriz:

- Los autovalores de la matriz de covarianza indican la varianza en la dirección correspondiente a los autovectores, es decir, indican la proporción de información original que contiene esa nueva característica.
- El autovector dominante (es decir, el de mayor autovalor) indica la orientación preferente de la distribución global. Debiendo tomarse W igual al autovector dominante, de modo que $c = w^T f(x, y)$.

Esto se ve reflejado en el efecto del siguiente ejemplo:

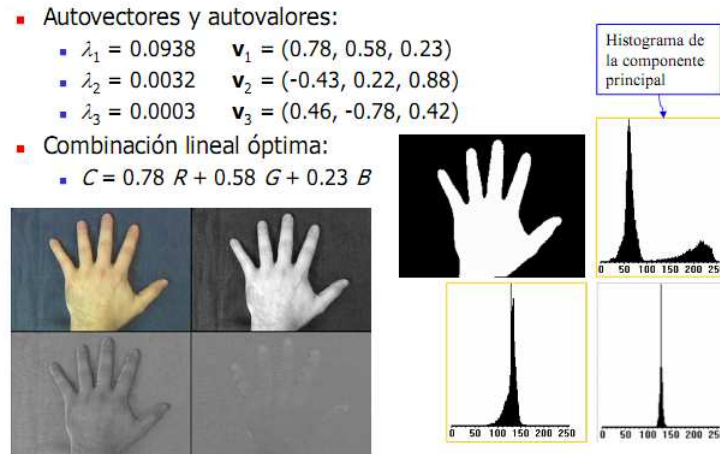


Figura 2.14: Segmentación de imágenes a color. .

2.1.2.6. Segmentación con técnicas de clusterización

En este caso estas técnicas se utilizarán para extraer las regiones, pero su relevancia recae en que se aplican con frecuencia para extraer la clasificación de las distintas regiones obtenidas, mediante otras técnicas de segmentación, en células normales y anormales.

Técnicas de clusterización:

K-means

La idea básica del Algoritmo [15] es obtener los K centros iniciales y formar grupos de todos los objetos de X a los centros más cercanos, después recalcular los centros. La función de MATLAB utilizada para la llamada de este algoritmo en nuestro software es la siguiente:

```
function [idx, C] = kmeans(matriz_entrenamiento, k);
```

Las variables asociadas son:

1. Entrada: `matriz_entrenamiento` y `k` (número de centroides que queremos).
2. Salida: `idx` (centroide a la que pertenece) y `C` (coordenadas de cada centroide para ser situado en el plano).

Cuando hablamos de agrupamiento (*clustering*) de datos, lo definimos como asociar un dato a un punto “*cluster* o grupo” que es usado para representar un conjunto de valores que tienen algo en común y se agrupan en función de ese determinado rasgo.

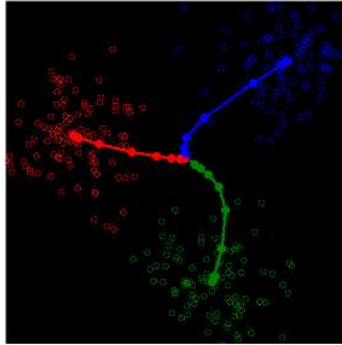


Figura 2.15: *Clusters*.

En esta figura se representa una posible salida del algoritmo *Kmeans*. En este ejemplo del algoritmo encontramos divididos en 3 grupos (*clusters*) todos los datos de la imagen que usara el ejemplo.

Así dicho algoritmo tiene cuatro partes importantes: en primer lugar, la matriz de datos; en segundo lugar, la matriz de distancias; en tercer lugar, la matriz de centroides; y en cuarto, y último lugar, la matriz de pertenencias.

Sus diferentes variantes se basan fundamentalmente en la forma de medir distancias entre los datos y los grupos, el criterio para definir la pertenencia de los datos a cada grupo y la forma de actualizar dichos grupos.

Matriz de datos

En esta matriz se almacena el conjunto de muestras de las que se pretende obtener los puntos significativos que representen grupos a clasificar. Siendo su orden de $n \times N$, donde N es el número de observaciones o muestras de que se dispone y n son los rasgos que caracterizan a cada muestra, adquirida del sistema al que representa. Como vemos en la siguiente figura, están situados los cluster obtenidos del entrenamiento y los datos de la matriz:

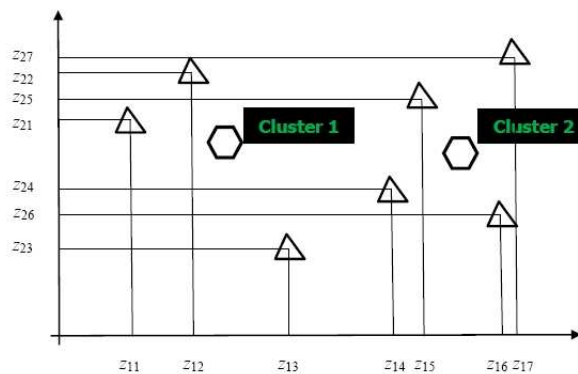


Figura 2.16: *Matriz de datos y clusters*.

En la figura encontramos dos tipos de iconos: los hexágonos hacen referencia a los centroides de los grupos o clusters, y los triángulos son los datos que poseemos. En función de la distancia a la que estén de un centroide u otro, pertenecerán a un grupo u otro.

Matriz de *clusters* o centroides

Los centroides que se generen durante la evolución del algoritmo de clustering van siendo almacenados en la matriz v ($n \times c$), siendo n el número de rasgos que caracterizan a cada cluster, en nuestro caso vectores de 16 columnas que se corresponden a las 16 características que el artículo implementado desea extraer y c el número de grupos que deben definirse antes de aplicar el algoritmo.

Esta matriz se actualiza continuamente durante la evolución del algoritmo, para ello debe previamente definirse a que grupo pertenece cada muestra.

Una vez tenemos el algoritmo entrenado, tenemos que pasarlo por una función que calcule la distancia de cada punto a evaluar, con los obtenidos tras el entrenamiento, para ver si pertenece a alguno de los distintos grupos, buscando distancia mínima.

Así podemos observar el siguiente ejemplo del algoritmo de *Kmeans*:

- En primer lugar mostramos los centroides extraídos de la matriz de entrenamiento

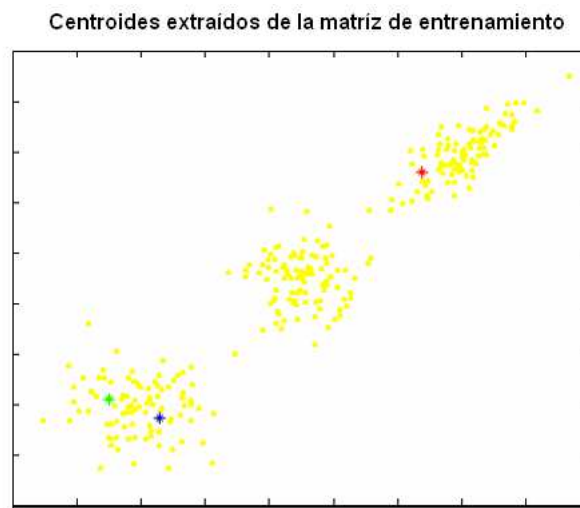


Figura 2.17: **Centroides de la matriz de entrenamiento.**

Muestra la posición de los centroides en la matriz total; estos centroides tienen unas coordenadas para después situar el dato y comprobar la distancia a los mismos y clasificar el dato analizado.

- Y a continuación mostramos la división en las distintas regiones:

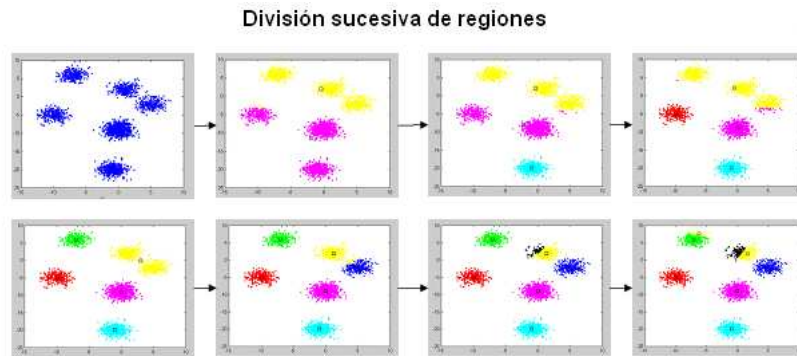


Figura 2.18: **División en regiones de *Kmeans*.** .

Proceso y resultado de la clasificación de regiones de una imagen.

Parametric fitting

Este algoritmo [16] se usa exclusivamente para la segmentación de las imágenes extraídas mediante la técnica de Papanicolaou. Este sigue un modelo paramétrico que minimiza la función de coste adaptándose a la forma de los objetos que deseamos extraer, en nuestro caso el modelo será elíptico por la forma de las células que tenemos. Los resultados de la segmentación demuestran que la propuesta de dicho algoritmo es muy exitoso en la segmentación de las imágenes donde las células no se superponen .

Hay muchos algoritmos no paramétricos, pero poco se ha hecho con este tipo de algoritmos debido a que precisa de una información inicial de los objetos que han de ser identificados.

En primer lugar usa la transformada de Hough, ya que es un método útil para el análisis de la forma. Después de detectar los bordes, la transformada de Hough se realiza para detectar una forma deseada.

A continuación se encuentran los objetos, mediante la búsqueda de los picos en el espacio paramétrico. A pesar de que las transformaciones Hough tienen éxito en la detección de patrones analíticos, tales como líneas, círculos y elipses, existen varias deficiencias con esta técnica cuando se utilizan para la segmentación de imágenes reales.

Además de los requisitos de almacenamiento excesivos y la complejidad computacional, las texturas de imagen, detección de bordes y el ruido se degradan haciendo difícil encontrar los picos adecuados en el espacio de parámetros.

Para mejorar el rendimiento de segmentación de imágenes reales, generalmente son de textura ruidosa tenemos que incorporar el conocimiento a priori de las formas celulares y también hacer uso de la información desde el interior de las regiones, en vez de sus fronteras.

Dicho algoritmo nos muestra el siguiente resultado en el ejemplo, presentado a continuación:

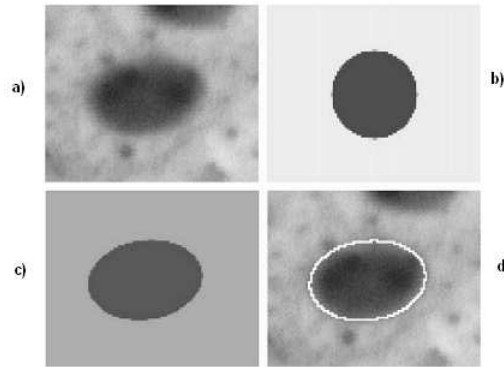


Figura 2.19: *Parametric Fitting*.

(a) imagen original (b) es la forma a la que tiene que asemejarse el objeto a extraer de la original (c) forma de la región extraída en función de la forma esperada (d) resultado del contorno extraído superpuesto. (Figura extraída de [16]).

Dicho método presenta como vemos en el artículo, grandes beneficios para imágenes con regiones solapadas frente a otros métodos como Kmeans.

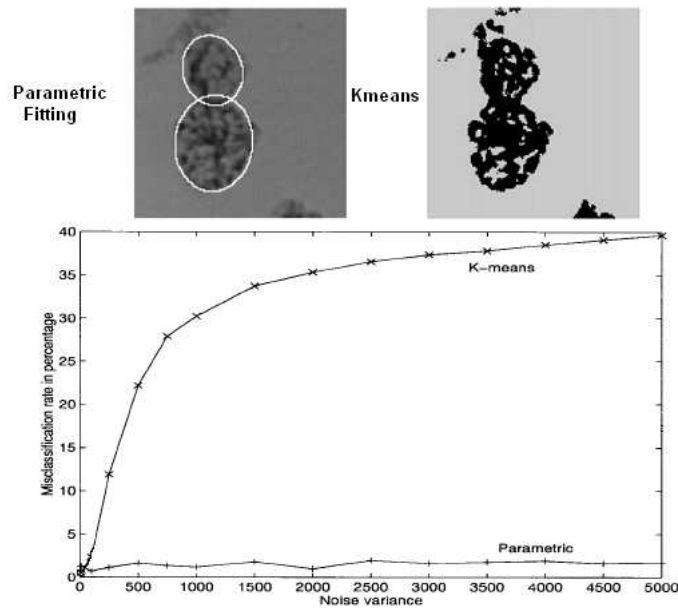


Figure 2.20: Comparación de imágenes con células solapadas de los algoritmos de “*Kmeans y Parametric Fitting*”.

Las figuras superiores son los resultados obtenidos para las imágenes originales de ambos métodos y la gráfica inferior muestra la comparación de los mismos (Cuanto más ruidosa es la imagen más problemas tiene *Kmeans*). (Figura extraída de [16]).

Como podemos observar *Parametric Fitting* es más robusto cuando encontramos células solapadas, teniendo así problemas aun.

Support Vector Machine(SVM) clasificado con Radial Basis Function(RBF)

Una máquina de soporte vectorial [17] construye un hiperplano o conjunto de hiperplanos en un gran espacio de dimensión infinita, que puede ser utilizado para la clasificación, regresión, u otras tareas.

Tiene como objetivo extraer los objetos que presentan características similares, como en nuestro caso de intensidad, pero a menudo dichos conjuntos cuando discriminan no son linealmente separables en ese espacio. (Véase la figura 2.21).

Los vectores que definen los hiperplanos se pueden elegir para ser combinaciones lineales con parámetros α_i de las imágenes de vectores de características que se producen en la base de datos. Con esta elección de un hiperplano, los puntos x en el espacio de características, que se asignan al hiperplano, se definen por la relación: $\sum_i \alpha_i K(x_i, x) = constante$.

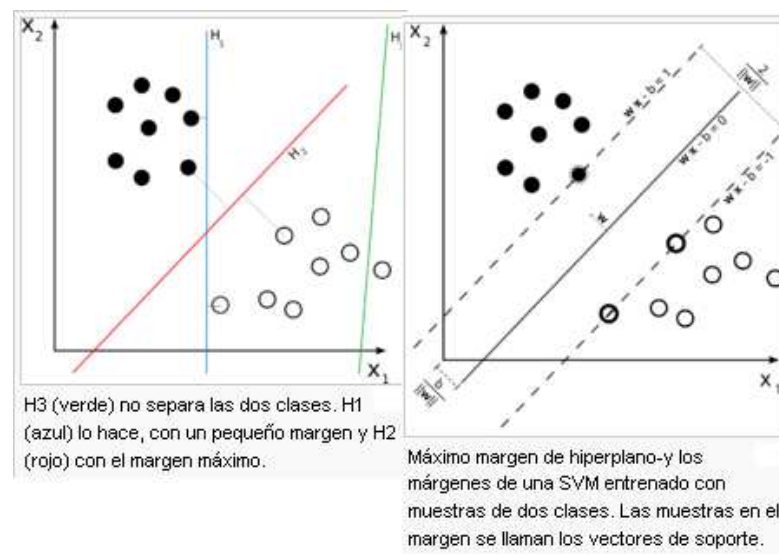


Figura 2.21: SVM.

En ocasiones las divisiones no son tan claras y, por tanto, tienen que realizar combinaciones no lineales. Sugiere una manera de crear clasificadores no lineales, mediante la aplicación del truco del *Kernel* establecemos un núcleo (como el *kernel* RBF).

Esto permite que el algoritmo se adapte al máximo, al margen del hiperplano, transformado en un espacio de características (véase la figura 2.22) .

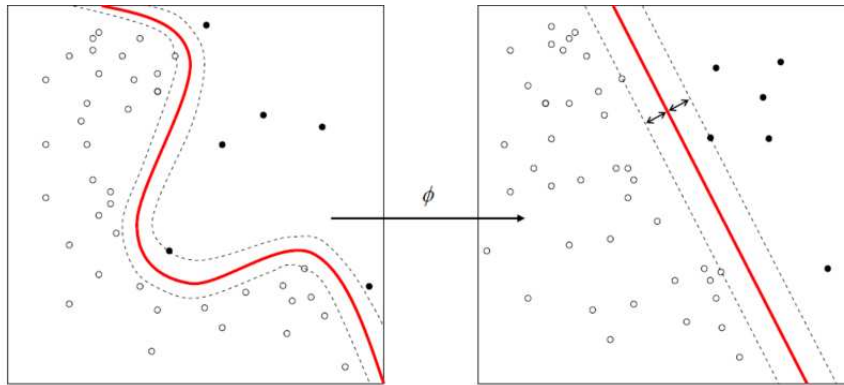


Figura 2.22: SVM con combinación no lineal de RBF.

Dónde la línea roja muestra el margen máximo para establecer los datos en un grupo o en otro. Las líneas discontinuas muestran el margen de error para la clasificación.

2.1.3. Selección de algoritmos

Tras realizar este estudio de las técnicas más utilizadas en la actualidad para segmentar imágenes médicas, debemos tener en cuenta que dichos tipos de segmentaciones no son igual de adecuadas para todos los tipos de imágenes y, por este motivo, vamos a dar los criterios por los que hemos elegido las técnicas que implementaremos. Y, tras esto, estas serán explicadas detenidamente en los siguientes capítulos.

En primer lugar, debemos ver cuáles son nuestros tipos de imágenes, para ello mostramos una de ejemplo:

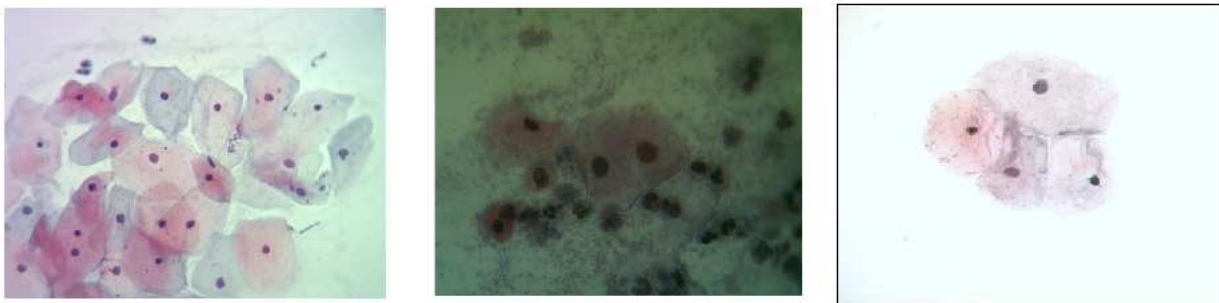


Figura 2.23: Tipos de Imágenes del estudio extraídas de nuestro dataset.

La primera de la izquierda es una imagen con muchas células, la del medio una imagen oscura y la tercera es una imagen clara y con pocas células.

A continuación, evaluaremos los problemas que presentan cada método para seleccionar los que implementaremos.

En primer lugar encontramos *GVF-Snake*; dicho método es uno de lo más robustos, pero tiene grandes dificultades al encontrar varias regiones que detectar, por tanto es especialmente adecuado para imágenes con células aisladas, lo cual no es nuestro objeto de estudio.

El siguiente de los métodos de segmentación por contornos activos, y como explicamos más arriba, presenta problemas cuando se trata de una imagen donde existen muchas regiones a detectar y además se encuentran unidas; por este motivo no consideramos que este método sea el deseado para nuestro tipo de imágenes.

La segmentación realizada mediante técnicas de agrupamiento, requiere en la mayoría de los casos un entrenamiento previo y precisa de gran capacidad de memoria, ya que las imágenes reales tienen gran coste computacional.

Hablando más específicamente de cada uno de los métodos explicados, debemos resaltar tres deficiencias, que presentan, que hacen que seleccionemos o descartemos dichas técnicas para ser implementadas.

En primer lugar, como observamos en 2.20, *Kmeans* presenta grandes dificultades cuando encontramos células solapadas, creando verdaderos problemas para extraer regiones. En el caso de *Parametric Fitting*, también hemos visto que requiere requisitos de almacenamientos excesivos y gran complejidad computacional, debido a que las imágenes reales son ruidosas, con texturas complicadas y esto dificulta la detección de bordes a dicho método. Y en último lugar encontramos *Support Vector Machine con un Kernel RBF*, que presenta grandes dificultades de extraer imágenes cuando las regiones a extraer se encuentran muy próximas.

Por los motivos anteriormente argumentados, descartamos implementar estos métodos para nuestro tipo de imágenes. Y por el contrario, implementaremos los métodos que hacen referencia a las características de intensidad de la imagen, ya que como explicamos en el comienzo de la memoria, nuestros algoritmos están basados en la idea de que las zonas más oscuras de la imagen hacen referencia a los núcleos; por este motivo implementaremos los algoritmos que enumeramos, y que a continuación introducimos para realizar una explicación detallada en los capítulos posteriores:

- *Watershed más Kmeans*: Algoritmo de extracción de características 4, donde no está *Kmeans* como técnica de segmentación por clusterización, sino como paso posterior al agrupamiento, siendo utilizado como clasificador de las regiones anteriormente encontradas por *Watershed*. De la misma manera el estudio que realiza el algoritmo que introducimos en la siguiente sección, las técnicas que comparamos de clusterización son utilizadas como clasificadores de regiones y no como segmentación de imágenes, aun siendo la misma implementación de la técnica.
- Segmentación por Umbralización.3.3.13
- Segmentación en imágenes a color.3.3.23

2.1.4. Introducción a los algoritmos seleccionados

2.1.4.1. Estudio de la segmentación para la extracción de núcleos en imágenes médicas

La segmentación de núcleos es una tarea clásica en el análisis de imágenes de citologías. Según se ha comentado anteriormente, esta etapa tiene un peso especial dentro del proceso global de diagnóstico: una buena segmentación de los núcleos permite una adecuada extracción de características, y por consiguiente, una clasificación más eficiente de los núcleos. La principal dificultad que existe en este proceso es la presencia en las imágenes de núcleos solapados. La separación de núcleos solapados se realiza normalmente de modo manual. En este trabajo se presenta un método automático para segmentar dichos tipos de núcleos. Este método está basado en la combinación de dos técnicas de segmentación bien conocidas (umbralización y transformada watershed) junto con un nuevo método de fusión de regiones diseñado específicamente para estas imágenes.

La palabra 'segmentación' tiene un significado que depende del uso y del contexto en el cual se utilice. El objetivo básico de los algoritmos de segmentación es definir una partición del espacio. En el contexto de la imagen, el espacio puede ser temporal (1D), espacial (2D) o espacio-temporal (3D). En consecuencia, este espacio se llama espacio de la decisión. Esta sección revisa los pasos principales implicados en un algoritmo de segmentación y las principales opciones a realizar. Un esquema general para la segmentación se puede considerar como la concatenación de tres pasos principales, representados en la siguiente figura:

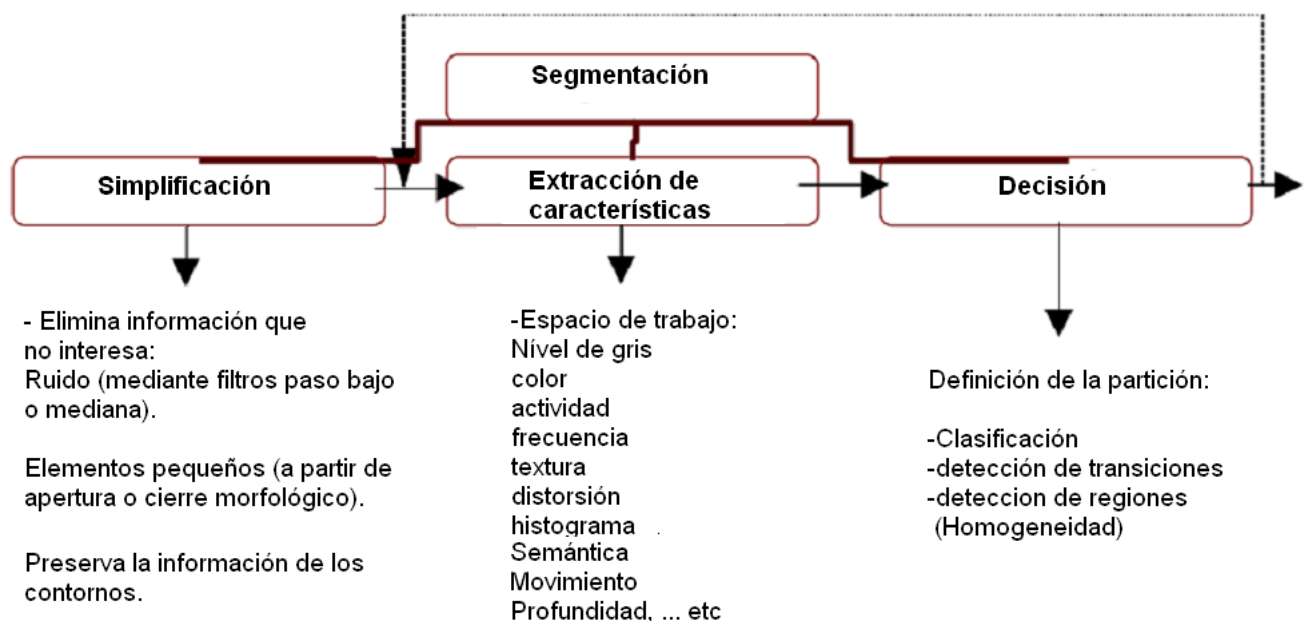


Figura 2.24: Proceso Global de la Segmentación de Imágenes.

Simplificación

La mayoría de las veces, los datos originales en una imagen contienen información que es irrelevante para una aplicación dada. En tales casos, los datos se deben simplificar, eliminando toda aquella información irrelevante. La simplificación controla la cantidad y naturaleza de la información que es preservada. Además, los datos simplificados deben contener áreas fáciles de segmentar. Por ejemplo, la simplificación puede reducir la complejidad de texturas, eliminando ruido o eliminar objetos muy pequeños para un tamaño dado.

Figura extraída de ¹



Figura 2.25: **Simplificación: Eliminación de ruido**

Extracción de características.

La segmentación se realiza sobre características específicas de los datos. La selección del espacio de características conduce al tipo de homogeneidad que se espera en la partición final. En algunas aplicaciones, los datos originales proveen directamente el espacio de características necesario para la segmentación. Consecuentemente, quizá se pueda introducir un ciclo en el proceso de segmentación, de manera que la estimación dependa de los resultados de la segmentación.

Decisión.

Finalmente para obtener una partición de los datos, el espacio de características debe ser analizado. En este paso decide la posición de los contornos que formaran parte del espacio de decisión. Los contornos separan áreas de datos que contienen elementos con las mismas características en el espacio de características seleccionado.

En nuestro caso encontramos tres algoritmos diferenciados para el proceso de segmentación.

- *Grey Level Method* [1]. 3
- *Energy Method* [1]. 3
- Algoritmo de extracción de características[2]. 4

¹(<http://www.mathworks.es/help/toolbox/images/ref/medfilt2.html>)

2.1.4.2. Introducción a los métodos *Grey Level Method* y *Energy Method*

Ambos algoritmos son implementados para la extracción de células malignas del examen de Papanicolaou (Pap). Extraeremos los núcleos de las células que son de tamaño anormalmente grande, forma extraña, así como los que representan gran densidad, aplicando este método para los diferentes tipos de células anormales.

Este algoritmo consta de dos métodos cuyos resultados luego se fusionan: *Grey Level Method* y *Energy Method*.

El primero de ellos está basado en la escala de grises de la imagen o intensidad, sobre la que se realiza un procesamiento previo de la imagen, para facilitar la extracción de los núcleos en la segmentación. Esta etapa extraerá los contornos y regiones que se consideraran núcleos y, como último paso de dicho método, extraeremos los núcleos anormales.

El segundo de los métodos está basado en un procesamiento previo de la imagen, pero en este caso centrado en extraer la energía de las tres componentes de color (R, G y B), por separado, para dejar las zonas de los núcleos más oscuras y realizar la misma segmentación que realiza el método anterior; una vez tenemos las regiones y contornos extraídos, clasificamos los núcleos en normales o anormales.

Una vez tenemos ambos resultados por separado, los juntamos obteniendo el resultado final del algoritmo.

El esquema de trabajo del artículo es:

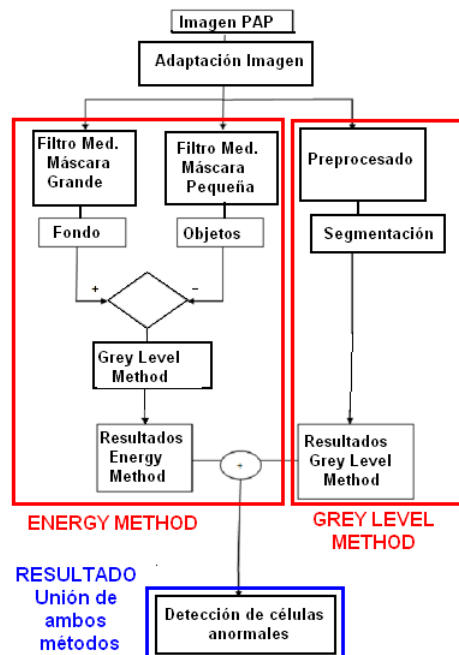


Figura 2.26: Diagrama de flujo del primer artículo implementado. (Figura extraída de[1]).

2.1.4.3. Introducción al algoritmo de extracción de características

Este algoritmo tiene como objetivo crear un sistema automatizado para la detección y determinación de los contornos de los núcleos de las células de Papanicolaou. La detección de las áreas candidatas a núcleos se basa en un proceso de reconstrucción de imagen morfológica y la segmentación de los contornos de los núcleos se logra con la aplicación del algoritmo de *Watershed* a la imagen en color, a la cual previamente hemos aplicado un gradiente morfológico.

Para la eliminación de falsos positivos, utilizamos las características de la forma, la textura y la intensidad de la imagen, extraídas de las regiones candidatas a núcleos (una etapa posterior de clasificación realizada para determinar los núcleos verdaderos) mediante el algoritmo de *Kmeans*.

Este algoritmo será entrenado con las 16 características de las regiones extraídas que cumplan el criterio de mRMR (mínima Redundancia Máxima Relevancia). El 70 % de las imágenes que poseemos en el dataset serán utilizadas para dicho entrenamiento . Una vez tengamos los centroides, utilizaremos el 30 % de nuestro dataset restante para extraer los resultados de nuestro algoritmo.

Y podremos realizar un estudio en base a los resultados obtenidos en el artículo [2], el cual muestra los resultados de dos métodos distintos al implementado en imágenes de Papanicolaou.

Capítulo 3

Alg. basado en niveles de gris y energía de las bandas de color

3.1. Introducción

El objetivo de este capítulo es presentar una explicación detallada del primero de los algoritmos que se ha implementado en este proyecto. Este algoritmo está implementado a partir de la descripción y datos concretos que aparecen en el artículo [1]. Estos se han seguido fielmente; a excepción de los que no se detallan en el mismo, los cuales hemos elegido según la experiencia o extrayendo datos de otros artículos relacionados. Por ello, los datos que no se dan en el artículo los resaltaremos en las etapas donde se tengan que usar. Este algoritmo se ha probado con un dataset de 88 imágenes médicas de los tipos que vimos en el capítulo anterior. Durante la explicación del algoritmo mostraremos los resultados parciales en cada una de las etapas.

El objetivo del algoritmo es obtener una clasificación de las regiones extraídas como núcleos en: normales o anormales; para ello seguirán los criterios de área y circularidad, ya que son unos de los criterios principales que siguen los expertos para clasificar las células.

Los algoritmos basados en imagen médica dependen en gran medida del tipo de imagen al que nos enfrentamos, y esto provoca que tengamos que añadir una etapa previa de adaptación de las imágenes que poseemos en nuestro dataset para poder ser tratadas por el algoritmo. Así, las imágenes que describe el artículo son imágenes uniformes, mientras que las que nosotros tenemos presentan mayor oscuridad en las esquinas al ser obtenidas por un microscopio con un patrón de iluminación no uniforme. Esto crea la necesidad de una adaptación que nos dé como resultado una imagen uniforme para introducir en el algoritmo.

Una vez tenemos la imagen adaptada al algoritmo explicaremos los dos métodos por los que está compuesto el mismo: *Grey level method* y *Energy method*, los cuales una vez hayan extraído los resultados por separado, los unificarán para dar la clasificación de las regiones en núcleos

normales o núcleos anormales, objetivo del algoritmo.

En el algoritmo del artículo se implementan ambos métodos, debido a que cada uno de ellos tienen distintos fundamentos y aportan una mejora distinta al mismo; *Grey Level Method* se basa en la característica que tienen los núcleos de ser las zonas más oscuras de toda la imagen; y *Energy Method* tiene en cuenta la posible situación de núcleos en zonas oscuras de la imagen y, que estos, sean confundidos con el fondo y no se extraigan correctamente. Por este motivo, el artículo aplica ambos métodos por separado y combina sus resultados para obtener el resultado final (Véase la figura 3.1 para observar el diagrama del algoritmo completo).

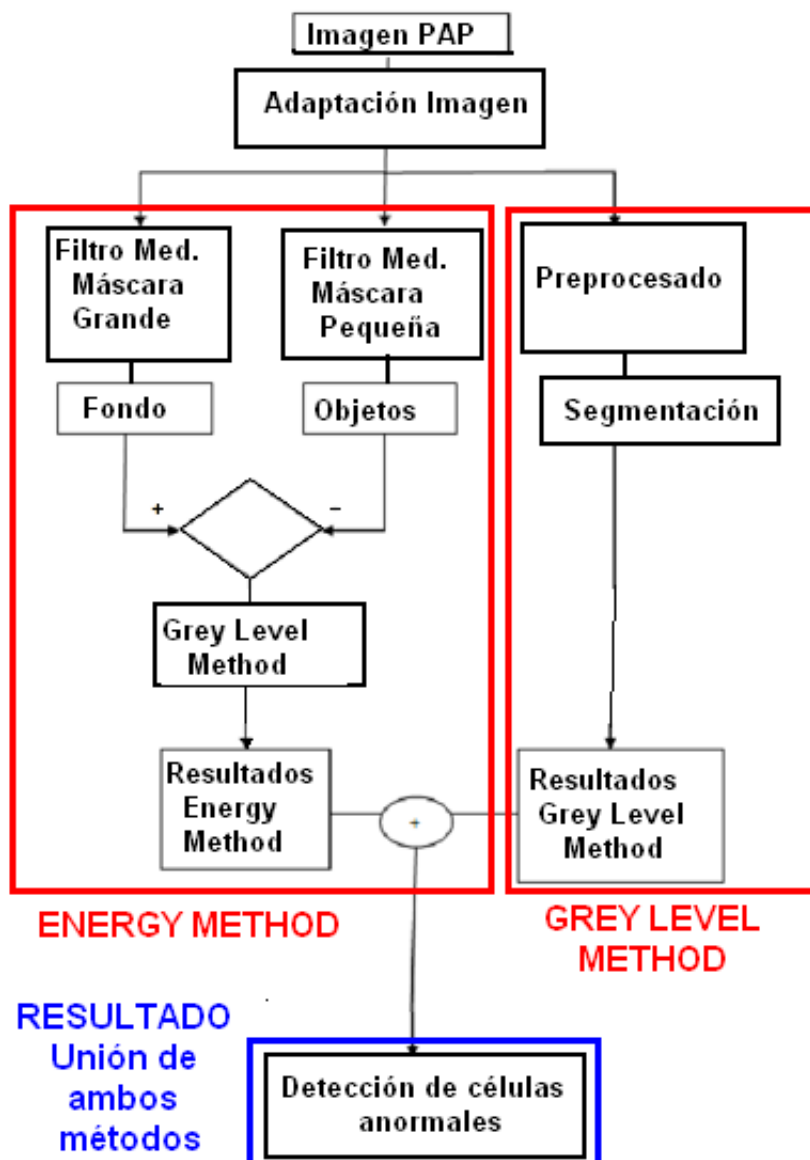


Figura 3.1: Diagrama de flujo.

Dividido en las siguientes etapas: adaptación de las imágenes del dataset, desarrollo del algoritmo repartido en dos partes: *Grey Level Method* y *Energy method*; y última etapa unificará los resultados finales.

3.2. Adaptación de las imágenes del dataset

Esta primera etapa tiene como objetivo adaptar nuestro dataset al algoritmo desarrollado. El fundamento de esta adaptación es que el artículo trabaja sobre imágenes uniformes, mientras que nuestro dataset está compuesto por imágenes Pap, que no lo son. En las imágenes de PAP existe un fenómeno donde las esquinas de la imagen son más oscuras que las zonas centrales, debido a que las imágenes son captadas a través de un microscopio, el cual posee una bombilla que enfoca a la zona central del porta provocando una mayor oscuridad en las esquinas.

En esta etapa y durante todo el método iremos modificando siempre su luminancia, Y ; nunca modificaremos las componentes de crominancia: U, V .

Para la adaptación de las imágenes implementamos un *script* que contiene una función para realizar la adaptación de la imagen; este *script* recibe la imagen PAP, y la imagen del porta blanco (una imagen sin células en la que se representa la distribución de la intensidad que aporta el microscopio) que presenta oscuridad en las esquinas.

Para ello, aplicamos un método para obtener la imagen compensada o uniforme, método basado en el algoritmo del baricentro. El procedimiento para alcanzar la imagen uniforme es obtener, en primer lugar, la imagen del porta blanco o vacío y su inversa. Con ambas obtenemos una relación que compensará la imagen de células; esta relación se aplica sobre el punto de máxima luminancia o baricentro de la imagen con células.

Este punto es necesario para aplicar la relación con su punto homólogo en el porta blanco. Para extraer dicho punto, aplicamos el algoritmo del baricentro, y, a partir de él, obtendremos la imagen con células uniforme o compensada. A continuación, una explicación más detallada:

1. Necesitamos la imagen con el porta blanco, de esta buscamos su inversa, extraída al restar a la luminancia media el valor de cada uno de los píxeles de la imagen del porta blanco. Una vez tenemos la imagen inversa, sumamos la original y la inversa y así tenemos una imagen uniforme con todos los píxeles al valor medio de la luminancia (véase la figura 3.2, donde a la imagen negativa le hemos sumado 128 para poder visualizarla, ya que los valores negativos en Matlab están representados como “negros” y no podríamos observar este efecto).

Cuando hablamos de compensar, significa, eliminar el efecto que introduce el microscopio que provoca oscuridad en las esquinas y dejar una imagen con una luminancia media uniforme en toda la imagen.

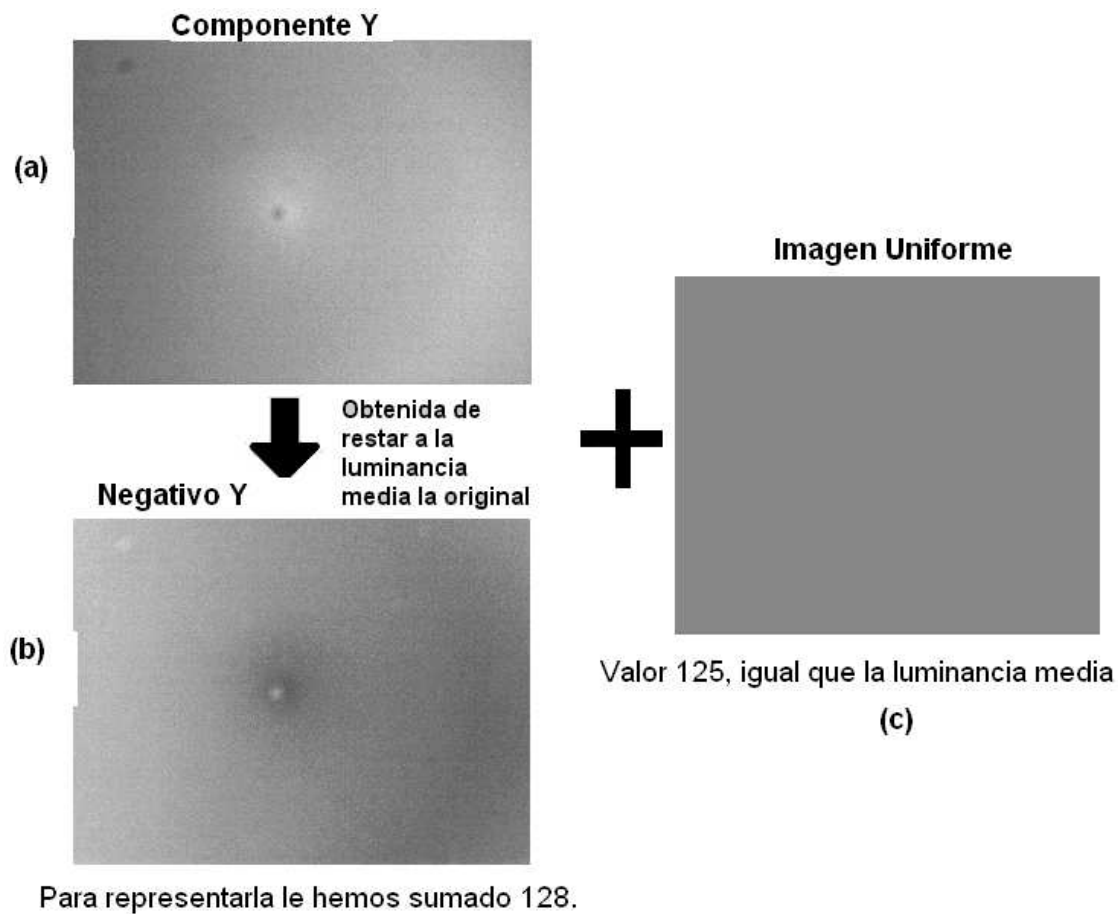


Figura 3.2: **Extracción de Imagen Uniforme.**

(a) luminancia de la imagen del porta en blanco (observamos un punto más oscuro en la zona central); (b) la inversa de la anterior (el centro de la imagen un punto con mayor claridad); y (c) suma de ambas que da como resultado una imagen uniforme.

2. En función de los valores obtenidos de la imagen original del porta blanco y su inversa, sacamos una relación; ésta se aplicará sobre el punto de máxima luminancia o baricentro de la imagen con células, necesario para establecer la relación con el punto homólogo en la imagen del porta blanco.

3. Una vez tenemos la imagen uniforme del porta blanco tenemos que compensar la imagen con células. Para ello aplicaremos el algoritmo del baricentro, explicado en la subsección siguiente 3.2, para encontrar el punto de máxima luminancia de la imagen con células; normalmente este se encuentra en el centro de la imagen pero no tiene por qué cumplirse siempre.

Algoritmo del baricentro

Realizamos una partición de la imagen en 3x3 bloques o cuadrantes y calculamos para cada uno su energía (esta es calculada sumando las intensidades de todos los píxeles del cuadrante donde nos encontramos). Una vez tenemos dichas energías, las comparamos; y el que presenta mayor energía, lo seleccionamos y volvemos a repetir el algoritmo en este cuadrante, hasta que el número de píxeles del cuadrante en el que nos encontramos sea menos que 100, este criterio ha sido especificado por nosotros, y del cuadrante que hemos obtenido calculamos su punto de máxima luminancia.

Véase la figura 3.3 para observar como se aplica dicho desarrollo para obtener una imagen uniforme de luminancia media en todas las partes.

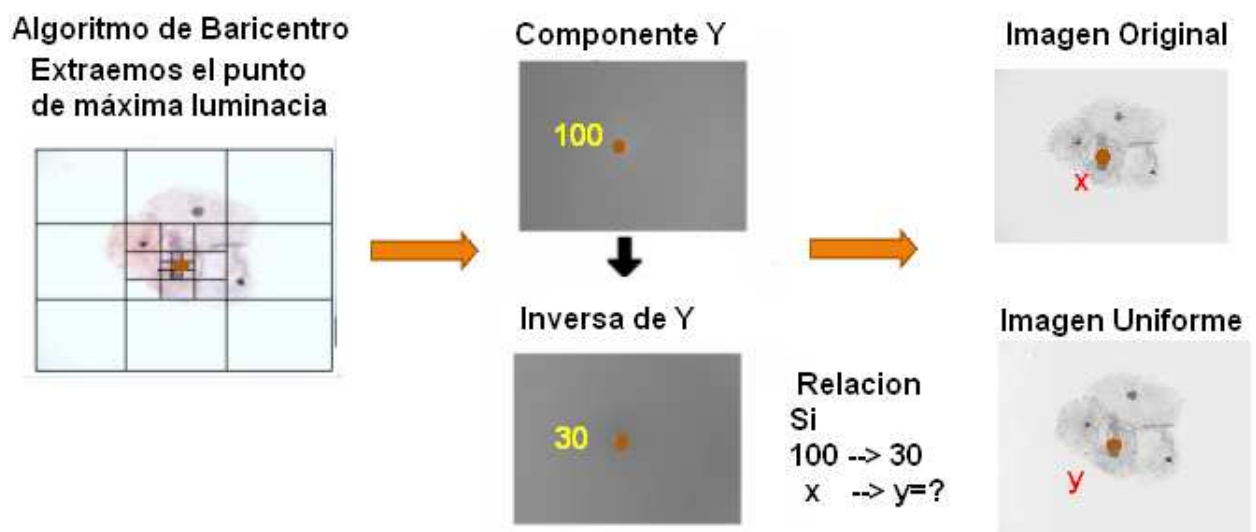


Figura 3.3: Diagrama del Algoritmo de baricentro

Una vez tenemos la luminancia modificada, volvemos a unir en una sola imagen las tres componentes YUV. Como dijimos anteriormente en el método, sólo modificaremos la luminancia, las componentes U, V de crominancia se quedarán igual. Una vez unificadas obtendremos la imagen uniforme (véase la figura 3.7 como ejemplo para observar la imagen uniforme resultante). De esta manera, teniendo como resultado la imagen uniforme o compensada, será introducida en el algoritmo implementado, ya que esta compensación es una de las características requeridas para poder aplicar el trabajo.

A continuación explicaremos la segunda etapa, en la cual comenzamos la replicación del trabajo, detalladamente dónde encontraremos dos partes, las cuales son cada uno de los métodos ya citados.

3.3. Desarrollo del algoritmo

En esta segunda etapa comenzaremos a replicar el algoritmo descrito en el artículo. Este algoritmo tiene como objetivo extraer los resultados parciales de cada uno de los métodos que implementaremos por separado. Estas son las dos partes que veremos en esta etapa.

3.3.1. *Grey Level Method*

En esta primera parte de la etapa nos encontramos con el método *Grey level method* que tiene como objetivo extraer las regiones consideradas núcleos y establecer una clasificación de las mismas. Esta clasificación estará basada en los criterios de circularidad y área que veremos más adelante.

Para llegar al objetivo del método propuesto precisamos de varias fases que observamos en el diagrama de flujo que encontramos en la figura 3.4, donde podemos observar las tres fases claramente diferenciadas: Preprocesado, Segmentación y Clasificación de las regiones encontradas.

En la fase de preprocesado, el método tiene como objetivo eliminar el ruido y los artefactos que dificultan la extracción de las regiones que nos interesan, pero siempre preservando la información necesaria de las regiones. En esta fase, y durante todo el método, iremos modificando siempre su luminancia (Y) nunca modificaremos las componentes de crominancia U,V.

En primer lugar, añadiremos al trabajo una clasificación del tipo de imagen en función de su luminancia media; esto lo necesitaremos para la etapa de umbralización. Esta luminancia es necesaria para calcular el umbral, ya que no se detalla en el artículo y debemos extraerlo a través de la experiencia.

Una vez tenemos clasificada la imagen, aplicaremos una ecualización y umbralización para resaltar las regiones núcleo, usando el umbral extraído con ayuda de la luminancia media.

La segunda fase de este método será la de segmentación de las regiones de posibles núcleos y extraeremos sus contornos a partir de la imagen umbralizada (imagen binarizada que utilizamos para extraer las regiones mediante la aplicación de componentes conexas y sus contornos mediante el algoritmo de Canny, como explica el trabajo).

La tercera, y última fase, de este método, es clasificar las regiones extraídas en función de su área y de su circularidad.

A continuación explicaremos estas tres fases de forma exhaustiva:

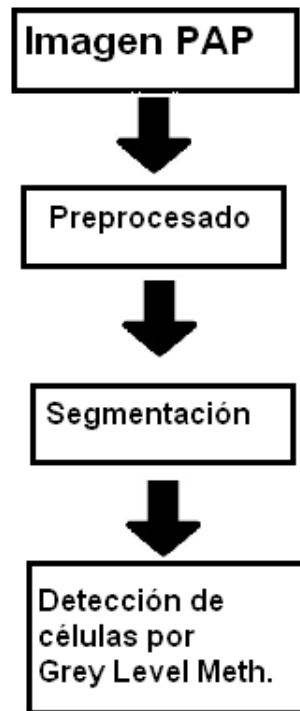


Figura 3.4: Diagrama de flujo “*Grey Level Method*” (Figura extraída de [1]).

3.3.1.1. Preprocesado

El fundamento principal del preprocesado es que este tipo de imágenes son muy ruidosas y esto provoca una peor detección de los núcleos en ellas. Por eso, se requiere forzosamente un preprocesado de la imagen. El preprocesado de las imágenes hace referencia a la etapa de simplificación, que vimos en el capítulo anterior, donde pretendemos eliminar ruido y artefactos que dificultan la extracción de las regiones que nos interesan, pero siempre preservando la información necesaria.

Esta fase estará dividida en los siguientes pasos: en primer lugar, realizamos una clasificación de las imágenes (esto lo añadiremos al artículo ya que será necesario para la umbralización); en segundo lugar, realizamos una ecualización del histograma; y por último, una umbralización para obtener la imagen binarizada que queremos como resultado de esta fase. Estos dos últimos pasos son los principales de esta fase.

Para comenzar a realizar el preprocesado de la imagen, debemos observar nuestro dataset, ya que esta etapa se adecúa en función de las imágenes. En él, hemos comprobado que existen dos tipos de imágenes: claras u oscuras (véase la figura 3.5 para observar dichos tipos de imágenes).

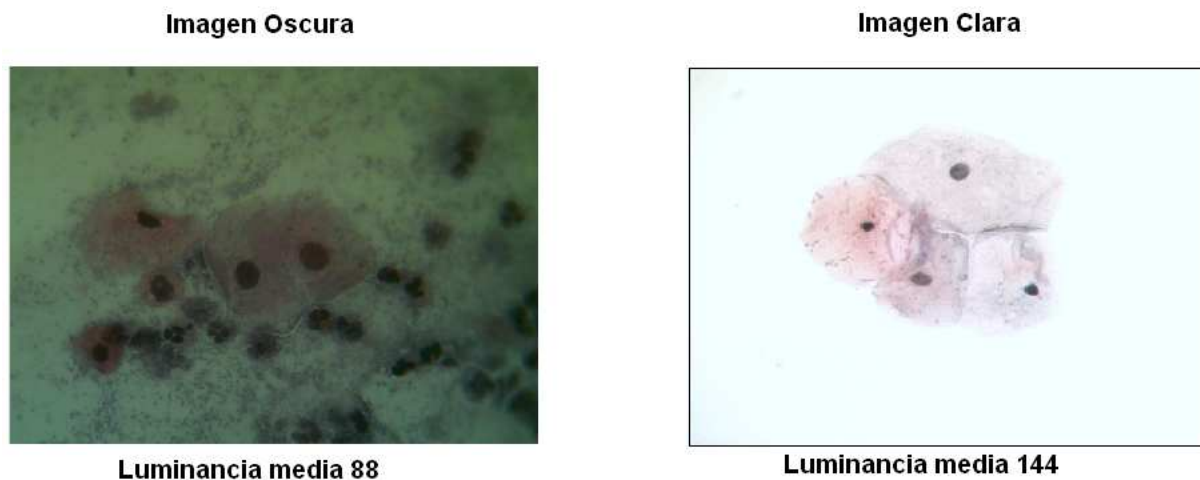


Figura 3.5: Ejemplos de Imagen Clara e Imagen oscura del dataset.

Añadiremos esta clasificación de las imágenes al trabajo debido a que necesitaremos conocer la luminancia media de la imagen para calcular el umbral que utilizaremos en el paso de umbralización del preprocesado. Este dato no está reflejado en el artículo pero sabemos que es necesario extraerlo tras nuestra experiencia en este campo.

La luminancia media se calcula dividiendo en primer lugar la imagen en sus tres componentes Y, U, V (véase la figura 3.6 para observarlas); de ellas la que tomaremos es Y, que hace referencia a la luminancia. Con dicha imagen calculamos la luminancia media, sumando todos sus datos y dividiéndolo entre el número de píxeles totales de la imagen, si esta es menor que 128, será considerada una imagen oscura, en caso contrario, será considerada una imagen clara.

Esta clasificación sólo se añade al artículo para poder extraer el umbral acorde al tipo de imagen, ya que este valor no se nos da en el desarrollo del artículo.

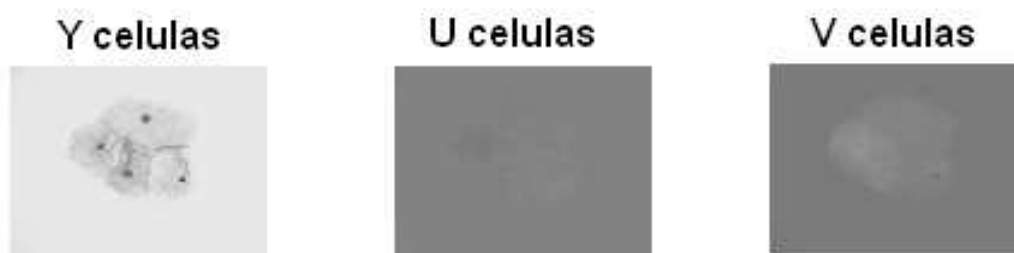


Figura 3.6: División de la imagen en sus componentes YUV.

Explicaremos el método con un ejemplo de imagen clara y tras explicarlo completamente con dicho tipo de imagen mostraremos en la sección de resultados un ejemplo de imagen oscura .

El preprocesado del artículo, como ya anticipamos, tiene dos pasos principales: una ecualización al histograma y, por último, umbralizar la imagen. La ecualización del histograma se aplica para mejorar su contraste y dejar más oscuras las zonas de los núcleos que el resto de la imagen y así facilitar su extracción.

En general, la **ecualización del histograma** de una imagen es una transformación que pretende obtener para una imagen un histograma con una distribución uniforme, es decir, todos los niveles de gris tienen el mismo número de píxeles (Véase la figura 3.7 para ver el resultado sobre nuestra imagen). En el apéndice de “Herramientas de procesamiento de imágenes” se explica detalladamente esta técnica.

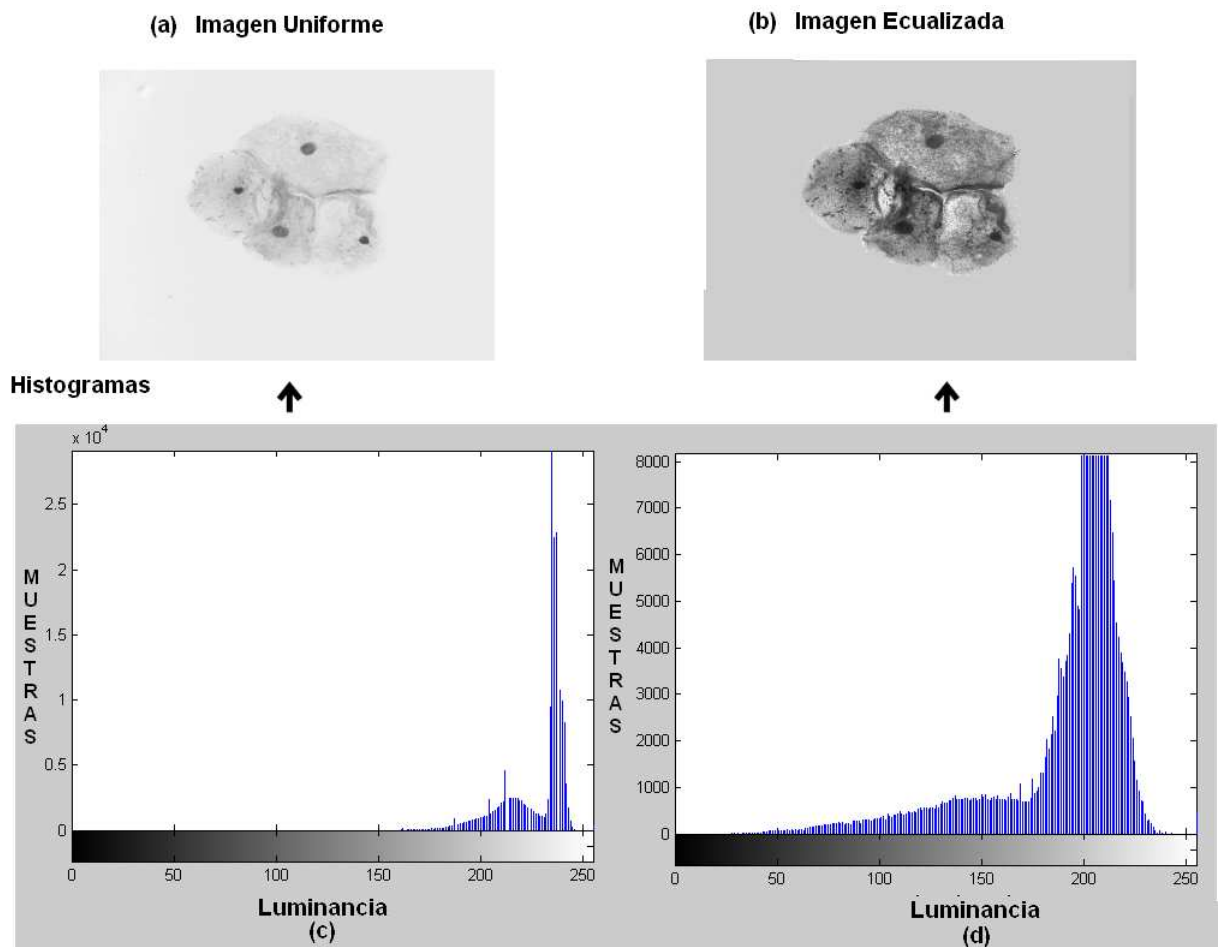


Figura 3.7: Ecualización sobre la imagen clara.

(a) la imagen uniforme obtenida tras el algoritmo del baricentro, (c) histograma de (a) (distribución de la intensidad de los píxeles(luminancia)), (b) una vez se ha aplicado la ecualización del histograma y (d) representa el histograma de la imagen ecualizada.

En la figura 3.7 observamos como en el histograma de la imagen uniforme los valores de intensidad se encuentran acumulados en torno a los valores más claros; estos valores dificultan la extracción del umbral para poder detectar las regiones como núcleo. Pero, por el contrario, en la gráfica

de la derecha observamos un histograma que tiene los valores de la intensidad más repartidos, pudiendo establecer el umbral de una manera más sencilla para extraer las regiones consideradas núcleos (valores más oscuros de la imagen).

Una vez tenemos la imagen ecualizada entramos en el paso final de la fase de preprocesado, la Umbralización. La umbralización es un procedimiento en el cual se obtiene una imagen binaria de la siguiente manera:

$$\begin{cases} Pixel(a,b) = 1 & Pixel(a,b) > Umbral \\ Pixel(a,b) = 0 & Pixel(a,b) < Umbral \end{cases}$$

Esto es, si el píxel seleccionado es mayor que el umbral, se considera que no es núcleo, y por tanto se queda como blanco, y en el caso contrario se considera núcleo y se sitúa como negro (véase la figura ?? para observar dicho resultado).

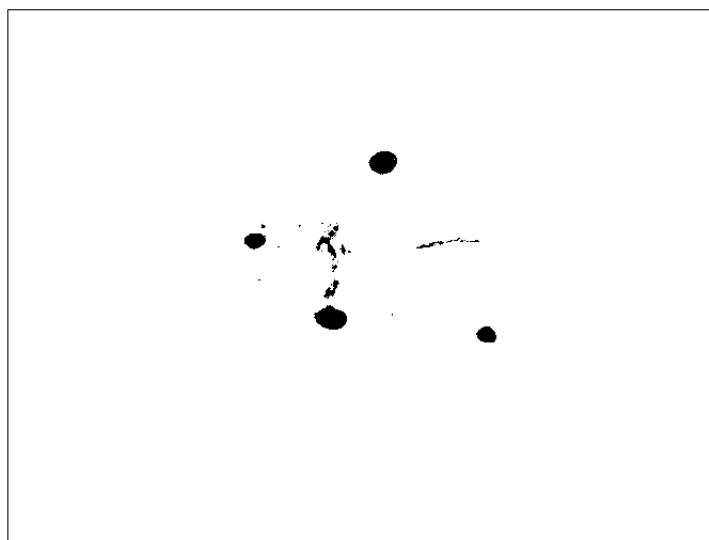


Figura 3.8: **Imagen Umbralizada.**

Como indicamos anteriormente, el artículo no facilita el dato del umbral que debemos utilizar, por ello, añadimos la parte de clasificación de las imágenes en: claras u oscuras, para poder facilitar la elección de dicho umbral. El artículo lo único que nos facilita sobre este paso del preprocesado es la aplicación de “*Line-Scanning*”; éste consiste en analizar fila a fila la imagen y comprobar que la zona de los núcleos presenta un claro descenso de intensidad, zona de mínimos. Y cuando observas esos mínimos podrás establecer un valor de umbral. (véase la figura 3.9 para observar este resultado). Aunque no lo utilizamos para la estimación del umbral, tan solo nos sirve para hacernos una idea de donde debería que estar situado el valor.

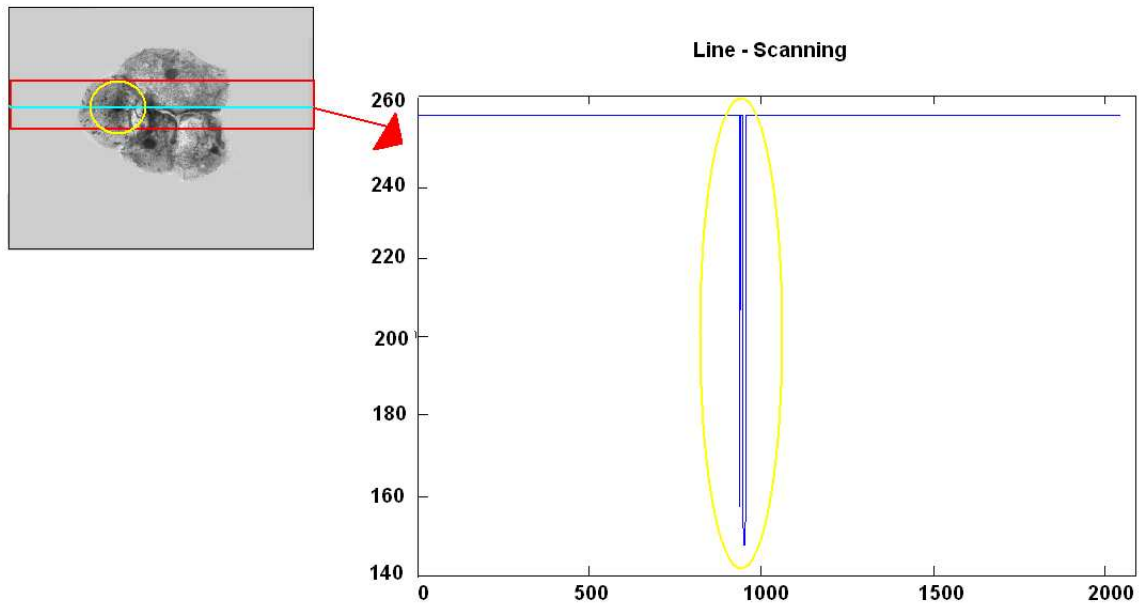


Figura 3.9: *Line-Scanning* Imagen Clara.

En este momento ya podemos escoger el valor del umbral, debemos hacerlo según la experiencia, ya que el artículo no especifica dicho dato, como ya hemos explicado. Por este motivo, nosotros hemos elegido que el umbral se detenga en el 10% de los valores acumulados del histograma de la imagen modificada, siendo distinto el valor de ambos tipos de imagen (clara u oscura) por eso realizamos la clasificación inicial.

Como resultado del preprocesado, obtenemos una imagen binarizada como vemos en la figura 3.8, que utilizaremos como máscara para la siguiente fase del método: segmentación. En esta fase utilizaremos la imagen binaria (la cual tiene resaltadas las regiones extraídas) como máscara para extraer sus contornos mediante el algoritmo de Canny y extraer sus regiones a través del algoritmo de componentes conexas.

3.3.1.2. Segmentación

El objetivo de la segmentación es extraer regiones con características similares, en este caso las regiones que nos importan son los núcleos que presenten las imágenes.

En la segmentación, encontramos dos pasos: extracción de contornos mediante el algoritmo de Canny, para señalar los núcleos extraídos; y el segundo paso, la extracción de las regiones que corresponden a los núcleos mediante la aplicación de componentes conexas.

El primer paso de la segmentación es la extracción de contornos; se lleva a cabo aplicando el método de Canny. En el apéndice de “Herramientas de procesamiento de imágenes ” se explica detalladamente esta técnica.

El método de Canny encuentra bordes mediante la búsqueda de máximos locales del gradiente de la imagen. El gradiente se calcula utilizando la derivada de un filtro gaussiano. El método utiliza dos umbrales, para detectar los bordes fuertes y débiles, e incluye los bordes débiles en la salida sólo si están conectados a los bordes fuertes. Mediante la función de Matlab: “**E=edge(imagen,'canny')**” extraemos los bordes.

A continuación, tras observar que al aplicar Canny como detallaba el artículo, obteníamos ruido en la imagen hemos tomado la decisión de aplicar las siguientes funciones de Matlab que se encargan de eliminar estas imperfecciones. Hemos tomado esta opción, debido a que las imágenes del artículo no son exactamente iguales que las que nosotros encontramos en nuestro dataset; esto implica que las imágenes que tenemos pueden ser más ruidosas y provoca errores que debemos subsanar.

Las siguientes funciones de Matlab que utilizaremos son: primero aplicaremos: “**BW2 = bwperim(BW1)**” (devuelve una imagen binaria que contiene sólo los píxeles del perímetro); segundo, “**B2=bwmorph(E2,'close')**” (Realiza el cierre morfológico (dilatación seguida por la erosión)); y por último, “**B=bwfill(B2,'holes')**”, **B1=bwmorph(B,'remove')**, **B3=bwmorph(B1,'thicken')**” (se encuentran los interiores de los contornos, se eliminan y se eliminan los exteriores que no son contornos) (Véase la figura 3.10 para observar que los errores han sido subsanados).

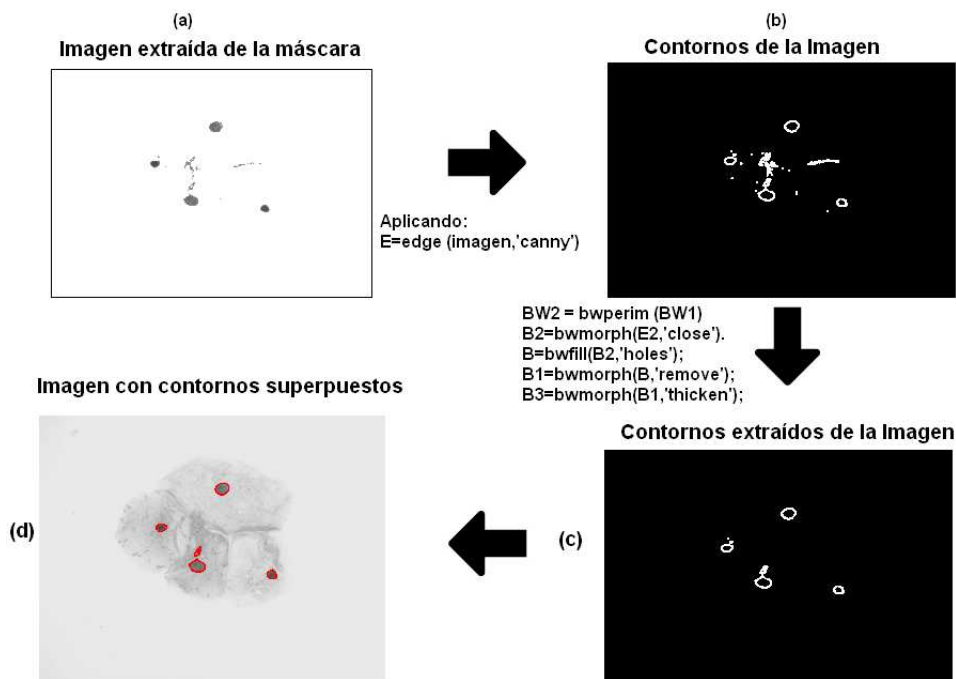


Figura 3.10: Imagen Clara con contornos.

En esta figura encontramos el proceso que siguen la imagen umbralizada para extraer los contornos. (a) imagen umbralizada, (b) imagen extraída de aplicar Canny (c) imagen tras aplicar las funciones (d) imagen original a color con los bordes extraídos sumados.

Una vez tenemos los contornos obtenidos, continuamos con la segmentación situándonos en el paso final de la misma, donde extraeremos las regiones consideradas núcleos mediante la extracción de componente conexas, es decir, las zonas más oscuras de la imagen.

Aplicaremos la función de Matlab que halla las componentes conexas (**bwconncomp**) la cual a través de sus características nos da el tamaño de las áreas. Como nos indica el artículo, serán núcleos anormales los que sean de un tamaño mayor y olvidaremos los que presenten un tamaño menor. Además, en el artículo, también se hace referencia a las formas de los núcleos y a los que presentan una circularidad baja, considerándose también núcleos anormales.

Componentes conexas “bwconncomp”

Esta función traza los límites de los objetos en una imagen binaria, que nosotros le pasamos, en este caso la umbralizada, y devuelve la información sobre ellos en una estructura con los siguientes campos:

1. Conectividad: La conectividad utilizado en el límite de rastreo.
2. ImageSize: El tamaño de la imagen BW (Black and White (Blanco y negro)).
3. NumObjects: El número de objetos de la imagen.
4. PixelIdxList: Cada elemento está representado como un vector de índices lineales del límite del objeto dado.

La conectividad utilizada en el trazado por defecto es 4, pero se puede cambiar estableciendo el parámetro de entrada de la conectividad a 8 (Véase la figura 3.11 para observar el resultado).

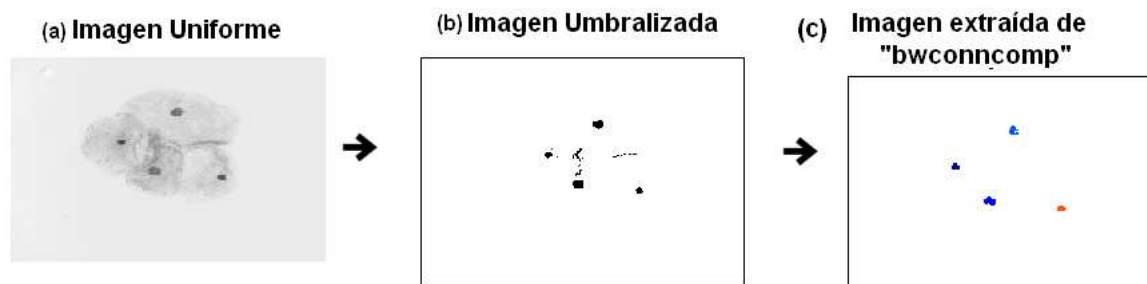


Figura 3.11: Imagen obtenida tras **bwconncomp**.

(a) imagen uniforme que antes hemos obtenido, (b) imagen umbralizada y (c) los resultados tras aplicar la función de Matlab “bwconncomp”.

3.3.1.3. Clasificación de regiones segmentadas

La fase de clasificación es la fase final de *Grey Level Method* donde estableceremos los criterios para dividir las regiones extraídas en dos tipos: núcleos normales y núcleos anormales. Así, tras

aplicar la función “bwconncomp”, aplicamos la función “regionsprop”, que permite extraer las características requeridas por el autor del artículo.

Estos métodos, tanto *Grey level method* que hemos explicado como *Energy method* que explicaremos a continuación, tienen como objetivos la extracción de las células anormales, en base a dos criterios (coincidentes con las características de la célula en las que el citopatólogo se fija): su tamaño y su forma anormal (circularidad). Además, otro factor para clasificar las células son los cambios biológicos de la célula y citoplasma, pero esto es más difícil de reconocer en imágenes fijas. Por lo tanto, nos centraremos en los dos primeros parámetros:

1. Tamaño de la célula-”Área”

- a) Cuando el tamaño de la célula es excesivamente grande, se considera célula anormal. El área del núcleo es el número de píxeles que encierra el perímetro o contorno de un núcleo. Asumimos que el núcleo es el orgánulo principal de las células eucariotas, contenido en el citoplasma y constituido esencialmente por cromatina.
- b) Extraemos dicho parámetro directamente de la función “regionsprop”. El criterio para esta característica es, según el artículo, si un área tiene un tamaño grande se considera célula anormal (una célula es grande, según el artículo, si está por encima de la media de las células de la imagen) El criterio establecido por nosotros es: si un área supera la media por encima del 50 % se considera núcleo anormal.

2. Forma anormal, basado en la Circularidad :

- a) La forma de las células pueden ser redonda, ovalada o elíptica, y se obtiene dividiendo el área entre el perímetro al cuadrado. Así la forma responderá a :

$$Circularidad = 4\pi * \frac{Area}{Perimetro^2}$$

Los valores pequeños del parámetro indican objetos alargados. Además, es invariante frente a traslaciones, giros y cambios de escala. Conviene normalizar sus valores al intervalo [0,1] dividiendo entre $1/(4\pi)$ para garantizar que el valor que le asigne al círculo sea 1, y un valor menor a cualesquiera otra figura geométrica, tanto menor cuanto más ovalada sea.

El siguiente esquema presenta los parámetros que deseamos obtener (véase la figura 3.12 para observar los criterios de clasificación):

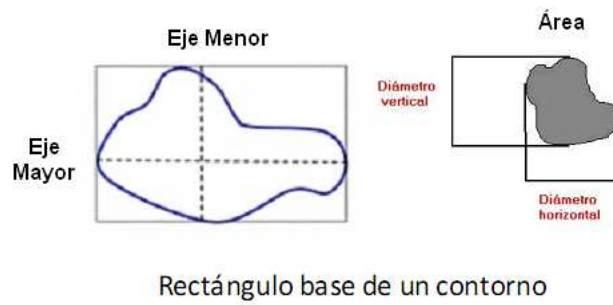


Figura 3.12: **Características de núcleo.**

Esta figura representa la manera de obtener las características que obtendremos de forma detallada.

Los valores utilizados en esta sección, debido a la ausencia de datos específicos del artículo, son los siguientes: en el caso del área, hemos calculado el tamaño de las áreas de todas las regiones extraídas de todo nuestro dataset y realizando su media con los datos obtenidos hemos extraído que si un área está por encima de 154 píxeles es considerada maligna, y en el caso contrario, benigna; en el caso de la circularidad, tenemos un rango de valores de $[0-1]$, cuanto más cerca del 1 más circularidad y tras la experiencia determinamos que cuando está por debajo de 0,3 es maligno, y en caso contrario benigno.

3.3.1.4. Resultados de *Grey Level Method*

Las pruebas realizadas con las 88 imágenes de nuestro dataset serán en base a estos criterios anteriormente especificados. Se pueden observar resultados parciales con una imagen oscura como ejemplo en las figuras: 3.13 y 3.14.

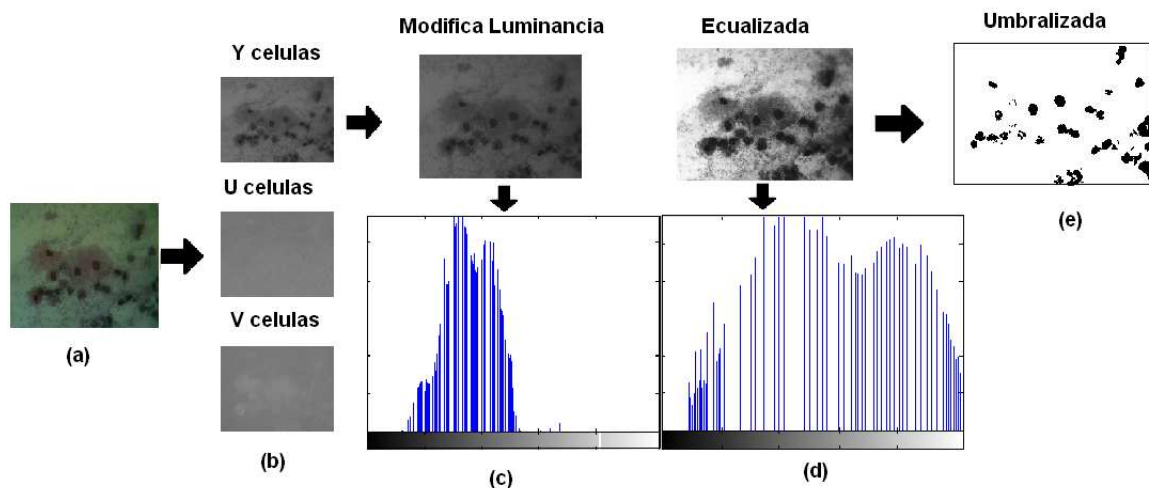


Figura 3.13: **Proceso Imagen Oscura.**

Muestra el proceso completo: (a) imagen original (b) las tres componentes: (Y, U, V) (c) (d) aplicación de la ecualización y (e) imagen umbralizada.

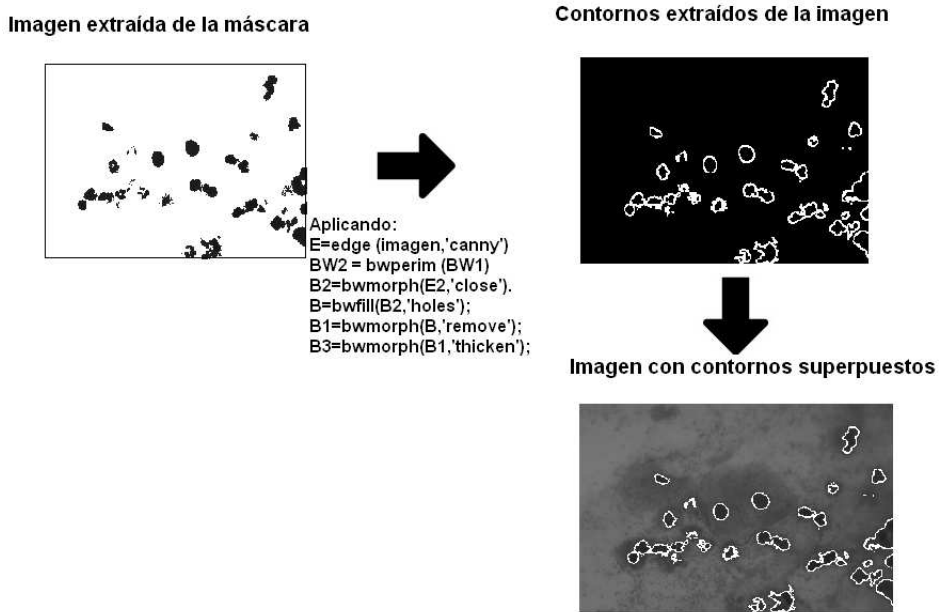


Figura 3.14: Contornos en Imagen Oscura.

Muestra el resultado parcial de la extracción de contornos, en solitario y sumados a la imagen original.

En las figuras 3.15, podemos observar resultados finales de ambos tipos de imágenes.

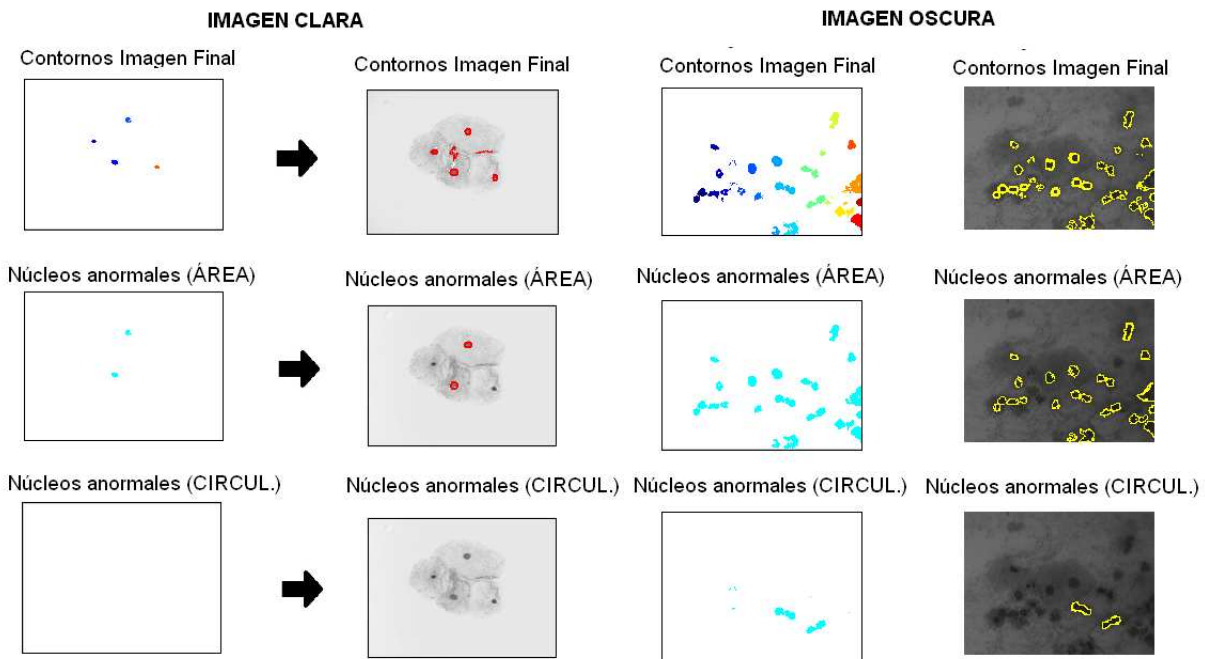


Figura 3.15: Selección de Núcleos Anormales

Encontramos tres resultados divididas en dos columnas: la columna de la izquierda son los resultados extraídos con “bwconncomp” y la de la derecha muestra los contornos resaltados, de los núcleos anormales, en amarillo en la imagen de luminancia de la original.

3.3.2. *Energy Method*

Este método, referenciado en [1], tiene el mismo objetivo que veíamos en *Grey Level Method*, obtener una clasificación de las regiones extraídas como núcleos.

Su fundamento, a diferencia del anterior, está basado en que los núcleos de las células anormales tienen unos niveles de gris bajos. Si se presentan en zonas más oscuras, muchas veces no son detectados con “*Grey Level Method*” (sección 3.3.1). Por ello, “*Energy Method*” añade un preprocesado al método anterior para contrarrestar esta situación. El desarrollo del método posterior es el mismo que seguimos con “*Grey Level Method*”.

En este método hay dos vías, como vemos en el diagrama de flujo de la figura (3.16), que son similares pero con un detalle que las diferencia. La primera de ellas consiste en aplicar un filtro de media con una máscara grande que sirve para extraer el fondo de la imagen; y en la segunda vía se aplica un filtro de media con máscara pequeña para extraer los objetos que queremos. Una vez tenemos las imágenes filtradas, las restamos posteriormente, para extraer los núcleos anormales aplicamos *Grey level method*. (Véase la figura 3.16 para observar el diagrama).

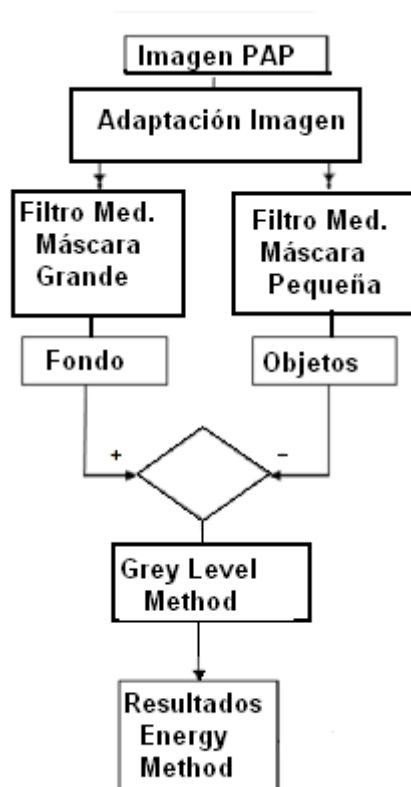


Figura 3.16: Diagrama de flujo de *Energy Method*. (Figura extraída de [1]).

3.3.2.1. Preprocesado

Este método trabaja con la imagen a color, no como en el caso anterior que se trabajaba con la luminancia de la imagen o escala de grises.

En primer lugar, aplica un filtrado de mediana con dos máscaras distintas, la primera de ellas es una máscara mayor, que extrae el fondo de la imagen, mientras que la segunda máscara es más pequeña y es utilizada para resaltar los puntos más oscuros.

En el procesamiento de imágenes, a menudo, es deseable ser capaz de realizar algún tipo de reducción de ruido. El filtro de mediana es un filtrado no-lineal utilizado para esto. Esta reducción de ruido es típica para mejorar los resultados de procesamiento posterior (por ejemplo, la detección de bordes en una imagen). El filtro mediana se utiliza mucho en procesamiento digital de imágenes, ya que conserva los bordes.

Por este motivo, aplicamos dichos filtros a cada una de las componentes de color de nuestra imagen, R G B, de la siguiente manera: (véase la figura 3.17 para observar dicha aplicación).

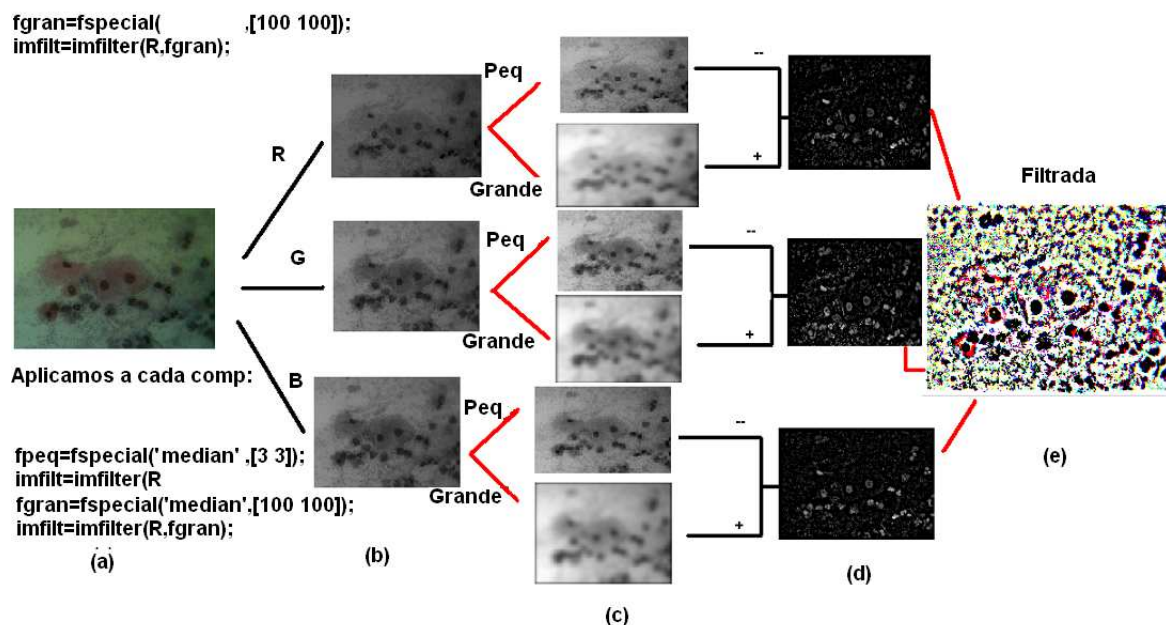


Figura 3.17: **Preprocesado de Energy Method: Filtros medianas.**

En la figura encontramos el desarrollo de la parte añadida a *Grey Level Method* del preprocesado: (a) muestra la imagen original, la cual se separa en sus tres componentes de color (b): R, G y B, una vez están separadas aplicamos los filtros (c) el resultado del filtro menor arriba y el del mayor abajo para cada una de las componentes, una vez tenemos aplicados los filtros extraemos el menor al mayor (d) y por último (e), muestra el resultado unificando de nuevo las componentes.

Para cada componente realizamos el siguiente procedimiento: aplicamos las dos máscaras, y cuando tenemos los dos resultados, extraemos el resultado de la máscara menor al de la máscara

mayor, obteniendo una imagen que resalta los mínimos de cada componente. Posteriormente, uniremos al resultado de las otras dos componentes obteniendo una imagen que mejora, el contraste entre los objetos y el fondo, manteniendo los bordes de la imagen. Esta mejora nos facilita la elección del umbral posterior, ya que tenemos resaltados los mínimos respecto del fondo pudiendo extraer el umbral de una manera más robusta.

La aplicación a las tres componentes por separado se debe a que resaltando los mínimos de cada una de ellas, resaltaremos los mínimos totales de la imagen de manera más pronunciada.

Como se apunta en el artículo, el tamaño del filtro y la elección del umbral son los factores más importantes, ya que el umbral viene marcado por una buena elección del tamaño de los filtros. Hemos elegido el tamaño, tal y como se describe en el artículo, en función del tamaño de los núcleos (el área media de los núcleos es de 80 píxeles) que presentan nuestras imágenes, y tras realizar varias pruebas. Los mejores resultados se han obtenido con un tamaño de 3x3 para la máscara menor y de 100x100 para la máscara mayor.

A continuación, mostramos un ejemplo de aplicación del filtro de media de la máscara menor (ya que es inviable mostrar el ejemplo de la máscara mayor):

Máscara de [3x3]:

$$\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{matrix}$$

Siendo el píxeles central el que vamos a dar un nuevo resultado en función de los 8 vecinos, el resto de la imagen no lo tenemos en cuenta.

Valores por lo que multiplica es $1/9$, quedándonos una matriz:

$$\begin{matrix} 0,111 & 0,111 & 0,111 \\ 0,111 & 0,111 & 0,111 \\ 0,111 & 0,111 & 0,111 \end{matrix}$$

Así si tenemos un píxel central, rodeado de su vecindad de esta manera;

$$\begin{matrix} 150 & 128 & 132 \\ 135 & 170 & 240 \\ 156 & 165 & 147 \end{matrix}$$

El píxeles cuyo valor es 170, ahora pasa a tomar el valor de:

$$Pix_{cent} = 150 * 0,111 + 128 * 0,111 + 132 * 0,111 + 135 * 0,111 + 170 * 0,111 + 240 * 0,111$$

$$+156 * 0,111 + 165 * 0,111 + 147 * 0,111 = 157,953 \simeq 158$$

Una vez hemos aplicado este filtro, tenemos una imagen simplificada como resultado de la fase de preprocesado que hemos añadido en *Energy method*. Los pasos posteriores a los filtros empleados (el menor de 3x3 y el mayor de 100x100), desde la ecualización y umbralización de la fase de preprocesado, la segmentación y la clasificación se realizan de la misma manera que en *Grey level method* (explicado en la sección 3.3.1). Pero en este caso la aplicación de dicho método, al tener una imagen mucho más simplificada, presenta un histograma más definido, que nos permite extraer mejor el umbral y obtener los núcleos.

El cálculo del umbral con el método de “Line-Scanning”, se verá optimizado gracias a que tras el filtrado, la diferencia entre máximos y mínimos es mucho mayor (véase la figura 3.18).

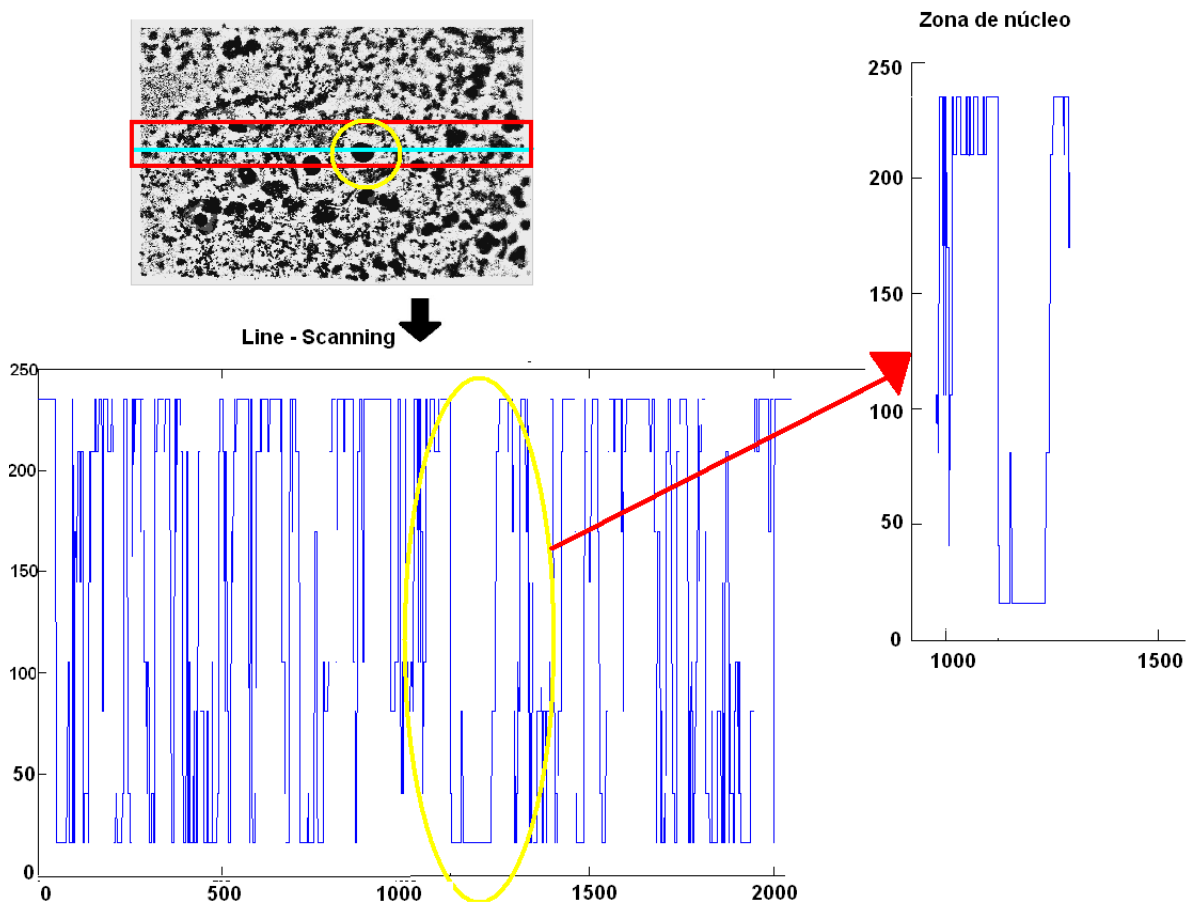


Figura 3.18: **Line-Scanning**.

Encontramos la imagen superior que es la imagen ecualizada, a la cual aplicamos Line-Scanning extrayendo la gráfica inferior y ampliamos la zona más relevante, mostrándola en la imagen de la derecha, para observar el mayor espacio entre máximos. Los valores mínimos son en torno a 10 de intensidad y los máximos en torno a 240, mientras que en el caso del otro método el mínimo estaba alrededor de 140 y el máximo a 255. Esto facilita la extracción del umbral.

Tras este paso umbralizamos y extraeremos los contornos mediante el algoritmo de Canny(al igual que en “*Grey Level Method*”).

Véase la figura 3.19 para observar los resultados del algoritmo tras realizar el preprocesado añadido.

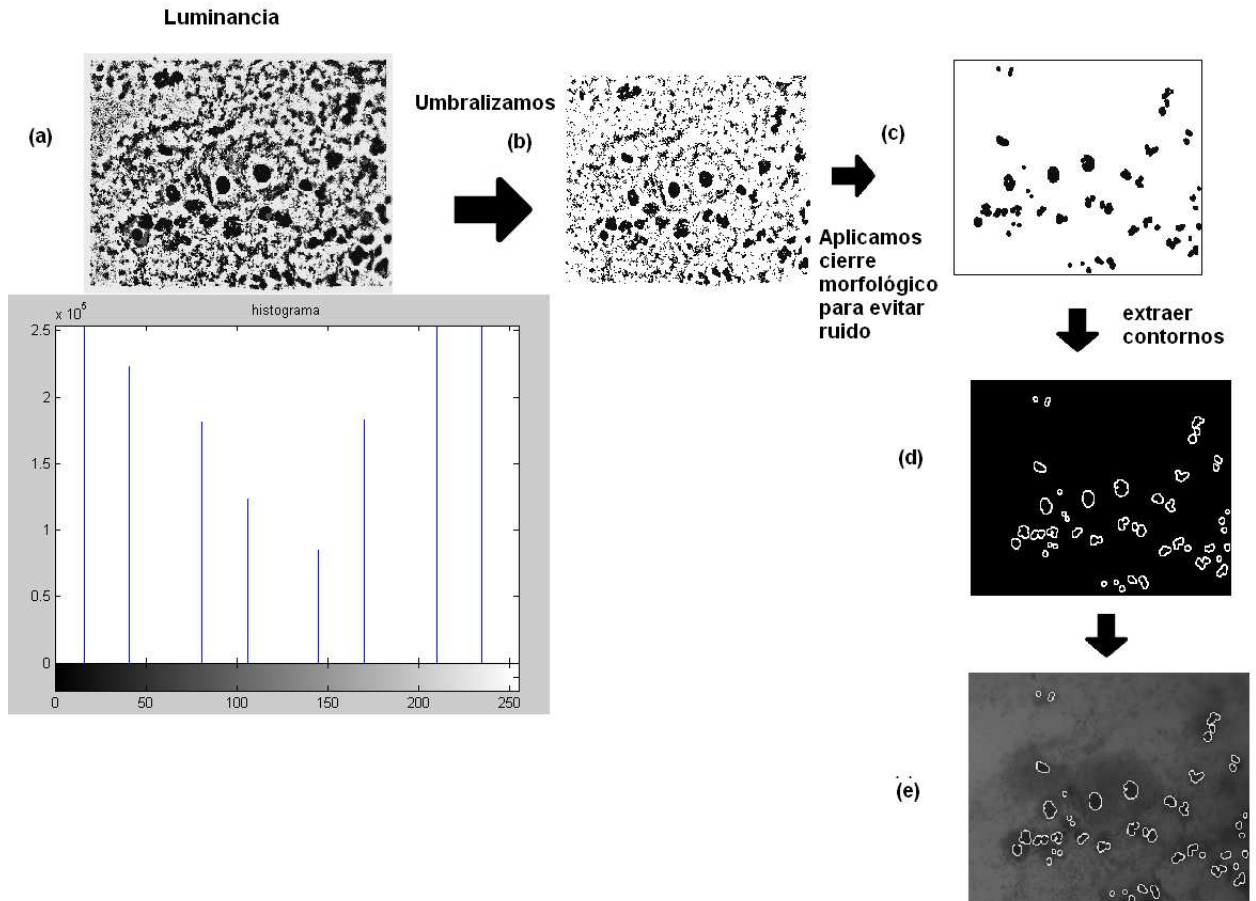


Figura 3.19: **Preprocesado + Segmentación de Energy Method.**

En la figura encontramos las siguientes imágenes: (a) Imagen ecualizada (componente de luminancia) y su histograma en la parte inferior, (b) Imagen umbralizada, (c) Aplicación de operaciones morfológicas, (d) Contornos extraídos y (e) Imagen original (luminancia) con sus contornos superpuestos.

Y una vez obtenidos estos resultados hasta la primera parte de la fase de segmentación, extraemos las regiones mediante “*bwconncomp*” terminando la fase de segmentación y finalizamos con la fase de clasificación (al igual que en “*Grey Level Method*”) (véase la sección de componente conexas explicada en 3.3.1.2 para poder obtener los resultados que se encuentran a continuación).

3.3.2.2. Resultados de *Energy Method*

Estos resultados han sido obtenidos con los mismos criterios de la etapa de clasificación que vimos en el apartado de resultados de “*Grey Level Method*” 3.3.1.3.

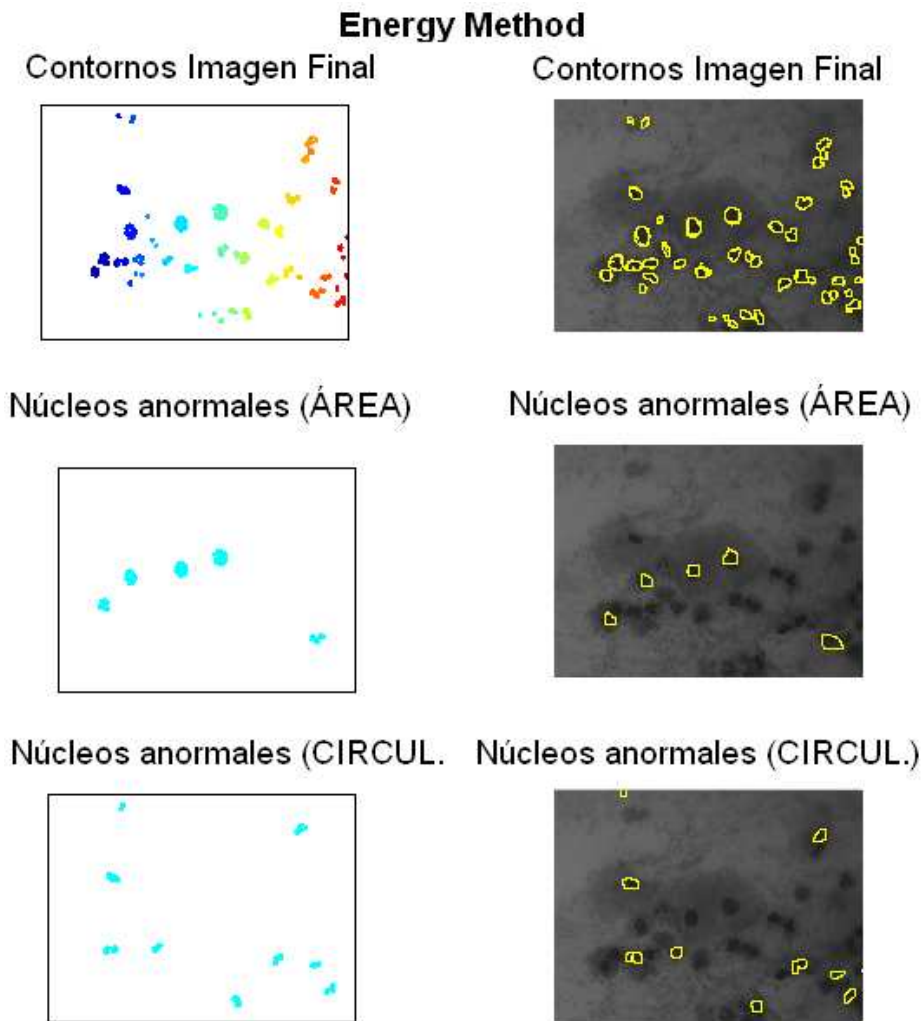


Figure 3.20: **Resultados de *Energy Method*.**

En la figura encontramos dos columnas: la columna de la izquierda muestra los resultados a través de la función “bwconncomp” y la de la derecha la imagen original (componente de luminancia) con los contornos superpuestos. La primera de las filas de la imagen contiene todas las regiones extraídas. La segunda de las filas muestra las células anormales según el criterio de área. Y la última, muestra los núcleos anormales según el criterio de circularidad.

3.4. Resultados finales del algoritmo

Finalmente, en esta última etapa, se unen los resultados de ambos métodos, es decir, los núcleos en los que coinciden ambos métodos, para obtener como resultado final las células anormales consideradas (véase la figura 3.22).

En primer lugar, mostramos los resultados obtenidos con cada método (véase la figura 3.21) y a continuación, mostramos los resultados unificados (véase la figura 3.22).

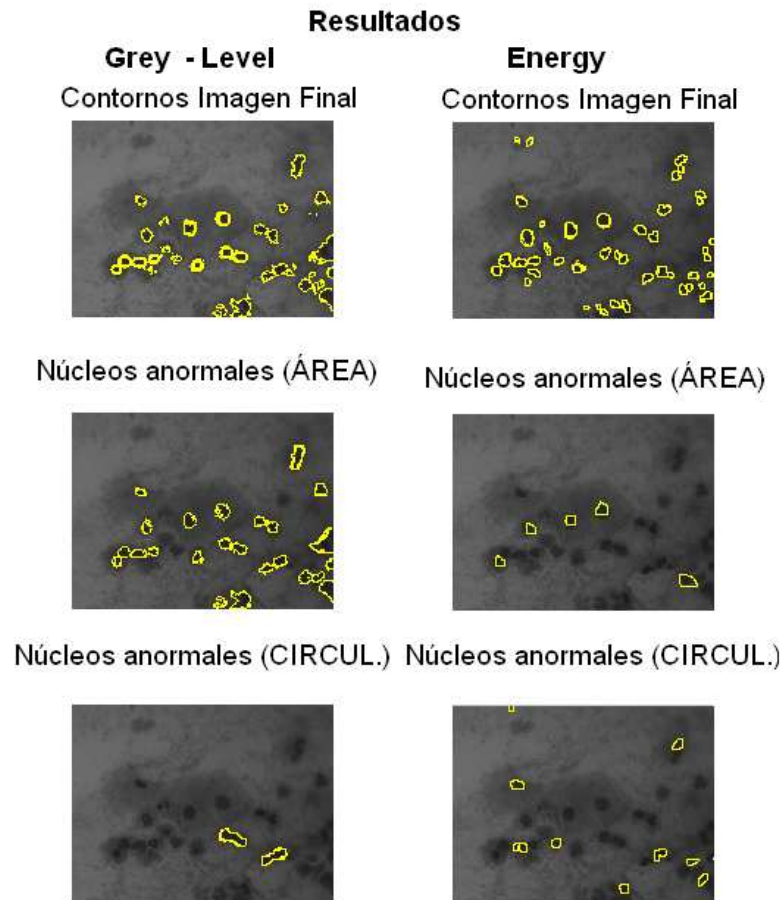


Figura 3.21: **Resultados finales de cada método por separado**

La columna de la izquierda muestra los resultados de *Grey Level Method*, y la de la derecha, los de *Energy Method*.

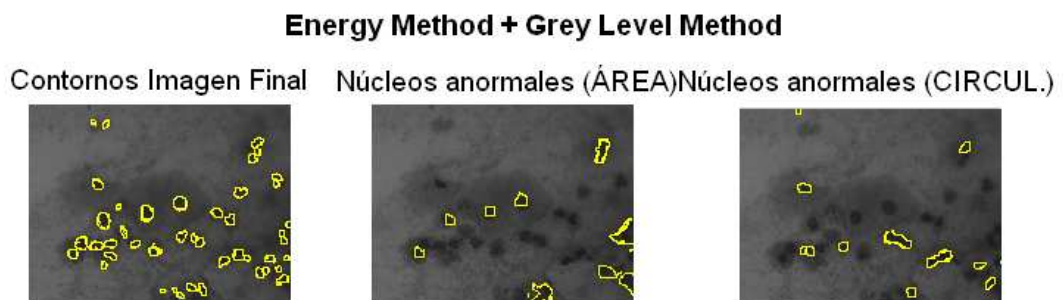


Figura 3.22: **Resultado final algoritmo *Grey Level Method* + *Energy Method*.**

La figura muestra tres imágenes: la primera de ellas (comenzando desde la izquierda), contiene todas las regiones extraídas; la segunda, la combinación de resultados de ambos métodos que muestran los núcleos anormales según el criterio de área; y por último, los resultados combinados de núcleos anormales según el criterio de circularidad.

3.5. Análisis Crítico del Algoritmo

Una limitación del algoritmo es que no están definidos muchos de los parámetros necesarios, como por ejemplo, el umbral o los criterios de clasificación. Esto puede producir problemas debido a que las imágenes médicas no presentan un patrón uniforme, lo que dificulta obtener buenos resultados.

Un claro ejemplo de lo anteriormente citado es elegir el umbral. Según la experiencia, este dato supone grandes diferencias entre una imagen clara con una media de luminancia por encima de 128, y una imagen oscura con una luminancia media por debajo de 128.

Además, a la hora de elegir los contornos de las células, también tenemos que escoger el número de píxeles conectados para extraer dicho contorno. Dado que el tamaño de las células no es uniforme y es desconocido, esta elección puede inducir a errores, es decir, que algunas células no sean seleccionadas.

Finalmente, el punto más conflictivo es a la hora de clasificar el tipo de células en: normales o anormales, ya que el criterio que propone el artículo y que explicamos en la etapa de clasificación de las células, en ambos métodos es incongruente:

Por un lado, el criterio basado en el tamaño del área, que aunque bien es cierto que los tamaños de los núcleos son una de las bases para clasificar este tipo de células en benignas o malignas, el modo en que lo realiza este artículo no parece el adecuado. Este artículo emplea una media del tamaño de todas las células que encontramos en las imágenes de nuestro dataset para clasificarlas en benignas o malignas, si supera la media son malignas y si no lo hace son benignas. Si en un caso en el que todas las células son benignas, realiza su media, la que sobrepase esa media, será considerada maligna aun no siéndolo; de la misma manera, si todas son malignas, el extraer su media, no va a darnos resultados reales, ya que nos darán todas buenas, menos una maligna, cuando en realidad todas las células son malignas.

De la misma manera, no parece adecuado el criterio de la circularidad, aunque es cierto que es más razonable que el anterior. La forma de las células permiten una buena clasificación de las mismas. Nosotros tenemos una guía de las formas evaluada de [0-1], donde 0 son las células más alargadas y con puntas (triangulares) y 1 es el círculo perfecto. Este artículo no presenta un valor exacto para especificar el límite de las células, para saber qué es célula benigna y cual no.

Por los motivos que acabamos de explicar observamos que este artículo, no nos proporcionará resultados útiles para nuestro estudio y precisamos de un trabajo más extenso y detallado como el que explicaremos en el siguiente capítulo.

3.6. Propuestas de Mejora del algoritmo

La primera de las mejoras que podemos realizar, haciendo referencia al análisis crítico realizado, sería hacer una clasificación inicial de si pertenece a una imagen clara o a una imagen oscura, y en función de que tipo sea, aplicarles un preprocesado u otro, para finalmente realizar los últimos pasos del algoritmo de la misma manera.

El preprocesado es muy dependiente del tipo de imagen: por esto, clasificarla nos ayudará a realizar el preprocesado adecuado a la imagen y extraeremos una imagen donde realicemos una mejor extracción de las regiones y sus contornos.

Como segunda mejora, haremos referencia al modo de extraer los tamaños medios de las áreas y de la circularidad para extraer las células anormales. Esto lo podemos realizar teniendo un dataset donde tengamos clasificados por un experto los núcleos extraídos como benignos o malignos. Una vez que tengamos esos datos, podríamos extraer una área y una circularidad, el tamaño de área que especifique exactamente a partir de que cuando son considerados núcleos anormales y a partir de la circularidad, esto informa de la forma del núcleo, son considerados con formas de núcleos anormales.

Capítulo 4

Alg. basado en la extracción de características de los núcleos

4.1. Introducción

En este capítulo desarrollaremos el algoritmo de extracción de características, el cual ha sido necesario implementar ante los pobres resultados del artículo anterior y para tener una visión más actual de los desarrollos en este ámbito. La elección fue implementar el algoritmo de extracción de características, elección basada en las razones expuestas en la sección 2.1.3, donde encontrábamos los distintos métodos de segmentación. Cada uno de los métodos era adecuado para un tipo de imagen. Así, el algoritmo de selección de características contenía el método más indicado para nuestro dataset: segmentación por *watershed* (método enfocado a imágenes con gran número de células y no sensible a la luminancia de la imagen).

Este algoritmo sustituye al algoritmo del capítulo anterior, tiene las mismas etapas del anterior (preprocesado y segmentación) y aporta una nueva clasificación de las regiones extraídas en: núcleos verdaderos u otras zonas de la imagen (que se confunden con los núcleos). Esta clasificación es un paso previo a la clasificación de las regiones, determinadas como núcleos verdaderos, para ser divididas en las dos clases que ya vimos: normales o anormales.

El objetivo de este capítulo es explicar detalladamente la implementación desarrollada del algoritmo de extracción de características. Este algoritmo ha sido replicado fielmente con los datos que nos aporta el trabajo [2]; ya que como explicamos en el capítulo anterior, buscábamos un trabajo más extenso y detallado.

Este trabajo ha sido probado por el dataset de 88 imágenes, las cuales distribuiremos en dos grupos: las 56 primeras las utilizaremos para realizar el entrenamiento necesario de la etapa final del trabajo (clasificación de las regiones); y las 32 restantes se ejecutaran para verificar el funcionamiento de la implementación desarrollada.

Durante la explicación del algoritmo mostraremos una imagen del dataset como ejemplo, para observar el resultado parcial de cada una de las etapas hasta obtener el resultado final.

El trabajo, que desarrollaremos, tiene como objetivo extraer regiones candidatas a núcleo y verificar si realmente se trata de un núcleo o, por el contrario, se trata de otra zona de la imagen que no corresponde a este grupo (a diferencia del método del capítulo anterior que extraía las regiones, y las clasificaba sin contemplar que la región sea verdaderamente un núcleo y no otra región). Estas zonas que tienden a confundirse son debidas a la etapa de tinción que vimos anteriormente.

La importancia de detectar los núcleos verdaderos es debido a que después de un largo estudio realizado por los expertos, como se documenta en [2], se llegó a la conclusión de que en ellos se puede observar la anormalidad o normalidad de la célula. Y no deben confundirse con el resto de zonas. Para clasificar las regiones extraídas en: núcleos reales u otras zonas, el artículo sigue el esquema que encontramos en el diagrama de la figura 4.1.

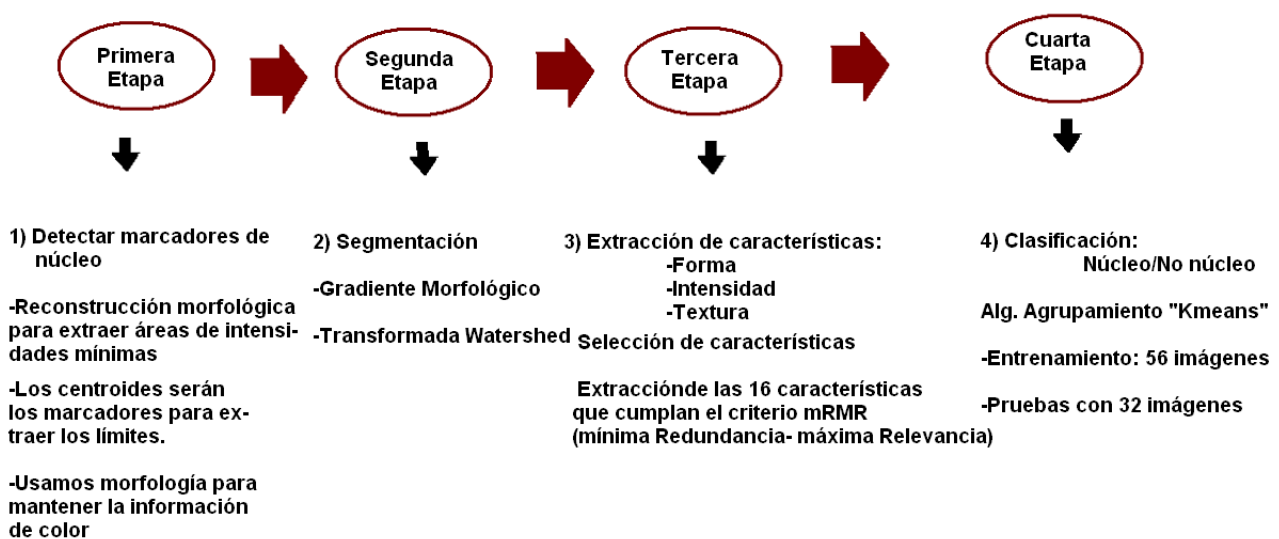


Figura 4.1: Diagrama de Flujo del algoritmo de extracción de características.

Este diagrama será el que sigamos durante todo este capítulo. En él encontramos cuatro partes claramente diferenciadas: detectar marcadores de núcleo, segmentación, extracción y selección de características y agrupación de características y clasificación de las regiones encontradas.

La primera de las partes la encontramos dividida a su vez en dos etapas: preprocesado y estimación de los centroides de los posibles núcleos. En el preprocesado se llevará a cabo una ecualización del histograma para mejorar el contraste, una umbralización y operaciones morfológicas para eliminar el ruido de la imagen. En cuanto a la estimación de centroides se aplican tres pa-

sos: transformación h-minima, reconstrucción morfológica y extracción de regiones de mínimos mediante el método *non regional maxima suppression*.

La segunda de las partes es la de segmentación, donde extraeremos las regiones y sus contornos. En esta parte se aplica el gradiente morfológico de color de la imagen y la transformada de *watershed*, para extraer regiones con características similares.

La tercera parte consiste en la extracción de características de las regiones. Primero, extraeremos las regiones; a partir de ellas obtenemos sus 57 características, que son de tres tipos: forma, textura e intensidad; y luego seleccionaremos las 16 características, entre las 57 posibles, que cumplan el criterio de mRMR (mínima redundancia-máxima relevancia).

La parte final realiza el agrupamiento de características, a través de *Kmeans* (algoritmo de agrupamiento); el cual tiene un entrenamiento con 56 imágenes y posteriormente, las pruebas del algoritmo las realizamos con las 32 imágenes restantes. Estas pruebas tienen como resultado una clasificación de las regiones extraídas en: núcleos u otras zonas ¹.

Una vez tenemos las regiones clasificadas comprobamos la efectividad del algoritmo; esta efectividad la analizamos con la ayuda de un médico que nos ha clasificado las regiones previamente en núcleos y no núcleos; estos resultados los comprobamos con los obtenidos por nuestro artículo y nos dará un porcentaje de acierto del algoritmo implementado para nuestro dataset, ya que como hemos indicado el tipo de imagen altera los resultados.

El porcentaje lo comparamos con otros métodos de estudio que ha realizado el trabajo implementado, estos son: GVF (*Gradient Vector Flow*) y ACM (*Active Contours Method*), ambos son métodos de segmentación, los cuales combinamos con los de los métodos de agrupamiento: SVM (*Support Vector Machine*) y *Spectral clustering*. Dichos métodos, estudiados por el artículo, son evaluados con 10.248 núcleos en 90 imágenes (dataset del artículo), dando claros resultados acerca de la efectividad de la detección y segmentación de los contornos de los núcleos.

A continuación, vemos detalladamente cada una de las partes del diagrama.

4.2. Detección de Marcadores de Núcleo

En esta primera parte del proceso realizamos un tratamiento previo de la imagen y estimamos los centroides candidatos, es decir, los marcadores de núcleo, tanto para células aisladas como para grupos de células.

El objetivo es definir las áreas de interés, es decir, las zonas de la imagen donde encontramos posibles núcleos. Necesitamos extraer como resultado de esta etapa una máscara binaria con dichas áreas resaltadas.

¹Las técnicas anteriormente mencionadas se explicarán detalladamente a continuación en el desarrollo del algoritmo.

Para lograr el objetivo de esta primera parte la dividimos en dos etapas: preprocesado y estimación de centroides.

4.2.1. Preprocesado

Esta primera etapa de la parte de detección de núcleos busca mejorar la imagen eliminando el ruido que contiene y resaltando las zonas de mínimos de intensidad.

El preprocesado consta de tres fases: ecualización (utiliza el método CLAHE), umbralización (realizada por el método Otsu) y operaciones morfológicas para la eliminación de ruido (operador de dilatación).

4.2.1.1. Ecualización

En esta fase mejoramos el contraste de la imagen en escala de grises resaltando los núcleos de las células con respecto al fondo de la imagen. Esto lo realizamos a través del método mencionado en el artículo, detallado en el trabajo [18], “CLAHE (*Contrast Limited Adaptive Histogram Equalization*)”. En el apéndice de “Herramientas de selección de características” se explica detalladamente esta técnica.

Este algoritmo reduce parte de la distorsión visual causado por el cambio de tamaño de una imagen.

Cuando aplicamos el algoritmo CLAHE a la imagen real que estamos mejorando, al ser una imagen RGB, lo aplicamos a cada una de sus tres bandas de color R, G y B, así conseguimos un mayor contraste de cada una de la tres componentes, provocando un mejor contraste final.

Una vez hemos ecualizado cada una de las componentes, volvemos a unir las encontrando, como esperábamos, una imagen muy contrastada (véase la figura 4.2 para observar el efecto sobre la imagen del dataset utilizada de ejemplo).

Una vez hemos mejorado el histograma de la imagen, ecualizándolo (obteniendo un mayor contraste); continuamos con la umbralización, fase intermedia del preprocesado, que pertenece a la primera parte de la etapa de detección de marcadores de núcleo, que buscará como salida una máscara binaria.

La máscara extraída, mediante la umbralización, contiene las regiones mínimas marcadas. Esto favorecerá para la extracción de los centroides posteriormente.

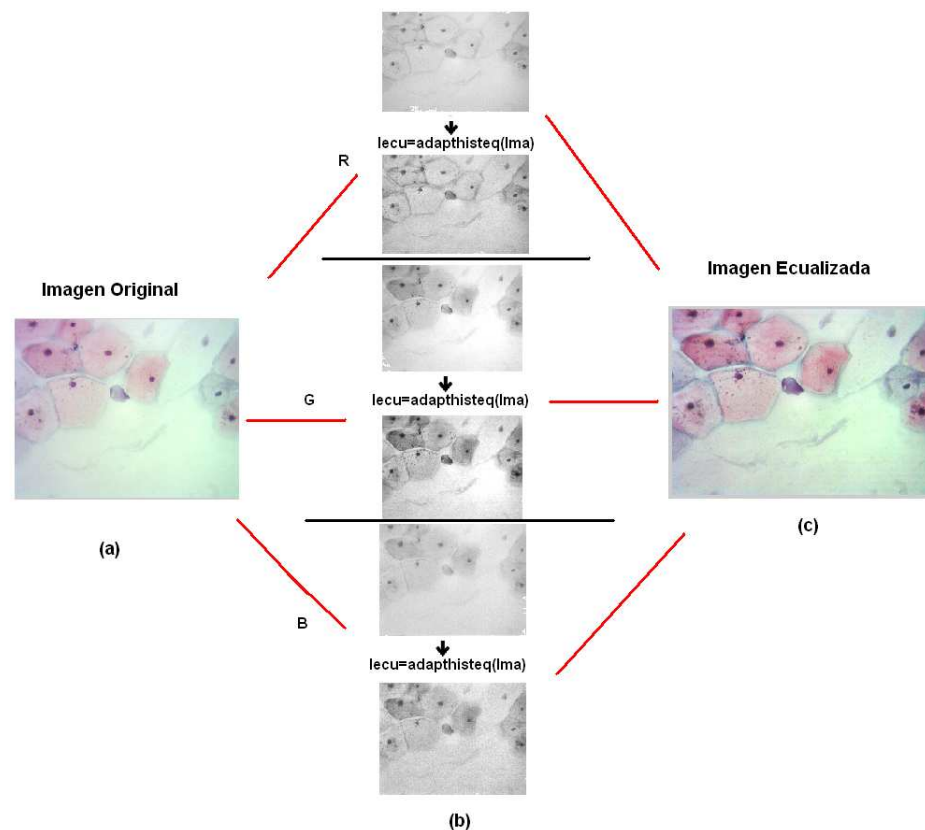


Figura 4.2: CLAHE aplicado a la imagen.

La figura expone tres fases: (a) Imagen originales, que se divide en las tres componentes de color (R,G,B), (b) a las cuales aplicaremos CLAHE por separado y (c) unificaremos las tres para ver el resultado.

4.2.1.2. Umbralización

La umbralización la llevamos a cabo mediante el método de Otsu, explicado en el artículo [19].

El **Método de Otsu**, es un algoritmo que obtiene un umbral para la segmentación de la imagen de manera automática. Dicho umbral se selecciona por un criterio discriminante, es decir, se selecciona el umbral a fin de maximizar la separabilidad de las clases resultantes en los niveles de gris, asumiendo que la imagen es bimodal. En el apéndice de “Herramientas de procesamiento de imágenes” se explica detalladamente esta técnica.

Este método lo aplicamos en nuestro algoritmo por separado a: R, G y B a través de las funciones de Matlab:

1. $levelR = graythresh(Rclahe)$. Nos da como resultado el umbral según el método otsu.
2. $BWR = im2bw(Rclahe,levelR)$. Nos da como resultado una imagen binaria, máscara, que contiene las áreas de la imagen que pertenecen a los posibles núcleos.

Al igual que en el caso de la ecualización, buscamos resaltar la zonas de mínimos en cada una de las componentes de color, como realiza el artículo aplica un umbral diferente a cada una de las aportaciones de color. Extrayendo tres máscaras. (véase la figura 4.3 para observar el resultado).

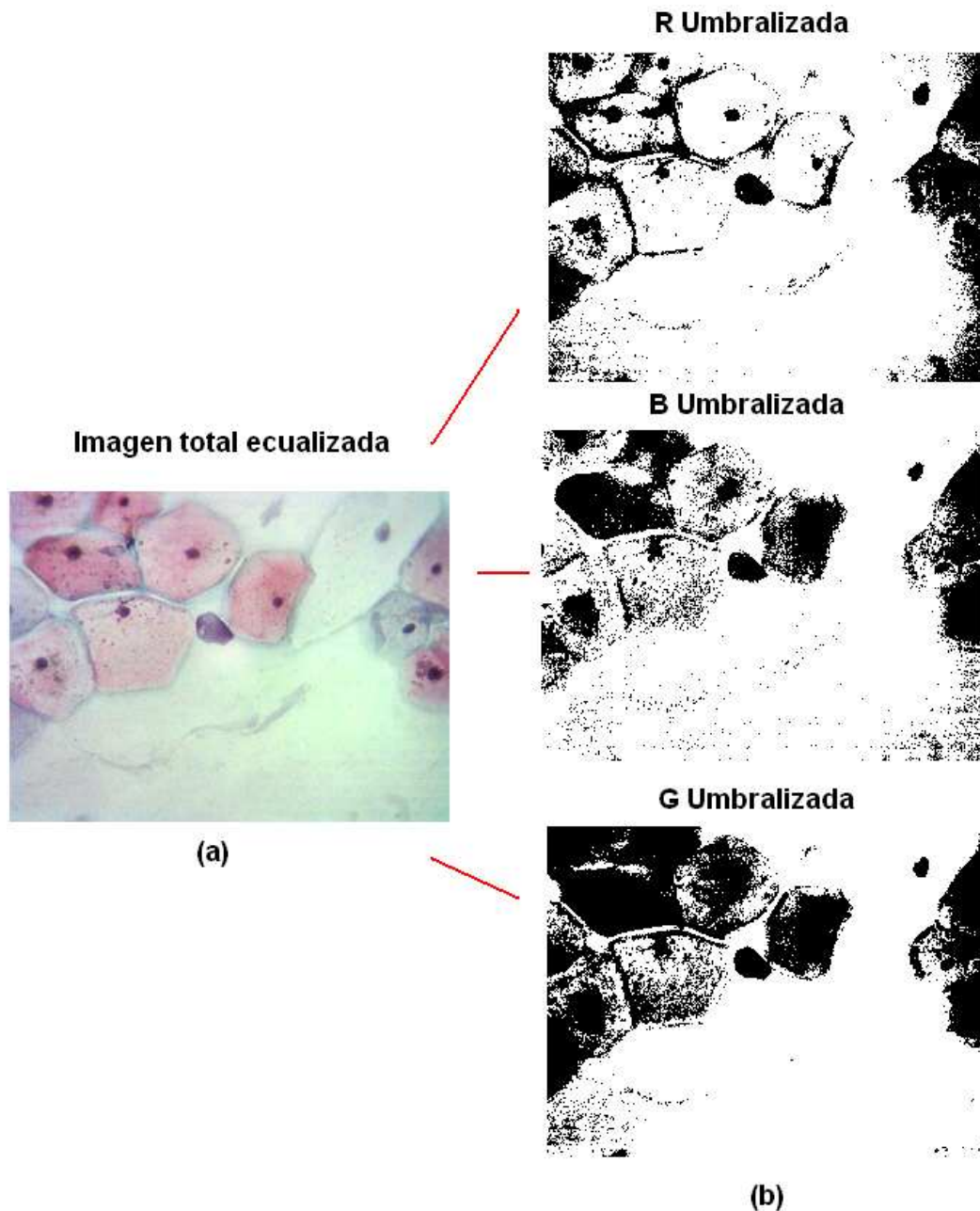


Figura 4.3: Umbralizada con Método Otsu.

La figura está compuesta por tres etapas: (a) imagen ecualizada a la cual aplicaremos (b) una umbralización a cada una de sus tres componentes por separado.

Una vez realizada la umbralización entramos en la parte final del preprocesado donde aplicaremos un operador morfológico de dilatación.

4.2.1.3. Dilatación.

Este operador lo utilizamos para eliminar las zonas de la imagen, que son pequeños objetos, consideradas ruido de la imagen.

Como en los casos anteriores de ecualización y umbralización se aplica por separado a cada componente (Véase las figuras 4.4, 4.5 para observar el resultado en nuestra imagen).

Aplicamos la operación morfológica de dilatación, que realiza una ampliación de los máximos, es decir, eliminación de los mínimos o ceros. En el apéndice de “Herramientas de procesamiento de imágenes” se explica detalladamente.

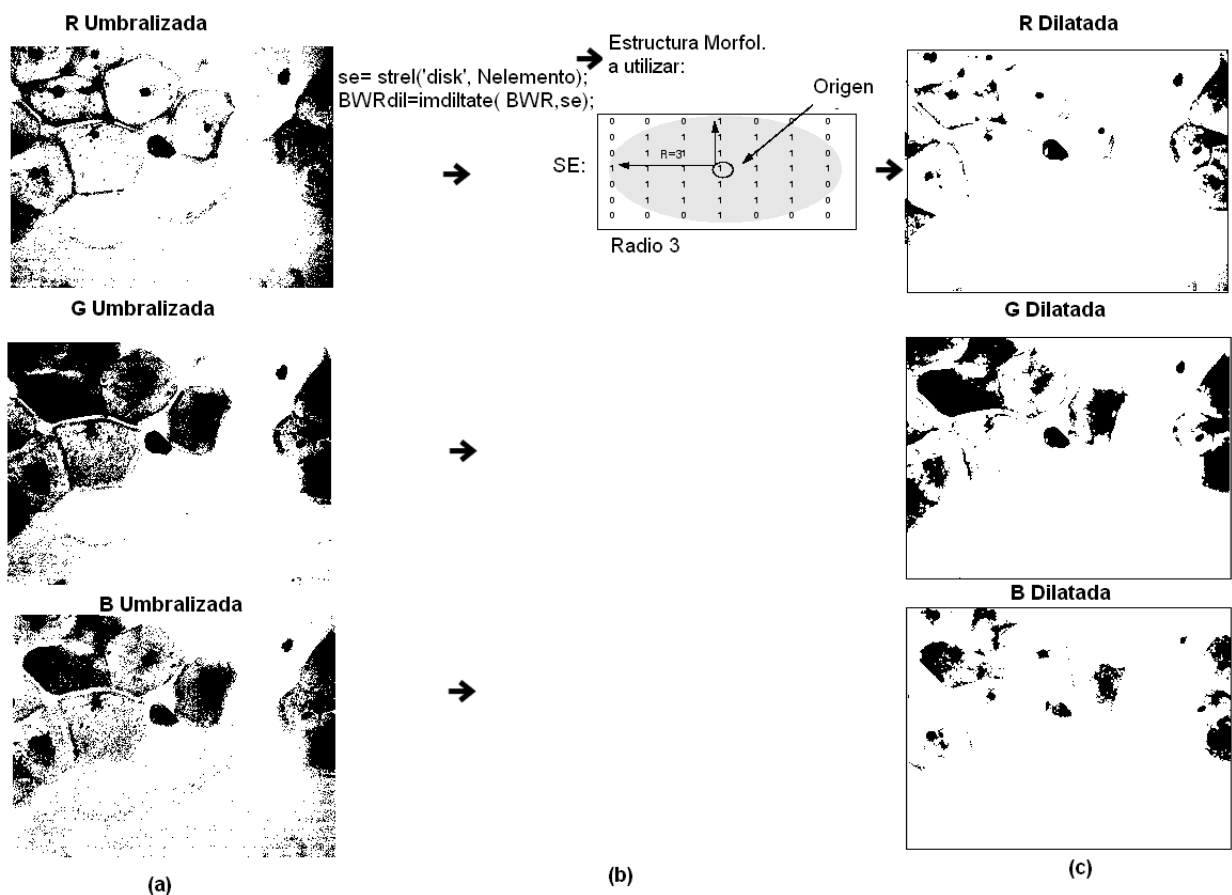


Figura 4.4: Operador morfológico de dilatación.

Muestra (a) las tres componentes umbralizadas por separado, (b) el elemento estructurante y (c) el resultado de aplicar la operación morfológica a cada una.

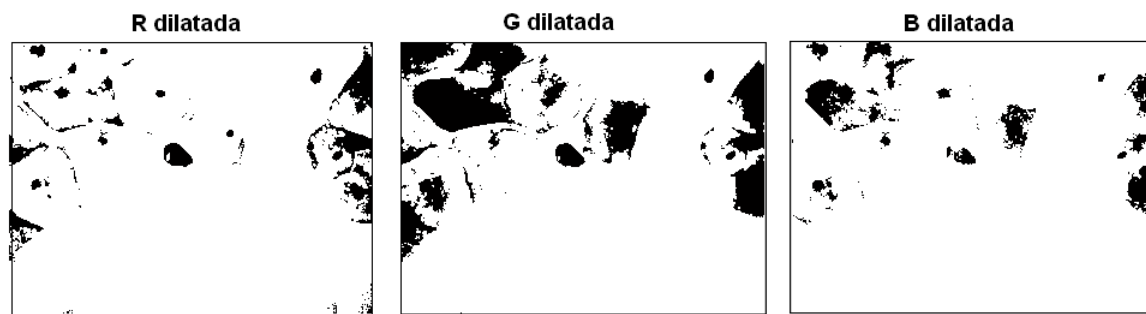


Figura 4.5: Unión de las tres componentes dilatadas.

Muestra las tres componentes tras haberlas aplicado el operador.

Tras realizar esta operación morfológica finalizamos el preprocesado, en el cual tenemos como resultado la máscara deseada; para continuar con la segunda etapa (estimación de los centroides candidatos) de la primera parte del algoritmo (Detección de marcadores de núcleo).

4.2.2. Estimación de los centroides de los núcleos candidatos

Después de la etapa de preprocesado, obtenemos una imagen con el fondo extraído dejando resaltados los objetos (imagen binaria resultante de aplicar el operador morfológico). De estos estimaremos sus centroides. Comenzaremos por definir a que hace referencia el centroide, este es la posición central de las regiones extraídas (región de mínimos) como posibles núcleos.

Más específicamente los pasos que llevaremos a cabo, para la estimación de los centroides, son los siguientes:

1. Transformación h-minima.4.2.2.1, tiene como objetivo seleccionar las zonas de mínimos y resaltarlas para utilizarla como máscara a la que queremos llegar en el siguiente paso.
2. Reconstrucción Morfológica [20].4.2.2.2, tiene como objetivo resaltar las zonas de mínimos, teniendo como referencia la máscara obtenida en el paso anterior.
3. Detección de regiones mínimas , a través del método “*Non Maxima Suppression*”[21].4.2.2.3. Una vez tenemos resaltadas las regiones de mínimos, podemos detectarlas con mayor facilidad.
4. Extracción de Centroides.4.2.2.4 Una vez detectadas las regiones mínimas, extraemos los centroides.

4.2.2.1. Transformación h-minima

Este método realiza una selección de las zonas mínimas de la imagen, para un determinado umbral (extraído de calcular el mínimo de la imagen). Esta técnica tiene como objetivo extraer

las zonas de mínimos y pronunciarlas para utilizarlas de referencia en el paso posterior. Aplicamos la función de Matlab “imhmin(Imagen, h)”, donde h es un escalar que nos marca el umbral (véase en la figura 4.6 un ejemplo de Mathworks²).

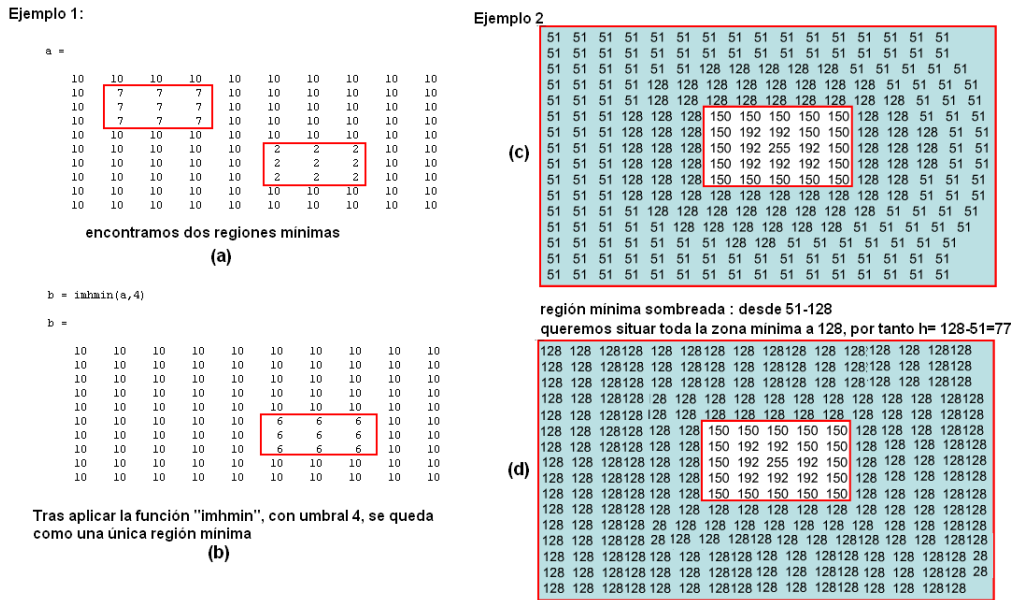


Figura 4.6: Función “Imhmin”.

En la figura encontramos dos ejemplos de dicha función; En el primero de los ejemplos (a) es la imagen original, donde está marcada las zonas de mínimos y (b) añade el valor introducido, obteniendo como resultado los mínimos totales. La distribución del segundo ejemplo está distribuida de la misma manera.

Más específicamente, en nuestra imagen, debemos aplicar dicho algoritmo a cada una de las componentes de color de la imagen original, como venimos haciendo, substrayendo a cada una de las componentes el valor que le corresponde (mínimo de cada componente) (véase la figura 4.7 para observar el efecto en nuestra imagen).

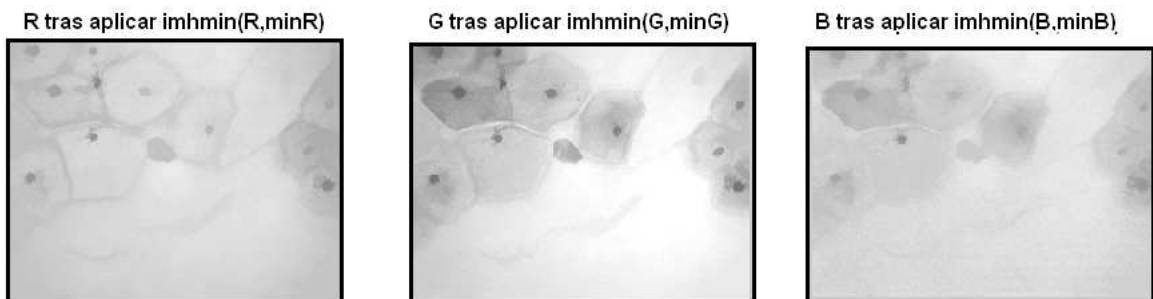


Figura 4.7: Aplicación de Transformada h-minima.

Muestra los resultados de la aplicación de la función “imhmin” a cada componente.

²<http://www.mathworks.es/help/toolbox/images/ref/imhmin.html>

4.2.2.2. Reconstrucción Morfológica

La reconstrucción morfológica, segundo paso de la etapa de extracción de centroides, tiene como objetivo resaltar las regiones de mínimos, teniendo como referencia la imagen obtenida en el paso anterior a la que deseamos llegar, para facilitar la detección de áreas de mínimos en el paso posterior.

La reconstrucción morfológica puede considerarse conceptualmente como dilataciones repetidas de la imagen original llamada marcador, hasta que el contorno de la imagen marcador encaja bajo una segunda imagen, llamada imagen máscara. En este caso, el marcador es un elemento estructurante circular de tamaño dos y la imagen máscara, la obtenida en el primer paso, la cual tiene resaltados las zonas de mínimos al haber restado en las mismas un valor uniforme (estos datos empleados son los que detalla el artículo que estamos replicando).

En la reconstrucción morfológica, los picos de la imagen de marcador se dilatan (Véase la figura 4.8 que ilustra este proceso en 1D). Cuando la dilatación siguiente deja de cambiar la imagen, el tratamiento se detiene. La dilatación final es la imagen reconstruida ³.

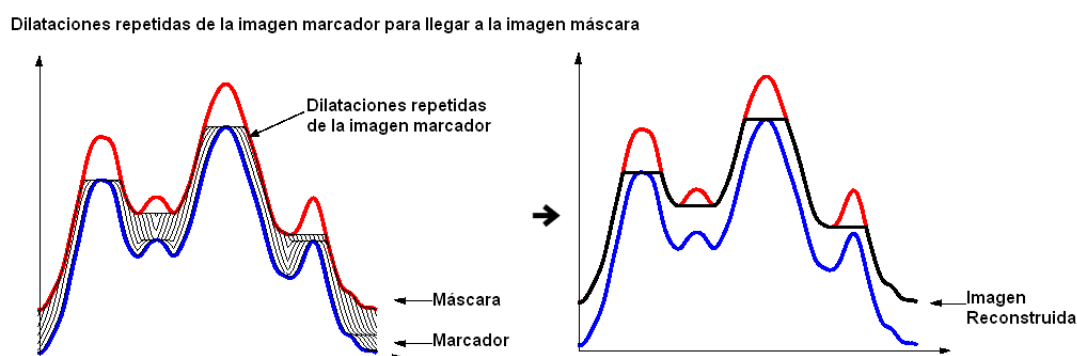


Figura 4.8: **Reconstrucción morfológica.**

La figura contiene dos gráficas para mostrar el proceso de la reconstrucción: la de la izquierda representa la imagen que tiene (azul) y a la que queremos llegar (roja), y la de la derecha contiene ambas curvas más la curva negra que representa el resultado final (Figura extraída de Mathworks).

La reconstrucción morfológica se basa en la dilatación morfológica (ahora la veremos con más detalle). Para esta, debemos tener en cuenta las siguientes características únicas:

- El procesamiento se basa en dos imágenes: un marcador y una máscara.
- Esta basado en el concepto de la conectividad, en lugar de un elemento estructural.
- Repite hasta que alcanza la estabilidad en el proceso, es decir, la imagen no cambia más.

³<http://www.mathworks.es/help/toolbox/images/f18-16264.html>

Así, aplicamos la función de Matlab “`imreconstruct (imagen,máscara)`” donde los elementos que la componen (como adelantamos) son:

- Imagen: Imagen original, a la que queremos resaltar las zonas de mínimos.
- Máscara: es la que hemos obtenido de aplicar la transformación de h-minima (esta es la extraída de la imagen original, tras pasar por el preprocesado y extraerle, como menciona en el artículo, una constante h que hemos elegido como el mínimo de cada componente, ya que no indican su valor) (véase la figura 4.9 para observar el efecto creado en nuestra imagen ejemplo).



Figura 4.9: **Reconstrucción Morfológica de la imagen.**

En la cual observamos, como presentan la zona de los núcleos de una manera más oscura y uniforme. Este es el camino para una buena extracción de las regiones núcleo.

4.2.2.3. Detección de regiones mínimas, a través del Método “Non Máxima Suppression”

Entramos en el tercer paso de la etapa de detección de centroides, en este paso estableceremos, a través del método mencionado, las regiones mínimas de la imagen, es decir, regiones de concentración de mínimos, donde todos los píxeles mínimos se establecerán a 1, mientras que el resto de la imagen lo hará a 0. También obtenemos el valor de máximos, que corresponde al fondo de la imagen.

Una vez tenemos este valor, cogemos el valor del píxel actual y se compara con este valor máximo. Si es menor se establece en blanco o 1, de lo contrario el valor se establece en 0 o negro (véase la figura 4.10 para observar un ejemplo clarificador).

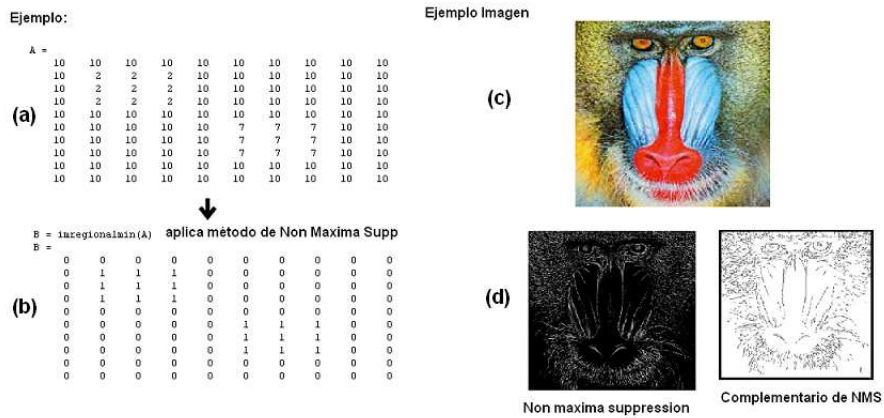


Figura 4.10: *Non Maxima Supresion*.

Dos ejemplos: primero, (a) imagen original y (b) aplica non maxima suppression a (a); segundo, (c) imagen real y (d) aplicar la función.

Aplicamos dicho método en nuestro algoritmo, a cada una de las tres componentes de color, y obtenemos el siguiente resultado:

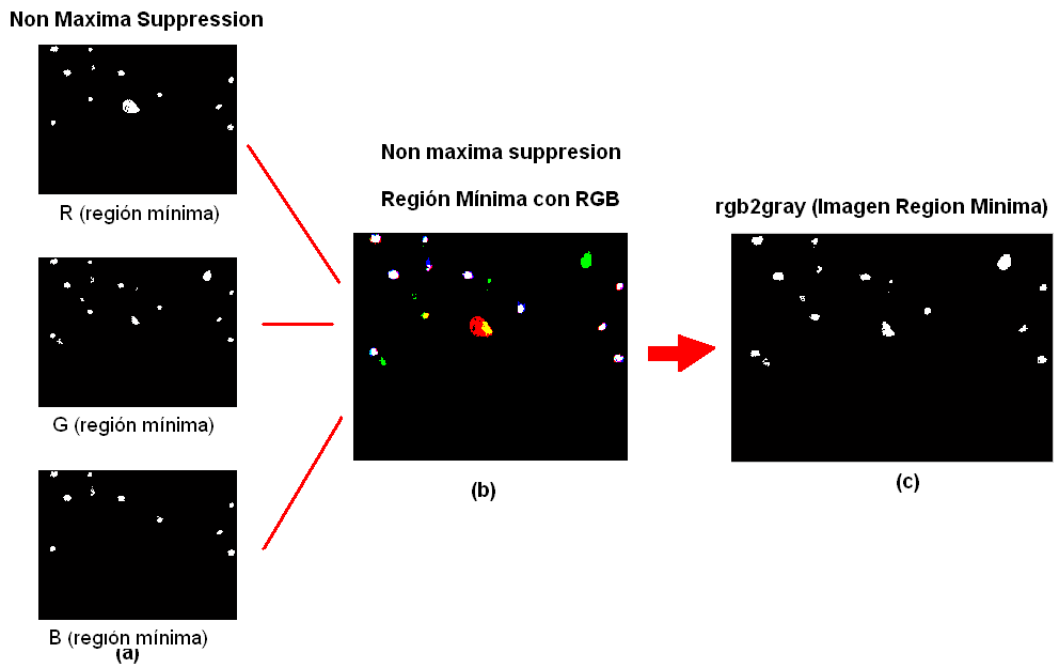


Figura 4.11: Método *Non Maxima Supresion* aplicada a la Imagen.

La figura contiene (a) las regiones mínimas extraídas en cada una de las tres componentes, (b) unificación de las tres componentes y (c) esta imagen que contiene las regiones puesta en escala de grises.

En este paso obtenemos las áreas mínimas de la imagen, de las cuales, situaremos sus centroides en el cuarto paso y último de la etapa de detección de centroides.

4.2.2.4. Extracción de Centroides

Esta es el paso final de la etapa de detección de centroides, finalizando con ella la primera parte del algoritmo: Detección de marcadores de núcleo.

En cada región de mínimos detectada, las coordenadas del centroide quedan definidas como:

$$r_c = (\bar{x}, \bar{y}) = \frac{1}{N} \sum_{i=1}^N (x_i, y_i)$$

donde N es el número de píxeles de contorno de la región mínima de la que estamos extrayendo sus centroides; x_i, y_i son las coordenadas del píxel i del contorno de la región mínima.

Aunque, en muchas ocasiones debido a la falta de homogeneidad y la concentración de colorante en la zona del núcleo, la detección de centroides múltiples dentro del área de un solo núcleo es posible. Para la determinación de un centroide único en cada núcleo debemos tener en cuenta la eliminación de dos (o más) centroides en una región de mínimos, serán eliminados los que estén en un radio menor al radio medio de un núcleo normal. Esto se logra con la aplicación de la regla dependiente de distancia, que se describe como: $\forall p = (x, y) \in R_c$

$$if\ exists\ q = \{(x_q, y_q) \mid D(p, q) \leq T\}$$

$$select\ r = \{p, q \mid \min \{I(p), I(q)\}\}$$

$$update\ R_c$$

$$until\ no\ change\ in\ R_c$$

En esta regla, R_c es el conjunto de todos los centroides, D es la distancia euclídea entre dos puntos, T es el umbral en el radio mínimo y I (t) es la intensidad de la imagen en el punto p.

Tras aplicar dicho método obtenemos el siguiente resultado (véase la figura 4.12).



Figura 4.12: Imagen original con el resultado de la extracción de Centroides.

Una vez hemos extraído los candidatos a centroides entramos en la siguiente parte del algoritmo: Segmentación.

4.3. Segmentación

Esta parte tiene como objetivo extraer las regiones de características similares, en este caso, las regiones de mínimos que hacen referencia a los posibles núcleos de la imagen.

Esta está compuesta a su vez por dos etapas: obtener el gradiente de la imagen, que usa la información de color de la imagen original o su escala de grises con el fin de mejorar los límites de los posibles núcleos; y la transformada de *watershed*.

A continuación, explicamos el gradiente de la imagen.

4.3.1. Gradiente de la imagen

Esta etapa es necesaria para aplicar el método de watershed, el cual requiere una imagen que contenga los límites de los núcleos pronunciados; esto se lo garantiza la aplicación del gradiente morfológico. La mayoría de los núcleos tienen generalmente:

1. Forma de elipse
2. Intensidad de los píxeles dentro de la zona núcleo inferiores a las que están fuera
3. Se espera alto gradiente de la imagen en los límites de los núcleos.

Así, existen dos tipos de gradiente que podemos aplicar: gradiente morfológico en color de la imagen y gradiente morfológico en escala de grises; de estos dos elegiremos el más adecuado para nuestro dataset. A continuación, los explicamos detalladamente.

4.3.1.1. Gradiente morfológico en color de la imagen

Hay variaciones amplias en la intensidad de los núcleos debido al procedimiento de tinción, dando un amplio intervalo de valores entre los que fluctúan el gradiente existente entre las fronteras núcleo /citoplasma. Por esto, el uso de un umbral después de la aplicación de los detectores de borde, en el caso del gradiente morfológico de color se aplica el método de Canny, produciría resultados ruidosos, porque los umbrales bajos darían lugar a la detección de bordes demasiados falsos, mientras que los valores altos daría lugar a la pérdida de algunos límites de núcleos verdaderos (véase la figura 4.13 para observar el efecto anterior).

Esto se debe a que el algoritmo de Canny, es muy sensible al efecto ruido, y esto crea que no se detecten bien los bordes en las imágenes donde está presente este efecto; por tanto, como

nuestras imágenes presentan mucho ruido no podemos aplicar este tipo de gradiente morfológico de color, detallado en [22] (véase el efecto que debería de obtenerse con un ejemplo de imagen con poco ruido en la figura 4.13).

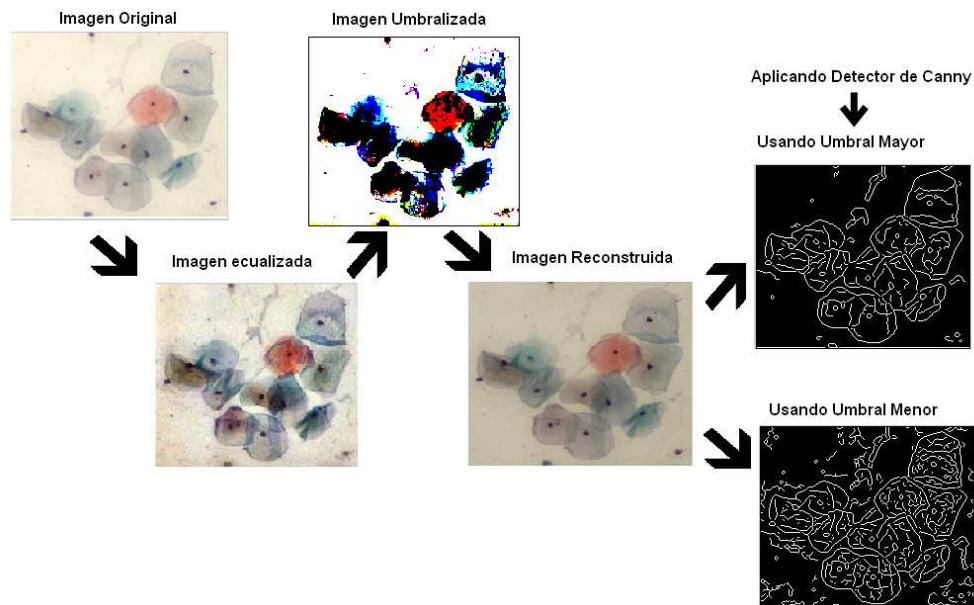


Figura 4.13: .Detector de Canny.

La figura contiene una imagen ejemplo con poco ruido del resultado de aplicar dicho método (con los pasos mostrados) en una imagen poco ruidosa.

Debido a que nuestras imágenes son muy ruidosas, pero precisamos del gradiente para aplicar *watershed*, no podemos utilizar el gradiente morfológico de color, por eso empleamos un gradiente morfológico en escala de grises que explicamos a continuación para continuar la parte de segmentación.

4.3.1.2. Gradiente Morfológico de una imagen en escala de grises

El gradiente morfológico tiene como objetivo resaltar los contornos. Este se puede obtener de diferentes maneras. En el apéndice de “Herramientas de procesamiento de imágenes” se explica detalladamente esta técnica.

De las diferentes maneras de obtener el gradiente elegimos la diferencia entre dilatación y erosión:

$$\nabla(f) = \delta_g(f) - \varepsilon_g(f)$$

Esta manera es la que marca el artículo que utilizamos. Véase la figura 4.14 para observar el efecto sobre nuestra imagen.

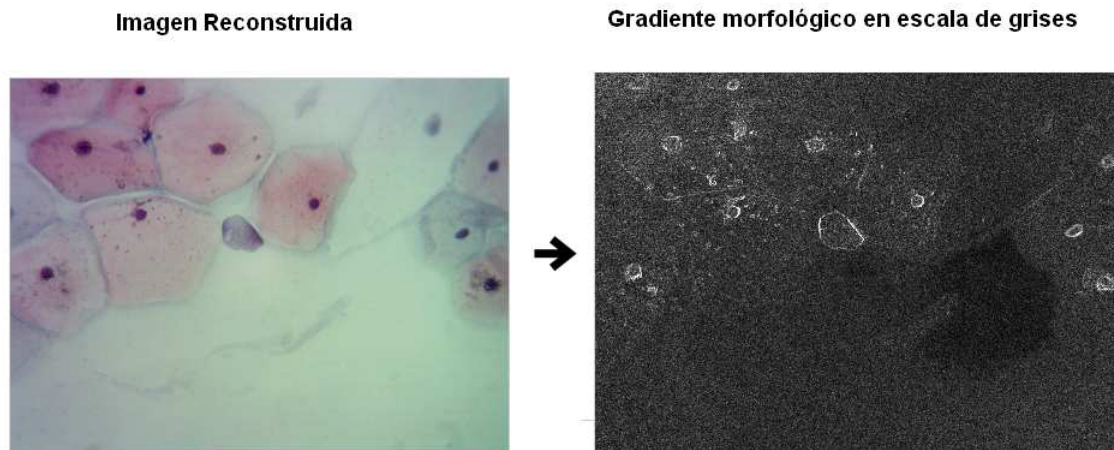


Figura 4.14: **Gradiente Morfológico en escala de grises.**

La figura muestra en la figura de la izquierda la imagen reconstruida que obtuvimos en el paso anterior y la de la derecha muestra el gradiente de dicha imagen.

Tras obtener el gradiente morfológico, que contiene los contornos de los núcleos de la imagen finalizamos la primera etapa de la parte de segmentación, y podemos aplicar “*Watershed*”.

4.3.2. *Transformada de Watershed*

Con esta transformada concluimos la parte de segmentación del algoritmo, artículo de referencia [6]. Dicha segmentación ha sido explicada detalladamente en la sección 2.1.2.3.

El resultado parcial del proceso en la parte de segmentación es el que obtenemos de la transformada de *watershed* (véase la figura 4.15 para observarlo).

Aplicándolo la función de Matlab:

`D= -bwdist(~Ycelbwimcomp);` Extraemos la matriz de distancias
`D(~Ycelbrimcomp)=-Inf;`
`W= watershed(D);`



Aplicándolo en la imagen en escala de grises:



Figura 4.15: *Watershed* .

La figura de la izquierda muestra el resultado de watershed y la de la derecha la imagen con los contornos extraídos, y superpuestos en la imagen en escala de grises, de la imagen obtenida a través de *watershed*.

4.4. Extracción y selección de características

Esta tercera parte del algoritmo está fundamentada porque aunque el resultado de la transformación de *watershed*, por lo general, se traduce en la correcta identificación de las posiciones de los núcleos de la imagen. Sin embargo, algunas áreas de falsos positivos también son detectados, debido a la existencia de un mínimo local. Esto es una consecuencia de la detección de los marcadores de núcleos que produce algunos centroides de mínimos regionales que no indican la existencia de núcleos. Por lo tanto, la eliminación de estas áreas es necesaria. Para ello utilizamos esta parte y la última para la separación de las áreas detectadas en dos clases: la clase núcleos verdaderos y el resto de los hallazgos.

Esta parte constará de dos etapas: extracción de características y selección de características; la primera de ellas está basada en la extracción de 52 características divididas en tres tipos: forma, intensidad y textura de cada una de las regiones; y la segunda de ellas será una selección de las 16 características de las 52 posibles que cumplan el criterio de mRMR (mínima redundancia - máxima relevancia).

4.4.1. Extracción de características

Situándonos al comienzo de la tercera parte del algoritmo debemos tener claro que el objetivo de la misma es obtener las características de las regiones que hemos obtenido en la parte de segmentación. Estas características son necesarias porque clarificarán la división entre núcleos y otros hallazgos.

Habiendo encontrado las áreas de los núcleos encerradas por los límites detectados, las características relativas a la forma, textura y la intensidad de las regiones detectadas se pueden determinar fácilmente. Sin embargo, la restricción del cálculo de estas características sólo para el área encerrada por los límites detectados no es suficiente, porque las regiones de mínimos locales que no corresponden a los núcleos de verdad también puede tener características similares.

En este momento es conveniente aprovechar que los núcleos son zonas más oscuras que el citoplasma y que las regiones de núcleos detectados presentan diferencias significativas respecto de su vecindad. Además, las regiones detectadas que no pertenecen a los núcleos se detectaron probablemente debido a la existencia de poca profundidad en sus niveles de intensidad de la zona del citoplasma en la imagen, siendo más propensos a presentar valores similares (véase la figura 4.16 para observarlo).

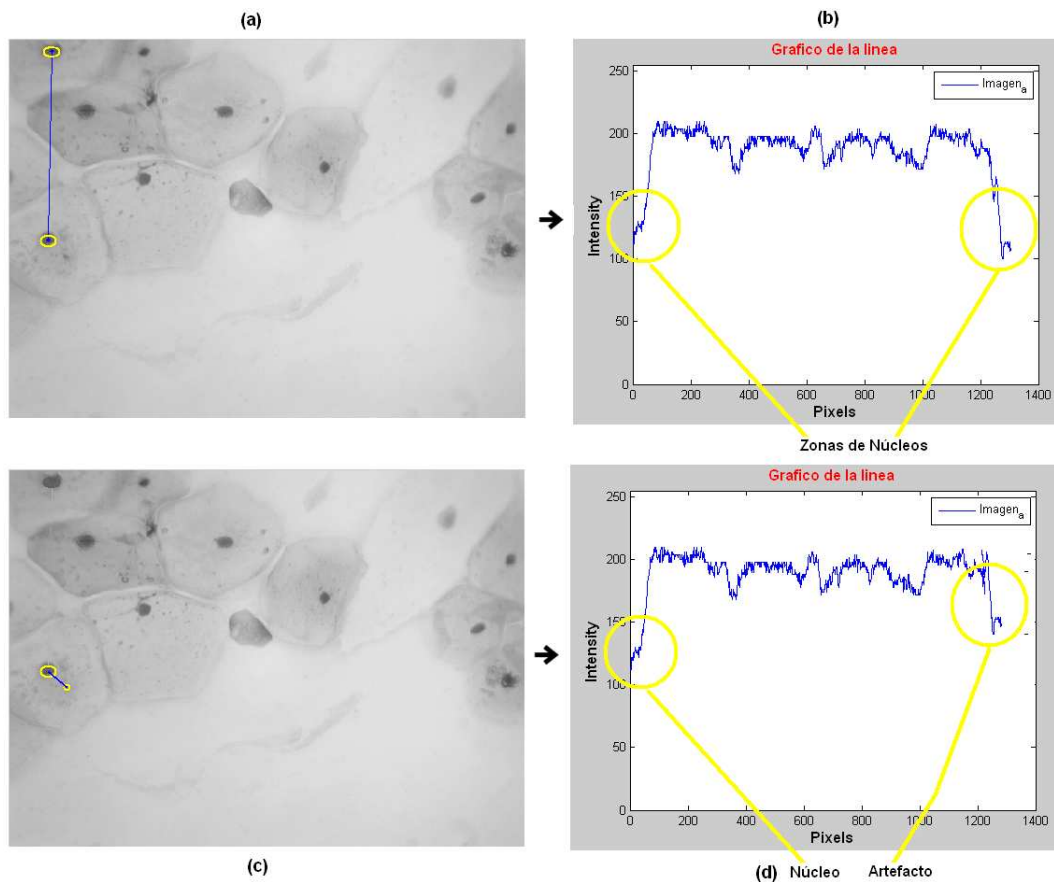


Figura 4.16: **Relación de las intensidades de los núcleos verdaderos con otras zonas.**

La imagen (a) muestra la unión de dos núcleos: mostrará el valor de todos los píxeles del camino, en (b) una gráfica que representa la intensidad de los píxeles que va recorriendo, (c) muestra la unión de un núcleo con una zona que no lo es y (d) muestra la gráfica del valor de los píxeles de la línea de unión. Observamos que la diferencia de intensidad entre un núcleo y otras zonas es mínimo.

Por lo explicado en el párrafo anterior, se propone el cálculo de las características de la vecindad de las áreas detectadas y de la caja que contiene la región completa dentro; más específicamente, para cada área detectada “A”, se calcula como el rectángulo máximo que contiene la región detectado “B”, y el vecindario “Ngh” se determina como el complemento A^c en B, que es $Ngh = A^c \cap B$ (véase la figura 4.17 para observar las distintas regiones).

Las distintas áreas que debemos tener en cuenta las extraemos a partir de la función: “bwconn-comp” 3.3.1.2 que vimos en el capítulo anterior. Ya que una vez que tenemos los resultados del algoritmo de watershed, tenemos las distintas áreas que son considerados posibles candidatos a núcleos, es decir, las áreas A.

Una vez tenemos dichas áreas A, a través de la función “regionsprop” aplicada a dichas zonas, extraemos de la propiedad de dicha función “Bounding box”: las coordenadas de la esquina superior izquierda del rectángulo y las medidas de su largo y de su ancho; pudiendo así dibujar la caja que contiene el área A, y así podemos extraer la zona B, es decir, la región mas su vecindad.

Y, en ultimo lugar, teniendo dichas zonas A (región) y B (región + vecindad) podemos extraer la vecindad: Ngh, definida como vinos anteriormente.

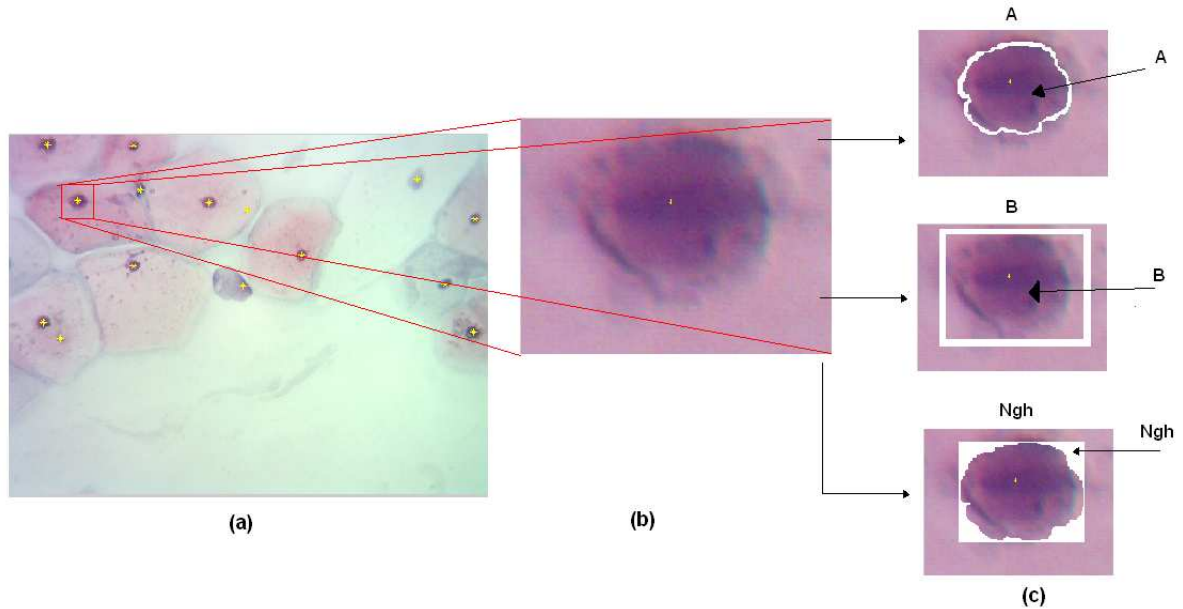


Figura 4.17: **A, B y Ngh.**

(a) muestra la imagen con centroides (b) amplía una de sus zonas para enseñar en (c) las zonas de las que vamos a extraer las características. Las zonas son: A, B y Ngh.

Para extraer dicha área Ngh, hemos implementado los siguientes pasos, ya que en el artículo no especificaban como obtenerla (véase la figura 4.18). Estos pasos son:

1. Tener una imagen binaria compuesta toda de unos con las dimensiones de la imagen total.
2. Aplicamos el operador morfológico de dilatación a la imagen binaria que tiene marcadas a 1 las regiones extraídas. Por la dilatación creamos un píxel más a todo el contorno de la imagen.
3. Recorremos las imágenes binarias de la zona B y la imagen binaria de la zona A dilatada. Ambas tienen a 1 las zonas deseadas.

a) Si $B(i,j)=A(i,j) \rightarrow Ngh(i,j)=0$

b) En caso contrario $Ngh(i,j)=1$

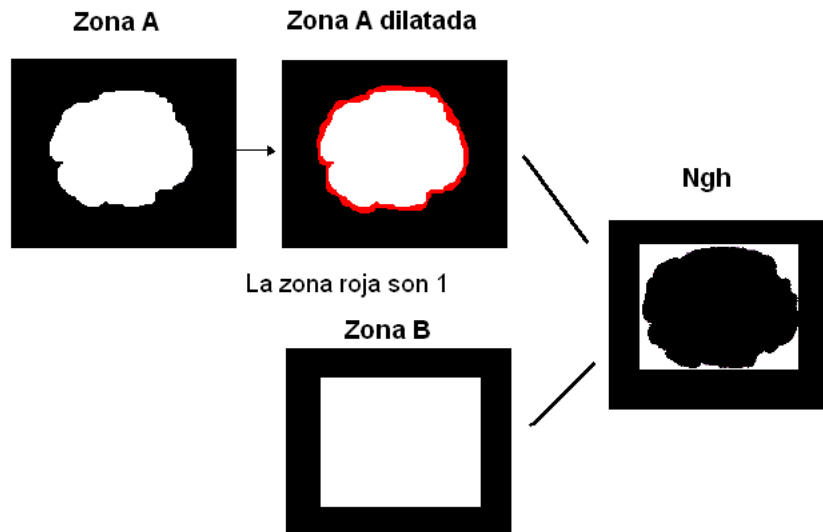


Figura 4.18: **Extracción de área Ngh.**

El resultado aplicado lo podemos ver en la figura 4.17.

Continuando con la parte de extracción y selección de características tenemos las tres zonas extraídas para la construcción del conjunto de características. Este conjunto de características se divide en tres grupos:

1. Características de forma
2. Textura
3. Intensidad de la disparidad.

La extracción de los tres grupos de características las detallamos a continuación.

4.4.1.1. Características de Forma

En primer lugar extraemos las características de forma, donde habitualmente las regiones detectadas que son núcleos presentan forma de elipse. Pero no en todos los casos es así, por este motivo se eligen seis características de la forma de la región extraída. Estas son: la circularidad, la excentricidad, la mayor y menor longitud de los ejes principales, el diámetro equivalente de un círculo con la misma área de la región, y el perímetro de la zona detectada (véase la figura 4.19 para observar dicho tipo).

Shape features.

Minor axis length ^a	$K = \sqrt{\frac{2(u_{20}+u_{02}-A)}{u_{11}}}$
Major axis length ^a	$L = \sqrt{\frac{2(u_{20}+u_{02}+A)}{u_{11}}}$
Eccentricity	$E = 2\sqrt{\frac{(\frac{L}{2})^2 - (\frac{K}{2})^2}{L}}$
Equivalent diameter	$ED = \frac{4 \times Area}{\pi}$
Perimeter	$P = \text{number of boundary points}$
Circularity	$C = \frac{4\pi \times Area}{P^2}$

Figura 4.19: **Características de forma** . Figura extraída de [2]

Dichas características las extraemos mediante la función que utilizamos en *Grey Level Method* para extraer el tamaño de las áreas y circularidad, esta es “bwconncomp” 3.3.1.2, introduciendo las imágenes binarias A, B o Ngh ,que contienen las regiones detectadas, para extraer sus características. Extraemos dichos datos en el *script* de Matlab y las almacenamos en un vector que también añadirá las características de textura y de intensidad.

4.4.1.2. Características de Textura

El análisis de la textura de las regiones detectadas se basa en las propiedades estadísticas del histograma de intensidad en las tres componentes de color: R, G y B, y el cálculo de algunos descriptores de textura como los patrones binarios locales (LBP) [7] [3]. En el apéndice de “Herramientas de procesamiento de imágenes” se explica detalladamente esta técnica.

La aplicación de patrones binarios locales es debido a su invariabilidad en escala de grises. Dichos patrones tienen en cuenta la vecindad de nuestros píxeles de las regiones, teniendo en cuenta la textura, dejándola uniforme.

Por otra parte, LBP lo aplicamos al histograma, este es normalizado, uniforme e invariante a la rotación, calculado para la región que contiene el cuadro delimitador (zona B) de las regiones segmentadas. LBP se ha calculado con dos topologías de vecinos diferentes: un círculo de radio unidad y una hipérbola con el eje mayor y menor de longitudes iguales a uno (véase la figura 4.20). En ambas topologías, el número de píxeles equidistantes fue de $P = 8$. La media y la desviación estándar de cada histograma se utilizaron como características.

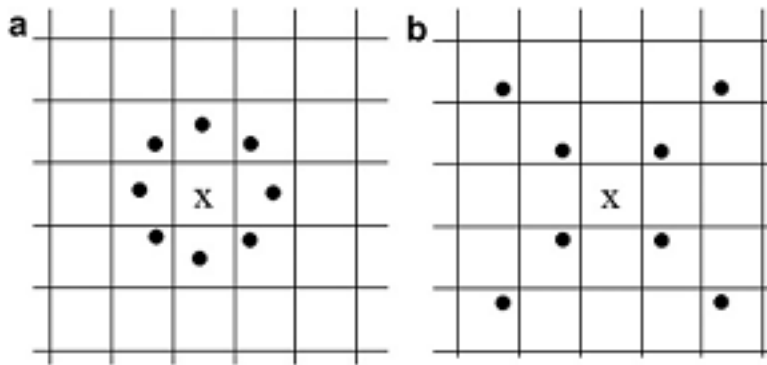


Figura 4.20: Topología de los vecinos: a)Circular b)Hipérbola. Figura extraída de[7][3]

Ambas topologías son elegidas porque lo detalla el artículo que estamos siguiendo y además este se apoya en los resultados obtenidos por el trabajo [7] que muestran que son las más óptimas para imágenes binarias o en escala de grises, como podemos ver si observamos el apéndice de “Herramientas de procesamiento de imágenes”.

Así, para todas las regiones segmentadas (A,B y Ngh) hemos calculado el tercer momento, la uniformidad, la entropía y la suavidad del histograma de intensidad.

Todas las características de textura se calculan para los tres canales de color y se resumen en la figura 4.21.

Table 2
Texture features.

Third moment ^a	$\mu_3 = \sum_{i=0}^{L-1} (z_i - m)^3 p(z_i)$
Uniformity	$U = \sum_{i=0}^{L-1} p^2(z_i)$
Entropy	$e = -\sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i)$
Smoothness	$R = 1 - \frac{1}{1+s^2}$, where $s = \sqrt{\sum_{i=0}^{L-1} (z_i - m)^2 p(z_i)}$
Mean histogram LBP_{circle}^{riu2}	See Appendix A
Std histogram LBP_{circle}^{riu2}	See Appendix A
Mean histogram $LBP_{hyperbola}^{riu2}$	See Appendix A
Std histogram $LBP_{hyperbola}^{riu2}$	See Appendix A

^aAppendix A: ver Apéndice A de [Plissiti2011838] y los artículos : [Nanni2010117, 1017623]

Figura 4.21: Tabla de características de texturas. Figura extraída de [2].

Una vez tenemos los patrones binarios extraeremos las características de los núcleos como vimos

en la figura 4.21. Estas características las almacenamos en el vector de características anterior (que contiene las características de forma). Y continuamos con la extracción de características de intensidad.

4.4.1.3. Características de Intensidad

Las características de intensidad son las últimas que extraemos en esta tercera parte del algoritmo.

Lo que caracteriza a la intensidad de cada región es el promedio del valor de intensidad de todos los píxeles de la región. Sin embargo, la intensidad media de los núcleos varía en un amplio intervalo y puede coincidir con las regiones de la superposición de células en la imagen.

Una característica de la intensidad equivalente que manifiesta la desigualdad entre la región detectada y sus alrededores es la diferencia de la intensidad media entre las regiones que muestra la figura 4.22. Esperamos valores altos de esta función cuando se refiere a las regiones núcleos, ya que el área de los núcleos es más oscuro que el citoplasma circundante. Los tres valores que obtenemos de las características de intensidad se calculan de forma independiente para el componente rojo, verde y azul de la imagen original.

Se realiza el estudio en cada una de las tres componentes debido a que las aportaciones de ellas son muy diferentes y aportan alto grado de información.

Intensity disparity features.

Foreground-background contrast in red ^a	$dR = m_{RED}^{Ngh} - m_{RED}^A$
Foreground-background contrast green ^a	$dG = m_{GREEN}^{Ngh} - m_{GREEN}^A$
Foreground-background contrast in blue ^a	$dB = m_{BLUE}^{Ngh} - m_{BLUE}^A$

Figura 4.22: **Tabla de características de intensidad.** Figura extraída de [2]
Muestra m_{color}^{region} valor medio de intensidad en un región color especificados.

Una vez tenemos las tres características de intensidad las añadimos al vector que tiene almacenadas las características de forma y textura. Dando como resultado un vector que contiene 57 características por cada región detectada (véase la tabla ??).

Num	Características	Num	Características
1	Minor axis length	30	Third moment "Ngh en Blue"
2	Major axis length	31	Uniformity "Ngh en Blue"
3	Eccentricity	32	Entropy "Ngh en Blue"
4	Equivalent diameter	33	Smoothness"Ngh en Blue"
5	Perimeter	34	Third moment "B en Red"
6	Circularity	35	Uniformity "B en Red"
7	Foreground-background contrast in red	36	Entropy "B en Red"
8	Foreground-background contrast in green	37	Smoothness"B en Red"
9	Foreground-background contrast in blue	38	Mean Hist LBP_{circle}^{riu2} "B en Red"
10	Third moment "A en Red"	39	Std Hist LBP_{circle}^{riu2} "B en Red"
11	Uniformity "A en Red"	40	Mean Hist $LBP_{hyperbola}^{riu2}$ "B en Red"
12	Entropy "A en Red"	41	Std Hist $LBP_{hyperbola}^{riu2}$ "B en Red"
13	Smoothness"A en Red"	42	Third moment "B en Green"
14	Third moment "A en Green"	43	Uniformity "B en Green"
15	Uniformity "A en Green"	44	Entropy "B en Green"
16	Entropy "A en Green"	45	Smoothness"B en Green"
17	Smoothness"A en Green"	46	Mean Hist LBP_{circle}^{riu2} "B en Green"
18	Third moment "A en Blue"	47	Std Hist LBP_{circle}^{riu2} "B en Green"
19	Uniformity "A en Blue"	48	Mean Hist $LBP_{hyperbola}^{riu2}$ "B en Green"
20	Entropy "A en Blue"	49	Std Hist $LBP_{hyperbola}^{riu2}$ "B en Green"
21	Smoothness"A en Blue"	50	Third moment "B en Blue"
22	Third moment "Ngh en Red"	51	Uniformity "B en Blue"
23	Uniformity "Ngh en Red"	52	Entropy "B en Blue"
24	Entropy "Ngh en Red"	53	Smoothness"B en Blue"
25	Smoothness"Ngh en Red"	54	Mean Hist LBP_{circle}^{riu2} "B en Blue"
26	Third moment "Ngh en Green"	55	Std Hist LBP_{circle}^{riu2} "B en Blue"
27	Uniformity "Ngh en Green"	56	Mean Hist $LBP_{hyperbola}^{riu2}$ "B en Blue"
28	Entropy "Ngh en Green"	57	Std Hist $LBP_{hyperbola}^{riu2}$ "B en Blue"
29	Smoothness"Ngh en Green"		

Cuadro 4.1: 57 Características seleccionadas.

4.4.2. Selección de características

Una vez tenemos las 57 características extraídas entramos en la etapa final de la tercera parte: Selección de características. Esta etapa tiene como objetivo seleccionar las 16 características que cumplan el criterio de mínima redundancia-máxima relevancia. En el apéndice de "Herramientas de selección de características" se explica detalladamente esta técnica.

Para obtener esta selección tenemos que contar con todas las regiones extraídas totales de todo nuestro dataset. Este consta de 88 imágenes, en las cuales encontramos un total de 857 núcleos reales que han sido señalados por un experto, frente a los detectados por el algoritmo que son 2.361, ya que como vimos en la introducción del artículo pueden cogerse núcleos reales o por el

contrario son otras zonas de la imagen que presentan una intensidad baja, confundándose con los núcleos.

De los núcleos seleccionados por el experto, como núcleos reales, almacenamos las coordenadas de sus centroides, para después poder clasificar cual es núcleo real y cual es otra zona de la imagen.

Así, nuestro algoritmo extrae para todas las zonas detectadas como posibles núcleos una matriz de 2361 regiones x 57 características/región.

Como ya tenemos la matriz de características completa podemos seleccionar las características. Para ello emplearemos el criterio de “Mínima Redundancia-Máxima Relevancia “mRMR”. En el apéndice de “Herramientas de selección de características” se explica detalladamente esta técnica.

Con dicho método obtendremos las 16 características seleccionadas que son óptimas para identificar que consideramos núcleo y que consideramos otras zonas de la imagen. El número de características seleccionadas es el dado por el trabajo que estamos replicando.

Dicho algoritmo está implementado en una función de Matlab “mrmr_miq_d”, en la cual introduciremos los 2361 regiones extraídas de las imágenes con las 57 características de cada una de ellas, y nos dará las 16 características más discriminantes. En este caso son:

Orden	Número de Característica	Característica
1	23	Uniformity “Ngh en Red”
2	27	Uniformity “Ngh en Green”
3	31	Uniformity “Ngh en Blue”
4	51	Uniformity “B en Blue”
5	35	Uniformity “B en Red”
6	43	Smoothness”B en Green”
7	45	Smoothness”B en Green”
8	53	Smoothness”B en Blue”
9	37	Smoothness”B en Red”
10	36	Entropy “B en Red”
11	44	Entropy “B en Green”
12	52	Entropy “B en Blue”
13	56	Third moment “B en Red”
14	34	Third moment “B en Red”
15	38	Mean Histogram LBP_{circle}^{riu2} ”B en Red”
16	39	Std Histogram LBP_{circle}^{riu2} ”B en Red”

Cuadro 4.2: 16 características más discriminantes extraídas con mRMR

4.5. Agrupamiento de características y clasificación de regiones

Entramos en la última parte del algoritmo, la cual tiene como objetivo extraer una clasificación de las regiones extraídas en núcleo/no núcleo. Para realizar dicha clasificación aplicamos un

algoritmo de agrupamiento: *Kmeans*.

Este algoritmo tiene dos etapas: la primera de ellas es una etapa de entrenamiento, donde como ya explicamos utilizamos las 56 imágenes primeras, y la segunda de ellas es la clasificación de las regiones de las 32 imágenes restantes mediante distancias.

4.5.1. Agrupamiento de características

En nuestro trabajo, empleamos el algoritmo de agrupamiento *Kmeans*. Debemos tener en cuenta que precisa de entrenamiento y este ha de realizarse con las 16 características de las regiones que verdaderamente son núcleos. Por este motivo al artículo le añadimos una columna final a la matriz que almacena las 16 características. Esta columna añadida clasificará cada región como núcleo (1) / no núcleo (0). Esta clasificación la hemos realizado con la matriz que contiene las coordenadas de los núcleos verdaderos de cada imagen.

En una imagen se extrae una región y comprobamos sus coordenadas en la matriz. Comparamos con la matriz de coordenadas de los núcleos verdaderos de esa imagen y si coinciden con alguna de las coordenadas lo clasificamos como núcleo. Esta cuenta con un margen de error de 50 píxeles en cada una de las coordenadas tanto en positivo como en negativo (valor elegido por nosotros).

Esto lo necesitamos para el entrenamiento, pero también nos facilitará sacar el porcentaje final de acierto de este algoritmo replicado.

Para la aplicación del algoritmo de agrupamiento "*Kmeans*" explicado detalladamente en la sección 2.1.2.6 construiremos un conjunto de datos de entrenamiento, extraído de nuestro dataset.

En nuestros experimentos debemos introducir los núcleos verdaderos; elegimos para entrenar el 70 % de los núcleos correctos, de los cuales cogemos sus 16 características más relevantes extraídas de mRMR e introducimos el número de centroides (número de grupos) que queremos que cree esta función. Una vez tenemos entrenado el algoritmo, aplicando la función "*kmeans*" de Matlab, nos devuelve dichos centroides.

El número de centroides ha sido elegido con la experiencia, en el apartado de resultados especificamos el número de centroides elegido.

Especificación de parámetros de Kmeans Las variables asociadas para nuestro caso específicamente son:

1. Entrada: *matriz_entrenamiento* (contiene las 16 características del 70% de los núcleos correctos) y *k* (número de centroides que queremos).
2. Salida: *idx* (centroide a la que pertenece) y *C* (coordenadas de cada centroide para ser situado en el plano).

4.5.2. Clasificación de regiones

Una vez tenemos entrenado el algoritmo de agrupamiento entramos en la parte final del algoritmo, donde clasificamos las regiones de las 32 imágenes de prueba que tenemos para evaluar.

Con dichas imágenes de prueba, y teniendo la matriz de centroides extraída de *kmeans*, miramos cada uno de los posibles núcleos de las 32 imágenes finales. Cogemos cada zona y miramos si cumple la condición de distancia mínima con los grupos incluidos en la matriz de centroides, si es así se considera núcleo y si supera la distancia se considera otra zona.

Esta distancia mínima ha sido establecida por la experiencia debido a que no estaba detallada en el artículo replicado, esta distancia estaba de 0 a 1, por tanto si estaba por debajo de 0,3 se consideraba núcleo y en caso contrario no.

Una vez tenemos las zonas clasificadas por distancias, de cuales son considerados núcleos y cuales no, comparamos con los datos que tenemos del dataset del experto, de cuales zonas él considera núcleos, y cuales considera otras zonas, y podemos extraer un porcentaje de acierto de nuestro algoritmo.

4.6. Resultados finales del algoritmo

Los resultados finales del algoritmo lo mostramos en la imagen, donde veremos clasificados los centroides de las regiones extraídas en: amarillo (núcleo verdadero) o azul (núcleo erróneo)(véase la figura 4.23).

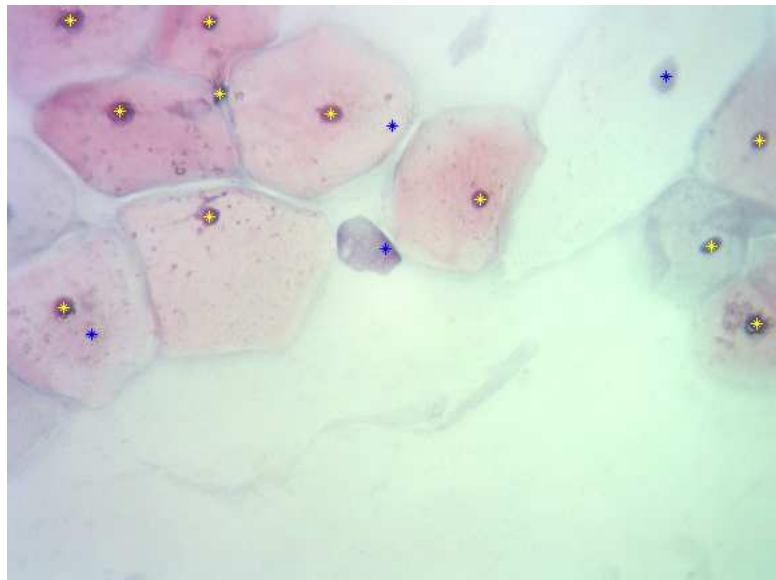


Figura 4.23: **Detección de Centroides erróneos de las regiones mínimas.**

Los centroides que están en amarillo son núcleos buenos y los que encontramos en azul son centroides de otras áreas que no son núcleos correctos, pero que se habían extraído como núcleos.

También, hemos extraído del algoritmo (añadido por nosotros) la tabla 4.3, tras aplicar la información que nos facilita el experto de las coordenadas de los núcleos reales en cada imagen. Con ellos, tras realizar la parte de segmentación que extrae las regiones, podemos extraer los resultados de la tabla.

	% Núcleos extraídos de más	Media de objetos extraídos de más
Imagen	82.8855 %	17,8 numobj/imagen

Cuadro 4.3: **Porcentajes Medios de regiones extraídos como núcleos de más en la imagen**

En la tabla podemos ver que tenemos una media alta de objetos considerados por el experto como zonas que no son núcleos reales, debemos aplicar el algoritmo explicado para poder extraer las zonas de núcleos correctos de las imágenes. Fundamentando así que el objetivo de este algoritmo sea extraer las zonas que son núcleos reales y cuales son otro tipo de zona.

Además, como hemos visto, durante todo el trabajo, tenemos como principal objetivo obtener las características que cumplen el criterio de mRMR (mínima redundancia-máxima relevancia). Estas son las que vimos en la tabla 4.2.

Una vez tenemos estas características, que pertenecen a los núcleos verdaderos de las primeras 56 imágenes de nuestro estudio, la utilizamos para componer nuestra matriz de entrenamiento. Esta matriz es el dato de entrada del algoritmo “*kmeans*” encargado de extraer los centroides, tantos como el número de grupos que hemos especificado, la matriz sería de 20 filas, numero de grupos elegidos, y de 16 columnas haciendo referencia a las 16 características seleccionadas. En nuestro estudio hemos hecho pruebas con distintos número de grupos {10,20,22,25} para seleccionar el más adecuado. Este, como hemos mencionado antes, es elegir 20 grupos, debido a que en torno a este valor se estabiliza la función de porcentaje de acierto, encontrando su valor máximo de acierto (véase la tabla 4.4 que tenemos más abajo para ver lo explicado).

Una vez tenemos los datos de los centroides, continuamos con las pruebas de las ultimas 32 imágenes, de las cuales nos quedamos solo con los datos de las 16 características extraídas de todas las regiones que sacamos de las 32 imágenes, y mediante una función de distancias euclídeas calculamos si pertenecen a algún centroide, de los extraídos mediante la matriz de entrenamiento, y comparamos con los datos que nos ha facilitado el experto, de cuales son núcleos y cuales por el contrario no. Pudiendo ver el porcentaje de acierto que presenta nuestro algoritmo (véase la tabla 4.4).

Kmeans	Watershed			
	Numero de grupos			
	10	20	22	25
Porcentaje acierto	41,8626%	83,9304%	83,9352	83,9366

Cuadro 4.4: Resultados en función del número de clusters empleados.

Cuando hablamos del porcentaje de acierto en la tabla 4.4, hacemos referencia al número de regiones extraídas clasificadas como núcleos, que realmente lo son, y a las regiones que no son núcleos y se clasifican como no núcleos, esto es, el número de regiones que se clasifican de manera correcta con respecto al total de regiones clasificadas.

A continuación, mostramos los resultados obtenidos por el artículo [2] (véase la figura 4.24).

Clustering performance.

	K-means		Spectral clustering		SVM	
	Global mRMR (%)	Leave-one-out mRMR (%)	Global mRMR (%)	Leave-one-out mRMR (%)	Global mRMR (%)	Leave-one-out mRMR (%)
Watersheds	84.09	84.36	82.64	82.93	82.46	82.52
ACM	80.09	79.64	76.84	77.00	81.87	81.95
GVF	77.83	78.76	77.20	77.33	80.20	80.28

Figura 4.24: Resultados artículo.

Observando dicho resultado, lo comparamos con los obtenidos por nosotros (que mostramos en la tabla 4.4) vemos que el artículo implementado presenta unos resultados muy acordes con los obtenidos por el autor, y además podemos resaltar que para el tipo de imágenes de nuestro dataset, el algoritmo empleado es muy adecuado.

De todos los tipos de segmentación estudiados por dicho artículo combinado con los distintos tipos de agrupamiento, nos encontramos ante el mejor de los casos posibles para la detección de núcleos reales en la imágenes que poseemos.

Convirtiéndose en un algoritmo muy fiable para utilizar en la detección de zonas que no son núcleos para imágenes del mismo tipo que las de nuestro dataset.

4.7. Análisis Crítico del Algoritmo

Este algoritmo es muy robusto como hemos observado. Pero presenta un problema con determinados tipos de imágenes

Cuando observamos imágenes donde encontramos células muy próximas, el algoritmo tiende a solaparlas estudiándolas como células únicas, esto provoca que los centroides no estén situados donde deberían, provocando que se clasifiquen como “no núcleos”, y además que presenten características que les haga ser consideradas como partes de la imagen que no son núcleos, cuando en realidad si lo son. Esto provoca que un dataset donde las células presenten lo anteriormente mencionado, su porcentaje de fiabilidad decaiga considerablemente.

4.8. Propuestas de Mejora del algoritmo

El algoritmo es muy completo, pero deja incompleto la parte de clasificación de los núcleos extraídos como verdaderos en normales/anormales. Esta podría ser una mejora para el algoritmo que lo haría más completo.

Capítulo 5

Conclusiones

5.1. Conclusiones

En este capítulo se desarrollan las conclusiones resultantes del presente trabajo. Se va a presentar las conclusiones a las que se ha llegado sobre cada uno de los objetivos de los dos algoritmos implementados planteados en los capítulos anteriores de esta memoria. Finalmente, se exponen las conclusiones finales sobre el objetivo principal de este proyecto.

5.1.1. Conclusiones del algoritmo Grey Level Method y Energy Method

Este algoritmo tenía como objetivo extraer una clasificación de las células halladas en células malignas o benignas del examen de Papanicolaou (Pap). El fundamento principal de este método es que los núcleos, generalmente, son zonas más oscuras que otras partes de la imagen. Aunque en las imágenes de PAP, existe un fenómeno donde las esquinas de la imagen son más oscuras que las zonas centrales, ya que el centro de la imagen es la zona de células a analizar que deseamos captar desde el microscopio. Esto nos hace requerir forzosamente un preprocesado de la imagen. Posteriormente al preprocesado realizaremos ambos métodos, como ya vimos en el capítulo 33, donde finalmente unimos ambos para extraer los resultados.

Extraeremos los núcleos de las células que son de tamaño anormalmente grande, forma extraña, así como los que representan gran densidad, aplicando este método para los diferentes tipos de células anormales.

Hemos comprobado a medida que implementábamos el algoritmo que los resultados obtenidos no son aplicables para la obtención de dicha clasificación de forma veraz. Esto se debe a que el algoritmo no presenta unos parámetros claros que especifiquen las características de tamaño o forma que clasifican las células en malignas o benignas. Si calculamos la media de los tamaños que presentan las regiones encontradas como núcleos, no se extraerán estas células de manera

adecuada. Al observar un conjunto de dataset donde tenemos todas las células que son malignas, la media seleccionará solo unas células como malignas, cuando en realidad lo son todas ellas. De esta misma manera podemos aplicarlo al otro caso de clasificación, mediante la forma, se especifica que :

$$Circularidad = 4\pi * \frac{Area}{Perimetro^2}$$

A través de ella extraemos valores comprendidos entre 0 y 1, donde 1 es un círculo perfecto y 0 es un objeto que presenta esquinas. Pero al igual que en el caso anterior, el artículo no especifica valores exactos de clasificación, provocando el mismo efecto que antes, es decir, la falta de extracción de resultados aplicables (o satisfactorios).

Además como hemos investigado, este tipo de imágenes presentan una tinción, que en ocasiones se acumula en ciertas zonas de la imagen, provocando encontrar zonas que se consideran núcleos, cuando no lo son, dando también datos irreales, ya que este efecto no lo tiene en cuenta.

En conclusión, el objetivo de este artículo principal para extraer dicha clasificación sí se cumple, aunque no se obtiene de manera adecuada para nuestro dataset debido a la falta de datos detallados para cualquier tipo de imagen.

Por todo esto decidimos buscar un algoritmo más amplio y específico, que se pueda aplicar a cualquier dataset.

5.1.2. Conclusiones del Algoritmo de extracción de características

Este algoritmo tiene como objetivo clasificar las zonas detectadas en zonas que son núcleos reales y zonas que son otros casos, como pueden ser, acumulación de tinte en la imagen, o solapamiento de células que hacen que esta zona sea más oscura pudiendo dar lugar a error.

El objetivo anterior se llevará a cabo mediante un método automatizado para la detección y determinación de los límites de los núcleos de las células convencionales de Papanicolaou (imágenes de frotis cervical). La detección de las áreas candidatas a núcleos se basa en un proceso de reconstrucción de imagen morfológica y la segmentación de los límites núcleos se logra con la aplicación del algoritmo de *Watershed* a la imagen en color a la cual previamente hemos aplicado un gradiente morfológico, utilizando los marcadores de los núcleos extraídos en la etapa de detección.

Para la eliminación de falsos positivos, utilizamos las características sobresalientes que caracterizan a la forma, la textura y la intensidad de la imagen, extraídas de las regiones candidatas a núcleos y una etapa posterior de clasificación para determinar los núcleos verdaderos.

Hemos examinado el funcionamiento de supervisión de *K-means*, técnica de clasificación, empleando características discriminatorias que han sido seleccionadas con un esquema de selección de rasgos basado en el criterio de mínima redundancia-máxima relevancia.

El método propuesto se evaluó en un conjunto de datos de 88 imágenes de prueba de Papanicolaou que contienen 2.361 núcleos de células reconocidos. De las cuales las 56 primeras imágenes de nuestro dataset se utilizaron para extraer una matriz de entrenamiento para establecer cuales son núcleos y cuales son otras zonas distintas, y las 32 imágenes restantes del dataset se utilizaron para la evaluación de dicho algoritmo. Obteniendo los siguientes datos:

	Watershed
Kmeans	Numero de clusters
	20
Porcentaje acierto	83,9304 %

Cuadro 5.1: Resultados del algoritmo de extracción de características

Estos datos son los obtenidos en la sección de resultados del algoritmo de extracción de características; hacen referencia al porcentaje de acierto de clasificar las regiones extraídas en la segmentación como núcleos verdaderos u otras regiones. Obteniendo que aproximadamente un 84 % de las regiones extraídas son clasificadas de manera correcta. Estos resultados son muy satisfactorios debido a que el porcentaje de acierto es muy alto en la clasificación de regiones. Esto proporciona una clara identificación de regiones, paso previo muy importante para clasificar los núcleos en: normales o anormales.

Como conclusiones hemos de resaltar que este algoritmo es muy robusto para nuestro dataset de imágenes como hemos observado, pero presenta problemas con otros tipos de imágenes. Cuando observamos imágenes donde encontramos células muy próximas el algoritmo tiende a solaparlas estudiándolas como células únicas. Esto provoca que los centroides no estén situados dónde deberían y además que presenten características que les haga ser consideradas como partes de la imagen que no son núcleos cuando en realidad sí lo son. Por ello, cuando se utilice un dataset en el que las células presenten solapamiento, el porcentaje se verá reducido.

5.1.3. Conclusiones finales

El objetivo de este proyecto es la implementación de un sistema automático de clasificación de los núcleos detectados en una imagen como benignos o malignos. Este objetivo del proyecto tiene como objetivo secundario: por un lado, el análisis comparativo de algoritmos de segmentación y por otro, el análisis comparativo de algoritmos de preprocesado previo de las imágenes para mejorar la calidad de las mismas, y localizar las imágenes más adecuadas y sospechosas de contener tejido canceroso. Este estudio consistió en evaluar los diferentes algoritmos implementados en términos de reconocimiento de los núcleos existentes en las imágenes. Este reconocimiento se llevo a cabo en una colección de 88 imágenes reales .

En conclusión, debemos resaltar que tras realizar el estudio y obtener los resultados, los algoritmos elegidos fueron los correctos frente al tipo de imágenes de nuestro dataset, aunque por

el contrario, también observamos que en el primero de los algoritmos, aunque llegamos a una clasificación de las células encontradas en las imágenes, no obtuvimos resultados aplicables por la falta de datos, precisando de un artículo más detallado como el segundo que implementamos. Y en el segundo de los algoritmos, aun encontrando buenos resultados para la clasificación de las regiones extraídas de las imágenes como núcleos, no llegamos a la clasificación de los núcleos extraídos como correctos, en el 85 % de los casos, en: núcleos de células normales o núcleos de células cancerígenas. Por tanto, ese es un punto no encontrado de nuestro objetivo principal del proyecto, aunque bien es cierto que este segundo algoritmo es un paso previo importante para la clasificación de las zonas extraídas del dataset.

5.2. Líneas de Trabajo Futuras

Tras realizar la evaluación de los algoritmos, proponemos las siguientes líneas de trabajo:

1. En el primer algoritmo podríamos aplicar las siguientes mejoras propuestas:
 - a) Realizar una clasificación inicial de las imágenes del dataset, que nos permitan aplicar un preprocesado para cada uno de los tipos y así aplicar el algoritmo final a las imágenes preprocesadas de cada tipo diferente.
 - b) Obtener un dataset con las células clasificadas por un experto, en benignas y malignas, que nos permita extraer los parámetros de forma y de tamaño para clasificar las células obtenidas.
 - c) En este algoritmo no se tiene en cuenta los problemas creados por la tinción en estas imágenes. Por este motivo podríamos combinar este algoritmo y el siguiente que hemos implementado para extraer mejores resultados.

2. En el segundo algoritmo hemos comprobado que es muy robusto como hemos observado para nuestro dataset, pero presenta un problema con determinados tipos de imágenes. Una limitación de dicho algoritmo es cuando encontramos imágenes donde se presentan células muy cercanas, el algoritmo tiende a unir las como una sola, esto acarrea que los centroides no estén situados donde deberían y además provocan que se consideren zonas que no son núcleos cuando en realidad si lo son. En base a esto las mejoras propuestas son:
 - a) Mejorar los casos donde aparecen células muy próximas en las imágenes, realizando un preprocesado, para mejorar la clasificación de las células en núcleos o en otras zonas.
 - b) Añadir al algoritmo una parte final para clasificar las zonas obtenidas como núcleos en células benignas o malignas, y así clasificar las imágenes. Investigando la manera de extraer el citoplasma y el núcleo por separado, y buscar dicha relación núcleo-citoplasma para clasificar las células, que según los estudios es la metodología más fiable. Aunque bien es cierto, que esta tarea es muy problemática y difícil.

Apéndice A

Herramientas de procesamiento de imágenes

A.1. Ecuación del histograma

La **ecuación del histograma** de una imagen es una transformación que pretende obtener, para una imagen, un histograma con una distribución uniforme, es decir, todos los niveles de gris tienen el mismo número de píxeles. La ecuación se realiza trabajando sobre los histogramas acumulados.

Pasos de la ecuación:

1. Obtener el histograma acumulado: extraemos en primer lugar, los datos del histograma de la imagen, y normalizamos dichos datos (dividimos entre el número total de píxeles de la imagen ¹), una vez tenemos esto realizado calculamos su histograma acumulado:

$$a) s(k) = \frac{1}{n} \sum_{j=0}^k h(j), \text{ donde } k=0 : L-1 \text{ y } n = \text{número total de píxeles.}$$

2. Obtenemos los nuevos valores de las intensidades, entonces el histograma acumulado para cada nivel de gris sería:

$$a) s(k) = \sum_{j=0}^k h(j) x \frac{\text{Máximo nivel de intensidad}}{\text{Número Total De Píxeles}}$$

3. Y reemplazamos las intensidades².

Ver la figura A.1 para observar un ejemplo de la ecuación del histograma.

¹ $\frac{1}{h}$ que encontramos en la ecuación de abajo

² Información e imágenes obtenidas de <http://www.slideshare.net/omarspp/imagen-e-histograma>

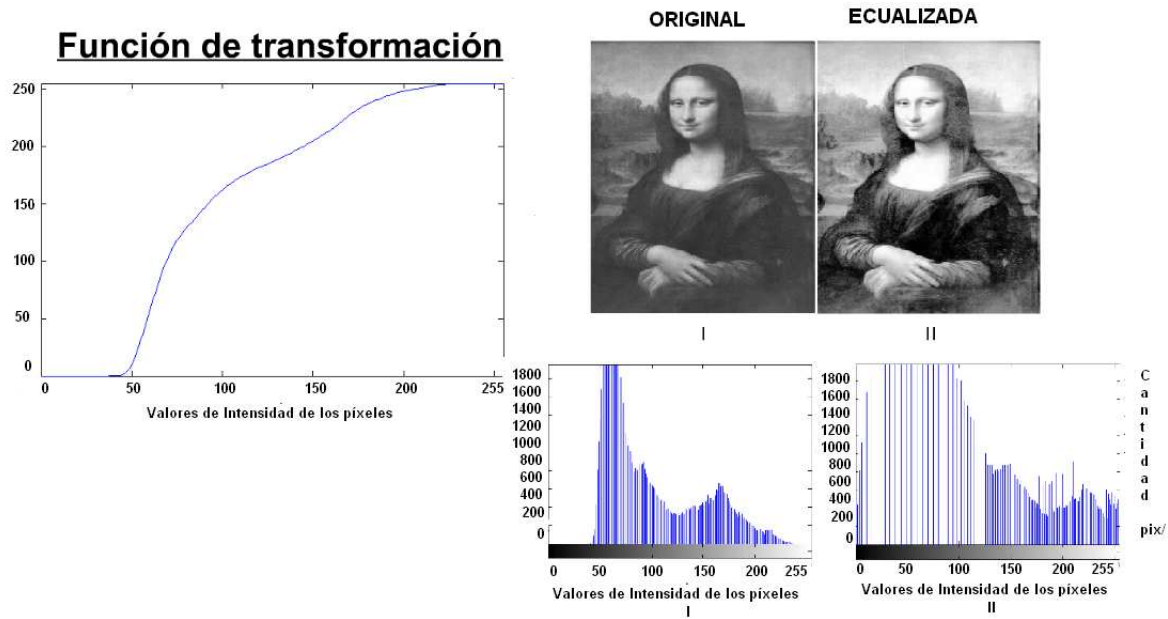


Figura A.1: Ejemplo de ecualización de histograma.

La gráfica de la izquierda muestra la función de transformación que se obtiene con el histograma acumulado. La (I) muestra la imagen original y su histograma (debajo de la misma) y (II) muestra el resultado de la ecualización y su histograma obtenido (debajo de la imagen).

A.2. Algoritmo de Canny

En el área de procesamiento de imágenes, la detección de los bordes de una imagen es de suma importancia y utilidad, pues facilita muchas tareas, entre ellas: el reconocimiento de objetos, la segmentación de regiones, etc.

El algoritmo de Canny es usado para detectar todos los bordes existentes en una imagen. Este algoritmo está considerado como uno de los mejores métodos de detección de contornos mediante el empleo de máscaras de convolución y está basado en la primera derivada. Los puntos de contorno son zonas de píxeles en las que existe un cambio brusco de nivel de gris.

El operador de detección de bordes de Canny fue desarrollado por un catedrático de la universidad de Berkeley (EEUU) en 1986 y se basa en un algoritmo de múltiples fases para detectar un amplio rango de bordes. Es, sin duda, el operador más utilizado en la detección de bordes.

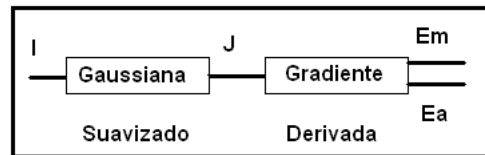
El objetivo de Canny era descubrir el algoritmo adecuado para la detección de bordes. La elección de un detector adecuado significará una buena detección, el algoritmo debe marcar tantos bordes reales como sea posible, los bordes marcados deben estar lo más cerca posible del borde en la imagen real, y por último una mínima respuesta, es decir, un borde dado debe ser marcado sólo una vez. El ruido que este presente en la imagen, no debería crear falsos bordes.

Esta técnica, que se caracteriza por estar optimizada para la detección de bordes diferenciales, consta de 3 fases pero algunos autores agregan una ultima fase, siendo en total 4 fases.

Fases de detección de bordes de Canny:

1. Obtención del gradiente
2. Supresión no máxima al resultado del gradiente
3. Histéresis de umbral a la supresión no máxima
4. Cierre de contornos abiertos

Obtención del gradiente (mostramos su diagrama de flujo):



Para obtener el gradiente se tiene que suavizar la imagen:

-Aplicar a la imagen I un suavizado gaussiano G o filtro gaussiano (también se puede aplicar otros filtros).

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (\text{A.1})$$

- Para calcular el kernel gaussiano G utilizar la siguiente ecuación. .

- Para cada píxel de la imagen J(i,j).

->Calcular las componentes del gradiente Jx y Jy.

->Calcular la magnitud de los bordes.

$$e_s(i, j) = \sqrt{J_x^2(i, j) + J_y^2(i, j)} \quad (\text{A.2})$$

->Estimar la orientación de la normal de los bordes.

$$e_o(i, j) = \arctan \frac{J_y}{J_x} \quad (\text{A.3})$$

Una vez tenemos esto, aplicamos Supresión no máxima[21] al resultado del gradiente, teniendo como objetivo obtener bordes de 1 píxel de grosor.

Aplicamos histéresis de umbral a la supresión no máxima, esto nos permite eliminar máximos procedentes de ruido. La salida es un conjunto de bordes conectados de contornos de la imagen, así como la magnitud y orientación.

Y, por último, procedemos al cierre de contornos abiertos (Algoritmo de Deriche y Cocquerez) [23].

1. La imagen de entrada es una imagen de contornos binarizada (1= borde; 0=no borde).
2. Para cada punto de borde de un extremo abierto se le asigna un código que determina las direcciones de búsqueda para el cierre del contorno.

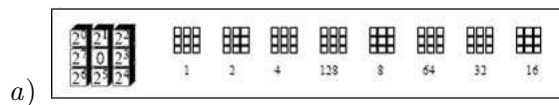


Figura A.2: Direcciones de Búsqueda para cerrar los contornos abiertos.

3. Para los píxeles marcados con este código se marca como píxel de borde el de máximo gradiente en las tres direcciones posibles.
4. Se repiten los pasos hasta que se cierren todos los contornos.

Véase el ejemplo de la aplicación del algoritmo de Canny en la figura A.3.



Figura A.3: Algoritmo de Canny.

La figura de la izquierda muestra la imagen original, mientras que la de la derecha muestra el resultado de la aplicación del algoritmo de Canny (Figura extraída del pie 3).

A.3. CLAHE (*Contrast Limited Adaptive Histogram Equalization*)

CLAHE opera en pequeñas regiones de la imagen, llamados azulejos, en lugar de toda la imagen. Contrasta cada azulejo, de manera que el histograma de la región de salida coincide aproximadamente con el histograma esperado (histograma con los valores de intensidad de los píxeles

distribuidos uniformemente por todo el rango de valores de intensidad). Los azulejos vecinos se combinan mediante la interpolación bilineal para eliminar las fronteras artificialmente.

El contraste, especialmente en las zonas homogéneas, puede ser limitado para evitar la amplificación de cualquier ruido que pueda estar presente en la imagen (véase la figura A.4 para observar este mediante un ejemplo).

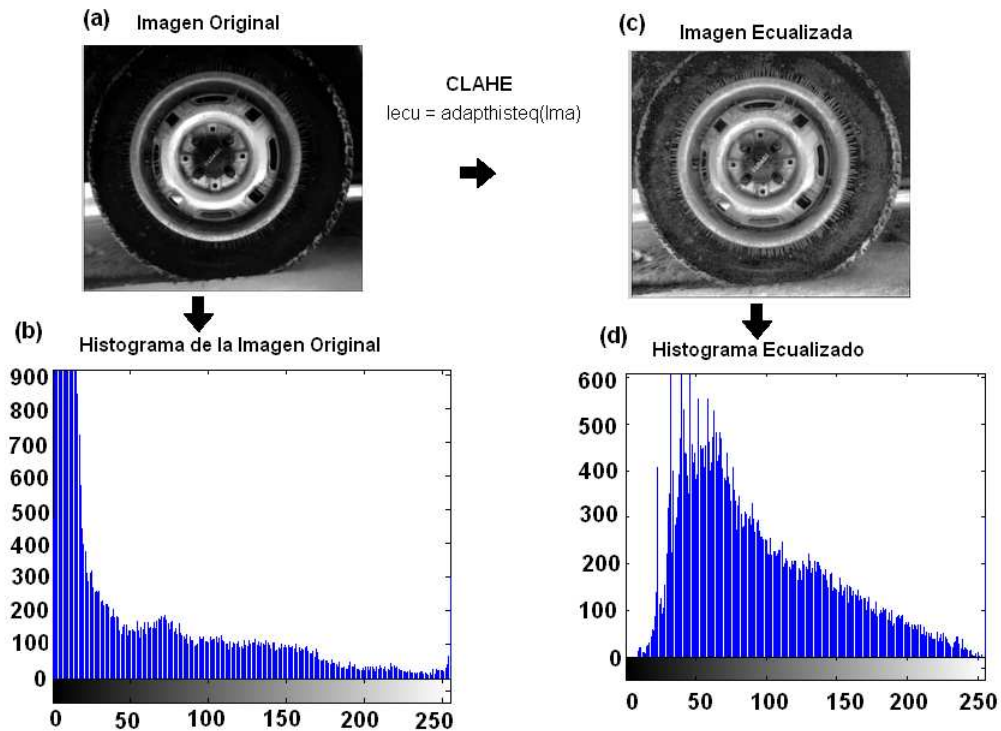


Figura A.4: Ejemplo de imagen mejorada mediante el algoritmo CLAHE.

Esta figura está compuesto por cuatro: (a) Imagen original , (b) Histograma de la imagen original, (c) Imagen resultante de la ecualización y (d) histograma de la imagen ecualizada.

La **interpolación bilineal** se aplica en el procesamiento de imágenes para transformar una imagen en la imagen perfecta, imagen con los objetos resaltados frente el fondo de la misma, calculando y asignando valores adecuados de intensidad a los píxeles. A diferencia de otras técnicas de interpolación, como son la interpolación del vecino más cercano y la interpolación bicúbica, la interpolación bilineal utiliza sólo los 4 valores de los píxeles más próximos que se encuentran en direcciones diagonales de un píxel dado, con el fin de encontrar los valores adecuados de intensidad de color de ese píxel. A continuación, toma un promedio ponderado de estos 4 píxeles para llegar a su final, valor interpolado.

Este algoritmo reduce parte de la distorsión visual causado por el cambio de tamaño de una imagen.

A.4. Método de Otsu

El **Método de Otsu**, es un algoritmo que obtiene un umbral para la segmentación de la imagen de manera automática. Dicho umbral se selecciona por un criterio discriminante, es decir, se selecciona el umbral a fin de maximizar la separabilidad de las clases resultantes en los niveles de gris.

Es importante, en el procesamiento de imagen, seleccionar un umbral adecuado para la extracción de objetos de su fondo. Una variedad de técnicas han sido propuestas en este sentido.

En un caso ideal, el histograma tiene un valle profundo y agudo entre dos picos, estos corresponden a los objetos y al fondo respectivamente, de modo que el umbral puede ser elegido en el fondo de este valle (véase la figura A.5 para observar esto). Sin embargo, para la mayoría de imágenes reales, a menudo es difícil detectar con precisión el fondo del valle, especialmente como en casos cuando el valle es plano y ancho, unido con el ruido, o cuando los dos picos son extremadamente desiguales en altura.

Ha habido algunas técnicas propuestas para superar estas dificultades, pero muchos requieren cálculos tediosos.

En cualquier caso, el método de Otsu para obtener el umbral se ha evaluado en la mayoría de los métodos propuestos hasta ahora, dando buenos resultados en el caso más elemental de la selección de umbral, en el que sólo los niveles de gris del histograma son suficientes, sin otros conocimientos a priori. Este método es considerado el criterio más adecuado para seleccionar automáticamente un umbral (véase la figura A.5).

Algoritmo:

En primer lugar normalizamos el histograma y se considera una distribución de probabilidad, con el fin de simplificar la discusión:

$$p_i = n_i/N \quad (\text{A.4})$$

dónde $p_i \geq 0$,

$$\sum_{i=1}^L p_i = 1 \quad (\text{A.5})$$

dónde L son los diferentes niveles de gris y $N=n_1 + n_2 + \dots + n_L$, donde n son el número de píxeles de cada nivel.

En segundo lugar, supongamos que tenemos divididos en dos clases los niveles de píxeles: C_0 y C_{-1} (de fondo y los objetos, o viceversa) por un umbral de nivel k; C_0 píxeles denota con los niveles de $[1, \dots, k]$ y C_1 denota píxeles con niveles de $[k + 1, L]$. A continuación, las probabilidades de ocurrencia de clase y los niveles medios de la clase, respectivamente, están dadas por:

$$w_0 = P_r(C_0) = \sum_{i=1}^k p_i = w(k) \quad (\text{A.6})$$

$$w_1 = P_r(C_1) = \sum_{i=k+1}^L p_i = 1 - w(k) \quad (\text{A.7})$$

y

$$\mu_0 = \sum_{i=1}^k iP_r(i|C_0) = \sum_{i=1}^k ip_i/w_0 = \mu(k)/w(k) \quad (\text{A.8})$$

$$\mu_1 = \sum_{i=k+1}^L iP_r(i|C_1) = \sum_{i=k+1}^L ip_i/w_1 = \frac{\mu_T - \mu(k)}{1 - w(k)} \quad (\text{A.9})$$

dónde $w(k) = \sum_{i=1}^k p_i$ y $\mu_k = \sum_{i=1}^k ip_i$

Así el momento cero y el primer momento son acumulativos hasta el nivel k-ésimo:

$$\mu_T = \mu(L) = \sum_{i=1}^L ip_i \quad (\text{A.10})$$

Siendo este el nivel medio total de la imagen original. Es fácil comprobar la siguiente relación para cualquier elección de k:

$$w_0\mu_0 + w_1\mu_1 = \mu_T, w_0 + w_1 = 1$$

Las varianzas de las distintas clases son:

$$\sigma_0^2 = \sum_{i=1}^k (i - \mu_0)^2 P_r(i|C_0) = \sum_{i=1}^k (i - \mu_0)^2 p_i/w_0 \quad (\text{A.11})$$

$$\sigma_1^2 = \sum_{i=k+1}^L (i - \mu_1)^2 P_r(i|C_1) = \sum_{i=k+1}^L (i - \mu_1)^2 p_i/w_1 \quad (\text{A.12})$$

Con el fin de evaluar el umbral adecuado (en el nivel k), se adoptarán las medidas discriminantes siguiendo el criterio (o medidas de separabilidad de clase) que se utilizan en el análisis discriminante:

$$\lambda = \sigma_B^2/\sigma_W^2, k = \sigma_T^2/\sigma_W^2, \eta = \sigma_B^2/\sigma_T^2$$

dónde:

$$\sigma_W^2 = w_0\sigma_0^2 + w_1\sigma_1^2 \quad (\text{A.13})$$

$$\sigma_B^2 = w_0(\mu_0 - \mu_T)^2 + w_1(\mu_1 - \mu_T)^2 = w_0w_1(\mu_1 - \mu_0)^2 \quad (\text{A.14})$$

Y

$$\sigma_T^2 = \sum_{i=1}^L (i - \mu_T)^2 p_i \quad (\text{A.15})$$

Los criterios discriminantes son maximizando λ , k y η , respectivamente ya que la relación básica siempre se mantiene:

$$\sigma_W^2 + \sigma_B^2 = \sigma_T^2 \quad (\text{A.16})$$

Después de experimentar con dicho método se observó que el mejor criterio de medida a adoptar para obtener el umbral en el nivel k , maximizando η .

$$\eta(k) = \sigma_B^2(k) / \sigma_T^2 \quad (\text{A.17})$$

$$\sigma_B^2(k) = \frac{[\mu_T^2 w(k) - \mu(k)]^2}{w(k)[1 - w(k)]} \quad (\text{A.18})$$

Obteniendo el umbral.

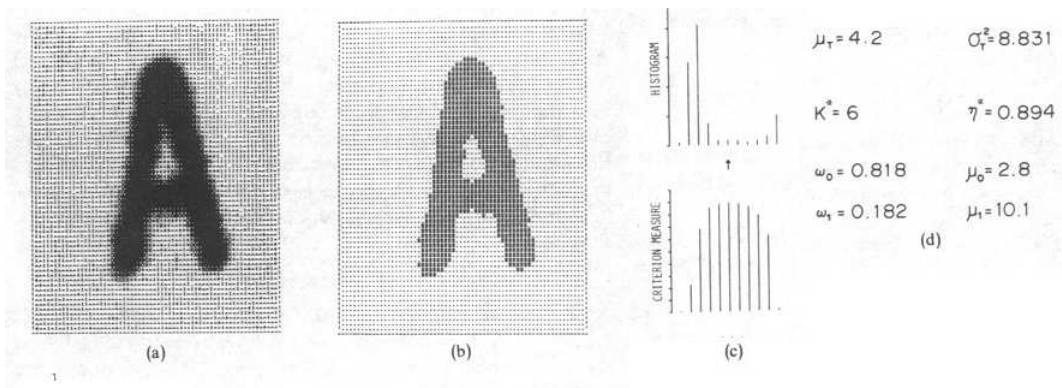


Figura A.5: Método de Otsu.

La figura muestra (a) Imagen original, (b) la imagen umbralizada, (c) los histogramas: de la imagen y el criterio seguido, y (d) los datos utilizados para obtener el resultado.

A.5. Operadores morfológicos: dilatación y erosión

A.5.1. Dilatación y Erosión ³

A.5.1.1. Dilatación

Observamos la imagen con un subconjunto de datos binarios, dónde se produce un ensanchamiento de los máximos, presentado el siguiente planteamiento:

$A \oplus B = B \oplus A = \bigcup_{a \in A} B_a$ o también:

$$y[n] = \bigvee_{k=-\infty}^{\infty} (x[k] + b[n - k]) = \bigvee_{k=-\infty}^{\infty} (x[n - k] + b[k]) = \bigvee_{k/b[k]=0}^{\infty} (x[n - k])$$

Esto es el máximo de los valores de la señal X que coinciden con los ceros del elemento estructurante b invertido, son los que continúan (véase la figura A.6).

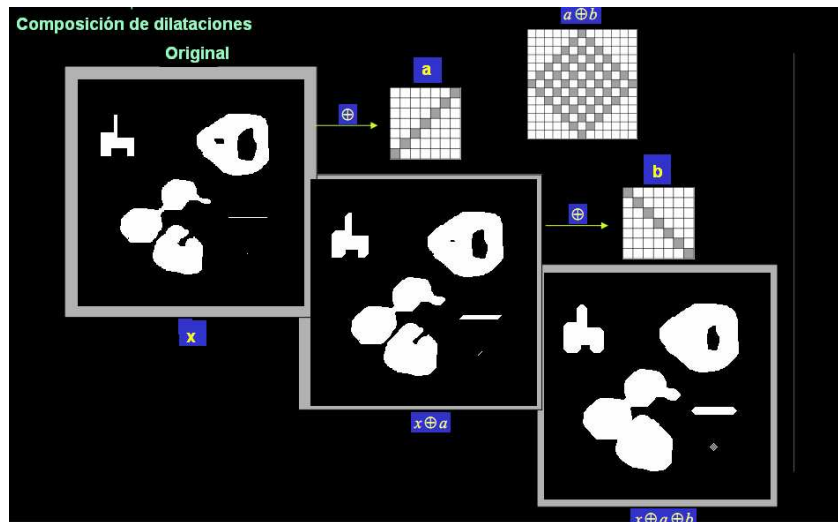


Figura A.6: Composición de dilatación.

Observamos tres imágenes principales en la figura: la primera de ellas (en la izquierda) muestra la imagen original, la segunda de ellas muestra la operación de dilatación en la misma en una dirección (elemento estructurante que se encuentra encima de ella) y por último, una imagen también dilatada, pero con un elemento estructurante en la dirección opuesta (mostrado encima de la misma).

Erosión Observamos la imagen con un subconjunto de datos binarios, dónde se produce un adelgazamiento de los máximos, presentado el siguiente planteamiento:

$$A \ominus B = \{z \in E | B_z \subseteq A\} \text{ o } y[n] = \bigwedge_{k=-\infty}^{\infty} (x[k] - b[n - k]) = \bigwedge_{k=-\infty}^{\infty} (x[n - k] - b[k]) = \bigwedge_{k/b[k]=0}^{\infty} (x[n + k])$$

En este caso es al contrario, los mínimos de los valores de la señal X que coinciden con los ceros del elemento estructurante B, son los que continúan (véase la figura A.7).

³Información e imágenes extraídas de apuntes de la asignatura de Temas Avanzados de Procesamientos de Señales, Tema 4 <http://arantxa.ii.uam.es/~jms/taps/#pr%C3%A1cticas>:

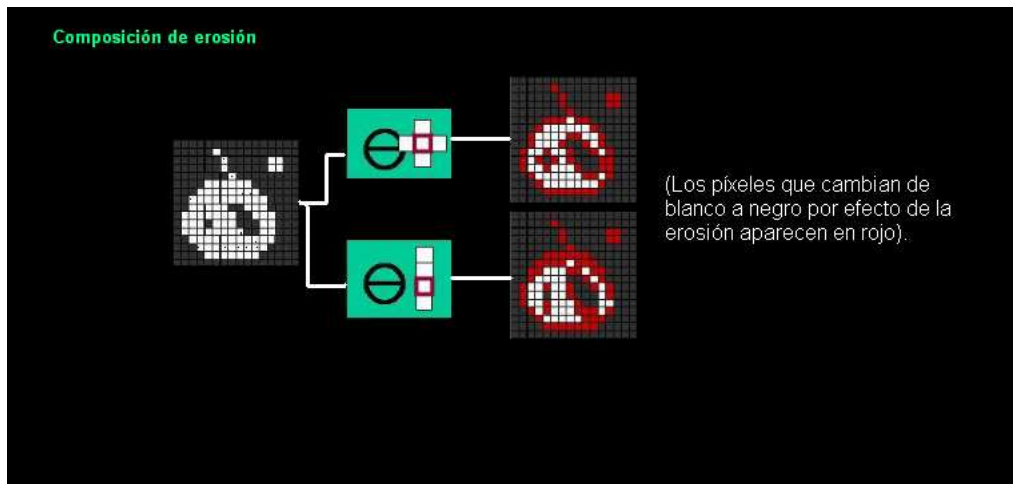


Figura A.7: **Composición de Erosión.**

Imagen original a la cual aplicamos una operación de erosión con dos elementos estructurantes, resultados en las imágenes de la derecha.

La motivación de aplicar dichos operadores morfológicos es:

1. Baja complejidad computacional: cálculo de mínimos y máximos en la señal de entrada.
2. El resultado para cada 'n' es un valor de la señal de entrada: estabilidad y conservación del rango y de la precisión.
3. Preservan el contraste en los contornos, dejándolos bien definidos.

Además presentan propiedades importantes:

Dilatación:

- Añade todos los puntos del fondo que tocan el borde de un objeto → la dilatación es extensiva.
- Rellena entrantes en los que no quepa el EE (Elemento Estructurante) (pequeños agujeros y bahías).

Erosión:

- La erosión es antiextensiva: reduce el tamaño del objeto.
- Elimina elementos en los que no quepa el elemento estructurante (pequeñas islas y protuberancias).

Además de estas propiedades también presentan las de distribución y composición:

■ *Distribución :*

$$(x \vee y) \oplus b = (x \oplus b) \vee (y \oplus b)$$

$$(x \wedge y) \ominus b = (x \ominus b) \wedge (y \ominus b)$$

■ *Composición:*

$$x \oplus a \oplus b = x \oplus c, \quad c = a \oplus b$$

$$x \ominus a \ominus b = x \ominus c, \quad c = a \oplus b$$

Figura A.8: Propiedades de Erosión: distribución y composición.

Véase los resultados en la figura A.9 que crean las operaciones morfológicas en una imagen.

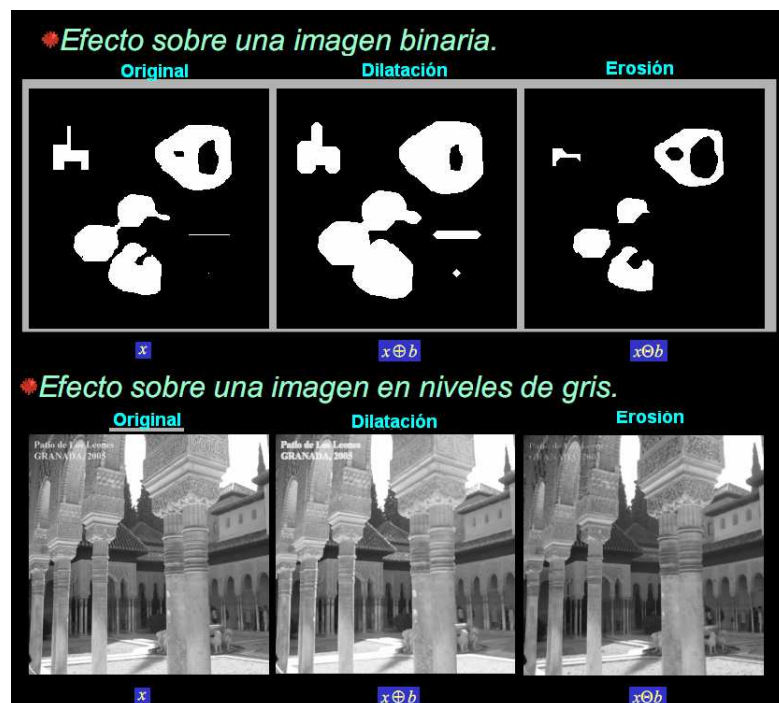


Figura A.9: Ejemplos de resultados de la aplicación de: dilatación y erosión.

A.6. Gradiente morfológico de color

El gradiente morfológico tiene como objetivo resaltar los contornos.

Dicho gradiente se puede obtener de diferentes maneras, la primera de ellas se puede obtener como :

$$\nabla(f) = \delta_g(f) - \varepsilon_g(f) \quad (\text{A.19})$$

donde $\delta_g(f)$ y $\varepsilon_g(f)$ es la dilatación en escala de grises y la erosión en escala de gris (véase más arriba del apéndice dónde se explican dichos operadores morfológicos), con un elemento estructurante g , de tamaño 3×3 como se indica en el artículo.

O también podemos definirla como:

$$\nabla(f) = \max \{f(x)\} - \min \{f(x)\} = \max \{|f(x) - f(y)|\} \quad (\text{A.20})$$

donde $\forall x, y \in g$, que es la diferencia de la máxima intensidad absoluta entre dos píxeles en el área del elemento de estructurante.

Para imágenes en color con píxeles denotados como tres vectores dimensionales el gradiente de color morfológica (CMG) se puede expresar como:

$$CMG = \max_{x_i, x_j \in G} \{\|x_i - x_j\|_p\} \quad (\text{A.21})$$

donde x_i, x_j son píxeles del elemento estructurante G .

Teniendo como resultados los contornos como apreciamos en la figura A.10 ⁴.

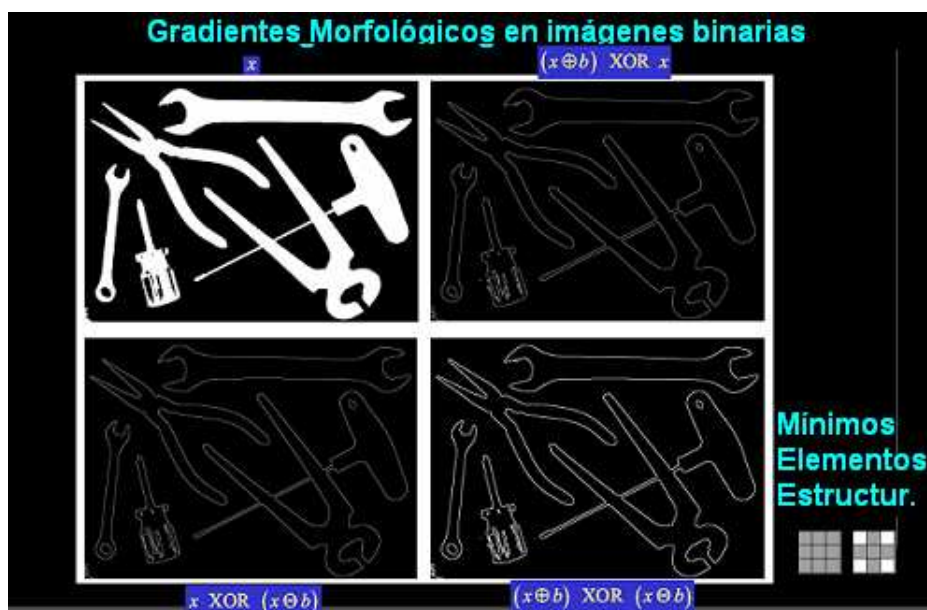


Figura A.10: Gradiente Morfológico.

Esta figura extrae de una imagen el gradiente morfológico, a través de dos elementos estructurantes (con uno de ellos se detecta las líneas horizontales y con el otro las verticales), luego suma los resultados para obtener el gradiente total.

⁴Información e imágenes extraídas de apuntes de la asignatura de Temas Avanzados de Procesamientos de Señales, Tema 4 <http://arantxa.ii.uam.es/~jms/taps/#pr%C3%A1cticas>:

A.7. Local Binary Patterns (LPB)

Local Binary Patterns Este operador es un método para describir texturas. El operador original LBP etiqueta cada uno de los píxeles de una imagen mediante el muestreo de matrices 3x3, comparando cada píxel con el valor central de la matriz y considerando el resultado como un número binario. A continuación, se calcula el histograma de las etiquetas y se concatenan los histogramas. Esto se puede utilizar como un descriptor de textura. Dado que LBP, por definición, es invariante a los cambios monótonos en escala de grises, se complementó con una medida ortogonal de contraste local.

La figura A.11 muestra cómo se calcula la medida de contraste C y el operador LBP. El contraste se calcula restando al promedio de los niveles de gris por debajo del píxel central al promedio de los niveles de gris por encima (o igual) que el píxel central. Las distribuciones bidimensionales del LBP y las medidas locales de contraste se utilizan como características de la imagen y es por ello, que al operador también se le llama LBP/C.

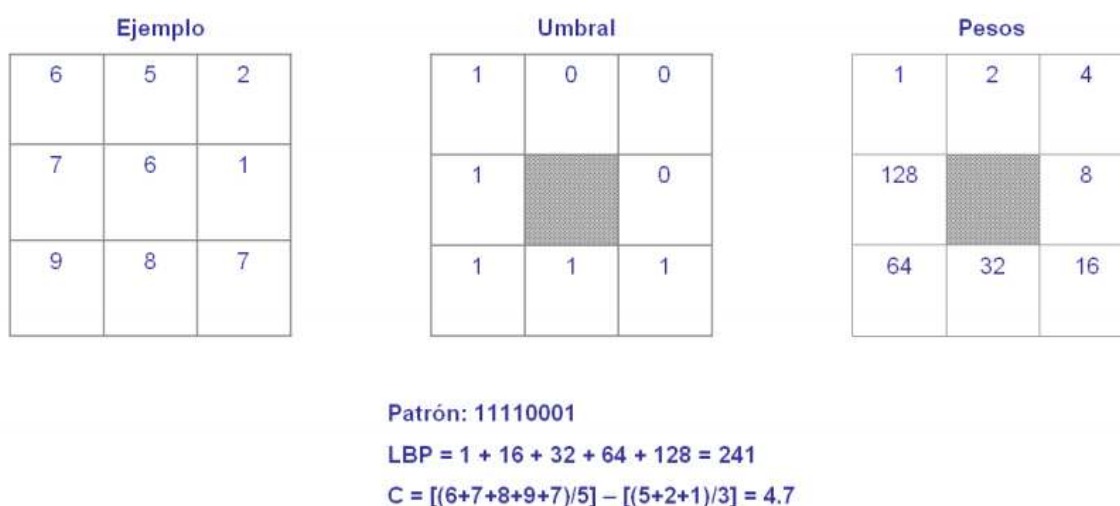


Figure A.11: **Procedimiento del cálculo del LBP original.** Figura extraída de [3].

Se clasifica como 1, sumamos su peso, si el píxel que estamos evaluando es mayor o igual que el central. En caso contrario, se clasifica como cero (no contamos el peso).

La evolución del operador LBP (detallada en [3]) estaba basada en extender el operador para el uso de diferentes tamaños, y añadir bilinealidad; el tamaño se quería extender no solo a ocho puntos, sino a muestreos circulares, hipérbolas, parábola, elipse, y espirales como muestra la figura A.12 extraída de [7], y la bilinealidad se consigue con la interpolación de los valores de los píxeles, lo que permite utilizar cualquier radio y por lo tanto cualquier número de píxeles vecinos. Para los entornos circulares se usa la notación (P,R), donde P indica el número de puntos de muestreo y R el radio del círculo. En la figura A.13 se pueden observar ejemplos para diferentes tamaños, en concreto para (8, 2); (12, 2,5) y (16, 4) como valores de (P,R).

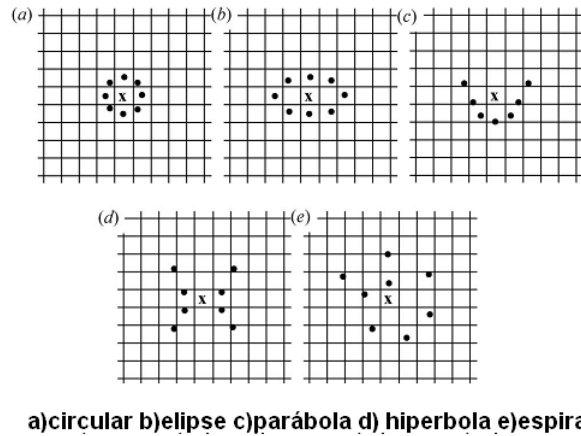


Figure A.12: Topologías de Vecindarios. Figura extraída de [7].

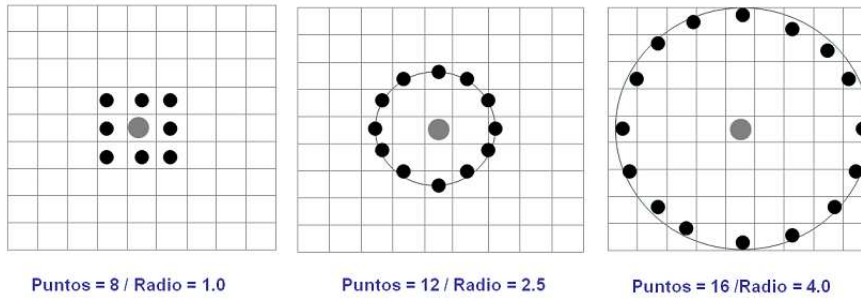


Figure A.13: Simetría circular para Conjuntos de Vecinos. Figura extraída de [7].

A.7.0.2. Algoritmo LBP [3]

Definimos T , Textura, como la distribución conjunta de los niveles de gris de $P + 1 (P > 0)$ píxeles de una imagen.

$$T = t(g_c, g_0, g_1, \dots, g_{p-1})$$

Donde g_c se corresponde con el nivel de gris del píxel central del muestreo de una matriz $n \times n$ y $g_p (p = 1, 2, \dots, p - 1)$, corresponde a los valores de gris de los píxeles P equidistantes en un círculo de radio $R (R > 0)$ que forman un conjunto circular y simétrico de los vecinos con respecto al centro.

La Figura A.13 muestra tres conjuntos vecinos circularmente simétricos para diferentes valores de P y R , como ya hemos indicado. Sin perder información podemos restar g_c de los puntos g_p , quedando:

$T = t(g_c, g_0 - g_c, g_1 - g_c, \dots, g_{p-1} - g_c)$, asumiendo que la diferencia es independiente, la distribución puede ser factorizada:

$T \approx t(g_c)t(g_0 - g_c, g_1 - g_c, \dots, g_{p-1} - g_c)$, como $t(g_c)$ describe la luminosidad general de una imagen, que no está relacionada con la textura de la imagen local, puede ser ignorada y queda:

$$T \approx t(g_0 - g_c, g_1 - g_c, \dots, g_{p-1} - g_c)$$

Aunque es invariante frente a cambios de escala de grises, las diferencias se ven afectados por el cambio de escala. Para lograr invariancia con respecto a cualquier transformación monótona de la escala de grises, sólo es necesario considerar los signos de las diferencias:

$T \approx t(s(g_0 - g_c), s(g_1 - g_c), \dots, s(g_{p-1} - g_c))$, donde:

$$S(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (\text{A.22})$$

Ahora, un peso 2^p es asignado a cada signo $s(g_p - g_c)$ transformando la diferencia en el muestreo en un único código LBP:

$$LBP_{P,R} = \sum_{p=0}^{p-1} s(g_p - g_c)2^p \quad (\text{A.23})$$

Así de este operador se han desarrollado extensiones reflejando el interés que en él han mostrado los diferentes grupos de investigación.

Patrones Uniformes Otra versión, es la llamada patrones uniformes, que puede ser usado para reducir la longitud del vector de características y para implementar un descriptor sencillo e invariante frente a rotaciones. Fue inspirada porque algunos patrones binarios son más frecuentes en las imágenes de textura que otros.

Un código LBP se llama homogéneo si el patrón binario contiene un máximo de dos transiciones a nivel de bits, de 0 a 1 o viceversa, cuando el patrón de bits es atravesado es.

En el cálculo de las etiquetas LBP, cuando los patrones uniformes son utilizados, se usa una etiqueta para cada uno de los patrones uniformes y todos los patrones no uniformes están etiquetados con una sola etiqueta. Por ejemplo, cuando se utiliza (8, R) de vecindad, hay un total de 256 patrones, 58 de los cuales son uniformes y el resto son no uniformes, por lo que resulta uno total de 59 etiquetas diferentes.

Invariante a rotaciones El operador LBPPR produce 2^p valores de salida diferentes, correspondientes a los 2^p patrones binarios diferentes que se pueden formar por los P píxeles vecinos.

Cuando se rota la imagen, los valores g_p de gris correspondientes se moverán a lo largo del perímetro del círculo alrededor de g_p . Puesto que g_0 está destinado a ser el valor de gris de elemento (0, R) a la derecha de g_c la rotación de un patrón binario en particular, naturalmente, da lugar a un LBPP,R diferente.

Esto no se aplica a los patrones que consta de sólo 0s (o 1s) que permanecen constantes en todos los ángulos de rotación. Para eliminar el efecto de la rotación, es decir, para asignar un identificador único para cada rotación un patrón binario local invariable se define:

$$LBP_{P,R}^{ri} = \min \{ROR(LBP_{P,R}, i) | i = 0, 1, \dots, P - 1\} \tag{A.24}$$

donde $ROR(x, i)$ realiza un desplazamiento circular, a nivel de bit, hacia la derecha P-bits de x i-veces.

En términos de imagen de bits, la ecuación anterior, corresponde simplemente a girar el conjunto de vecinos hacia la derecha tantas veces como el número máximo de los bits más significativos, a partir deg_{P-1} , esta 0.

$LBP_{P,R}^{ri}$ cuantifica las ocurrencias de rotaciones invariables de los patrones, correspondientes a ciertas micro características de la imagen, por lo que los patrones pueden ser considerados como detectores de características de textura. La figura inferior ilustra las 36 únicas rotaciones invariables de patrones binarios locales que pueden ocurrir en el caso de $P = 8$, es decir, $LBP_{8,R}^{ri}$ puede tener 36 valores distintos. Si $R = 1$, esto es, $LBP_{8,1}^{ri}$, se corresponde con el operador invariante, de escala de grises, y rotación que ha sido designado como LBPROT (véase la figura A.14 para observar este apartado).

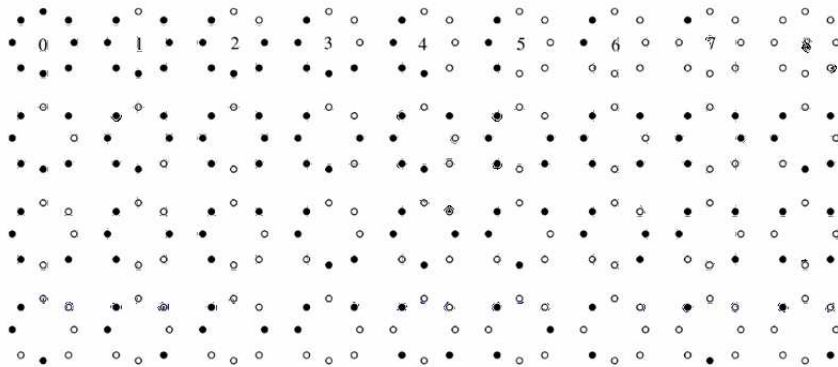


Figura A.14: **Simetría circular** $LBP_{8,R}^{ri}$. Figura extraída de [7].

Los círculos en blanco y negro son los valores de bits de 0 y 1 en la salida de 8 bits. La primera fila contiene los nueve patrones uniformes y los números dentro de ellos corresponden a su único código $LBP_{8,R}^{ri}$.

Invariancia a rotación mejorada con los patrones uniformes y cuantificación más fina del espacio angular La experiencia práctica, sin embargo, ha demostrado que LBPROT como

tal, no proporciona una muy buena discriminación.

Hay dos razones:

- Las frecuencias de ocurrencia de los 36 patrones individuales incorporados en LBPROT varían mucho.
- La cuantificación en bruto del espacio angular es en intervalos de 45 grados.

Se ha observado que existen ciertos patrones binarios que están presente en más del 90% de las texturas generadas por muestreos 3x3 píxeles. A estos patrones se les conoce con el nombre de patrones uniformes, que tienen una cosa en común: su estructura circular uniforme tiene pocas transiciones espaciales. Los patrones uniformes se muestran en la primera fila de la figura A.14. Funcionan como una micro plantilla, con un punto brillante (0), superficie plana o punto negro (8), y los bordes de diferente curvatura positiva y negativa (1-7).

Para definir formalmente los patrones uniformes, se introduce una medida de la uniformidad U (patrón), que se corresponde con el número de transiciones espaciales (a nivel de bits 0 / 1) o cambios del patrón.

Designamos a los patrones que tienen un valor de U al menos de 2 como patrones uniformes y tenemos el siguiente operador invariante a escala de grises y rotaciones, en lugar del usado hasta ahora $LBP_{P,R}^i$:

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c), & \text{si } U(LBP_{P,R}) \leq 2 \\ P + 1 & \text{en otro caso} \end{cases} \quad (\text{A.25})$$

donde tenemos que

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - (g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)| \quad (\text{A.26})$$

El superíndice, riu2, refleja el uso de los patrones invariable a rotaciones y uniformes que tienen un valor de U menor o igual a 2.

Por definición, $P + 1$ patrones binarios uniformes pueden existir en un conjunto de P píxeles vecinos con simetría circular. La ecuación (25) asigna una etiqueta única para cada una de ellas, correspondiente al número de bits '1' en el patrón ($0 \rightarrow P$), mientras que los patrones no uniformes agrupan bajo la etiqueta genérica ($P + 1$). En la figura A.14, las etiquetas de los patrones uniformes se indican en el interior de los patrones. En la práctica, el mapeo de $LBP_{P,R}$ a $LBP_{P,R}^{riu2}$, que tiene $P + 2$ valores de salida diferentes, será más eficaz que la búsqueda en una tabla de 2^P elementos.

La función de la textura final empleada en el análisis es el histograma de los resultados del operador (es decir, las etiquetas del patrón). La razón por la que el histograma de los patrones uniformes proporciona una mejor discriminación, en comparación con el histograma de todos los patrones individuales, se reduce a las diferencias en sus propiedades estadísticas. La proporción relativa de los patrones no uniformes con respecto a todos los patrones, acumulados en un histograma, es tan pequeña que sus probabilidades no se pueden estimar de forma fiable.

La inclusión de estimaciones de ruido y de análisis de disimilitud de las muestras y el modelo de histogramas podría deteriorar el rendimiento.

Hemos señalado anteriormente que la invariancia a rotaciones de $LBPROT(LBP_{8,1}^{ri})$ se ve obstaculizada por la cuantificación de rotaciones de 45 grados del espacio angular proporcionada por los conjuntos de vecinos de 8 píxeles. Una solución es usar una P mayor y la cuantificación del espacio angular se define por $(360^\circ/P)$.

Sin embargo, algunas consideraciones que deben tenerse en cuenta en la selección de P son: en primer lugar, P y R se relacionan en el sentido de que el entorno circular que se corresponde a una R dada contiene un número limitado de píxeles, que introduce un límite superior al número de puntos de muestreo no redundante en el entorno. En segundo lugar, una implementación eficiente, con una tabla de búsqueda de elementos 2^P establece un límite superior práctico para P .

Medida de la varianza de la rotación invariante del contraste de la textura local de una imagen El operador $LPB_{P,R}^{riu2}$ es una medida invariante en una escala de grises, es decir, su salida no se ve afectada por cualquier transformación monótona de la escala de grises. Es una excelente medida de la distribución espacial ya que por definición elimina el contraste. Si la invariancia de escala de grises no es necesaria y queremos incorporar el contraste de la textura de la imagen, esta se puede medir con una medida invariante de rotación de la varianza local:

$$VAR_{P,R} = \frac{1}{P} \sum_{p=0}^{P-1} (g_p - \mu)^2 \quad (A.27)$$

donde $\mu = \frac{1}{P} \sum_{p=0}^{P-1} g_p$

$VAR_{P,R}$ es, por definición, invariante a los cambios en la escala de grises. $LPB_{P,R}^{riu2}$ y $VAR_{P,R}$ son complementarios, se espera que su distribución conjunta $LPB_{P,R}^{riu2}/VAR_{P,R}$ sea una medida potente, invariante a rotaciones, de la textura de la imagen local.

Reconocimiento de núcleos mediante el operador LBP En este artículo [7] en particular, se presentan algunas variantes de (LBP), que son considerados en la técnica de descriptores de textura. Después de realizar una revisión detallada de la literatura acerca de las variantes

existentes de LPB, abordamos el más destacado, con sus pros y sus contras, se presenta los nuevos experimentos con varios LPB basados en descriptores y proponen un conjunto de descriptores de textura novedosas para la representación de imágenes médicas. El estándar de LPB se define como una medida invariable de la textura en escala de grises, derivada de la definición general de la textura en un vecindario. Nuestras variantes se obtienen considerando diferentes formas para el cálculo de vecindad, como hemos visto en la figura A.12 y diferentes codificaciones para la evaluación de los diferentes tipos de LPB en escala de grises.

Estos conjuntos de características se utilizan para la formación de un clasificador automático (un soporte independiente de vectores), extraído para realizar el entrenamiento de las técnicas de clusterización.

En dicho artículo se propone un algoritmo que detecta el cáncer a partir de 20 características extraídas de las imágenes individuales de las células (véase la figura A.15 para observar el resultado).

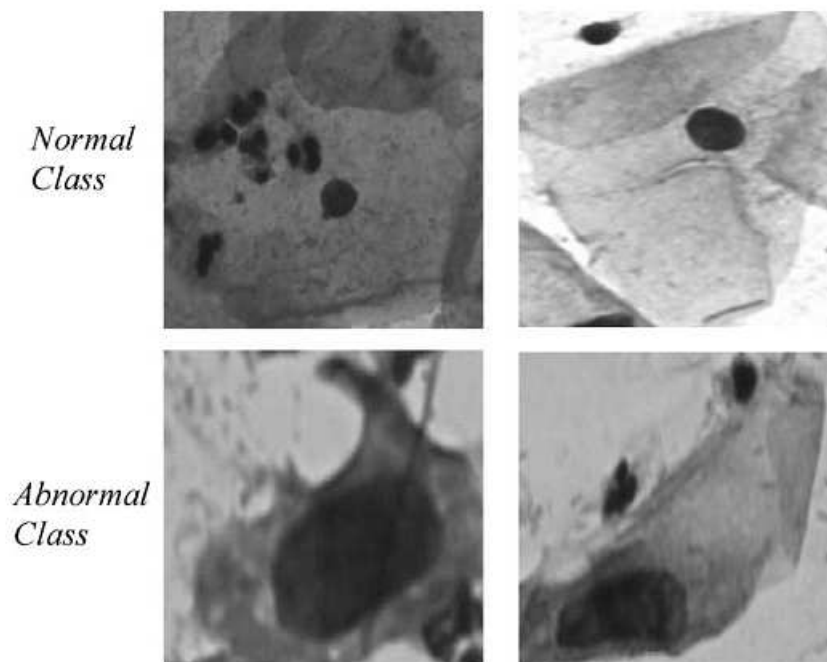


Figura A.15: **Tipos de células del artículo.** Figura extraída de [7]. Las imágenes superiores corresponden a células normales y las dos inferiores a células anormales.

Los resultados obtenidos en el artículo (véase la figura A.16).

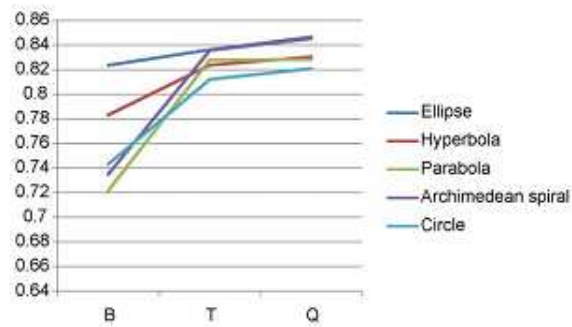


Figura A.16: **Resultados de imágenes PAP distintas topologías.** Figura extraída de [7]. Muestra los resultados para cada una de las topologías posibles. (B= Binaria, T= Ternaria(tres colores), Q=Cuaternaria(cuatro colores posibles)).

Como observamos en la imagen, los que ofrecen mejores resultados son los patrones de elipse, y en segundo lugar para la zona binaria, la hipérbola y la circular. En nuestro artículo a implementar nos piden estos dos últimos casos.

Apéndice B

Herramientas de selección de variables

B.1. mRMR(mínima Redundancia-Máxima Relevancia)

Este algoritmo [24] de selección de variables está basado en información mutua. En lo que sigue, supondremos un problema de clasificación en C clases formado por N patrones. Cada patrón se representa por el vector formado por el conjunto de M variables y un valor adicional que indica la clase a la que pertenece esto es, $x = \{v_i, c\}$ con $i = 1, \dots, M$ y $c = 0, \dots, C - 1$.

El objetivo de la selección de variables consiste en encontrar un subespacio de m variables ($m < M$) que permita clasificar correctamente el problema. La información mutua es la medida más general de independencia estadística entre dos variables aleatorias v_i y v_j , cuantifica la información que v_i y v_j comparten es decir, cómo afecta el conocimiento de una de ellas en la incertidumbre sobre la otra.

Formalmente, se define la información mutua entre dos variables aleatorias v_i y v_j a partir de las funciones de densidad marginales $p(v_i)$ y $p(v_j)$, y la función de densidad conjunta $p(v_i, v_j)$ como,

$$I(v_i; v_j) = \int \int p(v_i; v_j) \log \frac{p(v_i; v_j)}{p(v_i)p(v_j)} dv_i dv_j \quad (\text{B.1})$$

Esto verifica

$$I(v_i; v_j) \geq 0$$

$$I(v_i; v_j) = I(v_j; v_i)$$

Si v_i y v_j son independientes, el conocimiento de v_i no da ninguna información acerca de v_j y, por tanto, su información mutua es cero (la implicación contraria también es cierta, es decir, la información mutua es nula únicamente en el caso de variables independientes). En el otro extremo, si v_i y v_j son idénticamente distribuidas, el conocimiento de v_i implica un total conocimiento sobre v_j y viceversa. La principal dificultad para el cálculo de (2) radica en estimar las densidades

$p(v_i, v_j), p(v_i)$ y $p(v_j)$ a partir de los datos. Para variables categóricas, la ecuación (2) se reduce a un sumatorio y el cálculo de la información mutua es inmediato realizando un simple conteo.

$$I(v_i; v_j) = \sum \sum p(v_i = w_i; v_j = w_j) \log \frac{p(v_i = w_i; v_j = w_j)}{p(v_i = w_{ii})p(v_j = w_{jj})}$$

En cambio, la evaluación de la integral (2) para variables continuas es mucho más costosa. Una posible solución es incorporar un paso de discretización en el preprocesado de los datos pero, en algunos casos, determinar cuál es la discretización adecuada no es tarea fácil pudiendo recurrir entonces a otros métodos de estimación de funciones de densidad como máxima verosimilitud, estimación bayesiana o técnicas no paramétricas.

Así pues, el objetivo de un método de selección de variables basado en información mutua es encontrar un subconjunto S de m variables $S = v_1, \dots, v_m$ que conjuntamente tenga la máxima dependencia con la variable objetivo c ,

$$\max D(S, c), D = I(\{v_i, i = 1, \dots, m\}; c)$$

A pesar de que MaxDep (máxima dependencia entre variables) es un criterio adecuado desde un punto de vista teórico, la estimación de $p(v_i, \dots, v_m)$ y $p(v_1, \dots, v_m, c)$ es prohibitiva por dos motivos:

1. El número de ejemplos de que se dispone suele ser insuficiente y es computacionalmente costosa.
2. Únicamente es posible utilizar MaxDep cuando se desea seleccionar un número pequeño de variables y el número de patrones N es grande;

De esta manera, la insuficiencia de datos para la estimación de las funciones de densidad quedaría cubierta y su cálculo se realizaría en espacios de muy pocas dimensiones donde el coste computacional es asumible.

Como alternativa a MaxDep se puede seleccionar el conjunto de variables basándose en la dependencia con la clase (relevancia). El criterio de Máxima Relevancia, el que nosotros aplicaremos, consiste en encontrar el subconjunto de variables S que maximicen $D(S, c)$, donde D se ha aproximado como el valor medio de la información mutua entre cada una de las variables que componen el subconjunto S y la clase, esto es:

$$D = \frac{1}{|S|} \sum_{v_i \in S} I(v_i; c) \quad (3)$$

Por tanto, el subconjunto seleccionado por el algoritmo serán las m variables con mayor información mutua con la clase c . Este criterio se enmarca dentro de los métodos de filtro descritos

anteriormente. A continuación, explicamos el Criterio de mínima Redundancia y Máxima Relevancia (mRMR).

Es posible que las variables seleccionadas por MaxRel sean bastante redundantes, debido a una gran dependencia entre ellas. Si la dependencia entre dos variables es alta, su poder discriminativo no se ve muy afectado al eliminar una de ellas. Por tanto, no sólo es deseable maximizar la relevancia de las variables con la clase, sino también minimizar la redundancia entre ellas. Con tal fin, el criterio de mínima redundancia se define como:

$$\text{mín}R(S), R = \frac{1}{|S|^2} \sum_{v_i \in S} I(v_i; v_j) \quad (4)$$

El algoritmo mínima redundancia - máxima relevancia (mRMR) combina (3) y (4) de acuerdo a la expresión,

$$\text{máx}\phi(D, R), \phi = D - R \quad (5)$$

Los autores del algoritmo [24] sugieren un método iterativo para encontrar un conjunto de variables cercano al deseado ya que, en la práctica, es inviable tratar de evaluar la expresión (5) para todos los posibles subconjuntos de m variables en el espacio original. Supuesto encontrado el conjunto subóptimo S_{m-1} de $m-1$ variables, la siguiente variable a añadir será aquella que verifique

$$\text{máx}_{v_i \in X \setminus S_{m-1}} \left\{ I(v_j; c) - \frac{1}{m-1} \sum_{v_i \in S_{m-1}} I(v_i; v_j) \right\}$$

En la primera iteración del algoritmo el término de redundancia es nulo, entonces la primera variable seleccionada por mRMR es la misma que la escogida por MaxRel: aquella que tenga la máxima relevancia con la clase.

Bibliografía

- [1] C.-W. Chang, M.-Y. Lin, H.-J. Harn, Y.-C. Harn, C.-H. Chen, K.-H. Tsai, and C.-H. Hwang, “Automatic segmentation of abnormal cell nuclei from microscopic image analysis for cervical cancer screening,” in *Nano/Molecular Medicine and Engineering (NANOMED), 2009 IEEE International Conference on*, pp. 77 –80, oct. 2009.
- [2] M. E. Plissiti, C. Nikou, and A. Charchanti, “Combining shape, texture and intensity features for cell nuclei extraction in pap smear images,” *Pattern Recognition Letters*, vol. 32, no. 6, pp. 838 – 853, 2011.
- [3] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, pp. 971 –987, jul 2002.
- [4] C. Xu and J. Prince, “Snakes, shapes, and gradient vector flow,” *Image Processing, IEEE Transactions on*, vol. 7, pp. 359 –369, mar 1998.
- [5] K. Zhang, L. Zhang, H. Song, and W. Zhou, “Active contours with selective local or global segmentation: A new formulation and level set method,” *Image Vision Comput.*, vol. 28, pp. 668–676, April 2010.
- [6] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2nd ed., 2001.
- [7] L. Nanni, A. Lumini, and S. Brahmam, “Local binary patterns variants as texture descriptors for medical image analysis,” *Artificial Intelligence in Medicine*, vol. 49, no. 2, pp. 117 – 125, 2010.
- [8] B. J. F., “The life and contributions of doctor george nicholas papanicolaou,” *Surgery, gynecology and obstetrics*, vol. 174, no. 6, pp. 530–532, 1992. eng.
- [9] T. Papanicolaou.GN
- [10] Y. Marinakis, M. Marinaki, and G. Dounias, “Particle swarm optimization for pap-smear diagnosis,” *Expert Systems with Applications*, vol. 35, no. 4, pp. 1645–1656, 2008.

- [11] M. A. Duggan, "Papnet-assisted, primary screening of cervico-vaginal smears.," *European Journal of Gynaecological Oncology*, vol. 21, no. 1, pp. 35–42, 2000.
- [12] S. F. Patten, J. S. Lee, D. C. Wilbur, T. A. Bonfiglio, T. J. Colgan, R. M. Richart, H. Cramer, and S. Moinuddin, "The autopap 300 qc system multicenter clinical trials for use in quality control rescreening of cervical smears," *Cancer Cytopathology*, vol. 81, no. 6, pp. 337–342, 1997.
- [13] Y. Marinakis, M. Marinaki, and G. Dounias, "Particle swarm optimization for pap-smear diagnosis," *Expert Syst. Appl.*, vol. 35, pp. 1645–1656, November 2008.
- [14] M. Plissiti, C. Nikou, and A. Charchanti, "Automated detection of cell nuclei in pap smear images using morphological reconstruction and clustering," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 15, pp. 233 –241, march 2011.
- [15] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.
- [16] H.-S. Wu, J. Barba, and J. Gil, "A parametric fitting algorithm for segmentation of cell images," *Biomedical Engineering, IEEE Transactions on*, vol. 45, pp. 400 –407, march 1998.
- [17] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge, UK: Cambridge University Press, March 2000.
- [18] K. Zuiderveld, *Contrast limited adaptive histogram equalization*, pp. 474–485. San Diego, CA, USA: Academic Press Professional, Inc., 1994.
- [19] "A threshold selection method from gray-level histograms," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 9, pp. 62 –66, jan. 1979.
- [20] L. Vincent, "Morphological grayscale reconstruction in image analysis: applications and efficient algorithms," *Image Processing, IEEE Transactions on*, vol. 2, pp. 176 –201, apr 1993.
- [21] E. J. Breen and R. Jones, "Attribute openings, thinnings, and granulometries," *Comput. Vis. Image Underst.*, vol. 64, pp. 377–389, November 1996.
- [22] A. N. Evans, "Morphological gradient operators for colour images," 2004.
- [23] O. Monga, R. Deriche, G. Malandain, and J. P. Cocquerez, "Recursive Filtering and Edge Closing : Two Primary Tools for 3D Edge Detection," in *Proceedings First European Conference on Computer Vision*, (Antibes,France), pp. 56–65, Apr. 1990.
- [24] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 27, pp. 1226 –1238, aug. 2005.

Parte I

Presupuesto

1) Ejecución Material: 1650 €

Desglose por Conceptos:

- Compra de ordenador personal (Software incluido)..... 1.500 €
- Material de oficina..... 150 €

2) Gastos generales

- 16 % sobre Ejecución Material 264 €

3) Beneficio Industrial

- 6 % sobre Ejecución Material 99 €

4) Honorarios Proyecto

- 1500 horas a 18 € / hora 27000 €

5) Material fungible: 280 €

Desglose por Conceptos:

- Gastos de impresión 80 €
- Encuadernación 200 €

6) Presupuesto antes de Impuestos

- Subtotal Presupuesto 29293 €

7) I.V.A. aplicable

- 18 % Subtotal Presupuesto 5273 €

8) Total presupuesto • Total Presupuesto 34566 €

Madrid, Junio de 2012

El Ingeniero Jefe de Proyecto

Fdo.: María Martínez Samboal

Parte II

Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de un sistema de de detección automática de células cancerígenas. En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego. Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

Condiciones generales

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.
2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.
3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que suponvo en los casos de rescisión.
8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.
9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.
10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.
11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea

beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partidaalzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.
2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.
3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.
4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.
5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.
6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.
7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.
8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.

e este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará

conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella.

Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad "Presupuesto de Ejecución de Contrata" y anteriormente llamado "Presupuesto de Ejecución Material" que hoy designa otro concepto.

Condiciones particulares

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.
2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.
3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación, contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora.

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.
5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.
6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.
7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.
8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.
9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.
10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.
11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.
12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.