

INFLUENCIA DE LOS ESPACIOS DE COLOR EN LOS ALGORITMOS DE SEGMENTACIÓN APLICADOS A IMÁGENES MICROSCÓPICAS

comp^{AS}

Christopher Crespo León

Jeniffer Bonilla Bermeo

Yuliana León Bazán

Docentes de la Universidad de Guayaquil

**INFLUENCIA DE LOS ESPACIOS DE COLOR EN
LOS ALGORITMOS DE SEGMENTACIÓN
APLICADOS A IMÁGENES MICROSCÓPICAS**

Autores:

Christopher Crespo León

Jeniffer Bonilla Bermeo

Yuliana León Bazán

Docentes de la Universidad de Guayaquil



INFLUENCIA DE LOS ESPACIOS DE COLOR EN
LOS ALGORITMOS DE SEGMENTACIÓN
APLICADOS A IMÁGENES MICROSCÓPICAS

Autores

Christopher Crespo León
Jeniffer Bonilla Bermeo
Yuliana León Bazán

Primera edición: febrero 2018

Diseño de portada y diagramación:

Grupo Compás

Equipo Editorial

ISBN: 978-9942-770-34-9

Quedan rigurosamente prohibidas, bajo las sanciones en las leyes, la producción o almacenamiento total o parcial de la presente publicación, incluyendo el diseño de la portada, así como la transmisión de la misma por cualquiera de sus medios, tanto si es electrónico, como químico, mecánico, óptico, de grabación o bien de fotocopia, sin la autorización de los titulares del copyright.

Prólogo

Los autores estudian la influencia que tiene la segmentación basada en color en el análisis de imágenes microscópicas. Para el efecto se ha creado una herramienta que permita caracterizar el comportamiento de los algoritmos de segmentación enfocándose en la componente de color como parámetro de evaluación. Estos resultados son valorados analíticamente por medio de un método estadístico basado en el análisis de curvas ROC.

La implementación del aplicativo que realiza la evaluación de la efectividad de los algoritmos de segmentación sobre imágenes microscópicas puede ser descargado del siguiente enlace:

<http://bit.ly/2FPHweR>

La finalidad de compartir este material, es que el lector pueda familiarizarse con técnicas básicas de segmentación, así como la implementación de algoritmos más complejos.

Para alcanzar las metas propuestas el documento se ha organizado de la siguiente manera:

Se describe el proceso de adquisición de imágenes y su almacenamiento en la memoria del computador. Asimismo, se categorizan las diferentes técnicas de segmentación que pueden ser usadas para la detección de objetos. Se estudia la representación de una imagen en diferentes espacios de color, resaltando sus principales características y como estos se relacionan con los dispositivos y la percepción humana. Finalmente se propone como pueden ser aprovechados las componentes de los espacios de color en el proceso de segmentación.

Se trata de la necesidad de realizar una evaluación cuantitativa de los resultados generados por los algoritmos de segmentación en aplicaciones de visión por computador. Se discuten los diferentes tipos de técnicas de evaluación existentes. Se estudian de forma particular las curvas ROC (Receiving Operating Curves) que miden términos de sensibilidad y especificidad el rendimiento de los algoritmos utilizados. Estas métricas son definidas como la proporción de píxeles que pertenecen al objeto de interés detectados correctamente y la proporción de píxeles que no pertenecen al objeto de interés y no forman parte de el resultado

de la segmentación; resultados que idealmente deben coincidir con los píxeles clasificados manualmente.

Se investiga la amplia variedad de aplicaciones en donde interviene el análisis de imágenes microscópicas que son susceptibles a automatización usando técnicas de visión por computador. Se plantean las principales características que deben tener los equipos de análisis y digitalización de muestras. Finalmente se estudia la problemática que interviene durante la automatización del análisis de células de cáncer cervical y se concluye que aunque son útiles para el diagnóstico, la implementación de este tipo de técnicas conlleva un alto grado de complejidad.

Se aplica el modelo de análisis y diseño orientado a objetos, esta metodología permite modelar el sistema antes de ser implementado. En esta etapa se analiza los requerimientos funcionales del sistema y se diseña la solución que permita implementar la aplicación que cubra las expectativas del proyecto. Al término del capítulo se describen los módulos desarrollados después de la etapa de análisis y diseño.

Se describe la implementación del sistema. En la fase de pruebas se realiza un experimento que permite utilizar la herramienta para caracterizar el comportamiento de 2 algoritmos de segmentación ampliamente usados en el análisis de bioimágenes. En esta etapa se probaron 14 espacios de color en donde se analizaron sus componentes de manera aislada o en combinación, estas pruebas fueron realizadas sobre imágenes de células cervico uterinas. Los resultados del experimento se incluyen en una sección adicional. Se muestra el análisis de costos de implementar la infraestructura que permita el estudio de especímenes microscópicos. En la parte final se presentan los resultados y las conclusiones de la utilidad que brinda la herramienta.

Índice

PRÓLOGO	1
SEGMENTACION DE IMAGENES A COLOR	5
FORMACIÓN DE IMÁGENES.....	8
TÉCNICAS DE SEGMENTACIÓN	20
SEGMENTACIÓN BASADA EN EL HISTOGRAMA	21
SEGMENTACIÓN BASADA EN LA DETECCIÓN DE BORDES	23
SEGMENTACIÓN BASADA EN REGIONES.....	30
LA INFLUENCIA DE LOS ESPACIOS DE COLOR EN LOS PROCESOS DE SEGMENTACIÓN	32
MODELOS ORIENTADOS A LOS DISPOSITIVOS.....	36
ESPACIO DE COLOR RGB.....	36
ESPACIOS DE COLOR YIQ, YUV, YCBCR, YCC	37
ESPACIO DE COLOR CMY (K).....	40
MODELOS ORIENTADOS A REPRODUCIR LA PERCEPCIÓN HUMANA	42
ESPACIO DE COLOR HSI.....	43
ESPACIOS DE COLOR CIE	46
ESPACIO CIE XYZ	47
ESPACIO CIE L* U* V*	51
ESPACIO CIE L* A* B*	52
EVALUACION DE ALGORITMOS DE SEGMENTACIÓN	55
IMPORTANCIA DE LA EVALUACIÓN DE LOS ALGORITMOS DE SEGMENTACIÓN.....	55
TIPOS DE TÉCNICAS DE EVALUACIÓN DE ALGORITMOS DE SEGMENTACIÓN	58
MÉTODOS ANALÍTICOS.....	58
MÉTODOS EMPÍRICOS SUPERVISADOS.....	59
MÉTODOS EMPÍRICOS NO SUPERVISADOS	60
EVALUACIÓN BASADA EN EL ANÁLISIS DE CURVAS ROC	61
INTRODUCCIÓN A LAS CURVAS ROC	62
COMPARACIÓN BASADA EN EL ANÁLISIS DE CURVAS ROC	64
INFORMACIÓN CONTENIDA EN LA CURVA.....	67
VENTAJAS DE LAS CURVAS ROC	69
DESVENTAJA DE LAS CURVAS ROC	70
EVALUACIÓN DE LA EFICIENCIA BASADA EN EL ANÁLISIS DE CURVAS ROC	71
ANALISIS DE IMÁGENES MICROSCÓPICAS.....	73
INTRODUCCIÓN.....	73
EL MICROSCOPIO	76
PARTES DEL MICROSCOPIO	78
SISTEMA ÓPTICO.....	78
SISTEMA MECÁNICO	79
TIPOS DE MICROSCOPIOS	80
MICROSCOPIOS ÓPTICOS.....	83
MICROSCOPIOS ELECTRÓNICOS.....	84
MICROSCOPIOS ESTEREOSCÓPICOS	85
TÉCNICAS DE MICROSCOPIA.....	85
CAMPO CLARO.....	86
CAMPO OSCURO.....	87
CONTRASTE DE FASE.....	88
CONTRASTE DE VAREL.....	88
CONTRASTE DE INTERFERENCIA DIFERENCIAL.....	89

CONTRASTE DE POLARIZACIÓN.....	90
FLUORESCENCIA.....	90
DETECCIÓN Y CARACTERIZACIÓN AUTOMÁTICA DE CÉLULAS.....	91
VISIÓN POR COMPUTADOR EN ANÁLISIS DE CÉLULAS.....	92
DIAGNÓSTICO DEL CÁNCER UTERINO.....	97
INTERPRETACIÓN DE LOS RESULTADOS DE LA PRUEBA PAP.....	99
SISTEMA DE ADQUISICIÓN.....	103
MICROSCOPIO.....	103
CÁMARA Y SOFTWARE.....	104
ANÁLISIS Y DISEÑO DE HERRAMIENTA PARA EVALUAR ALGORITMOS DE SEGMENTACION.....	107
ESPECIFICACIÓN DE LOS REQUERIMIENTOS Y ALCANCE DEL SISTEMA.....	107
REQUERIMIENTOS FUNCIONALES.....	108
REQUERIMIENTOS NO FUNCIONALES.....	109
USUARIOS DEL SISTEMA.....	110
ANÁLISIS DE APLICACIONES Y HERRAMIENTAS DE DESARROLLO.....	112
ESPECIFICACIONES DE CASOS Y SUBCASOS DE USO.....	115
ESPECIFICACIÓN DE CASOS DE USO.....	116
ESPECIFICACIÓN DE SUBCASOS DE USO.....	122
DISEÑO LÓGICO DEL SISTEMA.....	131
MÓDULO DE GENERACIÓN DE PLANTILLAS.....	132
MÓDULO DE SEGMENTACIÓN SEMI-AUTOMÁTICA.....	135
MÓDULO DE EVALUACIÓN.....	137
DIAGRAMA DE SECUENCIA DEL SISTEMA.....	138
IMPLEMENTACION Y PRUEBAS DE LA HERRAMIENTA.....	145
INTRODUCCIÓN.....	145
PROCESO DE IMPLEMENTACIÓN.....	147
PROBLEMAS PRESENTADOS EN LA IMPLEMENTACIÓN.....	158
PRUEBAS REALIZADAS.....	160
METODOLOGÍA.....	161
ALGORITMOS UTILIZADOS.....	163
TIPOS DE IMÁGENES ANALIZADAS.....	171
RESULTADOS DE LOS EXPERIMENTOS.....	173
ANÁLISIS DE COSTOS.....	178
RESULTADOS ESPERADOS DE LOS ALGORITMOS DE SEGMENTACIÓN EVALUADOS	

Capítulo 1

CAPITULO I

SEGMENTACION DE IMAGENES A COLOR

La segmentación es el proceso que divide una imagen en regiones cuyos píxeles poseen atributos similares, por ejemplo, intensidad, textura, movimiento etc. Idealmente cada región segmentada corresponde a un objeto que se encuentre en la escena. En este capítulo se analiza la importancia que tienen los espacios de color en la segmentación de imágenes, para conseguir este objetivo se estudia el proceso de adquisición de imágenes, se explica como es almacenada la imagen en la memoria del computador antes de que se le proceda a aplicar un algoritmo de visión por computador. Luego se categorizan las diferentes técnicas de segmentación que pueden

ser usadas para la detección de objetos. Finalmente se estudia la representación de una imagen en diferentes espacios de color y como puede ser aprovechada por los algoritmos de segmentación para mejorar sus resultados.

La segmentación es uno de los componentes más importantes de un sistema automatizado de visión ya que permite extraer los objetos de la imagen para su posterior descripción y reconocimiento. Basándonos en este concepto tenemos dos tipos de segmentación: *total* y *parcial*. La primera asume una segmentación ideal en donde todas las regiones encontradas tienen una interpretación física con los objetos dentro de la imagen (ver Figura 1.1), la segunda y la más común, asume que son regiones localizadas que pueden o no formar parte de los objetos de interés.

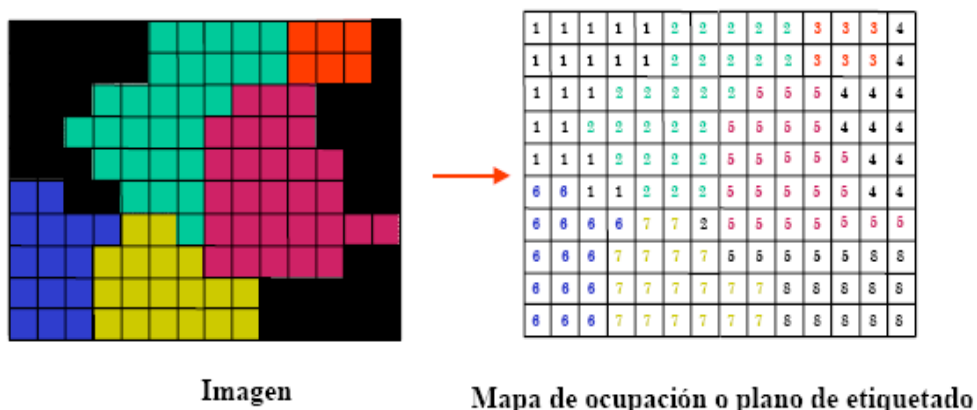


Figura 1.1.- Mapa de ocupación de regiones: Cada región en la imagen esta asociado con una etiqueta numérica.

Para lograr una segmentación completa se necesita aplicar procesamiento de alto nivel y tener un amplio conocimiento sobre el dominio del problema, por el contrario una segmentación parcial puede lograrse sin necesidad de ejecutar algoritmos complejos que demanden demasiado tiempo de procesador.

El factor fundamental en la segmentación de imágenes es la imagen como tal. En muchas de las aplicaciones de visión artificial se emplean imágenes en escala de grises, que capturan solo la intensidad de luz, el color se lo descarta y se trabaja sobre las diferencias en intensidad de los objetos, este proceso disminuye el volumen de información y el coste de procesamiento (ver Figura 1.7 (b) y 1.7 (d)). Entre las razones por las cuales los sistemas basados en el color son menos utilizados están: mayor volumen de datos (normalmente 3 veces más), encarecimiento del sistema (tanto en la cámara como en el digitalizador) y, que en muchas ocasiones, el color no aporta ninguna información relevante. Por ejemplo, para reconocer un rostro no es necesario la información del color ver Fig. 1.2. No obstante en determinadas aplicaciones los niveles de grises no son suficientes para solventar problemas que involucran directamente el color de los objetos. Un ejemplo típico donde el color es fundamental es cuando

se pretende clasificar frutas entre verdes y maduras; También son muy usadas en el análisis de imágenes médicas e imágenes microscópicas [23] [24], puesto que diferentes objetos dentro de este tipo de imágenes toman una coloración determinada dependiendo del tipo de tinsión usada en el análisis clínico.



Figura 1.2.- Imágenes en escala de grises, reconocer un rostro es posible sin necesidad del color.

Formación de Imágenes

El concepto de imagen está asociado a una función bidimensional $f(x, y)$, cuya amplitud es el grado de iluminación (intensidad de la luz) en el espacio de coordenadas (x, y) de la imagen para cada punto. El valor de esta función depende de la cantidad de luz que incide sobre la escena vista, y de la parte que sea reflejada y absorbida por los objetos que la componen.

La función de imagen es digitalizada en la memoria del computador, tanto espacialmente como en amplitud. La digitalización de las coordenadas espaciales x, y está asociada al concepto de muestreo,

mientras que la digitalización de la amplitud al de cuantificación de los niveles de gris.

El muestreo es la conversión que sufren las dos dimensiones espaciales de la señal analógica, y que genera la noción de píxel. La imagen en la memoria del computador se comporta como una matriz bidimensional donde el valor de los elementos de esta matriz corresponde a su intensidad, y la fila y la columna su posición en el espacio. La cuantificación es la conversión que sufre la amplitud de la señal analógica; así se genera el concepto de nivel de gris o intensidad. Para el caso de tener 256 niveles de gris (0,255), el 0 corresponde a un objeto no iluminado o que absorbe todos los rayos luminosos que inciden sobre él (negro), y el nivel 255 a un objeto muy iluminado o que refleja todos los rayos que inciden sobre él (blanco).

En el proceso de digitalización es preciso establecer el valor de M y N correspondientes al número de filas y columnas obtenidos durante el proceso de muestreo, así como el número niveles de gris asignados para cada píxel, estos valores son números enteros potencias de 2. En consecuencia el número de bits para almacenar la imagen digitalizada viene dado por Ec. 1.1, siendo 2^b el número de niveles de gris.

$$\text{num_bits_de_la_imagen} = N \times M \times b \quad \text{Ec. 1.1}$$

Uno de los conceptos más comunes en procesamiento de imágenes es el de resolución. El cual se asocia al número de muestras (cantidad de píxeles) y niveles de gris necesarios para tener una buena aproximación de la imagen. Es interesante apreciar el efecto de las variaciones de los parámetros citados sobre la calidad de una misma imagen, ver Figura.1.3, y la dificultad de establecer el valor óptimo de cada uno de ellos para cumplir con los requerimientos (calidad, tamaño, etc.) en diferentes aplicaciones. Si la imagen posee detalles de poca resolución, la cuantificación y muestreo estarán íntimamente ligados.

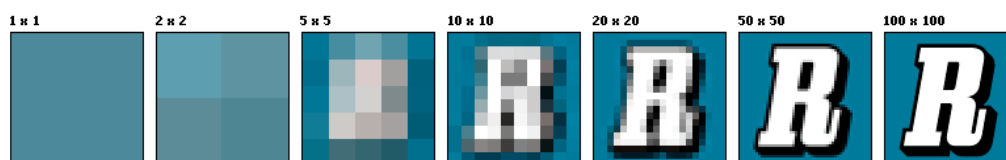


Figura 1.3.- Imagen obtenida a diferentes resoluciones.

La obtención de parámetros que nos permitan tener una alta calidad de imagen depende mayormente de la capacidad que tienen los equipos al momento de adquisición. En la actualidad los principales proveedores de equipos de digitalización como cámaras y videograbadoras resaltan la resolución y la cantidad de bits que

pueden estar almacenados en un píxel como algunas de sus principales características. Todo esto gracias a los componentes electrónico-ópticos que forman parte de ellos. Desde esta perspectiva, la digitalización de imágenes se entiende como el proceso de convertir una imagen proveniente del mundo 3D a una imagen de dos dimensiones.

De este modo en una cámara digital, Figura. 1.4, la luz penetra a través de la lente y llega al sensor en mayor o menor cantidad dependiendo de la velocidad de obturación y la apertura del diafragma que se haya seleccionado durante la adquisición.

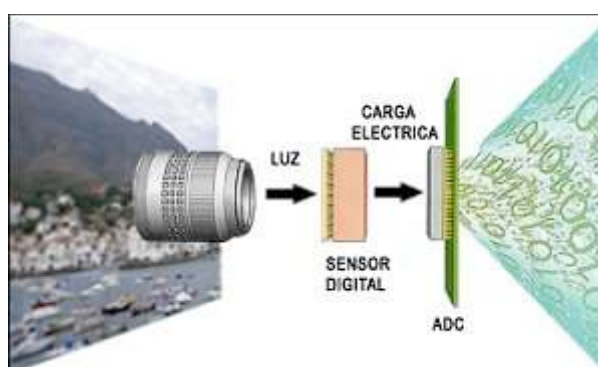


Figura 1.4.- Etapa de adquisición y digitalización de la imagen.

El sensor digital reacciona ante la luz generando una carga eléctrica y luego el convertidor analógico digital (ADC) se encarga de digitalizar esta señal. El sensor es una matriz formada por pequeños diodos de silicio; un material sensible a la luz, en el cual la luz incidente produce

una determinada carga eléctrica proporcional a su nivel de intensidad en cada uno de estos fotosensores. En el caso de adquirir una imagen en escala de gris la cámara utilizaría un solo sensor durante la captura de la imagen, por el contrario en imágenes a color se necesitaría usar tres sensores.

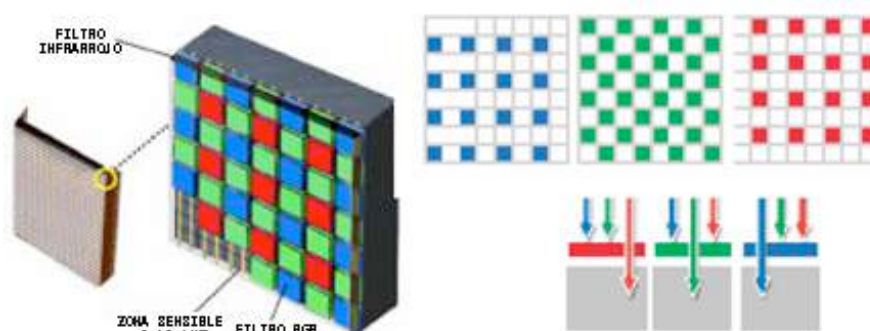


Figura 1.5.- Convertidor analógico digital – ADC para imágenes a color.

En la Figura.1.5 vemos un fotosensor a color el cual captura la información de tres longitudes de onda (llamados también canales) provenientes de la luz, los cuales corresponden a los colores primarios rojo, verde y azul, (RGB por sus siglas en ingles), gracias a la combinación de estos tres canales es posible obtener todos los colores capaces de ser representados por los dispositivos de digitalización (más información en la sección 1.4). La información de cada canal es almacenada en una matriz independiente, análogamente a la forma de cómo la imagen obtenida es almacenada en la memoria del computador.

Durante esta primera fase del proceso, donde la imagen es muestreada sobre el sensor, se determina la resolución espacial de la imagen. Pero para poder hablar de imagen digital es necesario convertir esta señal eléctrica -analógica en un código binario. De esta tarea se encarga el ADC. A partir de la información eléctrica, a cada píxel se le otorga un valor formado por ceros y unos que luego será interpretado como un color determinado en cada uno de los canales RGB de la imagen.

La profundidad de color es la que determina el número de cifras que podrá tener este código binario (valores de 0 y 1) y, por tanto, la cantidad de valores (colores) que tendrá la imagen.



Figura 1.6.- De arriba abajo y de izquierda a derecha, esta secuencia se compone de imágenes de 1, 8, 24 y 48 bits de profundidad de color.

Así, un ADC de 1 bit, sólo podría asignar dos valores: 0 ó 1, presencia de luz o ausencia de ella. Un ADC de 8 bits genera valores de 8 cifras, por lo que combinando ceros y unos podemos obtener un total de 256 resultados. 256 valores que corresponde al rango de las imágenes en escala de grises. No obstante, como estamos hablando de imagen en color, los dispositivos digitales vienen equipados con conversores de, por lo menos, 24 bits, esto es, 8 bits para cada canal. Con esta combinación de 24 bits, el resultado es de algo más de 16 millones de posibilidades con las que se puede representar los colores en diferentes tonalidades.

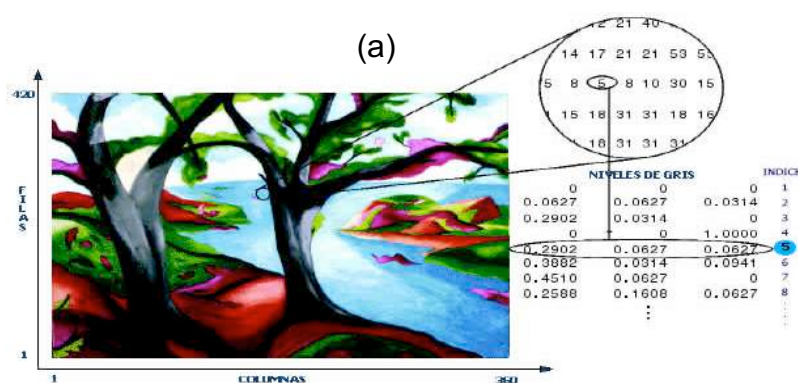
Existen cámaras y escáneres que trabajan a 12 ó 16 bits por canal (36 ó 48 bits). Mediante esta información extra contenida en cada píxel se consigue una imagen que resuelve mejor las zonas más complejas, por ejemplo en imágenes que tengan alto nivel de detalle como paisajes, fotos satelitales y permite obtener mejores resultados sobre todo al magnificar la imagen.

La imagen adquirida puede ser cargada en la memoria del computador de cuatro diferentes formas (Figura 1.7): en escala de gris, indexada, binaria, y en canales de color. A continuación

explicamos los 3 primeros tipos y dejamos las imágenes a color para la sección 1.4.

Las imágenes en niveles de grises y las imágenes binarias son almacenadas como una matriz bidimensional cuyas celdas representan intensidades (niveles de gris) por lo general de 0 a 255 o sólo dos posibles valores (0 y 1) respectivamente.

La imagen indexada es almacenada como 2 matrices, una que almacena información espacial y otra que almacena información del color, esta última es conocida como paleta de colores o mapa de colores, la cual puede ser predefinida por el usuario. Cada celda de la matriz contiene el índice correspondiente a un color en la paleta. Este esquema de almacenamiento permite ahorrar espacio al momento de cargar la imagen en la memoria del computador.



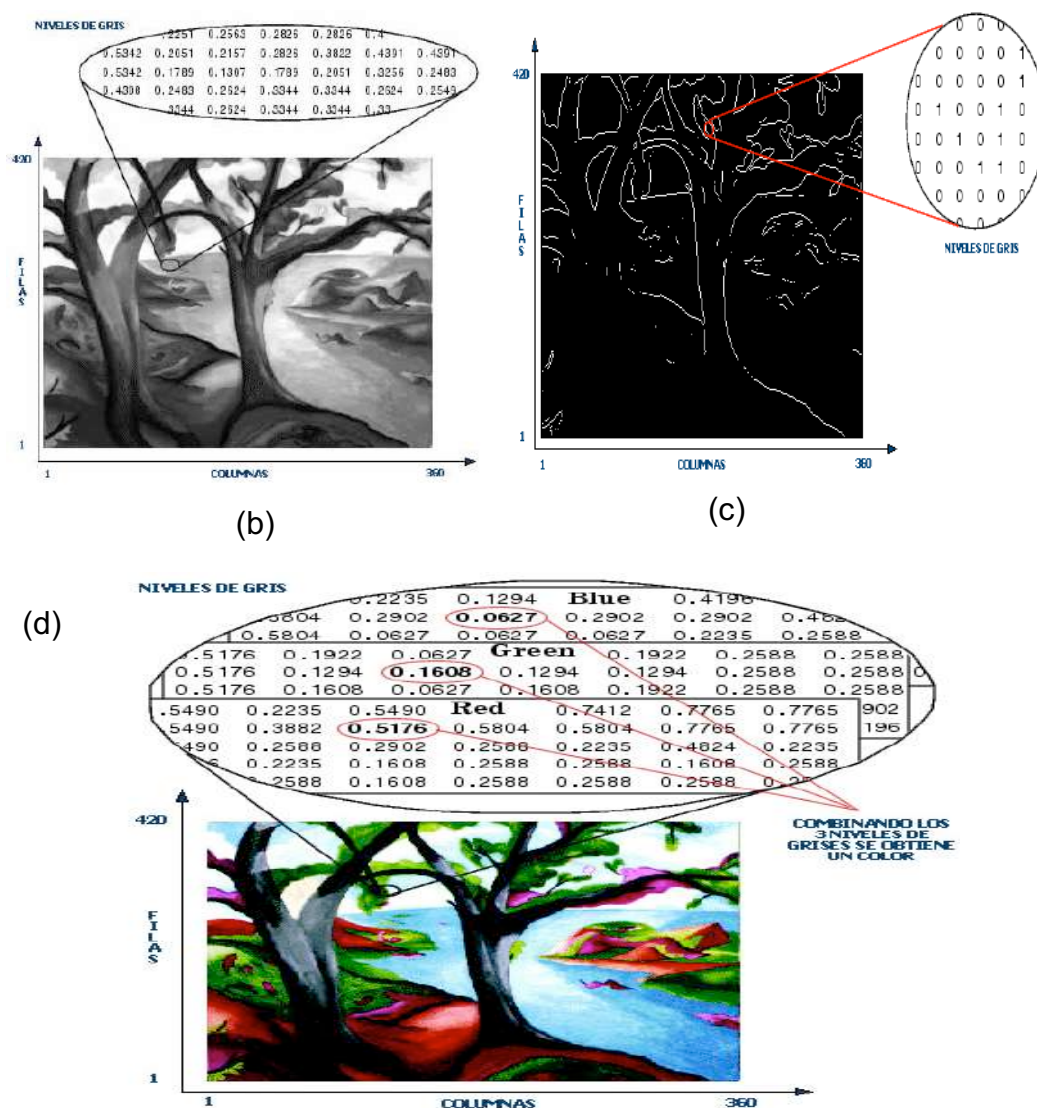


Figura 1.7.- Forma de almacenamiento de Imágenes: (a) Imagen Indexada; (b) Imagen en escala de gris; (c) Imagen binaria; (d) Imagen a color.

La información contenida en cada píxel puede ser representada por el computador en formato entero (valores discretos) o formato flotante (valores se encuentran en el rango de 0 a 1).

Una vez digitalizada la imagen, esta es almacenada en disco asignándole un formato de imagen, el cual posee una estructura que

permite guardar la imagen en forma legible para el programa o tipo de programa que la genero.

Existen muchos formatos de imagen tales como: BMP, GIF, TIFF, JPEG, PCX, IFF etc., que reflejan el desarrollo histórico de la informática (en *hardware* y en *software*) durante las últimas décadas.

Formatos como el PCX, o GIF fueron creados por las mismas empresas que desarrollaron los programas utilizados para trabajar con gráficos, otros entre los que contamos el **TIFF**, **JPEG**, **PNG**, han surgido del intento de normalizar el formato de las imágenes.

A pesar de esta "normalización", nos encontramos rodeados en infinidad de formatos de imagen distintos que son incompatibles entre sí. En algunas ocasiones, las imágenes sólo podrán ser abiertas con algunos programas específicos. Casi todos los formatos gráficos pueden soportar algún tipo de compresión, (técnica que permite mediante procesos y algoritmos matemáticos, reducir los tamaños de archivos para facilitar su transferencia o almacenamiento). En [16] se mencionan las distintas técnicas de compresión comúnmente usadas en imágenes. En la tabla 1.1 se resumen los principales formatos gráficos usados en la actualidad.

FORMATOS DE IMAGEN				
Formato	Profundidad de Color	Modos Color	comprim e	comentario
BMP (.bmp)	-1 (Mapa de bits) -4-8 bits (Escala grises) -8 bits (Color Indexado) -24 bits(RGB)	-Modos RGB -Color Indexado, -Escala de Grises, -Mapa de bits	NO (excepto en 4 y 8 bits)	-Formato estándar IBM PC, Uso: fondo escritorio, o imágenes sencillas de hasta 256 colores
GIF <i>Graphics Interchange Format</i> (.gif)	·8 bits (256 colores)	<i>Compuserve GIF:</i> -Mapa de bits -Escala de grises. -Color indexado <i>GIF89a (GIF Animado):</i> -Color Indexado -RGB	SI (LZW)	-Creado por Compuserve en 1987 -Gráficos color indexado -Posibilidad de visualización entrelazada (aparición gradual) -Transparencia y · Animación Uso: Internet
PICT (.pct; .pic)	-RGB: 16/32 bits -Escala Grises: 2,4, 8 bits	-Mapa de bits (sin canales alfa) -Escala de grises -Color Indexado -RGB (1 canal)	SI (sin pérdidas) (con <i>QuickTime</i> e: 4 opciones para JPEG)	-Estándar de Macintosh -Bueno para compresión imágenes con área color sólido Uso: Transferencia de gráficos entre aplicaciones
JPEG <i>Joint Photographic Expert Group</i> (.jpg; .jpe)	24 bits	-Escala de grises -RGB -CMYK	SI: DIVERSAS CALIDADES (con pérdidas)	-Junto con GIF y PNG el formato de Internet para gráficos y fotografías. -Formato de color verdadero en el que no se produce pérdida de color, aunque si se comprime se eliminan datos. Uso: fotografías Internet

FORMATOS DE IMAGEN				
Formato	Profundidad de Color	Modos Color	comprime	comentario
PHOTOSHOP (.psd)	32 bits	-Admite todos los Modos de Color -Canal Alfa y de Tintas Planas -Guías, trazados -Capas de ajuste, de texto, efectos capa	NO	-Propio de <i>Adobe Photoshop</i> -Guarda capas y selecciones (canales) Uso: Creación y Tratamiento Imagen
TARGA (.tga; .vda;.icb;.vst)	16, 24 y 32 bits	-Escala Grises -Color indexado -RGB (16 y 24 bits sin canales alfa); -RGB de 32 bits (un solo canal alfa)	NO	Uso: Exportación a edición profesional Video
PNG <i>Portable Networks Graphics</i> (.png)	24 bits	-Mapa de bits -Escala Grises, -Color Indexado, -RGB	SI (sin pérdidas)	-Mayor capacidad de almacenamiento y capacidades que el GIF -Genera transparencias de fondo sin bordes dentados -No muy extendido, con el tiempo sustituirá al GIF Uso: Internet
TIFF <i>Tag Image File Format</i> (.tif)	32 bits	-Mapa de bits sin canales Alfa -Escala de Grises con canales Alfa y archivos Lab	SI (LZW) (Se puede especificar si para IBMPC o Mac)	-Desarrollado por <i>Aldus Corporation</i> . -Reconocido por casi todos los programas de pintura y vectorización. -Compatible IBM PC y Mac.

		-Color Indexado -RGB con canales Alfa y archivos Lab ,CMYK		Uso: Imprenta e intercambio de archivos
Tabla 1.1 Formato de imágenes mas usados en la actualidad.				

Técnicas de Segmentación

La segmentación se basa en el estudio de tres propiedades principales sobre la imagen:

Similitud: Cada uno de los píxeles de un elemento de la imagen tiene valores parecidos para alguna propiedad como intensidad, color, textura, movimiento, etc.

Discontinuidad: Los objetos destacan del entorno y tienen bordes definidos.

Conectividad: Los píxeles pertenecientes al mismo objeto tienden a ser contiguos.

Basada en estas premisas podemos clasificar a los algoritmos de segmentación en tres grandes grupos:

- Segmentación basada en el Histograma
- Segmentación basada en detección de bordes
- Segmentación basada en crecimiento de regiones.

Por lo general durante la adquisición aparecen pequeñas entidades no deseables en la imagen, muchas veces no perceptibles a simple vista y que son detectados como objetos durante el proceso de segmentación. Comúnmente se las conoce como ruido y afectan el resultado de la segmentación. Es por ese motivo que muchas veces antes de realizar la segmentación en imágenes es preciso realizar pre-procesamiento para remover variaciones de intensidad indeseadas.

Segmentación basada en el Histograma

Estos métodos asocian cada píxel de la imagen a un determinado objeto según su nivel de gris o alguna propiedad derivada de ellos (varianza, media, etc.). Para aplicar esta técnica, es necesario determinar un umbral o un rango de umbrales en donde se encuentre agrupado la mayor cantidad de píxeles que posean la misma propiedad. El cual ayude a discriminar entre un objeto de otro. Una herramienta que permite visualizar de una manera sencilla la distribución de los valores de una determinada propiedad es el histograma. A la técnica que permite clasificar cada tipo de objeto basando en la asignación de umbrales sobre el histograma se la denomina umbralización. La asignación de umbral o rango de umbrales puede ser empíricamente fijada por el programador o

puede ser obtenida automáticamente usando técnicas basadas en el análisis de histogramas.

En los histogramas de la Figura. 1.8 se puede deducir que la zona donde están agrupados la mayor cantidad de píxeles que cumplan una determinada propiedad, pueden ser clasificados como un tipo de objeto. Basados en esta premisa se puede plantear que un tipo de objeto vendrá representado por un máximo local (moda) en el histograma. En consecuencia el proceso de segmentación se convierte en la búsqueda de picos en el histograma.

En su forma más simple la umbralización convierte una imagen con varios niveles de gris a una nueva con solo dos (valores de uno o cero), de manera que los objetos quedan discriminados del fondo (método simple). Sin embargo existen algoritmos que permiten definir rangos de umbrales (umbralización multinivel) los cuales permiten clasificar cada tipo de objeto relacionándolo con un rango específico, un ejemplo de esto es el algoritmo de Meanshift [32]. Generalmente durante esta clasificación cada píxel que pertenezca a un objeto es etiquetado con un valor numérico. Si bien la implantación de estos métodos suele llevar a soluciones relativamente rápidas y sencillas, adolecen el problema fundamental de ignorar las

nociones de proximidad y conectividad, suponiendo que los objetos se distinguen únicamente por su valor de intensidad o el de alguna otra propiedad evaluada.

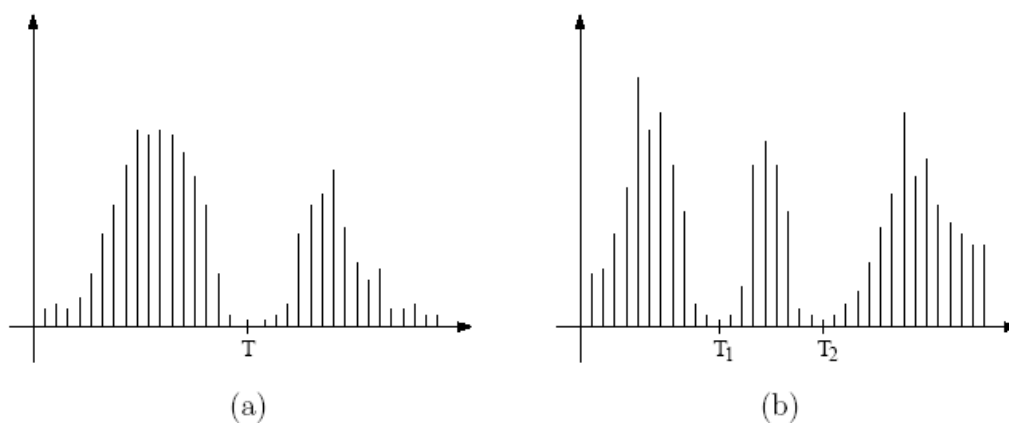


Figura. 1.8. Histogramas de nivel de gris que se pueden segmentar. (a) Con un único umbral T y (b) con múltiples umbrales T_1, T_2

Los métodos de umbralización son categorizados de acuerdo a la información que ellos tratan de aprovechar [1]: forma del histograma, agrupamientos de clases, entropía, atributos de los objetos, correlación espacial y análisis local de los niveles de gris.

Segmentación basada en la detección de bordes

Estos métodos se basan en suponer que existe una correspondencia entre las discontinuidades en intensidad de una imagen y los contornos de los objetos contenidos en la escena. El principal objetivo de estos algoritmos es la detección de píxeles (bordes) que

pertenezcan a un mismo contorno. La definición de contorno se basa en la percepción humana de que todos los objetos que se encuentren en la imagen están delimitados por píxeles contiguos que los cercan. Si bien los contornos de objetos suelen corresponder a gradientes de intensidad, existen muchos otros cambios de intensidad que no guardan relación con la interacción entre ellos sino a variaciones provocadas por la textura, efectos de iluminación y sombras, o porque el color de la superficie no es uniforme. En cualquier caso, se puede decir que los bordes detectados basados en el análisis de discontinuidades proveen una buena segmentación inicial. Que en algunos casos será insuficiente y en otros detectará una cantidad excesiva de posibles contornos, requiriendo por tanto algún tipo de procesamiento posterior.

La asignación de píxeles como contorno no se realiza de manera directa sino que primero se determina si un píxel pertenece o no a un contorno y en una fase posterior se agrupan y asignan. Para ello, definimos el borde como la frontera entre dos regiones con nivel de gris relativamente diferente, y suponemos que las regiones de interés son suficientemente homogéneas de modo que la transición entre dichas regiones se puede determinar empleando exclusivamente las

discontinuidades en el nivel de gris, para el caso de imágenes sin color. Esto se consigue aplicando operadores derivativos como: la derivada de primer orden (gradiente) que tiene su máximo local en la dirección del contorno y la derivada de segundo orden (laplaciana) que tiene información del contorno en sus cruces por cero.

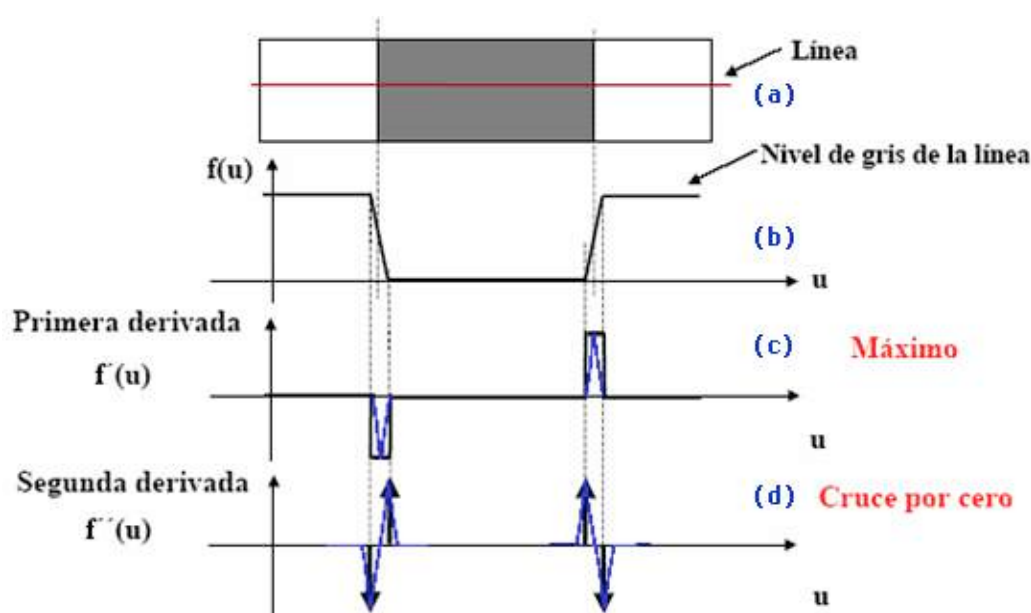


Figura. 1.9. Detección de bordes empleando operadores derivativos. (a) Extracción del perfil horizontal (b) Gráfica del perfil horizontal. (c) Primera derivada (d) Segunda derivada.

En la Figura. 1.9 se ilustra el proceso de detección de bordes empleando operadores derivativos, en la Figura. 1.9 (a) se extrae el perfil horizontal de una región de la imagen y se obtiene una gráfica lineal de las variaciones de los niveles de grises en los píxeles de esa región, ver Figura. 1.9 (b). A partir de esta gráfica se puede obtener los bordes de la imagen aplicando la primera y la segunda derivada. Los

bordes detectados coinciden con los máximos locales obtenidos en la primera derivada (Figura. 1.9 (c)), mientras en la segunda derivada son obtenidos en los cruces por cero (Figura. 1.9 (d)).

Sea una imagen continua $I(x,y)$ como se ilustra en la Figura.1.10, su derivada asume un máximo local a lo largo del contorno. Por lo tanto, una forma de detectar contornos consiste en medir el gradiente en una dirección r de la imagen I :

$$\frac{\partial I}{\partial r} = \frac{\partial I}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial I}{\partial y} \frac{\partial y}{\partial r} = I_x \cos\theta + I_y \operatorname{sen}\theta \quad \text{Ec. 1.2}$$

Donde θ es la dirección tangente al contorno y los parámetros I_x e I_y representan los gradientes ortogonales correspondientes a un sistema cartesiano.

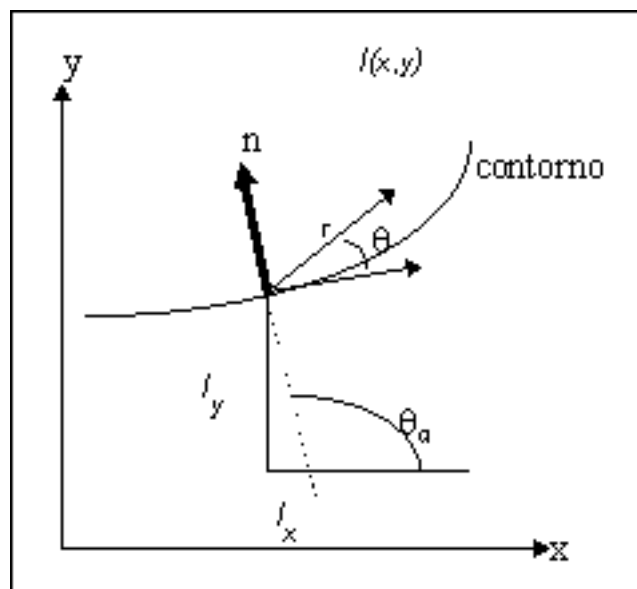


Figura.1.10 Gradiente $I(x,y)$

El valor máximo de la función $\partial I / \partial r$ en un punto (x,y) surge cuando:

$$(\partial I / \partial \theta) (\partial I / \partial r) = 0 \quad \text{Ec. 1.3}$$

Entonces:

$$-I_x \operatorname{sen} \theta_g + I_y \operatorname{cos} \theta_g = 0 \Rightarrow \theta_g = \tan^{-1} \left(\frac{I_x}{I_y} \right) \quad \text{Ec. 1.4}$$

$$\left(\frac{\partial I_x}{\partial I_y} \right)_{\max} = \sqrt{I_x^2 + I_y^2} \quad \text{Ec. 1.5}$$

Donde θ_g es la dirección tangencial al contorno de gradiente máximo. Basándose en estos conceptos se distinguen dos tipos de operadores de detección de contornos: los *operadores de gradiente* y los *operadores de compás*. Los primeros estiman el gradiente en dos direcciones ortogonales (por ejemplo, I_x e I_y) y los emplea para hallar el módulo y fase del gradiente según las ecuaciones (Ec.1.4 y Ec.1.5), mientras que los segundos miden el módulo del gradiente en un conjunto de direcciones y seleccionan el de mayor valor para cada punto.

Los operadores basados en el gradiente funcionan relativamente bien cuando existen variaciones abruptas en la imagen pero resultan incapaces de detectar variaciones graduales en la misma. En este segundo caso, es necesario recurrir al uso de los operadores basados en detectar los cruces por cero de la derivada de segundo orden,

siendo muy común el operador Laplaciano (Ec.1.6). La mayor sensibilidad en la detección de contornos que proporciona este método induce a su vez una mayor sensibilidad al ruido. Nótese que a diferencia del gradiente, la laplaciana no es un vector.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad \text{Ec. 1.6}$$

Para imágenes digitales, los operadores mencionados suelen llevarse a cabo mediante el empleo de máscaras de convolución (Ec.1.6) ya que estas pueden ser implementadas de manera mucho más eficiente que empleando su contrapartida analítica [22]. De esta manera sea **H** una máscara de tamaño $p \times p$ y **U** una imagen arbitraria, la aplicación de la máscara de convolución se obtiene mediante el producto interno de ambos en una localización (m,n) , siendo éste equivalente a la siguiente correlación:

$$\langle U, H \rangle_{m,n} = \sum_i \sum_j h(i, j) u(i + m, j + n) = u(m, n) * h(-m, -n) \quad \text{Ec. 1.7}$$

En la Figura.1.11 se muestra la relación de diversas máscaras comúnmente empleadas como aproximaciones discretas del operador gradiente.

Conjunto de Máscaras	H_1	H_2
Roberts	$\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Prewitt	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
Sobel	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

Figura.1.11 Máscaras comúnmente empleadas como operador gradiente.

Una vez obtenido el módulo del gradiente en cada píxel (imagen gradiente) suele aplicarse una operación de umbralización, donde se decide si el píxel pertenece o no a un borde de la imagen (imagen binaria de bordes). La elección del umbral es crítica: un umbral pequeño detectaría como borde fluctuaciones debidas al ruido y daría lugar a respuestas múltiples (borde de espesor mayor que un píxel); por el contrario, un umbral elevado no detectaría muchos de los bordes en la imagen. Lo mas adecuado, por tanto, sería emplear un umbral variable en cada píxel, en función del nivel de ruido y de la amplitud de la discontinuidad (relación señal-ruido). En la práctica, esto presenta serios inconvenientes que desaconsejan su utilización, como el elevado coste computacional que acarrea y la necesidad de contar con un adecuado modelo probabilístico.

La dirección del gradiente es perpendicular al borde en dicho punto y es utilizado en posteriores etapas en el proceso de visión (por ejemplo, para la segmentación de objetos mediante seguimiento de contornos y transformada de Hough).

Segmentación basada en regiones

Estos métodos explotan la noción de conectividad y permiten agrupar zonas de la imagen con puntos de intensidad similar en entidades discernidas (cada objeto es independiente del resto así sea de su misma clase). Con objeto de eliminar los efectos del ruido, se han desarrollado métodos iterativos orientados a unir entidades pequeñas en otras más grandes de valor medio de intensidad similar. Entre dichos métodos cabe destacar el algoritmo “*split and merge*” y sus variantes piramidales [17]. También cabe destacar los algoritmos basados en *programación dinámica* [18], la reciente aplicación al problema de la teoría de *campos aleatorios de markov* [19] y el uso de la transformada “*Watershed*” [20] dentro del ámbito de la *morfología matemática* [20].

El mayor inconveniente de estos métodos consiste en que son iterativos o recursivos, dificultando su implantación en sistemas que

requieren un tiempo de respuesta rápido [21]. También existe una pequeña dificultad a la hora de establecer el criterio por el cual una región pequeña es solapada por otra grande, exigiendo un compromiso entre la eliminación de ruido y la pérdida de detalle. Otra dificultad es la de determinar con precisión la frontera entre regiones cuando las variaciones de intensidad son graduales, produciendo a veces un efecto de ambigüedad [21] en los resultados.

El resultado de la segmentación depende que tan bien definidos se encuentren los criterios de similitud, discontinuidad y conectividad de los objetos que se encuentren en la imagen. En consecuencia, estos criterios también son válidos en el análisis de imágenes en color, debido a que gracias al color se puede discriminar un objeto de otro, además el contraste en color que existe entre ellos permite resaltar sus contornos, y a su vez los agrupa de tal manera que objetos de una misma clase posean valores similares en color. Es por esto que la segmentación de imágenes a color utiliza las mismas técnicas de segmentación de imágenes de escala de grises aplicadas sobre los diferentes componentes de espacios de color [25]. La idea es tratar de aprovechar las propiedades que tienen los espacios de color de

realzar algún aspecto (cromaticidad, intensidad) de la imagen, para ser usadas en la etapa de segmentación.

La influencia de los espacios de color en los procesos de segmentación

Podemos definir el color como una representación matemática de lo que percibimos visualmente. El proceso de ver es una interpretación que realiza el cerebro, a partir de la recepción de frecuencias electromagnéticas (espectro visible) en la retina, esta a su vez esta compuesta por receptores visuales llamados conos y bastones que son los responsables de percibir una imagen. Los conos son sensibles a tres longitudes de onda diferentes, rojo, verde y azul, las posibles combinaciones de ondas nos permiten distinguir entre millones de colores, por el contrario los bastones nos dan información sobre la figura o forma de las cosas, estos receptores solo ven en blanco y negro, y destacan sombras. Un ejemplo claro de apreciar la función que desempeñan los conos y bastones, se da cuando salimos de un cuarto oscuro hacia una habitación con luz, notamos que la imagen no se forma inmediatamente, esto se debe a que los conos tratan de graduar lentamente la cantidad de luz recibida, antes de que esta sea interpretada por el cerebro. Estos tres colores (rojo, verde, azul)

capaces de ser percibidos por los conos son denominados colores primarios.

Antes de definir los espacios de color es recomendable tomar en cuenta ciertos términos que están altamente relacionados con el concepto de color y que son ampliamente usados para describir la pureza de este. Estos términos se describen a continuación:

Brillo: Sensación que indica si un área está más o menos iluminada

Tono: Sensación que indica si un área parece similar al rojo, amarillo, verde o azul o a una proporción de dos de ellos.

Luminosidad o Intensidad: Brillo de una zona respecto a otra blanca de la imagen.

Croma: El color de un área respecto al brillo de un blanco de referencia

Saturación: Indica la concentración de color en el objeto.

Cromaticidad: El color es representado en términos de tono y saturación.

Basados en el modelo de la visión humana se han creado los diferentes espacios de color los cuales son clasificados en: *modelos orientados a los dispositivos* y *modelos orientados a reproducir la*

percepción humana. Los primeros están basados en la fisiología de la retina humana por ejemplo RGB, CMY, YIQ, YUV, YCbCr, YCC, y los otros se concentran en los parámetros síquicos de la percepción de los colores, como la luminosidad, el tono y la saturación; por ejemplo HSV y CIE.

Matemáticamente, el espacio de color se define por una base de N componentes vectoriales, por ejemplo, el espacio RGB esta formado por 3 componentes vectoriales: Rojo (R), Verde (G) y Azul (B), cuya combinación lineal genera todo el espacio de color y en consecuencia todos los posibles colores que pueden ser representados por ese espacio.

Algunos ejemplos en donde se emplea la información de color como parte de la segmentación lo encontramos en [23] y [25] que utiliza la información de las componentes de los espacios de color para la segmentación de imágenes biológicas, mientras que en [26] y [27] se combinan las componentes para crear un espacio de color híbrido, que permita segmentar objetos que tengan regiones uniformes.

Esta nueva tendencia de aprovechar la información de color se debe en gran medida al gran desarrollo que han tenido las computadoras

personales, en cuanto a velocidad de procesamiento y capacidad de almacenamiento se refiere y la disminución de costo de las cámaras digitales.

Modelos orientados a los dispositivos

Estos espacios de color son utilizados en dispositivos como: cámaras fotográficas, monitores, impresoras, etc. Pero tuvieron su origen en la transmisión de señales de televisión.

Espacio de color RGB

En este espacio se representa un color mediante la mezcla por adición de los tres colores de luz primarios: rojo (R), verde (G) y azul (B). Geométricamente el color está dado por la suma vectorial de sus componentes ($R + G + B$).

Gráficamente se representa por un cubo (Figura. 1.12). En la recta que une el origen con el valor máximo se encuentran los grises, y en el resto de los vértices se encuentran los colores saturados que son producidos por la combinación de dos colores primarios.

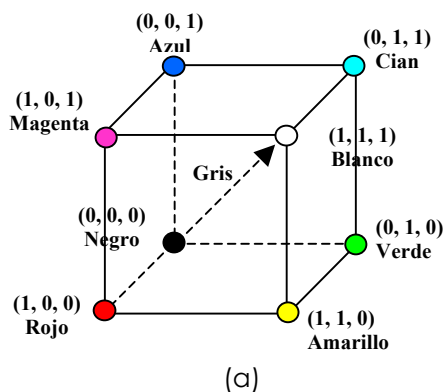


Figura 1.12.- Espacio de color RGB. (a). Interpretación geométrica del color. (b). Colores se forman de la mezcla de la luz blanca (síntesis aditiva)

Aunque el sistema RGB es el más intuitivo de todos y de hecho es en el que se basan las cámaras para adquirir imágenes en color, presenta un serio inconveniente: sus tres componentes mezclan información de cromaticidad y la intensidad. Por lo tanto, si se pretende realizar algún tipo de procesamiento basado en el color tendríamos que tomar en cuenta sus 3 componentes al mismo tiempo.

Espacios de color YIQ, YUV, YCbCr, YCC

Estos espacios se basan en la obtención de la luminancia y la cromaticidad. Están muy relacionados con los sistemas de color para la televisión: YIQ para el NTSC, YUV para el PAL e YCbCr para la televisión digital. Estos modelos de color son utilizados en algunas videocámaras de color.

En la Ec.1.8 a partir de la elección de los colores primarios y del nivel de blanco de referencia se puede obtener la expresión fundamental

de la luminancia monocroma, siendo RGB las señales de salida de la cámara de video tricolor.

$$Y = 0.30 R + 0.59 G + 0.11 B \quad \text{Ec. 1.8}$$

En cuanto a la cromaticidad, se han experimentado diversas formas de señal; inicialmente se trabajó con señales R y B de forma aislada, pero al no satisfacer el criterio de que la información de color debe desaparecer cuando el color es acromático (color gris), y cuando se transmite en monocromo (televisión en blanco y negro), se buscaron otras formas de señal que son las llamadas señales de diferencia de color, las cuales son usadas para obtener las señales de cromaticidad en el espacio YUV.

En el espacio YUV las señales simples de cromaticidad se obtienen al restar de las señales primarias (R,G,B) la señal de luminancia Y. Lo que produciría en total cuatro señales, de las cuales solo tres son necesarias para poder transmitir información. Por lo tanto la señal que se elimina es G-Y debido a sus menores coeficientes respecto a las señales R-Y y B-Y. En la Ec. 1.9 se encuentra la matriz de transformación que permite obtener las componentes YUV a partir de las componentes RGB.

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \text{Ec. 1.9}$$

$$U=0.493(B-Y) \quad \text{Ec. 1.10}$$

$$V=0.877(R-Y) \quad \text{Ec. 1.11}$$

La Ec. 1.10 y Ec.1.11 obtienen las componentes de cromaticidad U y V que pueden tomar valores positivos o negativos. Así mismo, permiten que la relación señal-ruido sea lo mejor posible y reduce el ancho de banda necesario para la transmisión. En la Ec. 1.12 se puede usar la transformación inversa para obtener el espacio RGB.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.000 & 1.140 \\ 1.000 & -0.396 & -0.581 \\ 1.000 & 2.029 & 0.000 \end{bmatrix} \begin{bmatrix} Y \\ U \\ V \end{bmatrix} \quad \text{Ec. 1.12}$$

El sistema NTSC usa las componentes YIQ que, de forma similar a la anterior, representan luminosidad y cromaticidad. La señal Y tiene un ancho de banda de 4.2 MHz, I 0.5 MHz y Q 1.5 MHz. El sistema de referencia forma un ángulo de 33° respecto a YUV. Las ecuaciones 1.13 y 1.14 nos permiten obtener las componentes YIQ a partir de las señales RGB y obtener la transformación inversa respectivamente.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.569 & -0.274 & -0.322 \\ 0.211 & -0.523 & -0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \text{Ec. 1.13}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.621 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.106 & -1.703 \end{bmatrix} \begin{bmatrix} Y \\ I \\ Q \end{bmatrix} \quad \text{Ec. 1.14}$$

Espacio de color CMY (K)

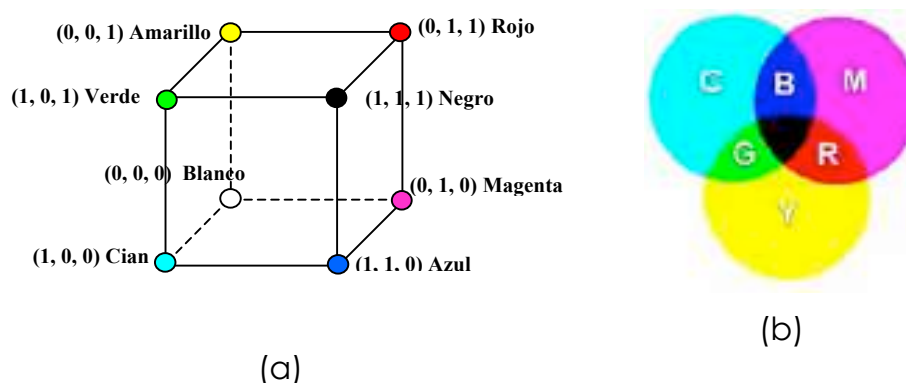


Figura 1.13.- Espacio de color CMY(K). (a) Interpretación geométrica del color. (b) Los colores se forman de la sustracción de sus pigmentos de tinta (sistema sustractivo).

En la Figura. 1.13 se ilustra la representación geométrica del espacio CMY, el cual esta formado por los colores primarios Cían (C), Magenta (M), y Amarillo (Y), ubicados en los ejes cartesianos. Es usado para la impresión. El CMY depende del dispositivo debido a que el color se lo obtiene por la mezcla de los pigmentos de tinta. Mientras que en RGB se obtiene un color con la suma de sus tres componentes de luz, el espacio CMY se lo obtiene restando las componentes RGB al color blanco. En las ecuaciones 1.15, 1.16 y 1.17 puede obtenerse las componentes CMY a partir del espacio RGB, por el contrario se

obtiene la transformación inversa usando las ecuaciones planteadas en 1.18, 1.19 y 1.20.

$$C=1 - R \quad \text{Ec. 1.15}$$

$$M=1 - G \quad \text{Ec. 1.16}$$

$$Y= 1 - B \quad \text{Ec. 1.17}$$

$$R=1-C \quad \text{Ec. 1.18}$$

$$G=1-M \quad \text{Ec. 1.19}$$

$$B=1-Y \quad \text{Ec. 1.20}$$

En el modelo CMY, el negro es creado por mezcla de todos los colores, y el blanco es la ausencia de cualquier color (asumiendo que el papel sea blanco). Una mezcla de cian, magenta y amarillo en realidad resulta en un color negro turbio por lo que normalmente se utiliza tinta negra de verdad. Cuando el negro es añadido, este modelo de color es denominado modelo CMYK. Recientemente, se ha demostrado que el modelo de color CMY es también más preciso para las mezclas de pigmento.

Las Ec.1.21, 1.22 ,1.23 y 1.24 son usadas para obtener las componentes CMYK a partir de las CMY, mientras que en las Ec. 1.25, 1.26, 1.27 se puede obtener la transformación inversa.

$$K = \text{mínimo}(C, M, Y) \quad \text{Ec. 1.21}$$

$$C = \frac{C - B}{1 - B} \quad \text{Ec. 1.22}$$

$$M = \frac{M - B}{1 - B} \quad \text{Ec. 1.23}$$

$$Y = \frac{Y - B}{1 - B} \quad \text{Ec. 1.24}$$

$$C = \text{mínimo}(1, C(1-K)+K) \quad \text{Ec. 1.25}$$

$$M = \text{mínimo}(1, M(1-K)+K) \quad \text{Ec. 1.26}$$

$$Y = \text{mínimo}(1, Y(1-K)+K) \quad \text{Ec. 1.27}$$

Modelos orientados a reproducir la percepción humana

Son modelos que intentan reflejar la percepción que tiene un humano promedio de los colores (aunque las señales básicas siguen siendo las

provenientes de los conos, el cerebro realiza un procesado posterior). Estos espacios tienen la propiedad de separar la información de cromaticidad de la información de luminancia lo cual permite distinguir el color independientemente de la iluminación (modelos no uniformes) por ejemplo, las combinaciones lineales o no, del espacio RGB (HLS, HSV, HSB) o del espacio de color CIE XYZ (ver sección 1.4.2.2). Si, además de separar la señal de luminancia de la señal de cromaticidad, se consigue que la sensación de diferencia percibida entre dos colores sea proporcional a la distancia Euclídea (medida entre los puntos que representan esos dos colores en el espacio tridimensional empleado), se habla de modelos orientados a reproducir la percepción humana uniformes, tales como $L^* u^* v^*$, $L^* a^* b^*$.

Espacio de color HSI

El espacio de color HSI se basa en el modo de percibir los colores que tenemos los humanos. Dicho sistema caracteriza el color en términos de tono o tinte (HUE), saturación (Saturation) e Intensidad (Intensity); componentes que se muestran favorables cuando el tono del color es importante para la segmentación de la imagen.

Las transformaciones matemáticas que permiten el paso del espacio RGB al HSI son:

$$I = \frac{R + G + B}{3} \quad \text{Ec. 1.28}$$

$$H = \left(\frac{\sqrt{3}(G - B)}{(R - G) + (R - B)} \right) \quad \text{Ec. 1.29}$$

$$S = 1 - \frac{\min(R, G, B)}{I} \quad \text{Ec. 1.30}$$

En la Ec.1.31 se expresa la ecuación del tono en base a los términos x y y . El signo obtenido es empleado para establecer el cuadrante al que pertenece el ángulo resultante. Se puede considerar que el tono indica el ángulo formado entre el eje de referencia (el correspondiente al color rojo) y el punto que ocuparía el color a analizar.

$$H = \arctan\left(\frac{x}{y}\right) \quad \text{Ec. 1.31}$$

Geométricamente, el HSI se representa con dos pirámides unidas por su base, Figura. 1.14. Dependiendo del valor de la intensidad se tendrá un corte con las pirámides obteniéndose un triángulo. Dentro del triángulo obtenido la componente H viene definida por su orientación

(el rojo es 0° , el verde 120° y el azul 240°) y la saturación indica la distancia al centro del triángulo.

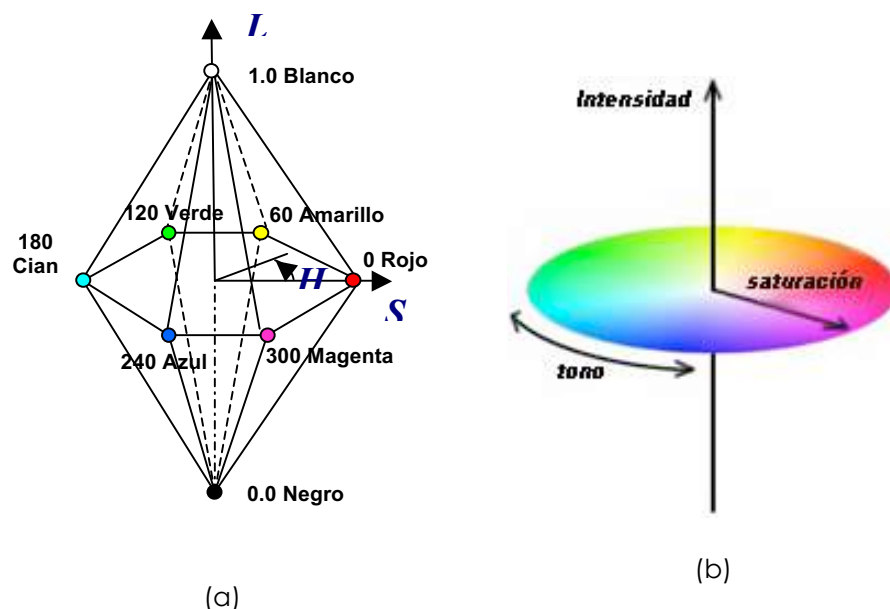


Figura 1.14.- Espacio de color HSI. (a). Representación geométrica del espacio HSI. (b). Proyección circular en el plano (se muestra la distribución de los colores).

Variaciones de este espacio de color: el HSL (Hue Saturation Lightness), HSV (Hue Saturation Value), HCl (Hue Chroma /Colorfulness Lightness), HVC (Hue Value Chroma), TSD (Hue Saturación Darkness), los cuales separan la componente de luminosidad de las de cromaticidad. Se caracterizan por ser variaciones lineales o no lineales del RGB, y son espacios perceptualmente no uniformes.

Los inconvenientes que presenta el uso de espacio HSI son: la no linealidad de las transformaciones realizadas, lo que implica un alto coste computacional para obtenerlas. La singularidad existente en el

entorno de eje definido por R-G-B (Saturación = 0), que provoca que los valores obtenidos para el tono sean inestables.

Una ventaja que presenta este espacio de color es que es posible modificar la luminosidad de la imagen, utilizando únicamente la componente I, esto demuestra que la componente de luminosidad es independiente de las otras dos componentes.

Espacios de color CIE

En 1931 la “**Commission Internationale de l'Éclairage**” definió un modelo que permite representar todos los colores mediante síntesis aditiva de tres fuentes primarias (Tristimulus), una combinación de estos tres colores pueden corresponderse con cualquier fuente monocromática de luz.

Uno de los primeros espacios de color definidos por CIE fue el CIE XYZ donde cada una de sus componentes primarias de color fueron definidas experimentalmente, basados directamente en mediciones de cómo el ojo humano percibe los colores [28]. Este espacio sirvió de base para la construcción de otros espacios de color.

Espacio CIE XYZ

El primer espacio de color estandarizado es el XYZ. Sus componentes no son reales, sino imaginarias, pero cualquier color se puede definir como combinación de ellas. Así mismo, toda la información de intensidad esta comprendida únicamente en la componente Y, mientras en las componentes XZ se encuentra la cromaticidad.

Geoméricamente, se lo representa como una figura en forma de herradura, Figura.1.15.

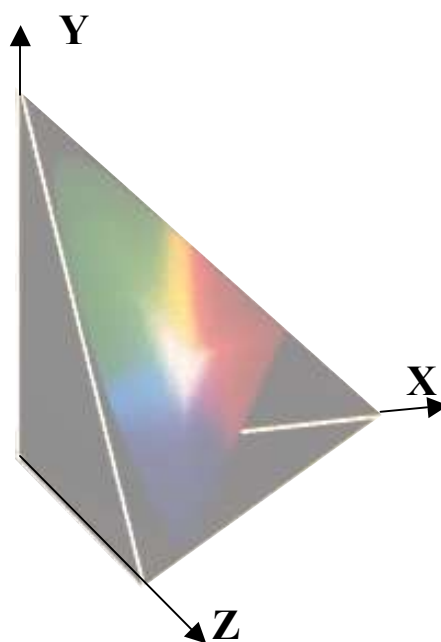


Figura 1.15.- Representación geométrica del espacio CIE XYZ

La Ec. 1.32 es usada para obtener las componentes XYZ a partir del espacio RGB, por el contrario la conversión inversa es obtenida usando Ec.1.33.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.431 & 0.342 & 0.178 \\ 0.222 & 0.707 & 0.071 \\ 0.020 & 0.130 & 0.939 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \text{Ec. 1.32}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.063 & -1.393 & -0.476 \\ -0.969 & 1.876 & 0.042 \\ 0.068 & -0.229 & 1.069 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad \text{Ec. 1.33}$$

Este espacio es independiente del dispositivo que se este usando. Se suele trabajar con valores normalizados entre cero y uno (Ec. 1.34). Basados en estas ecuaciones normalizadas es posible obtener las componentes XZ para una determinada luminosidad Y empleando las ecuaciones en 1.35.

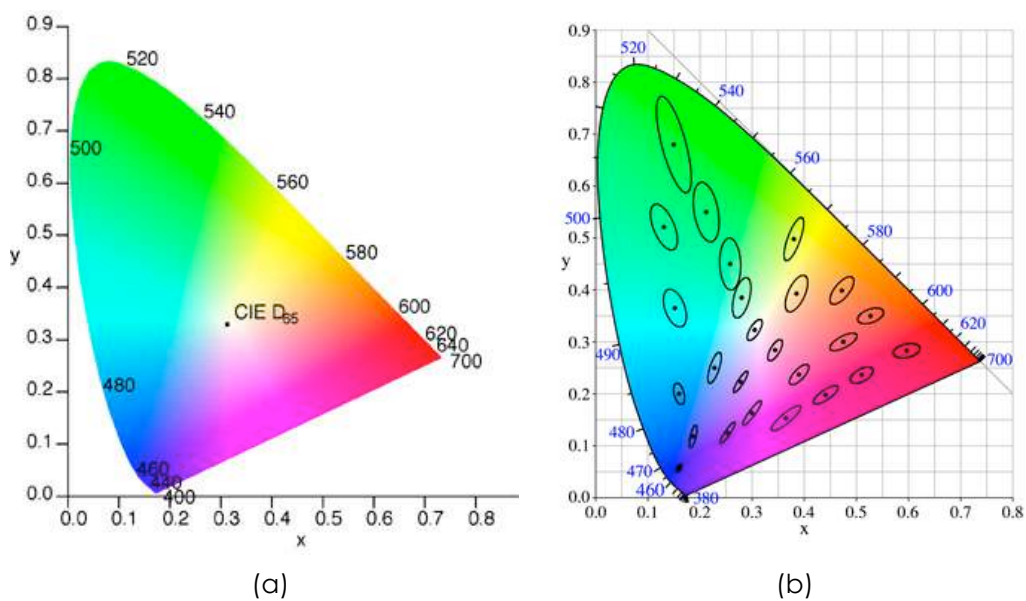
$$x = \frac{X}{X+Y+Z} \quad y = \frac{Y}{X+Y+Z} \quad z = \frac{Z}{X+Y+Z} \quad \text{Ec. 1.34}$$

$$X = \frac{xY}{y} \quad Z = \frac{zY}{y} \quad \text{Ec. 1.35}$$

Las ecuaciones planteadas en 1.34 permiten definir el color únicamente por las componentes x y y, que forman lo que se conoce como diagrama cromático (Fig. 1.15), y debido a que la suma de los

tres componentes es igual a 1, se puede eliminar la z con esta ecuación. Se puede notar que el espacio xy es una proyección del espacio CIE XYZ sobre el plano $X+Y+Z=1$, cada punto x, y corresponde a muchos puntos en el espacio original. La información faltante es la luminancia Y . Por lo que, surge un nuevo espacio de color derivado del XYZ que es el xyY , el cual define la cromaticidad por la componentes x y y y la luminancia por la componente Y .

Puede observarse como los colores se distribuyen en la curva en función de su longitud de onda (Figura. 1.16 (a)). Este sistema es no uniforme, tal como probó McAdam [29]. Este explica que el número de colores significativamente diferentes de uno determinado depende de las componentes de este color (Figura. 1.16 (b)), lo cual no es deseable para separar colores. Para corregir esta situación se crearon los espacios CIE $L^* U^* V^*$ y CIE $L^* a^* b^*$ espacios que son perceptualmente uniformes y más adecuados para la segmentación, ya que, en estos espacios el parecido entre dos colores puede ser medido por la distancia euclídea.



(a) (b)
 Figura 1.16.- Diagrama Cromático. (a).Distribución del color (espacio xyY). (b). Separación de McAdam

Espacio CIE L* u* v*

El espacio CIE L* u* v* se basa en las siguientes relaciones donde, de nuevo, L* es la información de la luminosidad y u*v* corresponden a la cromaticidad:

$$L^* = 116 \left(\frac{Y}{Y_n} \right)^{1/3} \quad \text{Si } \frac{Y}{Y_n} > 0.008856 \quad \text{Ec. 1.36}$$

$$L^* = 903.3 \frac{Y}{Y_n} \quad \text{Si } \frac{Y}{Y_n} < 0.008856 \quad \text{Ec. 1.37}$$

$$u^* = 13L^* (u' - u'_n) \quad \text{Ec. 1.38}$$

$$v^* = 13L^* (v' - v'_n) \quad \text{Ec. 1.39}$$

Siendo:

$$u' = \frac{4X}{(X + 15Y + 3Z)} \quad \text{Ec. 1.40}$$

$$v' = \frac{9Y}{(X + 15Y + 3Z)} \quad \text{Ec. 1.41}$$

Y u'_n , v'_n tienen las mismas ecuaciones y son los valores para la referencia del blanco.

Espacio CIE L* a* b*

En el espacio CIE L* a* b*, L* es la luminosidad, a* la referencia con respecto a la relación rojo/verde y b* respecto a la relación amarillo/azul. Las ecuaciones que lo definen dependen de la relación Y/Yn y son:

$$\text{Si } \frac{Y}{Y_n} > 0.008856$$

$$L^* = 116 \left(\frac{Y}{Y_n} \right)^{1/3} - 16 \quad \text{Ec. 1.42}$$

$$a^* = 500 \left(\left(\frac{X}{X_n} \right)^{1/3} - \left(\frac{Y}{Y_n} \right)^{1/3} \right) \quad \text{Ec. 1.43}$$

$$b^* = 200 \left(\left(\frac{X}{X_n} \right)^{1/3} - \left(\frac{Z}{Z_n} \right)^{1/3} \right) \quad \text{Ec. 1.44}$$

$$\text{Si } \frac{Y}{Y_n} \leq 0.008856$$

$$L^* = 903.3 \frac{Y}{Y_n} \quad \text{Ec. 1.45}$$

$$a^* = 7.87 \left(\frac{X}{X_n} - \frac{Y}{Y_n} \right) \quad \text{Ec. 1.46}$$

$$b^* = 7.87 \left(\frac{Y}{Y_n} - \frac{Z}{Z_n} \right) \quad \text{Ec. 1.47}$$

El cromatismo en este espacio queda definido por:

$$\text{croma} = \sqrt{(a^*)^2 + (b^*)^2} \quad \text{Ec. 1.48}$$

Mientras que el tono o tinte:

$$tono = \arctan\left(\frac{b^*}{a^*}\right)$$

Ec. 1.49

Capítulo 2

CAPITULO II

EVALUACION DE ALGORITMOS DE SEGMENTACIÓN

En este capítulo, se analiza la relevancia de realizar una evaluación cuantitativa de los resultados generados por los algoritmos de segmentación en aplicaciones de visión por computador. Se discuten las diferentes técnicas de evaluación y su aplicación en la segmentación de imágenes, en particular las curvas ROC (Receiving Operating Curves) método seleccionado para este trabajo. Idealmente el resultado de la segmentación debe coincidir con los píxeles clasificados manualmente por un experto.

Importancia de la evaluación de los algoritmos de segmentación

Dada la naturaleza variable de una imagen y las características de diseño de las técnicas de segmentación (ver Figura 2.1), es imprescindible la evaluación objetiva de algoritmos antes de la

implementación de cualquier solución de software basada en procesamiento de imágenes o la publicación de resultados de investigación de alguna nueva técnica [9].

Un ejemplo claro de esta problemática lo constituyen las imágenes microscópicas [8].

A pesar del continuo avance en el desarrollo de nuevas técnicas de segmentación su incorporación en aplicaciones tecnológicas se ve limitado por la falta de una evaluación cuantitativa de su rendimiento sobre una base de imágenes común [7]. Esta subjetividad constituye actualmente uno de los mayores problemas para el desarrollo del análisis de imágenes [6].

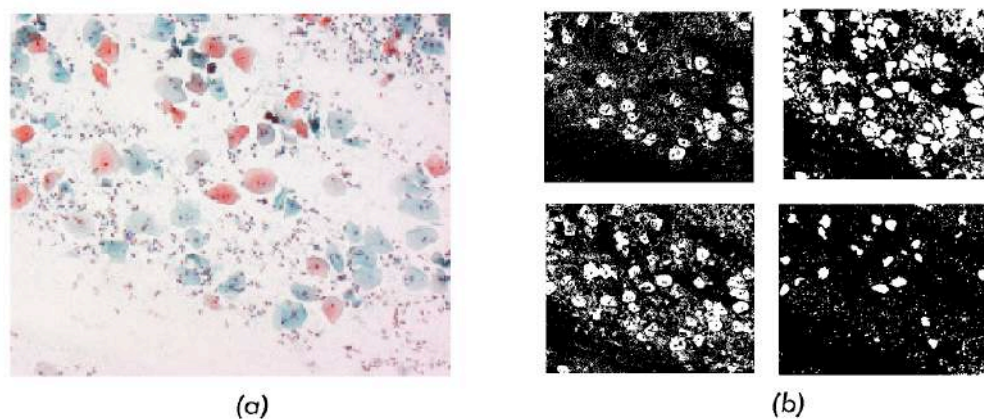


Figura. 2.1. (a) Imagen cérvico uterina (b) Resultados obtenidos de la segmentación aplicando diferentes algoritmos.

Muchas publicaciones de técnicas de segmentación en congresos internacionales desarrolladas para cubrir alguna necesidad

específica, ya sea en el área de control de calidad, medicina, análisis de imágenes microscópicas o detección de objetos, son muy difíciles de reproducir con el mismo porcentaje de efectividad con la que el autor afirma haber conseguido durante la etapa experimental, esto se da, debido a que muchos de los resultados obtenidos por esos experimentos se realizan sobre imágenes ideales, es decir imágenes no muy complejas con una alta calidad de adquisición y que en un ambiente de producción es difícil conseguir. Visto de este modo, la elección de un algoritmo de segmentación que funcione con un alto nivel de eficiencia, sobre un banco imágenes que cuente con las mismas características de adquisición es una tarea fundamental. Este permitirá manejar un gran volumen de datos y extraer información cuantitativa valiosa. La misma que puede ser registrada para su posterior análisis.

La creación de una herramienta que permita caracterizar el comportamiento de un algoritmo de segmentación, que es el objetivo de esta tesis, servirá de soporte al investigador y agilizará el proceso de implementación de una técnica de segmentación que pueda ser usada como parte de un sistema de visión más avanzado

Tipos de técnicas de evaluación de algoritmos de segmentación

Una variedad de técnicas han sido propuestas para evaluar cuantitativamente el funcionamiento y el resultado de los algoritmos de segmentación [4], estos pueden categorizarse en tres grupos: (1) métodos analíticos, (2) métodos empíricos supervisados y (3) métodos empíricos no supervisados

Métodos Analíticos

Los métodos analíticos caracterizan a los algoritmos de segmentación independientemente del resultado de la segmentación, evaluando su eficacia en base a sus propiedades y principios. En general, estos métodos trabajan solamente evaluando ciertas características de los algoritmos de segmentación, tales como la estrategia de procesamiento (paralelo, secuencial, iterativo, combinado), complejidad, eficacia del proceso y la calidad de la segmentación. Estas características no son generalmente decisivas para diferenciar el desempeño entre los algoritmos de segmentación. Por lo tanto, los métodos analíticos generalmente no son una buena alternativa para evaluar la segmentación de objetos en un sistema de reconocimiento. Ejemplos de estos algoritmos los podemos encontrar en [30],[31].

Métodos empíricos supervisados

Los métodos empíricos supervisados también conocidos como métodos relativos o de discrepancia evalúan los algoritmos de segmentación en base a la comparación entre la imagen resultante del proceso de segmentación con la imagen plantilla segmentada manualmente, este método comúnmente se lo conoce como estándar de oro [10]. Una ventaja de los métodos empíricos supervisados sobre los no supervisados es que la comparación directa entre la imagen segmentada y la plantilla proporciona un resultado completamente confiable o de alta resolución, y en general, son usualmente utilizados como métodos de evaluación objetiva. Sin embargo, la generación manual de la plantilla es un trabajo difícil y subjetivo y demorado. Y para la mayoría de las imágenes, especialmente las imágenes con escenarios de paisajes (con alto nivel de detalle) no podríamos garantizar que una segmentación generada manualmente sea mejor que otra. Además, estos métodos de evaluación no pueden ser utilizados como parte de un sistema de reconocimiento de objetos en tiempo real puesto que estos necesitan tener previamente generada una plantilla.

Métodos empíricos no supervisados

Los métodos empíricos no supervisados [5], también conocidos como métodos de evaluación stand-alone o independientes, evalúan cuantitativamente los resultados de los algoritmos de segmentación de acuerdo a la perspectiva humana sobre que propiedades debe cumplir una segmentación ideal. La ventaja de estos métodos es que no requieren conocimiento previo de una segmentación correcta (es decir no necesitan ser comparados con una plantilla generada manualmente). La idea fundamental en cuanto a que propiedades constituyen una segmentación ideal es basada en: medidas de uniformidad entre región (por ejemplo: variación de color, error cuadrático del color y textura), medidas de contraste entre regiones (por ejemplo, diferencia media del color entre regiones, diferencia media del color a lo largo de los bordes), medidas semánticas (por ejemplo la forma de la región y evaluación de contornos). Toda esta información puede ser usada como conocimiento previo de que tipo de imágenes serán analizadas.

Evaluación basada en el análisis de curvas ROC

En este trabajo se empleó una técnica estadística basada en el análisis de curvas ROC este tipo de análisis puede categorizarse como un método empírico supervisado (puesto que usa una plantilla durante el proceso de evaluación), el cual permite describir la eficacia de un algoritmo de segmentación de una manera visual, además permite escoger un umbral sobre el cual una máquina o una persona sin experiencia previa, podrían clasificar los resultados de la segmentación como satisfactorios.

El análisis de curvas ROC mide en términos de sensibilidad y especificidad el rendimiento de los algoritmos utilizados. Estas métricas son definidas como la proporción de píxeles que pertenecen al objeto de interés detectados correctamente y la proporción de píxeles que no pertenecen al objeto de interés y no forman parte del resultado de la segmentación detectados correctamente; resultados que deben coincidir con los píxeles clasificados manualmente por un especialista, encapsulados en una plantilla.

Introducción a las curvas ROC

La metodología ROC fue desarrollada en el contexto de la detección de señales electrónicas en los inicios de la década de los 50 y su primera aplicación fue motivada por problemas prácticos en la detección de señales por radar. A mediados de los 60 se habían usado las curvas ROC en psicología y psicofísica experimental [10].

Leo Lusted, un radiólogo, fue el primero en usarlas en el proceso de toma de decisiones médicas, en 1967, y comenzó a aplicarlas en estudios con imágenes, en 1969.

El análisis de curvas ROC es comúnmente utilizado en la caracterización de la eficacia que tienen los métodos de diagnóstico para la detección temprana de alguna enfermedad. Esta mide la habilidad de la prueba para distinguir entre estados alternativos de salud. Como la distinción entre enfermedad y salud, entre enfermedad benigna y maligna. Esta idea ha sido aplicada en el área de visión por computador en la caracterización de algoritmos de segmentación [11], en la caracterización de algoritmos de búsquedas [12], y otras aplicaciones en donde se pretenda evaluar la efectividad de un algoritmo dado. La limitación de este tipo de análisis es que no

posee la capacidad de clasificar en niveles el grado de segmentación obtenido por cada uno de los parámetros evaluados en el algoritmo, sólo podemos discriminar que parámetros dan buenos resultados versus que parámetros producen pobres resultados. Esto se debe a que este análisis solo permite evaluar resultados del tipo dicotómico (una cuyos resultados se puedan interpretar directamente como positivos o negativos), por lo tanto su precisión puede ser medida en términos de sensibilidad y especificidad, en el caso de los algoritmos de segmentación se entiende como la capacidad de discriminar que un píxel pertenezca al objeto de interés o que no forme parte de el, estos resultados deben coincidir con los píxeles clasificados manualmente por un especialista.

La importancia de estos conceptos es absolutamente reconocida, pero reportar un solo valor de sensibilidad y especificidad es una simplificación del problema. Consecuentemente, solo el rango completo de los pares sensibilidad/1-especificidad proporciona un cuadro íntegro de la eficacia de un parámetro o algoritmo durante el proceso de selección.

Comparación basada en el análisis de curvas ROC

La base del análisis ROC es sencillo [13]: se trata de comprobar la eficacia de un proceso de decisión entre dos alternativas, y se analizan los conceptos contenidos en una tabla de doble entrada en la cual se compara el estado real (píxeles clasificados en una plantilla) y el estado "predicho" (píxeles detectados automáticamente por el algoritmo):

		Estado Real	
		Positivo	Negativo
Estado Predicho	Positivo	Verdaderos Positivos (VP)	Falsos Positivos (FP) Falsas Alarmas
	Negativo	Falsos Negativos (FN) (omisiones)	Verdaderos Negativos (VN)

Tabla 2.1 Tabla de contingencia del resultado de una prueba con respecto a su estado real

En la tabla 2.1 se describe completamente el comportamiento de un sistema con dos únicos posibles resultados (positivo o negativo). Los valores contenidos en esta tabla de contingencia pueden ser obtenidos mediante la aplicación de las siguientes fórmulas: en las ecuaciones Ec. 2.1 y Ec. 2.3 es posible obtener el porcentaje de

aciertos, obtenidos al comparar el estado real con el estado predicho que correspondan correctamente con la clasificación realizada en la plantilla. Mientras que en las ecuaciones Ec. 2.2 y Ec. 2.4 se obtiene el total de píxeles clasificados en la plantilla en base al porcentaje de aciertos y desaciertos conseguidos durante la etapa de evaluación al ser comparados con la plantilla.

$$\begin{array}{l} \text{(Fracción de verdaderos positivos)} \\ \mathbf{Sensibilidad} = VP / (VP + FN) = FVP \end{array} \quad \text{Ec. 2.1}$$

$$(VP + FN) = \text{Total positivos (Estado Real)} \quad \text{Ec. 2.2}$$

$$\begin{array}{l} \text{(Fracción de verdaderos negativos)} \\ \mathbf{Especificidad} = VN / (VN + FP) = FVN \end{array} \quad \text{Ec. 2.3}$$

$$(VN + FP) = \text{Total Negativos (Estado Real)} \quad \text{Ec. 2.4}$$

$$\mathbf{Especificidad} = 1 - FFP \text{(Fracción de falsos positivos)} \quad \text{Ec. 2.5}$$

Se define la sensibilidad (Ec.2.1) o fracción de verdaderos positivos (FVP) como la probabilidad de detectar correctamente un píxel comparado con el estado real. La especificidad (Ec.2.3, Ec.2.5) también conocida como fracción de verdaderos negativos (FVN) se la define como la probabilidad de detectar que un píxel no forme parte de la segmentación y que coincida con su estado real.

La construcción de las curvas ROC (Fig.2.2), se lo realiza graficando 1-Especificidad o lo que es lo mismo Fracción de Falsos positivos (FFP), en el eje abscisas (eje X) y la fracción de verdaderos positivos (FVP) o sensibilidad, en el eje de las ordenadas (eje Y).

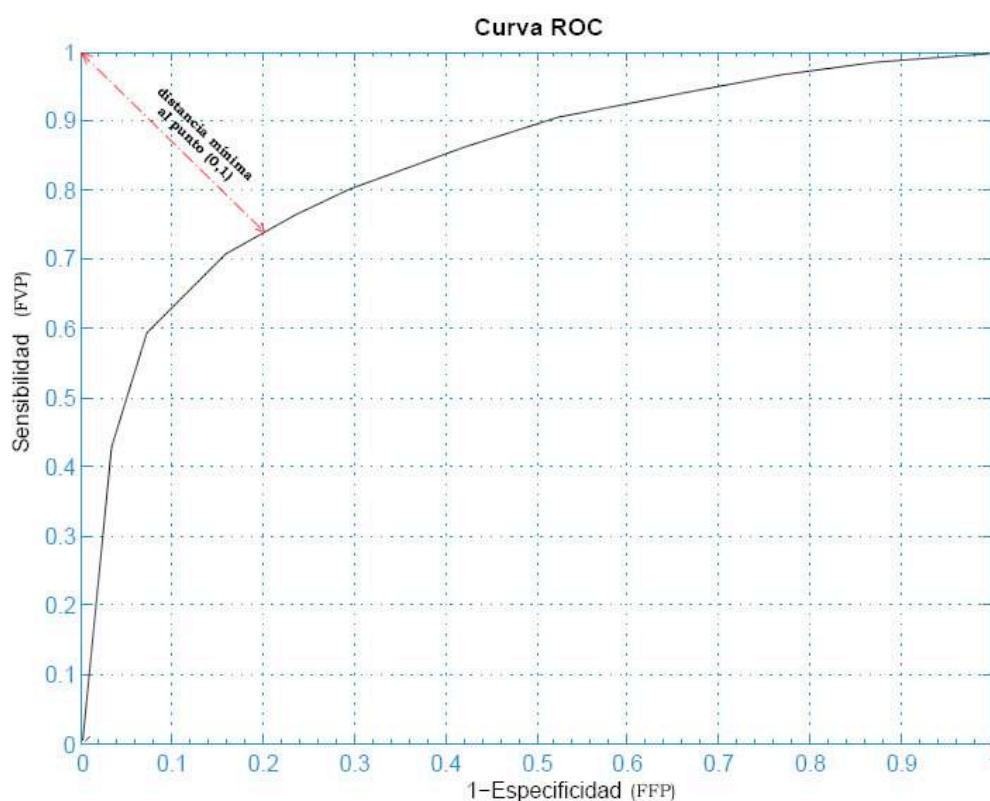


Figura 2.2 Curvas ROC correspondiente a la distribución teórica de los resultados de una prueba

El par (FFP, FVP) usado para graficar cada punto de la curva puede ser obtenido mediante la Ec. 2.6 y la Ec. 2.7. Estos términos son usados para discriminar los resultados obtenidos durante la evaluación.

$$FFP = \frac{\text{Negativos_incorrectamente_clasificados}}{\text{Total_Negativos (Estado Real)}} \quad \text{Ec. 2.6}$$

$$FVP = \frac{\text{Positivos_correctamente_clasificados}}{\text{Total_Positivos (Estado Real)}} \quad \text{Ec. 2.7}$$

En donde *Negativos_incorrectamente _clasificados*= FP y los *Positivos_correctamente_clasificados*= VP.

Información contenida en la curva

El FVP óptimo representa el punto sobre la curva ROC, el cual tiene la menor distancia al punto (0,1). Cuando esta distancia es cero entonces se obtendrá el FVP más alto (punto (0,1)), el cual representa que todos los píxeles fueron correctamente segmentados por el algoritmo. Mientras que el punto situado más a la derecha, donde el FFP toma su valor máximo, representa que el algoritmo ha segmentado todos los píxeles que no corresponden al objeto de interés. Por consiguiente un punto con un FVP alto y un bajo FFP (punto ubicado en la coordenada (0,1) en la gráfica) representa el mejor resultado que puede obtenerse durante la evaluación.

Si la prueba fuese perfecta, existirá una región en la que cualquier punto de la curva tiene sensibilidad igual a 1 y especificidad igual a 1. Por consiguiente la gráfica solo tendría el punto (0,1) para todos los

valores evaluados. Este punto corresponde a FFP de valor 0 y a un FVP de valor 1.

Si la prueba fuese mala, la curva obtenida sería la diagonal que va desde el punto (0,0) al punto (1,1). Por lo tanto, para cada FVP se obtiene un valor igual en FFP. Valores que no permiten discriminar la eficacia de los parámetros evaluados.

Las pruebas habituales tienen curvas intermedias (curva regular y buena). En la figura 2.3 se muestran los diferentes tipos de curvas que pueden ser obtenidas durante el proceso de evaluación.

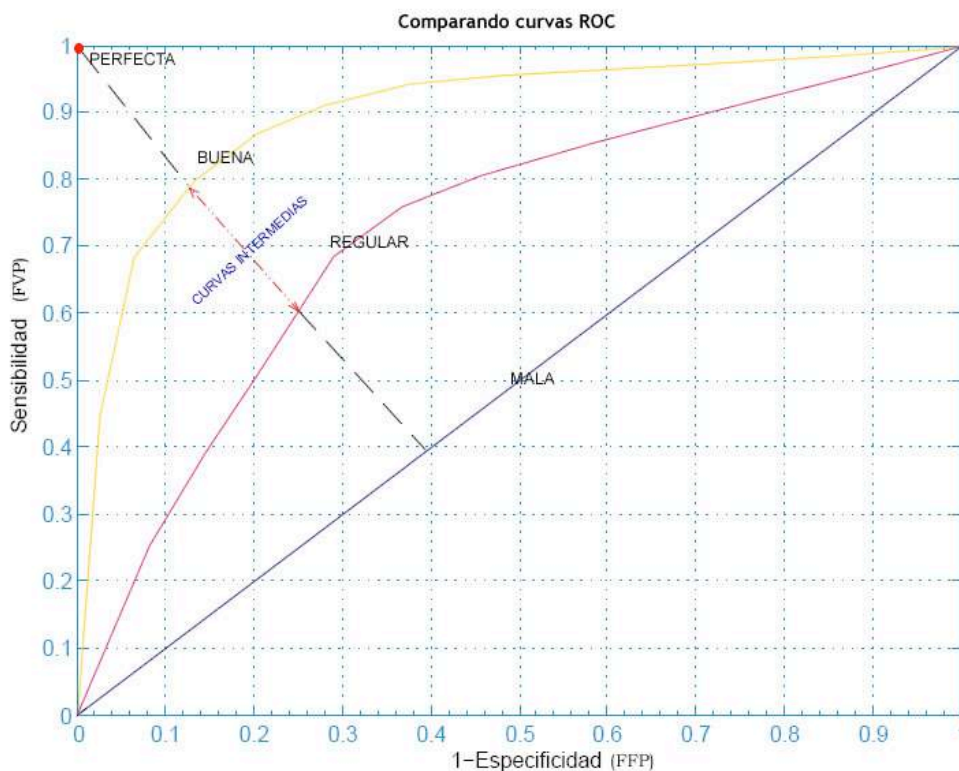


Figura 2.3 Tipos de curvas ROC que pueden ser obtenidas

Ventajas de las curvas ROC

- Es una representación simple, y fácilmente comprensible, de la precisión de una técnica aplicada. Posee la habilidad de discriminar a través de todo el rango de valores.
- La elección de un parámetro o técnica de segmentación se caracteriza solamente por sus valores de (sensibilidad y 1-especificidad) puntos graficados en las curvas ROC.
- Proporciona una comparación visual directa entre pruebas sobre una escala común.
- Puede ser aplicado para evaluar la eficacia de cualquier tipo de técnica o prueba cuyos resultados sean medidas en escala tanto ordinal, por intervalo o continua.
- Proporciona un indicador de criterio de decisión (punto de corte o umbral para la toma de decisión), que permite incluir probabilidades (incluso estimaciones subjetivas de la probabilidad) y costos o utilidades. Este método permite establecer razonadamente reglas o mecanismos óptimos de toma de decisión en la elección de una técnica en particular.

Desventaja de las curvas ROC

- No se muestra los umbrales de decisión reales.
- No se muestra el número de muestras evaluadas, y a medida que el tamaño de la muestra decrece, la representación gráfica tiende a volverse progresivamente mellada y desigual.
- La generación de la figura y el cálculo de los parámetros son difícilmente manejables sin programas de computación, los que no están ampliamente disponibles.

Durante el estudio de imágenes microscópicas, especialmente en el análisis de muestras teñidas, como es el caso de células, es imprescindible la detección de la mayor cantidad de objetos (citoplasmas y núcleos) que permitan, en algún análisis posterior diagnosticar la existencia de anomalías (ver sección 3.4.2 diagnóstico de cáncer uterino). Para conseguir este objetivo se ha planteado el uso de las curvas ROC como método de evaluación de técnicas de segmentación. Puesto que estas permiten realizar un análisis de la efectividad de los resultados producidos por los algoritmos de segmentación, sean estos de diferente naturaleza, evaluados sobre una escala común. Además posee la propiedad de clasificar los resultados mediante el uso de términos descriptivos (bueno, malo,

regular) de la efectividad que poseen los algoritmos evaluados, definiendo umbrales de decisión, que permitan a personas sin experiencia seleccionar el mejor parámetro o algoritmo de visión independientemente del tipo de imágenes a analizar o la técnica utilizada.

Evaluación de la eficiencia basada en el análisis de curvas ROC

Para la elección de que parámetro o técnica produce una mejor efectividad se define un umbral basado en 2 criterios: umbral definido por los valores de sensibilidad y especificidad, o umbral definido por la distancia al punto óptimo (distancia ROC) .

En la ecuación Ec. 2.8 se define un umbral centrándonos únicamente en el par ordenado (FVP, FFP) obtenidos como resultados del proceso de evaluación. El objetivo principal es encontrar un valor en la curva que tenga un alto FVP (podría ser mayor que un umbral T1) y un bajo FFP (menor que un umbral T2).

$$FVP > T1 \ \& \ FFP < T2 \qquad \qquad \qquad \text{Ec. 2.8}$$

Otro método para evaluar los parámetros es calcular la distancia de cada par ordenado hacia el punto (0,1) (ver Fig.2.4) que es el punto

donde se tiene un mejor resultado de la prueba (donde se generan el mayor número de aciertos). A esta distancia la llamaremos d_{roc} [12] la cual puede ser calculada usando la Ec. 2.9 .

$$d_{roc} = \sqrt{(1 - FVP)^2 + FFP^2} \quad \text{Ec. 2.9}$$

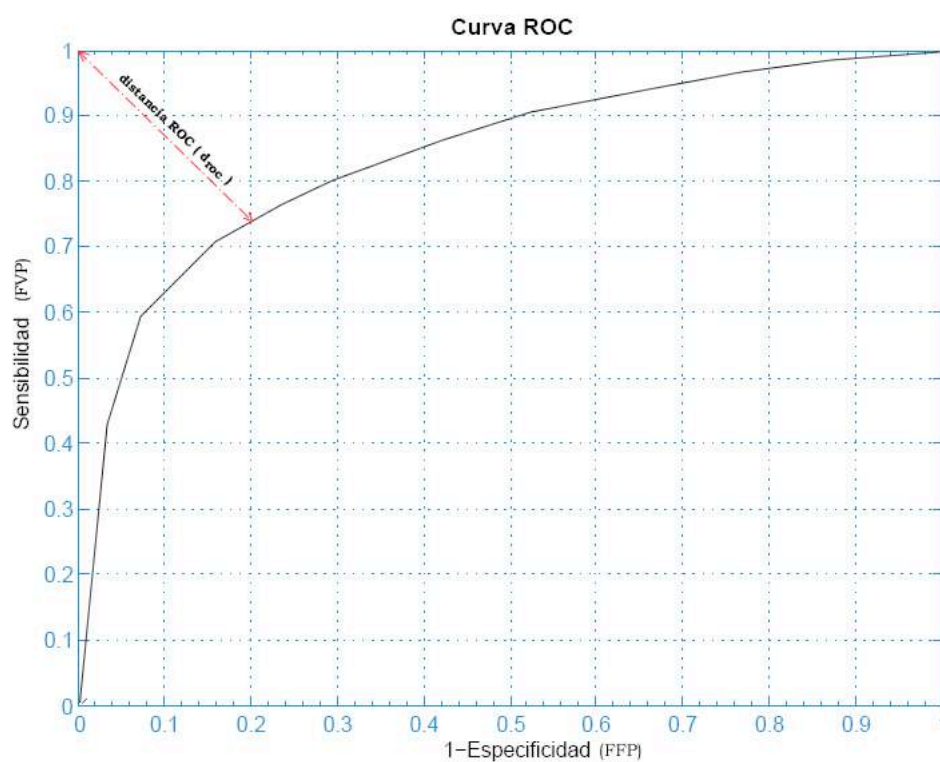


Figura 2.4 Eficiencia basada en la distancia ROC.

Capítulo 3

CAPITULO III

ANALISIS DE IMÁGENES MICROSCOPICAS

Introducción

El análisis de imágenes microscópicas es ampliamente empleado en la investigación científica, y con el transcurso del tiempo ha tenido una gran influencia en los progresos de la ciencia y la tecnología. Sus aplicaciones se extienden a otras áreas como:

- **Geología:** En Investigaciones geo-mineras, cristalográficas, mineralógicas y petrológicas, basadas en el estudio morfológico y estructural de las muestras.
- **Estudio de materiales:** Para la caracterización micro-estructural de materiales. Identificación y análisis de fases cristalinas y transiciones de fases en diversos materiales. Composición de superficies y tamaño de grano. Valoración del deterioro de materiales, determinación del grado de cristalinidad y presencia de defectos.

Capítulo 3

Identificación del tipo de degradación: fatiga, corrosión, fragilización, etc.

- **Paleontología y Arqueología:** Caracterización de aspectos morfológicos de restos de materiales, estudio de jeroglíficos, análisis cefalométrico, identificación de restos vegetales, animales y estudio de microorganismos fosilizados.
- **Control de Calidad:** En el seguimiento morfológico de procesos y su aplicación en el control de calidad de productos en las industrias de fibras y calzado, en la construcción de circuitos electrónicos, y fabricación de piezas electrónicas y mecánicas.
- **Biología:** En el diagnóstico de los diferentes tipos de anomalías cromosómicas (en genética), en el análisis de deformaciones de células (oncología), en el estudio de muestras de plantas, animales y contaminación de suelo y agua entre otros.
- **Estudio químico y estructural de obras de arte,** alteración de monumentos, identificación de pigmentos (restauración, autenticación). Se puede establecer la época de un cuadro, escuela pictórica y posibles intervenciones a las que ha sido sometido,

tales como aplicación de repintes, o tratamientos de restauración anteriores.

Todos estos ejemplos son posibles campos que pueden ser automatizados aplicando algoritmos de visión por computador. En particular, en la identificación de objetos en imágenes. Tal como se estudió en el capítulo dos, las técnicas de segmentación a color son utilizadas durante la detección de objetos en el análisis de células. El objetivo de este trabajo es encontrar el componente de color y el algoritmo de segmentación que detecte correctamente la mayor cantidad de objetos. En los experimentos realizados con la herramienta se trabajó con imágenes de células cérvico uterinas generadas a partir de muestras proporcionadas por APROFE (Asociación Pro Bienestar de la Familia Ecuatoriana).

En este capítulo, revisamos los diferentes tipos de microscopios que pueden ser utilizados para el análisis de muestras, centrándonos particularmente en el microscopio óptico convencional. También se revisan las diferentes técnicas de microscopía que pueden ser configuradas sobre éste y como pueden ayudar a mejorar la calidad de la muestra. Después, se analiza la importancia que tiene la detección y caracterización de células como parte del proceso de

análisis y se mencionan los principales sistemas que han surgido para automatizarlo en el pasado. Asimismo, revisamos como se realiza la recolección de la muestra durante la prueba de papanicolao y como pueden ser interpretados estos resultados. Finalmente se detallan las características de los equipos de visión utilizados durante el proceso de adquisición de muestras.

El Microscopio

El microscopio es un instrumento óptico que amplifica la imagen de un objeto pequeño. Mediante un sistema de lentes y fuentes de iluminación. Los microscopios pueden aumentar de 100 a cientos de miles de veces el tamaño original de una muestra.

Muchos factores importantes se deben tomar en cuenta antes de poder elegir un microscopio como por ejemplo el tipo aplicación, las capacidades de magnificación de las muestras, el tipo de imagen que se desea capturar ya sea en el plano x-y o imágenes tridimensionales, los diferentes accesorios que se pueden adaptar al microscopio y que ayuden a mejorar la calidad de adquisición de las muestras, entre otros.

Para las diversas aplicaciones que se emplee el microscopio una característica de igual importancia que sólo visualizar muestras para ser analizadas, es la capacidad de capturar, mostrar, y preservar imágenes de las muestras observadas.

Partes del Microscopio

Antes de analizar los distintos tipos de microscopios que pueden ser usados para realizar el análisis de imágenes, nos familiarizaremos con las partes básicas con que cuenta un microscopio óptico común (ver Figura.3.1). Para este estudio se las clasificó en: *sistema óptico* y *sistema mecánico*.

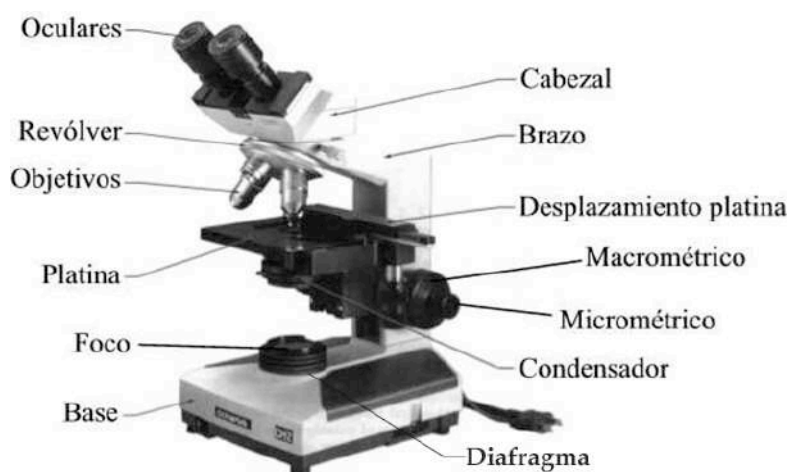


Figura. 3.1 Partes básicas de un microscopio

Sistema óptico

Oculares: Lo forman las lentes situadas en la parte superior del microscopio. Estos lentes son los que quedan cerca del ojo del observador. Su función es ampliar la imagen del objetivo.

Objetivos: Lo forman las lentes situadas cerca de la muestra a analizar. Su función es ampliar la imagen de la muestra. El número de objetivos

con los que puede contar un microscopio esta entre 2 a 5, estos permiten variar la magnificación de la imagen de la muestra en un rango establecido por el tipo de microscopio usado.

Condensador: Lente que concentra los rayos luminosos provenientes del foco y del medio ambiente y que inciden sobre la muestra a analizar.

Diafragma: Regula la cantidad de luz que entra por el condensador.

Foco: Dirige los rayos luminosos hacia el condensador.

Sistema mecánico

Soporte: Mantiene la parte óptica del microscopio y está formado por el brazo y el pie o base.

Platina: Lugar donde se deposita la muestra que se va analizar.

Cabezal: Soporta los sistemas de lentes oculares.

Revólver: Contiene los lentes objetivos y permite que al girar el revolver se pueda cambiar de un objetivo a otro.

Tornillos de enfoque: Esta formado por el macrométrico y micrométrico. El macrométrico aproxima el enfoque a la muestra y micrométrico consigue el enfoque correcto de esta.

Desplazamiento de platina: Formado por un carril que permite el desplazamiento linealmente milimétrico del cristal que contiene la muestra.

Tipos de Microscopios

Los microscopios se clasifican en tres grandes grupos: *Ópticos*, *electrónicos* y *estereoscópicos*. Estos pueden ser configurados con diferentes tipos de accesorios, que permiten trabajar con una técnica de microscopía en particular, esta configuración depende del tipo de muestra a analizar. Por lo general, el microscopio es nombrado según la técnica de microscopía empleada. Independientemente del tipo de microscopio usado, al nombre del microscopio se le añade la palabra simple o monocular, binocular, trinocular, según este tenga uno, dos, o tres oculares (ver Figura. 3.2). Habitualmente, si tiene 3 oculares el tercer ocular es usado por la cámara. A los microscopios con más de un ocular se los llama también *microscopios compuestos*.

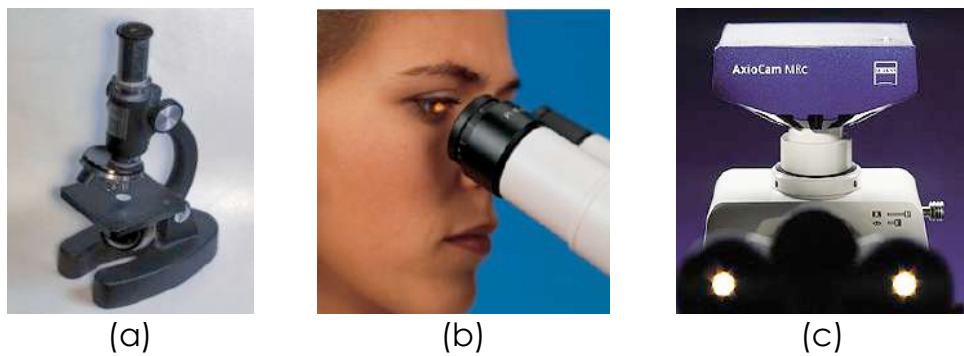


Figura. 3.2 Número de oculares usados en los microscopios: (a) Con un ocular (b) Con dos oculares. (c) Con 3 oculares uno de ellos usado por la cámara.

En la Figura. 3.3 se presenta una clasificación de los tres grandes grupos de microscopios que son comúnmente usados en el análisis de muestras y las diferentes técnicas de microscopía con que pueden ser configurados.

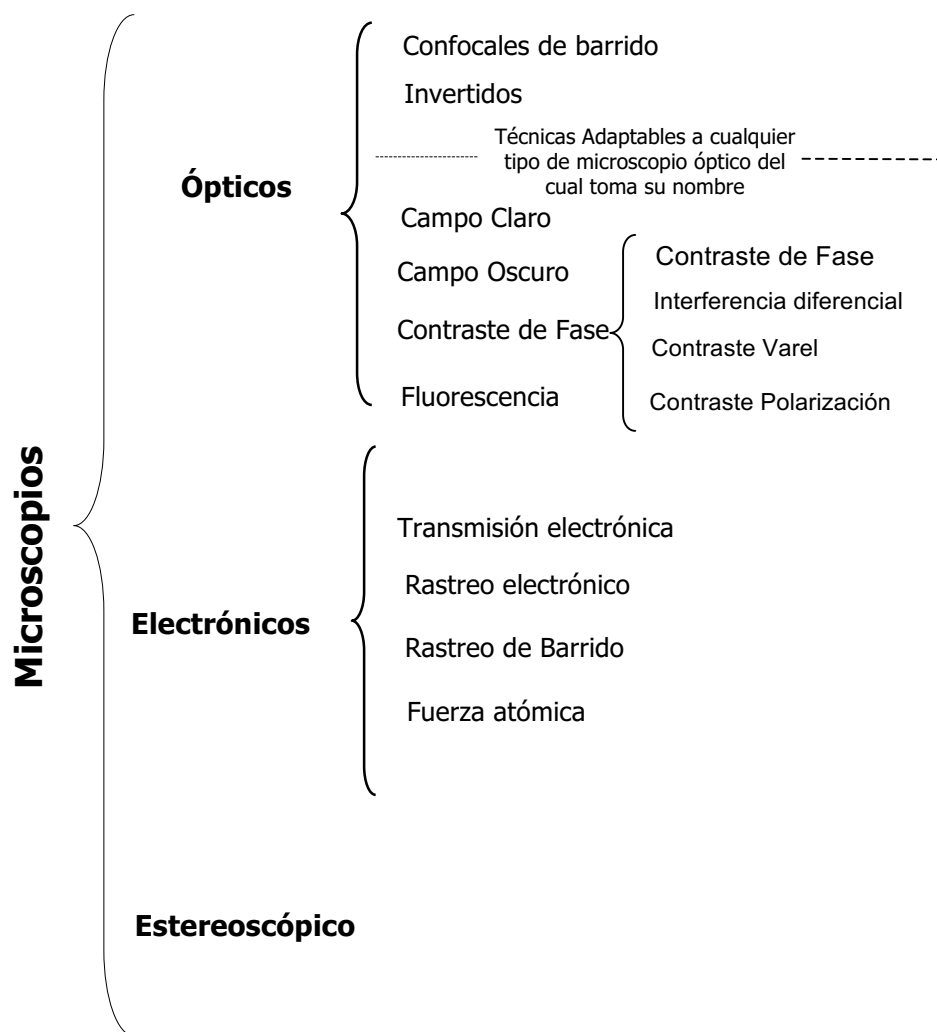


Figura. 3.3 Clasificación de los microscopios

El tipo de iluminación usada en los microscopios puede ser de *luz incidente*, es decir, desde abajo, particularmente útil en los microscopios usados en áreas biológicas; o de *luz transmitida*, cuando es proyectada desde arriba, usada típicamente por los microscopios estereoscópicos. Algunos de estos modelos estereoscópicos poseen también doble iluminación.

Microscopios ópticos

Este microscopio posee un número de objetivos de variado aumento. Típicamente el rango de aumento de un microscopio óptico está entre 10x y 1000x. El investigador observa la muestra en una imagen bidimensional usualmente al revés e invertida. Es ampliamente usado como instrumental normal de trabajo, tanto en laboratorios de análisis clínico-patológico, como en los laboratorios de investigación básica.

A diferencia de los microscopios ópticos convencionales se encuentra los microscopios ópticos confocales que emplea un sistema de iluminación con rayo láser y usa complejos sistemas electromecánicos para la generación de la imagen, que es reconstruida y visualizada por un computador. Las imágenes obtenidas son notablemente superiores a las que se obtienen con el microscopio óptico convencional. Pero su elevado costo impide su utilización.

Otra variante del microscopio óptico es el microscopio óptico invertido este puede obtener una imagen real es decir no invertida como en los microscopios convencionales. Esto se debe a que tienen los lentes objetivos ubicados en la parte inferior de la muestra a analizar.

Los otros tipos de microscopios cuyo nombre lo adquieren de la técnica de microscopía empleada se los puede obtener cambiando ciertos accesorios al microscopio óptico.

Microscopios electrónicos

En el microscopio electrónico los rayos luminosos del microscopio convencional son reemplazados por un haz de electrones. El aumento puede alcanzar en este caso hasta 100 veces el microscopio óptico convencional. La restricción de los preparados de las muestras es que en cada paso se trabaja con especímenes de magnitud 3-4 órdenes menores o más finos que los utilizados para el microscopio óptico.

Permite estudiar la estructura de materiales y células a nivel atómico, es usado por investigadores en análisis ultraestructural en organelas, células, tejidos y materiales inorgánicos. Este ayuda a comprender como están estructurados los materiales y sistemas biológicos a nivel de nanopartículas.

Vale recalcar que la óptica usada en el microscopio electrónico es muy similar a la usada en el microscopio óptico.

Microscopios estereoscópicos

Diferentes a los microscopios ópticos que ofrecen una imagen en 2 dimensiones, los microscopios estéreos muestran una imagen estereoscópica (en 3 dimensiones) construida (recta y no invertida). Este tipo de microscopio es particularmente usado por los biólogos en disecciones, por los técnicos para reparar tarjetas de circuitos, por los paleontólogos para limpiar y examinar fósiles o cualquiera quien necesite trabajar con las manos sobre pequeños objetos.

La mayoría de los microscopios estéreos son usados con aumentos de 5x hasta 50x, pero con los correctos accesorios se puede lograr aumentos aproximados de hasta 400x.

Técnicas de microscopía

La microscopía es la técnica de producir imágenes visibles de estructuras o detalles demasiado pequeños para ser percibidos a simple vista. Generalmente implica la difracción, reflexión o refracción de radiación incidente en el sujeto de estudio.

En la microscopía de luz clásica, esto implica el paso de luz transmitida a través o reflejada desde el sujeto mediante una serie de lentes, para poder ser detectada directamente por el ojo o impresa en una placa fotográfica.

Para el efecto en cada tipo de microscopio se han desarrollado diferentes técnicas que mejoran la calidad de adquisición de las muestras a analizar, esto implica que para que sean implementadas se necesita adecuar el microscopio con ciertos accesorios.

En esta sección enfocaremos en mayor detalle las técnicas de microscopía usadas por los microscopios ópticos que usan luz transmitida o reflejada para el análisis de sus especímenes.

Campo Claro

Método clásico de microscopía usado especialmente en medicina y biología, por ejemplo para especímenes histológicos teñidos (muestras muertas), secciones de tejidos y muestras (ver Figura. 3.4).

Accesorios adicionales requeridos: Ninguno (el microscopio óptico convencional viene implementado con esta técnica) [14].

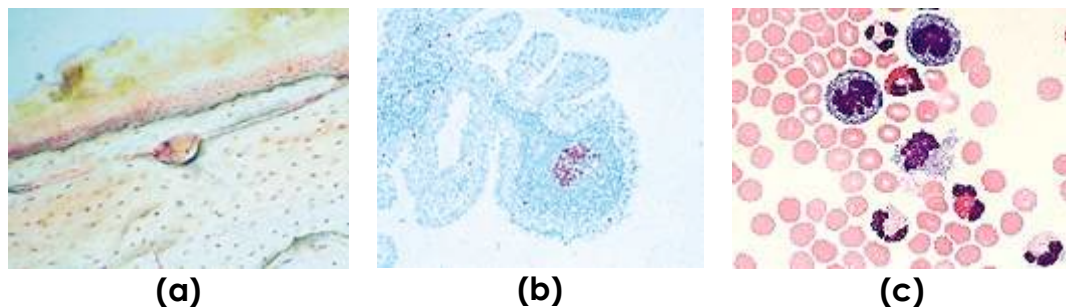


Figura. 3.4 (a) Sección lisa de hueso. (b) Cystadenolymphoma (proliferación de células cancerígenas) (c) Sangre humana.

Campo Oscuro

Este método de contraste puede visualizar características de estructuras finas, e incluso más pequeñas de las que pueden ser visualizadas en los microscopios con campo claro. Es ampliamente usado en aplicaciones de metalografía y cristalografía. Dichas muestras deben ser examinadas con luz reflejada (ver Figura. 3.5).

Accesorios adicionales requeridos: Condensador especial, algunas veces aceite de inmersión [14].

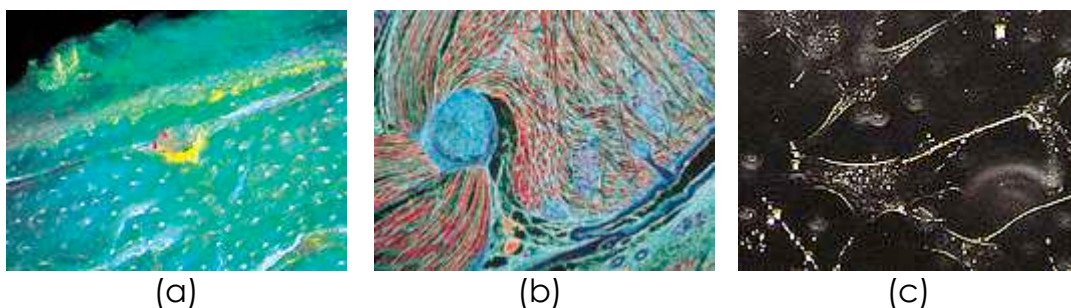


Figura. 3.5 (a) Sección lisa del hueso. (b) Hueso con forma U del ratón. (c) Neuronas.

Contraste de Fase

El método de contraste de fase es adecuado para visualizar características de estructuras muy finas en tejidos y células aisladas que se encuentren dentro de especímenes no teñidos muy delgados (<5 μm) que tengan muy poco contraste (ver Figura. 3.6).

Accesorios adicionales requeridos: Objetivos especiales y condensadores especiales [14].

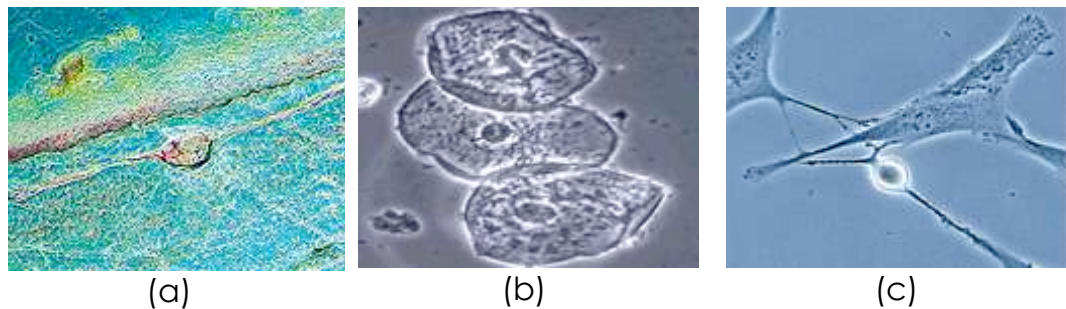


Figura. 3.6 (a) Sección lisa del hueso. (b) Especimen de fase delgado. (c) Neuronas.

Contraste de Varel

Este método de contraste muestra estructuras de especímenes en relieve, es insensible a efectos de polarización, y adecuado para especímenes mantenidos en recipientes plásticos (ver Figura. 3.7).

Accesorios adicionales requeridos: Objetivos especiales y condensadores especiales [14].

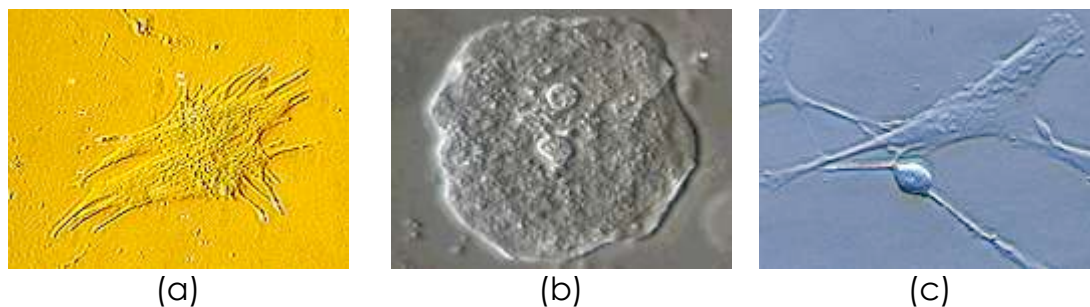


Figura. 3.7 (a) Célula miocardia de la rata. (b) Especimen de fase delgada en recipiente plástico.
(c) Neuronas.

Contraste de interferencia diferencial

El método de contraste de interferencia diferencial de Nomarski, también conocido como DIC (differential interference contrast) por sus siglas en ingles, muestra las diferencias de fase a manera de relieve. El método es excelente para especímenes densos no teñidos ($> 5 \mu\text{m}$) (ver Figura. 3.8).

Accesorios adicionales requeridos: Accesorios de iluminación especiales (relativamente caros) y desviadores de rayos de luz para formación de imagen como prismas y polarizadores [14].

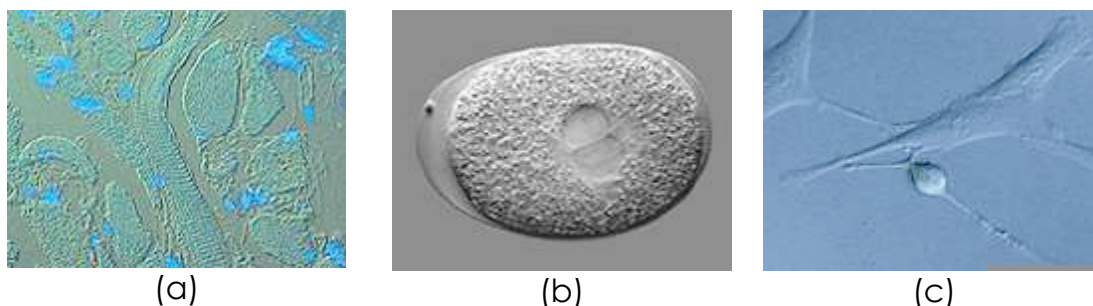


Figura. 3.8 (a) Lengua teñida con fluorescencia (DAPI). (b) Embrión. (c) Neuronas.

Contraste de polarización

Permite la identificación de estructuras birrefringentes (estructuras cristalinas) en especímenes biológicos. Esta técnica es usada también en metalografía y cristalografía. Estos análisis son realizados con luz reflejada y transmitida (ver Figura.3.9).

Accesorios adicionales requeridos: Polarizador, analizador, compensador [14].

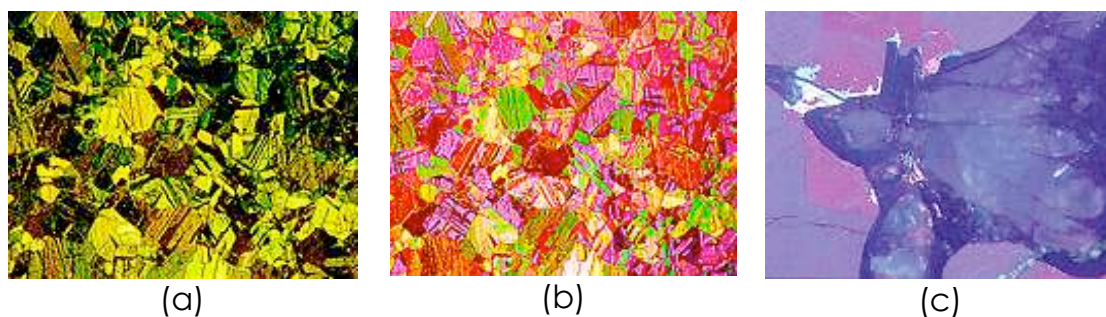


Figura. 3.9 (a) Espécimen de cobre fragmentado usando luz polarizada sin compensador. (b) Espécimen de cobre fragmentado usando luz polarizada con compensador, (c) Mineral de sulfuro.

Fluorescencia

El método de fluorescencia usa fluoro cromos y técnicas especiales de etiquetado (tinción) para visualizar diminutos tejidos y estructuras de células. Expone los especímenes a luz azul, violeta o ultravioleta, como resultado el objeto fluórese (ver Figura. 3.10).

Accesorios adicionales requeridos: Iluminación de luz reflejada, filtros, selectores dicróicos de emisión, fuente de excitación de luz [14].

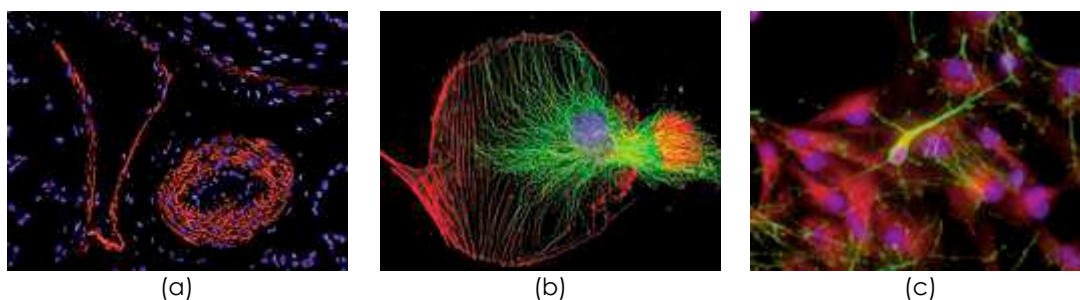


Figura. 3.10 (a) Duodeno de la rata. (b) Células de Endotelio (capa de células que cubre el interior de los vasos sanguíneos). (c) Células embrionarias de una rata.

Detección y caracterización automática de células

Las principales motivaciones para realizar el análisis de células es:

- Incrementar el número de muestras procesadas a fin poder establecer programas masivos.
- Establecer medidas cuantitativas de los aspectos significativos de la célula a fin de reducir la subjetividad de la prueba.

El proceso de análisis visual de imágenes microscópicas es una tarea tediosa que involucra la observación de una gran cantidad de muestras, los resultados de este tipo de análisis dependen de diversos factores como la fatiga y el cansancio. Es por este motivo que desde hace muchos años se ha trabajado en el diseño de técnicas e implementación de algoritmos de detección de objetos de interés con mayor o menor éxito. Sin embargo es problemático todavía tener un algoritmo de segmentación que funcione bien en todas las

situaciones. Ante este problema, en esta tesis se ha creado una herramienta que evalué la eficacia que poseen los algoritmos de segmentación sobre un cierto tipo de muestra, y basado en sus resultados se pueda seleccionar el algoritmo que brinde óptimos resultados.

La tarea del análisis de muestras de células cervicales aparenta ser una tarea susceptible para automatización y de gran atractivo comercial. En el pasado, muchas empresas han invertido grandes cantidades de dinero en la creación de sistemas que faciliten la detección temprana del cáncer uterino. Pero hasta ahora no han tenido el éxito esperado.

Visión por computador en análisis de células

En el área de análisis de imágenes médicas uno de los retos a través del tiempo ha sido la caracterización de las lesiones por medio de imágenes. Esto ha sido posible gracias a la generación de nuevas técnicas computacionales y al desarrollo tecnológico que han permitido la creación de aplicaciones automáticas y semi-automáticas, las cuales han sido empleadas en diversas áreas de la medicina.

Muchos de los esfuerzos para facilitar la detección temprana de cáncer cérvico uterino se han centrado en dos principales áreas: Mejorar las técnicas de recolección/preparación de la muestra y la automatización del diagnóstico. En la Figura. 3.11 se presenta un cuadro de las técnicas planteadas para lograr este propósito.

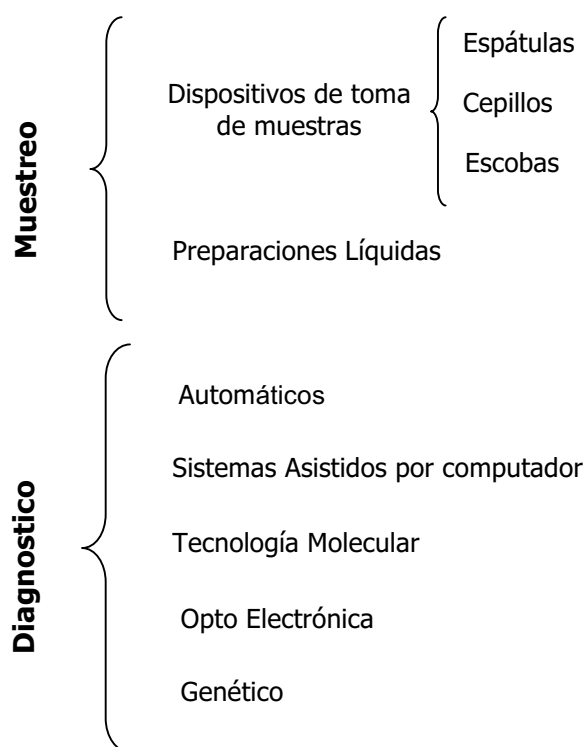


Figura. 3.11 Tecnologías en donde se han aplicado mejoras a fin de contribuir al trabajo del investigador en citopatología.

Para solucionar esta problemática muchas compañías han desarrollado sistemas comerciales que son usados principalmente en los Estados Unidos y Japón.

Estos sistemas dentro de los Estados Unidos fueron evaluados y aprobados por la FDA (*Food and Drug Administration*) y se consideran

como una ayuda en el diagnóstico del cáncer cervico-uterino, además se encuentran cubiertos por algunos seguros médicos.

En la tabla 3.1 se expone como han evolucionado los sistemas basados de análisis de imágenes a través de los años y los resultados que han aportado en esta área de investigación.

Periodo	Sistema	Observación
50's	Cyto-analyzer [EEUU]	Totalmente automático (alimentador de slides y auto enfoque, cámara de video análoga Circuitos de procesamiento especiales). Salida=histogramas de tamaño de células, lógica fija, caro y no preciso.
70's	CYBEST CYDAT TICAS LEYTAS SAMBA CELLO	Primeros sistemas en usar computadoras. Sufrían de limitación en poder de computación, estos sistemas se usaron para exploración de algoritmos y cuantificación de la morfología celular y de sus patrones.
80's	BioPEPR Fazytan Cerviscan LEYTAS DIASSCANNER	Prototipos de dispositivos totalmente automáticos, algunos reportaron rendimiento dentro del rango de el análisis convencional. Ninguno alcanzó el mercado. Muchos grupos observaron que la sensibilidad y especificidad eran altamente dependiente de la calidad de la muestra.
90's	CERVIVIF CYTOSAVAN CYTOSCAN-110	Nuevas técnicas de preparación de muestras fueron desarrolladas. Nuevas tinciones fueron desarrolladas para hacer que la intensidad sea lineal con respecto a la cantidad de DNA. Nuevas características nucleares fueron extraídas pero la sensibilidad estaba limitada por la resolución óptica del microscopio.
Tabla. 3.1 Evolución de los sistemas comerciales utilizados para la detección temprana del cáncer.		

A mediados de los 90's algunas compañías comenzaron a ofrecer productos comerciales para análisis masivo, entre ellos:

- PapNet, de Neuromedical systems
- Autopap de NeoPath,
- AutoCYTE SCREEN de AutoCyte,
- TracCELL 2000 de Accumed International.

PapNet y Autopap fueron autorizados por la FDA (Food and Drug Administration) para control de calidad de muestras y Autopap fue el único autorizado para análisis primario por la FDA. Accumed International fue adquirida por Molecular Diagnostic y trabaja ahora en marcadores genéticos para detección de cáncer. Después de años de disputas sobre violación de propiedad intelectual NeoPath, AutoCyte y Neuromedical systems se unieron en una sola compañía Tripath Imaging.

Tripath comercializó hasta el 2000, FocalPoint basado en Autopap, ahora sus ingresos provienen mayormente de las preparaciones líquidas. Esta técnica evita que las células se sobrepongan unas con otras, mejorando la calidad de la muestra y en consecuencia su análisis. Gran parte de las soluciones planteadas por los sistemas

mencionados anteriormente, usaban preparaciones líquidas antes de procesar la imagen y pronunciar un diagnóstico.

Los sistemas automatizados de visión por computador han permitido obtener un avance importante en tareas de detección y caracterización de células, pero a pesar de esto, existen tareas que no han sido resueltas satisfactoriamente todavía. Entre las que se pueden mencionar:

- Células agrupadas son hasta ahora imposibles de analizar individualmente, solo es posible extraer información contextual.
- Bajo poder discriminatorio de los cambios en los núcleos entre células normales y anormales.
- El análisis de células individuales a partir de muestras requiere propiedades cognitivas del sistema de percepción humano no reproducibles por computadoras hasta ahora.

Una estimación de costos por pacientes en Estados Unidos demuestra que: la implementación de alguna de estas tecnologías incrementa el valor de la prueba convencional de 1 a 4 veces su valor nominal. La inversión en este tipo de sistemas no es justificable debido a que no brindan una solución completa al problema y encarece el valor de la

prueba. Es más, el método tradicional, aunque es subjetivo presenta buenos resultados. A esa conclusión también arribaron los consumidores y es por ello que su uso es limitado.

Diagnóstico del cáncer uterino

En nuestro medio las pruebas comúnmente realizadas para la pronta detección del cáncer cervical se basan en dos tipos de exámenes de diagnóstico: el *Papanicolau* (PAP), en donde se analizan estáticamente imágenes celulares (nivel microscópico) y la *Colposcopia*, donde se analiza temporalmente la superficie de la cerviz (nivel macroscópico).

Para nuestros experimentos nos centraremos únicamente en el análisis de imágenes estáticas obtenidas por medio de la digitalización de muestras conseguidas durante el examen de Papanicolau. La prueba de Papanicolaou (llamada también prueba de Pap o frotis de Pap) es un examen que el médico realiza para chequear signos de cáncer en la cerviz (cuello de la matriz). La cerviz es parte del útero (matriz). La prueba de Papanicolau verifica si hay cambios anormales en las células cervicales y puede detectar un cambio precanceroso

en las células. Este cambio se llama displasia cervical. Las células anormales pueden convertirse en cáncer si la displasia no es descubierta y tratada a tiempo. Además el PAP puede detectar infecciones virales del cuello del útero, como el virus del papiloma humano (HPV) y herpes genitales, puede descubrir infecciones vaginales, como las causadas por levaduras o tricomonas. Algunas veces, puede dar información sobre las hormonas, especialmente sobre la progesterona y el estrógeno.

Antes de revisar los tipos de anomalías que pueden ser encontrados durante el análisis de células cervical describiremos el procedimiento a seguir para la recolección de la muestra:

Con un espéculo (instrumento que se usa para abrir la vagina), se toma una muestra de las células del cuello uterino. Se utiliza un hisopo (espátula de madera o pipeta) de algodón para obtener células de la superficie de la muestra obtenida. Estas células se colocan sobre una lámina de vidrio y se envía al laboratorio para que un citopatólogo las analice. En la Figura.3.12 vemos un ejemplo.

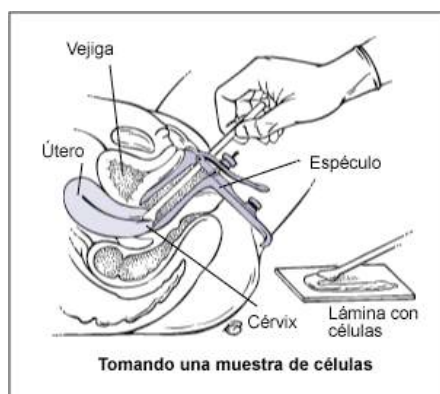


Figura. 3.12 Procedimiento realizado para la recolección de la muestra.

Interpretación de los resultados de la prueba PAP

Las células están compuestas de dos elementos principales: el núcleo y el citoplasma. Estos poseen una característica particular: el citoplasma es siempre de mayor tamaño que el núcleo. Idealmente todos los núcleos de la muestra son de tamaño similar, al igual que todos los citoplasmas. El objetivo del citopatólogo, es encontrar núcleos que posean deformaciones (en tamaño y forma) que los diferencie de otros objetos de su misma clase. A este tipo de deformaciones es lo que se conoce como displasia. Entre más marcada sea esta deformación la gravedad de la displasia aumenta. Existe un valor numérico que nos permite relacionar el tamaño entre el núcleo y el citoplasma denominado índice núcleo-citoplasma. Se lo calcula dividiendo el tamaño del núcleo sobre el tamaño del citoplasma. Un valor pequeño indica que no existe deformación en los

núcleos. Por el contrario un valor grande implica que existe una anomalía en esa célula. Además de estos elementos podemos ver en la Figura. 3.13 que existen otros tipos de estructuras celulares de menor tamaño que no influyen en el resultado del diagnóstico.

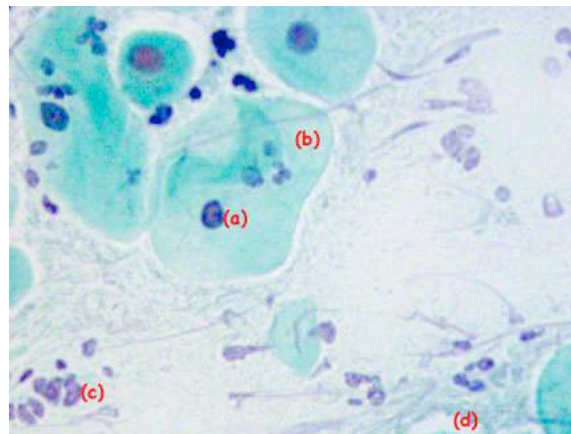


Figura. 3.13. Muestra de células cérvico uterina. (a) Núcleo de la célula. (b) Citoplasma. (c) Algún tipo de bacteria. (d) Diversos elementos celulares.

Negativo o Normal: En la Figura 3.14 se puede notar que todos los núcleos en la muestra son de tamaño similar y además el índice núcleo-citoplasma es bajo,

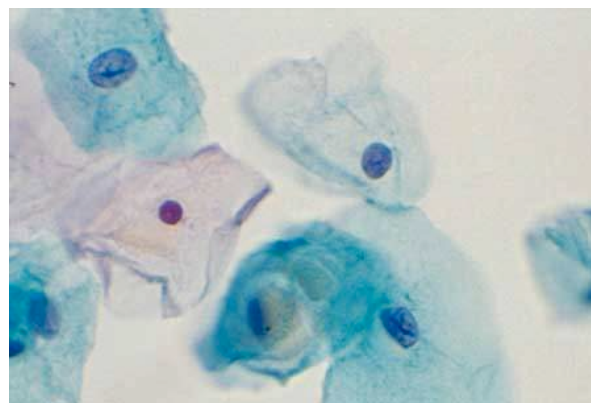


Figura. 3.14. Células sanas con resultado negativo o normal

Displasia: El índice núcleo-citoplasma es un poco elevado en comparación con las otras células (Figura.3.15). Además la saturación en el color es más granular y los bordes de los núcleos ligeramente irregulares. Dependiendo del grado de estas alteraciones la displasia se la clasifica en tres etapas: leve, moderada y grave.

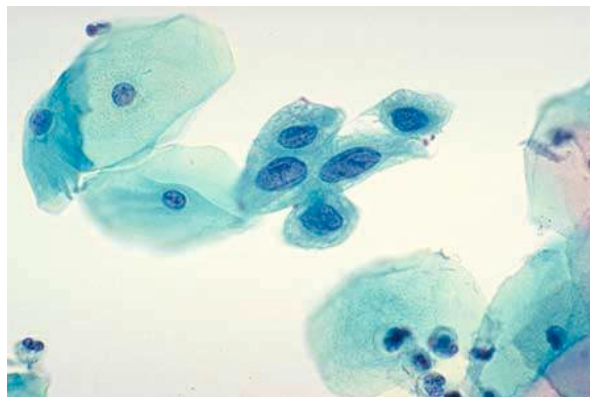


Figura.3.15 Displasia leve de las células cervicales.

Cáncer: El fondo de la muestra está parcialmente recubierto por glóbulos blancos y rojos. Los núcleos presentan una irregularidad importante de tamaño y forma, se pierde por completo la relación de tamaño entre núcleo y citoplasma. Como se lo puede apreciar en la Figura.3.16.

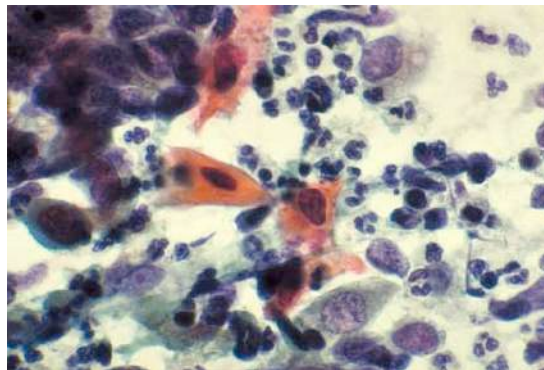


Figura. 3.16. Células cancerosas.

No existe todavía un método de clasificación que permita una clara discriminación entre el grado de gravedad de la displasia, pero se han propuestos sistemas que usan términos descriptivos acerca de las anomalías de las células, como el sistema Bethesda. Ninguno de estos sistemas se ha adoptado como estándar en la comunidad médica, pues no se llega a un consenso acerca de cual sistema es mejor.

El diagnóstico del citopatólogo depende mayormente de la correcta detección de un gran número de células que lo faculten a expresar un criterio acerca de la uniformidad o no, en tamaño de los objetos de estudio. Y basado en esta información, poder discriminar el grado de enfermedad. Por tal motivo se ha visto la necesidad de realizar una evaluación cuantitativa de los resultados de los algoritmos de segmentación que faciliten la tarea de búsqueda de células anormales durante el análisis y además ratifiquen los resultados

obtenidos durante el diagnóstico. Especialmente cuando factores como el cansancio y la repetitividad del examen producto de analizar un gran número de muestras sean motivos de equivocación en el diagnóstico.

Sistema de adquisición

Para trabajar con imágenes microscópicas, los principales componentes que se necesitan para su adquisición y digitalización son:

- Microscopio
- Cámara y software de adquisición
- Computador con tarjeta digitalizadora firewire (puerto de alta velocidad).

Microscopio



Figura. 3.17 Microscopio Carl Zeiss, modelo Axioskop 2 plus, equipado con cámara.

Según lo revisado en la sección 3.2.2 el microscopio óptico convencional (ver Figura. 3.17) es el más indicado para el análisis de muestras de células cérvico uterinas. Tomando en cuenta que se va a trabajar con muestras teñidas durante el análisis de las muestras, se configuró el microscopio para que trabaje en campo claro. Adicionalmente el microscopio escogido tiene la capacidad de ser configurado para trabajar con todos los tipos de técnicas de microscopía. En la tabla 3.2 se describe el microscopio utilizado para la adquisición de las muestras.

MICROSCOPIO	
NOMBRE:	MICROSCOPIO TRINOCULAR
MARCA:	CARL ZEISS
MODELO:	AXIOSKOP 2 PLUS
CARACTERÍSTICAS:	Microscopio configurado para campo claro en luz transmitida a este equipo se le puede adaptar fluorescencia a futuro. Objetivos achroplan 10x y objetivo plan neofluar 40x.
Tabla. 3.2 Características del microscopio utilizado.	

Cámara y software



Figura. 3.18 Cámara AxioCam MRc5.

La cámara AxioCam MRc5 de alta definición de la empresa Zeiss fue usada durante la etapa de adquisición de muestras (ver Figura. 3.18). Esta cámara es empleada en una amplia variedad de áreas como medicina, biología y ciencia de materiales. En esta sección se detalla las principales características que posee la cámara escogida:

- Captura digital de imágenes a color: 36-bit color verdadero.
- Alta velocidad de adquisición: Transmisión puerto firewire/IEE1394 velocidad de 400 mega-bits. Se transmiten 400 bits por segundo (no píxeles). Es adecuado si se pretende analizar células vivas.
- Alta resolución: 5 Mega-píxeles.
- Sistema de disparo interno y externo. Este permite tomar una foto sobre la secuencia de video.
- Velocidad máxima de digitalización 18 MHz (tiempo que le toma adquirir un cuadro de imagen).
- No requiere fuente de alimentación externa, ya que esta es suministrada a través del puerto firewire. El modelo escogido sólo la requiere, si este es conectado con una portátil. En ese caso se usa un HUB FireWire / IEEE 1394.

- Su pequeño tamaño permite una óptima integración a cualquier sistema, aprox. 11 cm. x 8 cm. x 4,5 cm. y pesa 370 g.
- Operación libre de vibración (no contiene sistemas mecánicos de enfriamiento, como ventiladores, obturadores o filtros con engranajes).
- La cámara usa el adaptador del tipo C, comúnmente empleado para adaptarla al microscopio. Gran parte de los microscopios usan esta tipo de adaptador.
- El Software de adquisición AxioVision permite operar todas la funciones provistas por la cámara.
- El software de desarrollo es propietario, lo que implica que se tiene que pagar un costo adicional si se pretende implementar módulos de software adicionales. Por el contrario otras cámaras proveen entornos de desarrollo libres.

Para mayor información de las principales características que debe poseer una cámara que sea empleada en el análisis de muestras refiérase a [15].

Capítulo 4

CAPITULO IV

ANÁLISIS Y DISEÑO DE HERRAMIENTA PARA EVALUAR ALGORITMOS DE SEGMENTACION

Especificación de los requerimientos y alcance del sistema.

Es necesario antes de implementar una solución de software hacer el levantamiento de información con la que se va a trabajar y definir los límites del sistema, una manera de conseguir esta información es determinar los requerimientos necesarios para que la aplicación funcione, así como también los requerimientos que no intervengan directamente en el desarrollo de la aplicación.

La herramienta que se plantea esta enfocada en la evaluación de técnicas de segmentación sobre imágenes de células, antes de implementarla es necesario entender los objetivos que se pretenden

alcanzar, información que es obtenida en la fase de análisis del sistema.

Requerimientos funcionales

Se requiere una herramienta que evalúe la efectividad que poseen los algoritmos de segmentación al ser aplicados sobre un banco de imágenes común, los resultados o gráficos obtenidos de la evaluación deben permitir caracterizar el comportamiento de las técnicas evaluadas y brindar al investigador un criterio que le permita seleccionar el mejor parámetro o técnica que logre óptimos resultados durante el proceso de detección de objetos. Se espera que con el uso de esta herramienta se pueda disminuir el tiempo de desarrollo de soluciones en visión por computador, justificar por medio de resultados cuantitativos la efectividad de una solución planteada y obtener un claro conocimiento acerca del comportamiento de las técnicas evaluadas.

También se requiere una aplicación que permita generar una plantilla en base a la manipulación de contornos y parámetros de algoritmos de umbralización aplicados sobre la imagen, que permitan la segmentación manual de objetos, esta debe poder clasificar en una

misma plantilla diferentes tipos de objetos y guardar esta información para su posterior modificación.

La generación de plantillas es importante debido a que estas aportan información valiosa acerca de los objetos que se pretenden segmentar, como el tamaño, color, números de objetos, forma y porcentaje de píxeles que ocupa una clase de objeto en particular. La plantilla debe ser generada en formato binario, para facilitar la extracción de información cuantitativa de los objetos segmentados manualmente. Esta información captura la experiencia del experto y sirve de referencia antes de implementar las técnicas de segmentación que se pretendan evaluar.

Las pruebas de la herramienta serán usando imágenes de células. Para la cual debe implementarse técnicas de segmentación comúnmente usadas para el análisis de este tipo de muestras.

Requerimientos no funcionales

Es necesario establecer requerimientos externos que son importantes durante el uso de la aplicación, estos pueden ser tanto del tipo económico, de facilidad de uso, instalación, escalabilidad, desarrollo entre otros. Las principales se detallan en esta sección.

Es necesario que usuarios ajenos al área de visión por computador puedan aprender el uso de la herramienta en poco tiempo y que las decisiones tomadas por estos sean respaldadas con resultados cuantitativos obtenidos de la evaluación.

El desarrollo de nuevos algoritmos de segmentación para su posterior evaluación debe ser realizado en algún tipo de herramienta que facilite su implementación y posea documentación suficiente acerca de las librerías de procesamiento de imagen utilizadas en el desarrollo.

El diseño de la herramienta debe permitir evaluar el mayor número de algoritmos posibles y no limitar la cantidad de muestras a analizar, con el fin de dar un resultado global de la prueba. En consecuencia de esto, el proceso de evaluación será por lotes.

Usuarios del sistema.

Los potenciales usuarios del sistema son investigadores y profesores.

El investigador usa la herramienta como fuente de verificación confiable que le permita comparar la efectividad de su algoritmo con los ya existentes, este estudia su comportamiento y realiza la búsqueda del mejor algoritmo o parámetro que obtenga buenos

resultados al ser ejecutados sobre un banco de imágenes común, de modo que basado en estas soluciones, se puedan crear prototipos rápidos que permitan demostrar teorías sencillas, además los resultados generados de la evaluación pueden ser citados en informes técnicos.

Los profesores son usuarios que usan el sistema con fines educativos, el uso de la herramienta les permite caracterizar el comportamiento de los algoritmos de segmentación y comparar los resultados obtenidos. Asimismo, le ayuda a familiarizarse con las principales técnicas de segmentación existentes de manera práctica y visual. Esta también ayuda a entender la importancia del proceso de segmentación en el desarrollo de aplicaciones basadas en visión por computador. Adicionalmente la importancia que tiene la generación de plantillas para describir cuantitativamente la muestra les permite una mejor comprensión de los sujetos a estudio. Todas estas experiencias adquiridas por el uso del sistema pueden ser transmitidas a los estudiantes en el aula de clases.

Análisis de aplicaciones y herramientas de desarrollo

Antes de implementar la herramienta requerida se probaron diferentes alternativas: el uso de lenguajes multiplataformas de código abierto como QT, GTK y Java. El uso de herramientas para la plataforma Windows como visual C++. En el primer caso estas herramientas usan para el desarrollo el esquema MVC, modelo vista controlador, el objetivo principal es separar la parte funcional de la parte gráfica, estas vienen con un diseñador de interfaces como QT Designer, Glade y Swing respectivamente, mientras que la parte funcional puede ser implementada con C o C++ para los dos primeros y Java para el tercero. Para desarrollar aplicaciones de visión por computador y para disminuir el tiempo de desarrollo es necesario utilizar librerías de procesamiento digital de imágenes [34], como IPL (Image Processing Library) [35], openCV [36], Java 2D [37] las dos primeras usan código C y tienen su implementación tanto para Linux como para Windows, la última es independiente de la plataforma debido a que no trabaja directamente con el sistema operativo sino a través de su máquina virtual. Además es importante tomar en cuenta que antes de obtener ejecutables multiplataformas el código debe ser compilado en el sistema operativo pertinente, lo que implica instalar software de

emulación adicional en un sistema operativo si se pretende generar todos los ejecutables dentro de una misma plataforma. También se probó con herramientas Microsoft como visual C++ donde se pueden integrar las mismas librerías de procesamiento de imágenes.

La principal desventaja del uso de estas soluciones es encontrar un conjunto de librerías de desarrollo bien documentadas que contenga la mayor cantidad de algoritmos implementados, además se requiere la codificación de grandes porciones de código si se necesita usar librerías de otro tipo tales como: estadísticas, cálculo matemático, algebra o interfaces para graficar funciones matemáticas.

También se probó con herramientas de desarrollo de uso investigativo como Matlab 7.0 tanto la versión para Windows como para Linux, este un lenguaje interpretado orientado a la manipulación de matrices y de amplio uso en la industria y centros de investigación en todo el mundo, este posee una sintaxis simple al momento de codificar algoritmos y provee un entorno de desarrollo fácil de instalar. Matlab viene con un amplio número de librerías (toolbox) bien documentadas para el desarrollo de aplicaciones. En particular, las de procesamiento digital de imágenes; el único inconveniente que presenta es la limitación que tiene en el desarrollo de interfaces gráficas, por lo que

se intentó combinarlo con lenguajes como C y Java, pero esta idea no fue factible debido a que existía incompatibilidad con ciertas librerías, que lo hacían inestable para desarrollar una aplicación completa. Una característica interesante es que se puede crear códigos ejecutables a partir de algoritmos desarrollados en Matlab; incluso pequeñas aplicaciones que cuenten con interfaces gráficas.

De todas las soluciones planteadas según los requerimientos establecidos, se optó por Matlab como herramienta de desarrollo con el fin de aprovechar la gran variedad de librerías con las que cuenta, como las de procesamiento digital de imagen, análisis estadísticos, manejo matemático de matrices, interfaces para gráficos estadístico tanto de dos a n dimensiones, redes neuronales entre otras. Aunque la creación de interfaces gráficas no es su fuerte se implementaron controles adicionales, rutinas de manejo del mouse y se mejoraron las rutinas de visualización de imágenes con el fin de crear una aplicación fácil de usar. Además la instalación, uso de la herramienta y creación de nuevos algoritmos no es una tarea compleja.

Debido al amplio rango de funciones bien documentadas que posee Matlab para el desarrollo de aplicaciones en visión por computador, las prestaciones más importantes que brindan a este proyecto son:

implementar la mayor cantidad de algoritmos de segmentación, disminuir el tiempo de implementación de algoritmos y crear una herramienta que permita la evaluación y caracterización de algoritmo de segmentación.

Es importante notar que las soluciones encontradas durante la evaluación pueden ser implementadas como pequeños prototipos ejecutables con interfaces gráficas no muy complejas, tanto para la plataforma Windows como Linux, y de esta manera realizar pequeñas pruebas en tiempo real antes de implementar una solución definitiva.

Matlab es ampliamente usado por investigadores, ingenieros y científicos de todo el mundo para la enseñanza, simulaciones, creación de nuevas técnicas y pruebas de hipótesis planteadas, puesto que posee potentes herramientas de cálculo matemático y su uso esta orientado a un sin número de aplicaciones.

Especificaciones de casos y subcasos de uso.

En esta sección se especifican los casos de uso y escenarios que provee el sistema, estas son herramientas UML (Unified Modeling Language – Lenguaje de modelamiento unificado) que permiten

especificar, visualizar, construir y documentar sistemas. Los casos de uso representan la interacción entre un usuario y el sistema.

Los usuarios pueden ser personas o máquinas que participan en los casos de uso cuando realizan una tarea en el sistema.

Los escenarios son descriptores formales del flujo de eventos que ocurren durante un caso de uso, los cuales son descritos mediante diagramas de secuencias del sistema en la sección 4.6.

Especificación de casos de uso.

A continuación se listan los casos de uso que se han considerado en el sistema.

1. Usuario genera segmentación manual.
2. Usuario guarda resultados de los algoritmos de umbralización.
3. Usuario genera plantilla.
4. Usuario implementa algoritmos de segmentación para evaluación.
5. Usuario realiza la evaluación y obtiene los resultados.

A continuación se describen la funcionalidad que capturan los casos de uso del sistema.

Nombre: 1. Generar segmentación manual.

Descripción: El usuario puede crear, manipular y eliminar contornos sobre la imagen, estos permiten clasificar diferentes tipos de objetos mediante la asignación de colores.

Notas: Los contornos generados manualmente sobre la imagen son guardados y están relacionados con esta.

Se permite verificar visualmente que los contornos estén cerrados pasando el mouse sobre ellos.

Se puede visualizar el centroide o el área de los objetos segmentados manualmente.

Subcasos de uso:

1.1 Se crea, modifica o elimina el contorno.

1.2 Se puede cambiar el color de los contornos dibujados.

1.3 Se puede deshacer y rehacer acciones realizadas durante la creación de contornos.

1.4 Se puede visualizar el centroide y el área de los objetos segmentados.

Nombre: 2. Guardar resultados de los algoritmos de umbralización.

Descripción: Este módulo, permite ejecutar algoritmos de umbralización tanto manuales como automáticos sobre imágenes representadas en diferentes espacios de color. Además es posible guardar los umbrales y criterios usados para la segmentación. Para que sean probados sobre otro tipo de imagen.

Notas: Se puede generar una imagen binaria resultado de la aplicación de los algoritmos de umbralización.

Permite visualizar el histograma y los umbrales escogidos antes de ejecutar el criterio de umbralización.

Permite analizar la distribución del color después de aplicar un algoritmo de umbralización mediante la comparación de la imagen segmentada sobrepuesta la original.

Subcasos de uso:

2.1 El usuario puede visualizar el histograma de una componente o espacio de color de cualquier imagen.

2.2 Se realiza la segmentación por umbralización definiendo los umbrales manualmente u obteniéndolos mediante la aplicación de un algoritmo de umbralización seleccionado.

2.3 Guardar los umbrales obtenidos ya sea manual o por la aplicación de un algoritmo de umbralización para cada espacio o plano en el que fue probado.

2.4 Aplicación de un umbral guardado sobre otro tipo de imagen.

2.5 Generación de imagen binaria a partir del criterio de umbralización escogido.

Nombre: 3. Usuario genera plantilla

Descripción: El usuario genera una imagen binaria ya sea de contornos o de áreas, que son obtenidas a partir de líneas creadas durante segmentación manual de objetos.

Notas: Todos los objetos que poseen un mismo color son de la misma clase.

Los objetos segmentados manualmente que posean un mismo color tienen un nombre que los identifica, el cual es usado para generar la imagen binaria resultante y puede ser editado.

La aplicación siempre genera dos plantillas: la de contornos y la de áreas de los objetos segmentados manualmente.

Subcasos de uso:

3.1 Asignación de nombres a cada tipo de objeto segmentado.

3.2 Visualización de las imágenes binarias de los objetos segmentados manualmente.

3.3 Generación de plantillas de contornos y áreas obtenidos a partir de los objetos segmentados.

Nombre: 4. Implementar algoritmos de segmentación para evaluación.

Descripción: El usuario implementa algoritmos de segmentación para ser evaluados, este debe generar una imagen binaria por cada parámetro que se pretenda evaluar.

Notas: Este es un proceso por lotes.

El usuario puede basarse en la plantilla generada para sacar características comunes de los objetos segmentados y fijar parámetros.

El usuario debe generar la imagen binaria con un nombre representativo para cada parámetro que se pretenda evaluar.

Subcasos de uso:

4.1 Generar imágenes binarias fijando parámetros característicos de la imagen a evaluar.

4.2 Generar imágenes binarias variando aleatoriamente todos los parámetros.

Nombre: 5. Usuario realiza la evaluación y obtiene los resultados.

Descripción: Se realiza la evaluación de los resultados producidos por los algoritmos de segmentación con su respectiva plantilla generada manualmente. Aquí se evalúa el rendimiento de los algoritmos por

medio del análisis de curvas ROC y tablas con información obtenidas de la evaluación.

Notas: Este módulo permite comparar las imágenes binarias de manera visual.

Se genera un archivo de texto con los resultados de la prueba.

Es posible obtener información de un punto de la curva haciéndole click sobre el punto o pasando el mouse sobre el.

Subcasos de uso:

5.1 Se visualizan las imágenes evaluadas y los resultados obtenidos de la prueba.

Especificación de subcasos de uso.

A continuación se realiza la especificación de subcasos de uso que pueden ser obtenidos al interactuar el usuario directamente con la herramienta.

Caso de uso 1: Generar segmentación manual.

Subcaso de uso 1.1: Se crea, modifica o elimina contornos.

Es posible generar contornos que permitan clasificar los objetos contenidos sobre la imagen. Para esto se debe usar herramientas que permitan crear, modificar y eliminar líneas. La adición de líneas debe permitir agregar segmentos nuevos a contornos que no estén

cerrados, la eliminación de líneas debe permitir eliminar secciones del contorno, para que usando una combinación de estas dos herramientas permitan la modificación de los contornos dibujados sobre la imagen.

Notas: Los contornos creados están relacionados con la imagen sobre la que se las creó.

Esta información es guardada para su posterior modificación.

Es posible generar contornos de diferente color.

Subcaso de uso 1.2: Se puede cambiar el color de los contornos.

Se debe poder contar con herramientas que permitan cambiar el color de los objetos clasificados manualmente. Es necesario contar con una lista de colores que posibilite clasificar los diferentes objetos de la muestra como una clase de objeto en particular.

Notas: Todos los objetos del mismo color agrupan objetos de una misma clase, lo que implica que deben ser almacenados junto con los objetos que posean su mismo color.

Los cambios realizados en el color de los objetos deben actualizar los controles que se visualizan en la sección de generación de plantillas.

Subcaso de uso 1.3: Se pueden deshacer y rehacer acciones realizadas durante la creación de los contornos.

Es necesario registrar las acciones realizadas durante la creación de contornos. Esto permite corregir errores de edición.

Notas: Estas herramientas solo serán usadas para registrar tareas realizadas durante la generación de contornos.

Subcaso de uso 1.4: Se puede visualizar el centroide y el área de los objetos segmentados.

Es necesario que se extraiga información de los objetos segmentados manualmente como el centroide y el área, información que debe ser visualizada sobre la imagen.

Notas: El tipo de información que puede ser obtenida de los objetos segmentados depende de las necesidades del usuario.

Caso de uso 2: Guardar resultados de los algoritmos de umbralización.

Subcaso 2.1: El usuario puede visualizar el histograma de una componente o espacio de color de cualquier imagen.

El usuario puede seleccionar un espacio o plano de color que genere una nueva representación de la imagen. Además, se debe visualizar el histograma que muestre la distribución de los valores contenidos en

los píxeles del espacio o plano de color de la imagen con la que se este trabajando.

Nota: Se deben actualizar los controles que grafican los umbrales sobre el histograma, así como la información de los nuevos límites que contienen a todos los valores de los píxeles de la imagen seleccionada.

Al seleccionar un espacio o plano de color se debe cargar la información de los umbrales previamente guardada.

Subcaso 2.2: Se realiza la segmentación por umbralización definiendo los umbrales manualmente u obteniéndolos mediante la aplicación de un algoritmo de umbralización seleccionado.

Permitir al usuario seleccionar un umbral o un rango de umbrales que puedan ser visualizados gráficamente sobre el histograma, También se puede estimar un umbral aplicando un algoritmo de umbralización automático, el cual sirve de referencia. Una vez seleccionado el umbral se pueden aplicar los siguientes criterios. Mayor igual que un umbral T1, menor igual que un umbral T2, que este entre T1 y T2, o que no se encuentre entre estos dos valores.

Nota: Se debe visualizar la máscara obtenida del proceso de umbralización.

Subcaso 2.3: Guardar umbrales obtenidos para cada espacio o plano color en el que fue probado.

Se debe de almacenar los umbrales y criterios de umbralización por cada plano o espacio de color que se este visualizando, con el objetivo de probar si ese mismo umbral segmenta los mismos tipos de objetos al ser aplicados sobre una imagen del mismo tipo. Además se debe guardar esos valores con un nombre que los identifique.

Notas: El umbral debe ser guardado sólo cuando se aplique un criterio de umbralización y se haya visualizado la máscara binaria generada.

Si el umbral al ser cargado sobre otra imagen del mismo tipo no se encuentra en el rango de valores que delimiten los píxeles contenidos en la imagen, se presenta un mensaje de error y se coloca los umbrales en estos límites.

Nota: Se debe cargar la lista de umbrales guardados cada vez que se selecciones un espacio o plano de color.

Subcaso 2.4: Aplicación de un umbral guardado sobre otro tipo de imagen.

Cada vez que se tenga un umbral cargado, este debe ser aplicado automáticamente cuando se cargue otro tipo de imagen. Si la imagen no coincide con el espacio o plano de color cargado este

debe realizar todo el proceso, primero el de conversión al espacio de color seleccionado y después aplicar el umbral.

Nota: Aplicar el criterio de umbralización sobre la imagen visualizada, cuando se seleccione un umbral guardado.

Subcaso 2.5: Generación de imagen binaria a partir del criterio de umbralización escogido.

La aplicación debe generar una imagen binaria a partir de la máscara generada por los criterios de umbralización, a esta imagen se le debe asignar un nombre que la identifique.

Nota: Esta imagen debe ser guardada en formato bmp.

No se puede generar el archivo con la imagen binaria, si no se ha ejecutado previamente un criterio de umbralización.

Caso de uso 3: Usuario genera plantilla.

Subcaso 3.1: Asignación de nombres a cada tipo de objeto.

La aplicación debe ser capaz de asignar un nombre a los objetos segmentados, dependiendo del color del contorno con que fueron dibujados. Este es el nombre del archivo que se generará a partir de los objetos segmentados manualmente.

Nota: Se debe presentar una lista con los tipos de objetos segmentados manualmente los cuales se diferencia por el color de los contornos.

Subcaso 3.2: Visualización de las imágenes binarias de los objetos segmentados manualmente.

Cada vez que se seleccione un tipo de objeto del cual se desee generar una imagen binaria, este debe ser visualizado tanto en una sección de vista previa, como en la sección de visualización principal. Además, se debe poseer una opción que permita visualizar los contornos o las áreas de los objetos segmentados.

Notas: Se debe sobreponer la máscara generada sobre la imagen original, esto ayuda a visualizar que los objetos de interés hayan sido segmentados correctamente.

Subcaso 3.3: Generación de plantillas de contornos y áreas obtenidos a partir de los objetos segmentados.

Se debe presentar una lista con los nombres de los archivos generados tanto de contornos como de áreas, estos corresponden a todos los tipos de objetos segmentados manualmente. Las imágenes binarias generadas deben ser guardadas en carpetas diferentes.

Notas:

El nombre de los archivos generados tiene la extensión bmp.

Todas las imágenes binarias de contorno generadas terminan con la palabra outline, mientras que las de área terminan con mask.

Caso de uso 4: Implementar algoritmos de segmentación para evaluación.

Subcaso 4.1: Generar imágenes binarias fijando parámetros característicos de la imagen a evaluar.

Asunciones:

Se deben definir los algoritmos que serán implementados. Si es posible se puede fijar parámetros basándose en información cuantitativa extraída de la plantilla y variar solamente los parámetros que se pretendan evaluar.

Notas: Se genera una imagen binaria por cada parámetro que se ha variado.

Se almacena en una carpeta todos los resultados obtenidos de la ejecución de los algoritmos.

Subcaso 4.2: Generar imágenes binarias variando aleatoriamente todos los parámetros.

Antes de implementar los algoritmos de segmentación se deben definir una lista con la combinación de los parámetros que se desea

evaluar, por cada entrada en esta lista se debe generar un archivo cuyo nombre identifique a los parámetros evaluados.

Notas: Se genera una imagen binaria por cada parámetro que se ha variado.

Se almacena en una carpeta todos los resultados obtenidos de la implementación de los algoritmos.

Caso de uso 5: Usuario realiza la evaluación y obtiene los resultados.

Subcaso 5.1: Se visualizan las imágenes evaluadas y los resultados obtenidos de la prueba.

Una vez que se realiza la evaluación, se podrá visualizar las imágenes evaluadas las cuales pueden ser sobrepuestas sobre la plantilla, lo cual permite constatar los resultados obtenidos durante la evaluación. Además, se podrá obtener información de cada punto de la curva ROC.

Nota: Se deben presentar los resultados obtenidos por orden de efectividad.

Es posible seleccionar con el puntero del mouse los puntos generados y obtener información de los resultados obtenidos en ese punto.

Se debe generar un archivo de texto con los resultados de la prueba.

Diseño lógico del sistema

En esta sección se describe como se relacionan los módulos del sistema, en la Figura 4.1 se presentan los tres módulos principales con que cuenta la herramienta: módulo generador de plantillas, módulo de segmentación semi-automática y módulo de evaluación. El primero captura la información del experto en el área, el segundo genera todas las posibles soluciones que permiten detectar los objetos de interés. Finalmente el módulo de evaluación coge los resultados obtenidos de ambos módulos y los compara generando la gráfica ROC y una tabla con los resultados obtenidos.

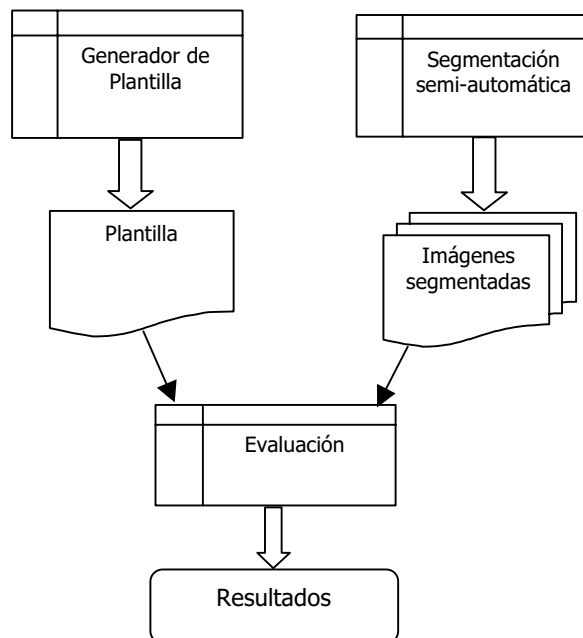


Figura 4.1. Diseño lógico del sistema.

Módulo de generación de plantillas

Este módulo permite la creación de una imagen binaria a partir de objetos segmentados manualmente usando herramientas de edición de contornos o algoritmos de umbralización. Además provee una sección donde se puede estudiar la distribución del color sobre el histograma del espacio o plano de color seleccionado. En las Figuras 4.2 se muestra un esquema de las funcionalidades que brinda este módulo. Este módulo cuenta con un conjunto de imágenes las cuales pueden ser segmentadas ya sea manualmente o por la aplicación de algoritmos de umbralización, ambos resultados pueden ser guardados dentro del proyecto. Finalmente se generan imágenes binarias resultado del proceso de segmentación.

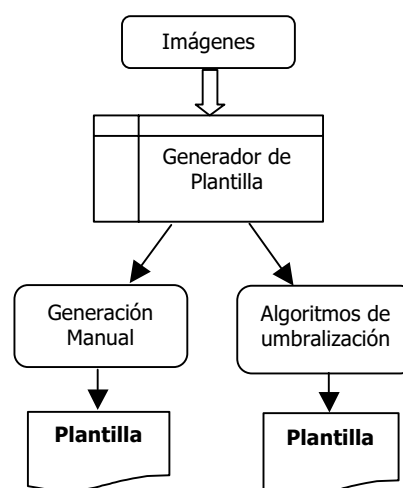


Figura 4.2. Diseño lógico del módulo generador de plantillas.

La aplicación debe permitir crear, guardar y abrir un proyecto existente, que contenga información de los objetos segmentados manualmente y de los umbrales que fueron probados sobre diferentes espacios de color.

La herramienta debe contener una sección en donde se pueda agregar o eliminar imágenes al proyecto, por cada imagen cargada se debe mostrar su respectivo histograma, asimismo el usuario puede seleccionar el espacio o plano de color con el que desee trabajar. Es importante mostrar información de la imagen cargada como el nombre, espacio y plano de color seleccionado.

La sección de visualizar la imagen (sobre la que se está trabajando) debe ser común para todas las funcionalidades que brinde el sistema. Este debe permitir mostrar cinco tipos de imágenes: (1) la imagen real, que es la imagen tal como la percibe el usuario, (2) la imagen actual, que es la imagen representada en el plano o espacio de color seleccionado, (3) la imagen máscara, que es la imagen binaria generada ya sea por la aplicación de un método de umbralización o por la segmentación manual, (4) la máscara sobrepuesta la imagen original, la cual permite obtener una idea de los objetos que han sido segmentados por la máscara y (5) la imagen con contornos

generados usando herramientas de segmentación manual. Independientemente del tipo de imagen que se este visualizando se debe poder ampliar la imagen y poder navegar sobre esta.

La herramienta debe poseer un submódulo que permita la creación de contornos de diferentes colores, usando herramientas de dibujo sobre la imagen, también se pueden aplicar algoritmos que faciliten la creación de contornos. Este también debe poseer herramientas que permitan deshacer y rehacer las últimas acciones realizadas por el usuario.

Para facilitar la creación de contornos se debe poder ampliar el tamaño de la imagen de modo que se pueda segmentar objetos más pequeños. Además este debe poder visualizar el área y el punto central de cada objeto segmentado manualmente.

Otra funcionalidad importante que debe brindar la herramienta es que se pueda segmentar la imagen aplicando umbrales definidos manualmente u obtenidos aplicando algoritmos de umbralización. Estos deben ser graficados sobre el histograma del espacio o plano de color seleccionado. Además, se debe mostrar información del valor máximo y mínimo de los píxeles contenidos dentro de la imagen

mostrada. Una vez obtenido el umbral se puede aplicar los siguientes criterios de umbralización: que el valor de los píxeles de una imagen sean mayor igual que un umbral T1, menor igual que un umbral T2, entre un umbral T1 y T2 o que no se encuentre entre este rango. Adicionalmente, se debe almacenar los umbrales y criterios de umbralización usados para segmentar objetos en cada espacio o plano de color en donde estos fueron aplicados.

Finalmente, este debe poseer un módulo que permita generar las imágenes binarias a partir de los objetos segmentados manualmente. Las imágenes generadas deben poseer un nombre que identifique el tipo de objetos segmentados. Asimismo por cada clase de objeto segmentado se debe generar una imagen binaria de contornos y de área.

Módulo de segmentación semi-automática

En la Figura 4.3.se muestra un esquema de cómo funciona el módulo de segmentación semi-automática. Se define una lista de parámetros los cuales van a variar dependiendo de lo que se quiera evaluar. Esta lista de parámetros es pasada al algoritmo de segmentación implementado y se genera una imagen binaria por cada parámetro de la lista.

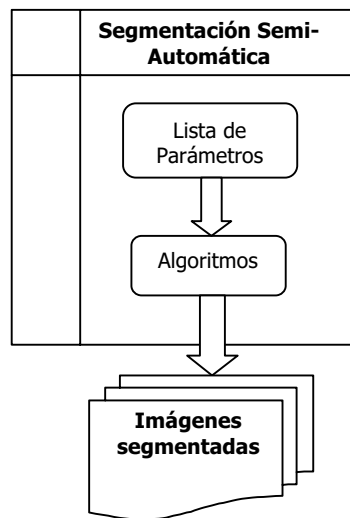


Figura 4.3. Diseño lógico del módulo de segmentación semi-automática.

En este módulo se implementa algoritmos de segmentación y se define los parámetros que se desean evaluar. Es posible fijar parámetros en los algoritmos implementados extrayendo información de la plantilla generada, estos valores deben ser característicos de los sujetos a analizar. Esto permite estudiar la influencia de un parámetro en particular.

También se pueden definir combinaciones de los parámetros que se deseen evaluar. Es necesario que las imágenes generadas por la variación de estos parámetros contengan un nombre que identifiquen los parámetros evaluados.

Los algoritmos implementados deben usar procesamiento por lotes para generar los resultados de la segmentación, esto permite que se procese gran cantidad de información y se genere un gran volumen de datos.

Módulo de evaluación

Este módulo realiza el proceso de comparación entre la plantilla y los resultados generados por los algoritmos de segmentación implementados. Este genera la curva ROC y una tabla con los resultados de la evaluación (ver Figura 4.4). Asimismo permite constatar los resultados obtenidos, al sobreponer la plantilla con la imagen binaria generada.

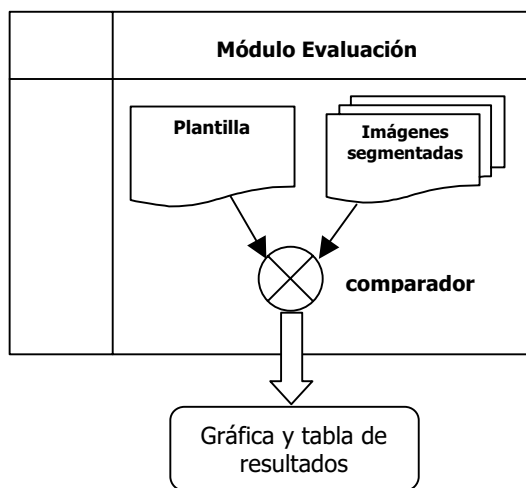


Figura 4.4. Diseño lógico del módulo de evaluación.

Diagrama de secuencia del sistema

Estos diagramas describen gráficamente la interacción entre los objetos del sistema. En estos diagramas se ponen los objetos en la parte superior a lo largo del eje horizontal y una escala de tiempo va en el eje vertical.

A continuación se muestran los principales diagramas de secuencia del sistema, que fueron propuestos en la etapa de análisis. Antes de explicar los diagramas de secuencia del sistema se describen las principales funciones y actores que interactúan en el sistema.

Investigador: este es el usuario que interactúa con la herramienta para obtener un resultado de valor medible.

Coordinador: función que permite administrar las tareas que serán ejecutadas por el usuario del sistema.

Imágenes: objeto que contiene información de la imagen visualizada.

Acciones: función que maneja la lista de acciones realizadas.

Proyecto: función que maneja la información y el comportamiento de todos los controles contenidos en la herramienta.

Umbralización: función que almacena y coordina todo lo referente al submódulo de umbralización.

Plantilla: función que almacena y coordina todo lo referente al submódulo de generación de plantilla.

Algoritmo: algoritmo implementado el cual posee una estructura dada.

Imagen plantilla: Imagen generada con la herramienta de segmentación manual.

Lista plantillas: objeto que mantiene información de las plantillas.

Lista imágenes generadas: objeto que mantiene información de imágenes binarias generadas.

Evaluación: función que se encarga de actualizar los controles contenidos en la interfaz del módulo de evaluación.

Generador resultado: función que realiza cálculos y genera gráfica ROC.

Repositorio: carpeta donde se almacena las imágenes binarias.

Una vez definido las principales funciones y actores que interactúan directamente en el sistema se describen los principales escenarios del sistema.

En la Figura. 4.5 se muestra la secuencia que se debe seguir para crear un contorno sobre la imagen. El investigador selecciona la tarea de generar contornos eligiendo una herramienta para la creación de

líneas, después usa esta herramienta sobre la imagen cargada, la acción realizada sobre la imagen es almacenada en una lista de acciones. El coordinador actualiza los controles del proyecto y le da una retroalimentación de la acción fue realizada con éxito.

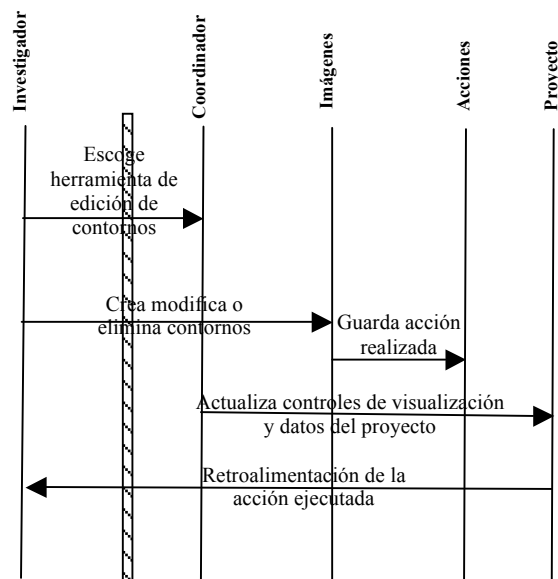


Figura 4.5. Generar segmentación manual con éxito.

En la Figura. 4.6 se muestra la secuencia realizada para generar una imagen binaria a partir de la ejecución de un método de umbralización. Para esto el investigador selecciona un umbral manualmente o toma de referencia uno generado por un algoritmo de segmentación, esta información es enviada a la función que coordina la tarea realizada y aplica el criterio de umbralización seleccionado por el usuario, este criterio sólo es aplicado si la función de umbralización verifica que se encuentra una imagen cargada.

Esta también actualiza los controles de visualización con la imagen binaria generada y muestra la información del criterio de umbralización ejecutado. Una vez obtenido este resultado el usuario elige guardar esta imagen en disco. La función de umbralización es la encargada de generar esta imagen en un repositorio, además le dice a la función proyecto que notifique al usuario que la imagen fue creada en disco.

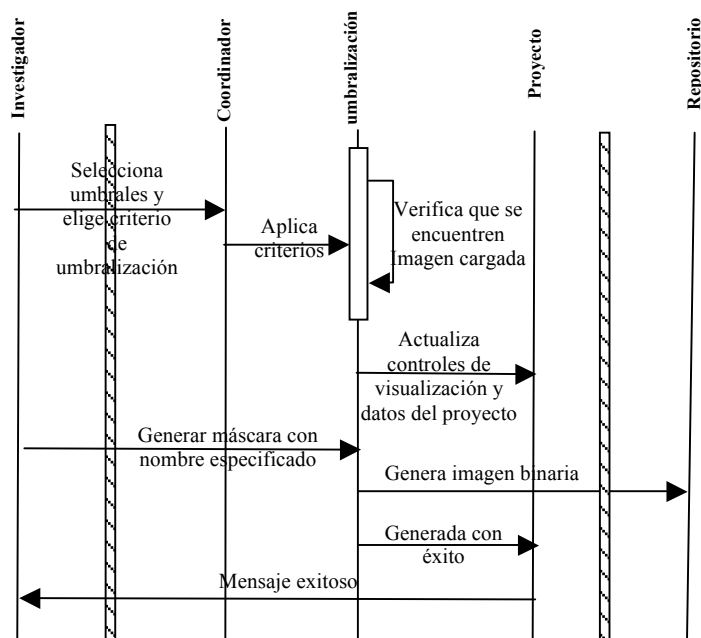


Figura 4.6. Generación exitosa de la imagen binaria a partir de un criterio de umbralización escogido.

En la Figura. 4.7 se muestra el escenario de generar plantillas a partir de los objetos segmentados manualmente. Para lo cual, el investigador realiza la tarea de cargar una imagen que posea los contornos generados, datos que al ser cargados actualizan los

controles usando la función de plantilla, la cual verifica que existan contornos generados sobre la imagen y actualiza los controles de visualización de la imagen. El usuario elige en el módulo de plantillas el tipo de objeto del cual desea generar la imagen binaria. Después solicita a la función plantillas que genere las imágenes en disco. Esta función crea las imágenes tanto de contornos como de área en un repositorio y además le dice a la función proyecto que notifique al usuario que las imágenes se han generado con éxito.

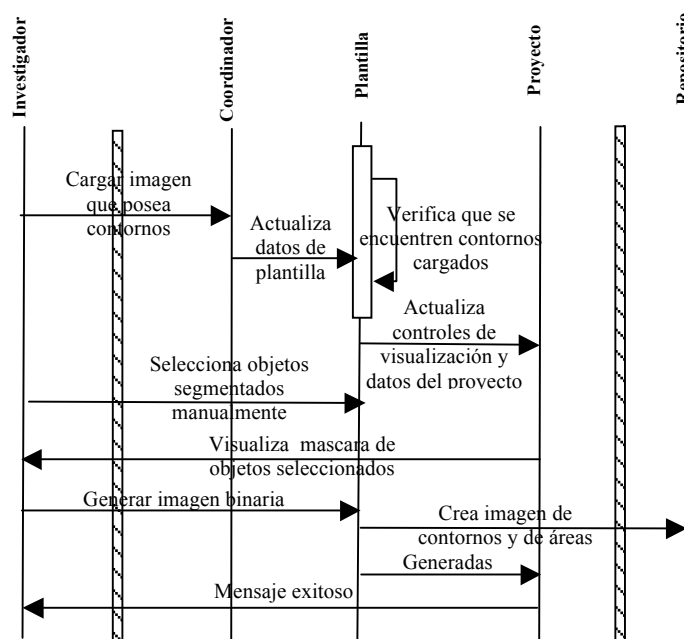


Figura 4.7. Generación exitosa de plantilla de contornos y áreas. Obtenidos a partir de los objetos segmentados.

En la Figura. 4.8 se muestra el escenario en donde se describe como los algoritmos implementados generan las imágenes binarias

correspondientes por cada parámetro a evaluar. En esta etapa se fijan parámetros que son característicos de los objetos de estudio. El investigador extrae información de la imagen plantilla y define los parámetros que serán fijos. En esta etapa se construye una lista con los parámetros que se desea evaluar, esta lista es pasada al algoritmo, el cual es ejecutado y genera las imágenes binarias en disco, una vez que el algoritmo termina envía un mensaje al coordinador para que verifique que todas las imágenes se hayan creado con éxito. Además, esta función notifica al usuario que el algoritmo terminó de ejecutarse.

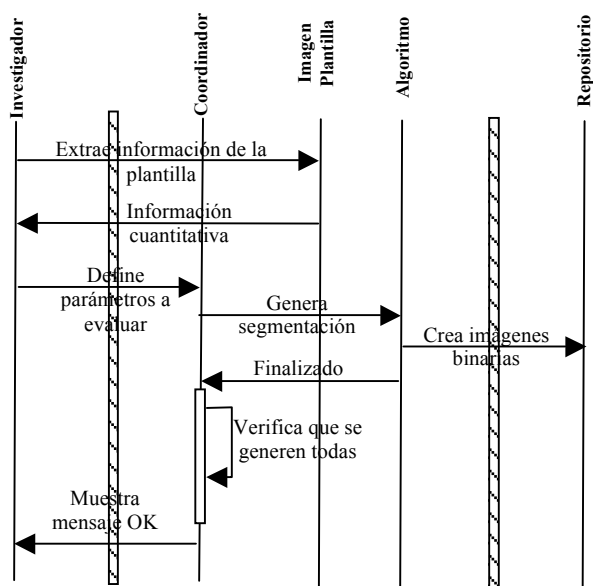


Figura 4.8. Generar imágenes binarias con éxito fijando parámetros característicos de la imagen a evaluar.

En la Figura 4.9 se muestra el escenario en donde la evaluación es ejecutada satisfactoriamente. Una vez que el usuario carga la plantilla y los resultados obtenidos por los algoritmos de segmentación la función coordinadora carga la lista de plantillas y la lista de imágenes generadas y ejecuta la evaluación, tarea que es realizada por el módulo de evaluación. Este realiza el proceso de comparación y envía los resultados obtenidos a la función generador de resultados, la cual genera la curva ROC y actualiza todos los controles con los resultados de la prueba. Además, genera un archivo de texto con los resultados de la evaluación.

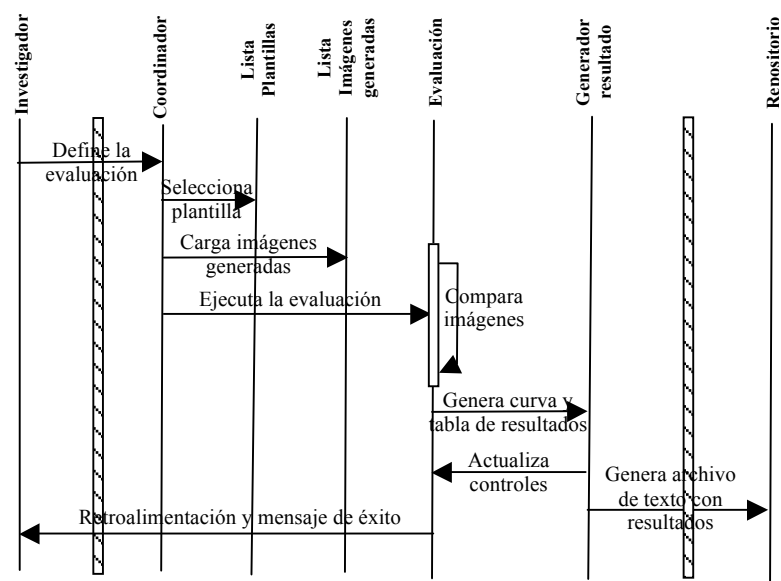


Figura 4.9. La evaluación es ejecutada satisfactoriamente.

Capítulo 5

CAPITULO V

IMPLEMENTACION Y PRUEBAS DE LA HERRAMIENTA

Introducción

En este capítulo, se describe la implementación de la herramienta que permite caracterizar el comportamiento de algoritmos de segmentación, según los requerimientos establecidos durante la etapa de análisis y diseño. Para esto se codificaron tres módulos importantes:

- Módulo de segmentación semi-automático.
- Módulo de segmentación automático.
- Módulo de evaluación.

Estos se relacionan de la siguiente manera: el primer módulo fue creado para generar la plantilla, la cual captura la información del resultado que se espera obtener.

El segundo módulo es donde se codifican los algoritmos que van a ser evaluados y se establece los parámetros que van a ir variando al momento de su ejecución. En consecuencia se obtiene una imagen binaria por cada parámetro generado. Estas imágenes generadas corresponden a los posibles resultados que pueden ser obtenidos por la implementación de una técnica particular.

Finalmente el módulo de evaluación realiza la comparación de la plantilla generada en el módulo uno y los resultados de la segmentación generados automáticamente en el módulo dos. Para cada parámetro comparado se genera un valor de sensibilidad y especificidad, valores necesarios para graficar la curva ROC. Una tabla descriptiva de los resultados obtenidos en la curva es también mostrada como referencia. El empleo de esta herramienta nos permite caracterizar el comportamiento de uno o más algoritmos de segmentación sobre una base de imágenes común.

Para conseguir este propósito, se describe el proceso de implementación seguido para el desarrollo de la herramienta, durante la fase de prueba se propone un experimento en donde se evalúa la influencia del color de dos técnicas de segmentación aplicados sobre imágenes de células y se presentan los resultados obtenidos. Asimismo se detalla los problemas presentados durante la implementación. También se elabora un análisis de costos de los equipos usados durante la etapa de adquisición. Finalmente, se presentan los resultados y conclusiones de los principales aportes que brinda la herramienta creada para este proyecto.

Proceso de Implementación.

En la primera parte del trabajo se revisó documentación diversa como: segmentación de imágenes a color, algoritmos de umbralización, técnicas de evaluación, análisis estadístico usando curvas ROC, análisis de imágenes médicas, artículos sobre cáncer uterino, textos referentes a dispositivos de adquisición como microscopio y cámara, revisión de entornos de desarrollo.

Se planteó el desarrollo de una herramienta que evalúe los resultados de los algoritmos segmentación y sirvan de soporte al investigador en

la elección de la técnica de segmentación adecuada para cada tipo de imágenes a analizar. Esta idea surge debido a que los resultados obtenidos del proceso de segmentación depende de varios factores (ver sección 3.6).

Durante la fase de desarrollo se probaron diferentes herramientas, tal como se revisó en la sección 4.3, donde se implementaron pequeños prototipos funcionales usando QT, GTK, Java herramientas de desarrollo multiplataforma, también se instaló una infraestructura que permita el desarrollo con estas herramientas para el sistema operativo Windows, sobre el cual se probó visual C++, además se probó Matlab que es una herramienta de cálculo numérico orientado al análisis de matrices y vectores. Entre todas estas alternativas se optó por Matlab para el desarrollo de la aplicación.

Entre las diversas pruebas realizadas con esta herramienta se intentó integrarlo con código C, C++ y java, para el desarrollo de la interfaz gráfica, pero la incompatibilidad existentes con ciertas librerías, dificultan el desarrollo de la aplicación.

Una de las principales ventajas que ofrece la integración con C o C++ es que se puede implementar código que no es posible implementar

con Matlab como por ejemplo controladores para el manejo de una cámara en particular. Además es posible crear ejecutables a partir de código implementado en Matlab, aquí se pudo generar el compilable de una aplicación con interfaz gráfica totalmente creada en Matlab, la cual pudo ser ejecutada tanto en la plataforma Windows y Linux sin necesidad de tener Matlab instalado. Hay que tomar en cuenta, que como Matlab es un lenguaje interpretado, esta conversión primero traduce el código de Matlab a código C o C++ y después genera el ejecutable.

En la segunda fase, antes de codificar la aplicación, se aprendió a usar el microscopio y se digitalizó las muestras que fueron provistas por APROFE, donde se adquirieron imágenes de cinco muestras con diferente grado de malignidad, según lo revisado en la sección 3.4.2.1. En este proyecto se optó por trabajar con imágenes que no posean anomalías, debido a que estas no presentan complejidades en contenido lo que facilita su estudio y comprensión del análisis.

En la tercera fase, se codificaron interfaces gráficas en Matlab, que nos permitan de manera simple la segmentación manual de imágenes para la generación de plantillas.

Este módulo consta de una interfaz básica para la visualización y magnificación de la imagen, además se pueden sobreponer los resultados de la segmentación sobre la imagen original para verificar la generación correcta de la plantilla. Permite administrar un proyecto y guardar los cambios realizados. También, provee cuatro funcionalidades a través de cuatro interfaces: imágenes, segmentación, umbralización y plantilla. En imágenes se puede agregar, eliminar y aplicar los resultados de algoritmos de umbralización sobre diferentes imágenes y comparar los resultados visualmente. En segmentación se implementó rutinas para el manejo del mouse, que faciliten la creación, modificación y eliminación de contornos. También se puede clasificar objetos por colores y generar contornos con algoritmos que seleccionen píxeles similares a partir de un punto específico de la imagen. En umbralización se implementó una pequeña herramienta que permita visualizar el histograma de un determinado espacio de color y visualmente aplicar umbrales manualmente sobre la imagen del histograma. Se implementaron diez algoritmos de umbralización automáticos que permiten separar los objetos del fondo [21], se puede guardar los umbrales encontrados para ser probados sobre otro tipo de imagen. Este genera una

máscara a partir de la ejecución de algoritmos de umbralización sobre imágenes. La finalidad de este sub-módulo es estudiar como se distribuyen los valores numéricos asignados a cada píxel en cada componente de color.

El módulo de plantilla permite generar una imagen ya sea de contornos abiertos o cerrados para cada tipo de objeto clasificado sobre la plantilla y a partir de estos generar una imagen binaria de contornos o de las áreas que ocupan los objetos.

En el módulo dos se implementaron dos algoritmos de segmentación ampliamente usados en el análisis de células, *Watershed* y *Meanshift*. Se generaron resultados variando el parámetro de color y asignándole al resto de parámetros valores predeterminados dependiendo de la muestra a trabajar con la finalidad que se evalúe la influencia del color durante la segmentación.

Finalmente en el módulo tres se implementó una técnica de evaluación basada en el análisis de curvas ROC, donde se grafica y se resume los resultados de la evaluación, además es posible visualizar y comparar los resultados obtenidos con la imagen original.

La aplicación fue probada en cada etapa de desarrollo, en el módulo uno se probó que: guarde los contornos generados, añada, modifique o elimine contornos, segmente diversos tipos de objetos en una sola plantilla y guarde los resultados de los algoritmos de umbralización. En el módulo dos se verificó que cada algoritmo genere una imagen binaria, resultado de la variación de sus parámetros de entrada. En el módulo tres que genere correctamente los resultados de sensibilidad y especificidad, valores necesarios para la construcción de la curva. Además que visualice la curva y permita comparar los resultados de la segmentación con los de la plantilla generada.

Una vez terminado el proceso de implementación de la herramienta, se presentan las interfases creadas y se describen la funcionalidad que brindan cada una de ellas.

El módulo de generación de plantillas cuenta con cuatro submódulos. En la Figura 5.1 se muestra la pantalla principal de la aplicación. La pestaña de Imágenes permite añadir y eliminar imágenes al proyecto, también brinda la posibilidad de visualizar información del histograma y del espacio o plano de color con el que se está trabajando.

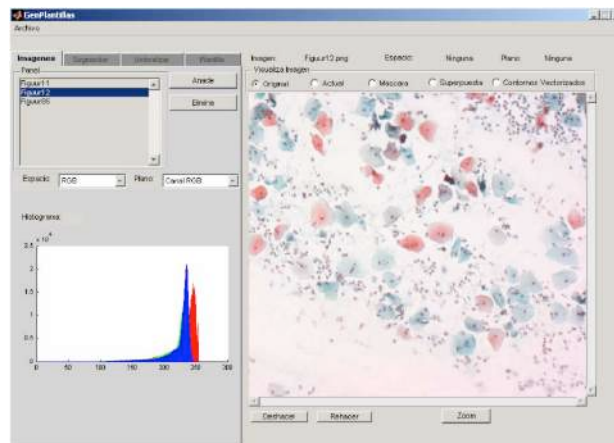


Figura 5.1. Módulo para cargar imágenes.

En la Figura. 5.2 se muestra la pestaña Segmentar, que cuenta con herramientas de edición de contornos que permiten segmentar manualmente objetos con diferente color, asimismo se puede visualizar el área y el centroide de los objetos segmentados. Además registra las últimas acciones realizadas por el usuario durante el proceso de segmentación manual.

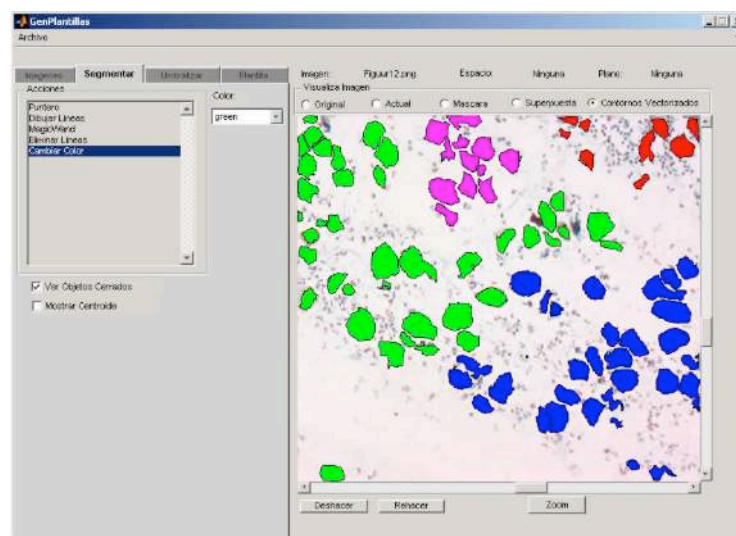


Figura 5.2. Módulo segmentación manual. Se muestran los diferentes objetos segmentados.

En la Figura 5.3 se muestra los distintos objetos segmentados con la herramienta de generación de contornos. En este escenario se ha segmentado una sola clase de objetos, los cuales son identificados con el color verde.

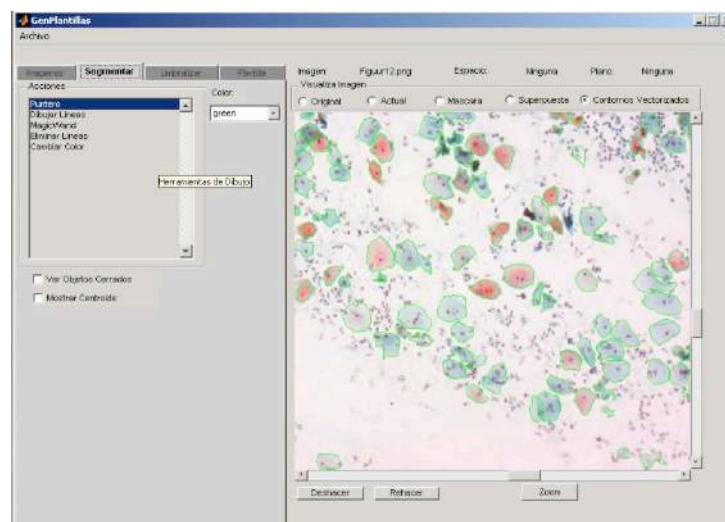


Figura 5.3. Segmentación manual. Se segmenta la imagen mediante herramientas de edición de contornos.

En la Figura. 5.4 se presenta la pestaña de umbralizar, en este escenario se definió un rango de umbrales sobre un plano de color determinado y se aplicó un criterio de umbralización. El resultado de la umbralización es la máscara binaria obtenida al lado derecho de la aplicación.

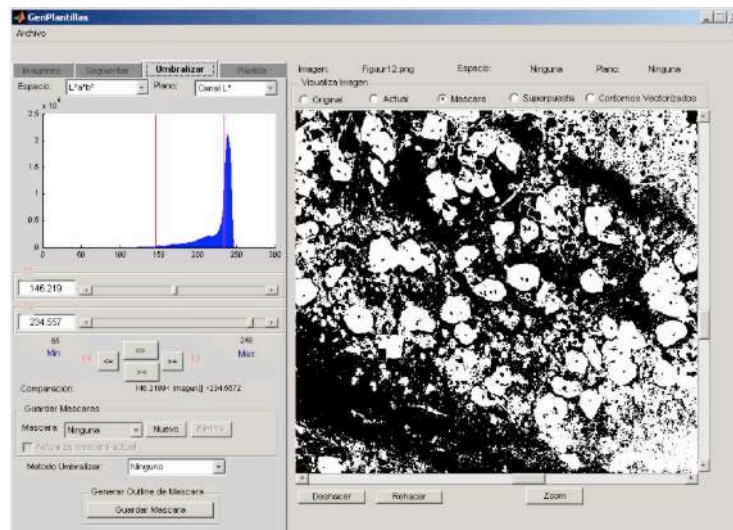


Figura 5.4. Módulo de umbralización de imágenes.

En la Figura 5.5 se muestra la pestaña de Plantilla. Este permite generar una vista previa de la imagen binaria de los objetos segmentados manualmente. Este submódulo permite obtener dos imágenes binarias por cada clase de objeto segmentado, una de contorno y la otra de las áreas que ocupan los objetos.

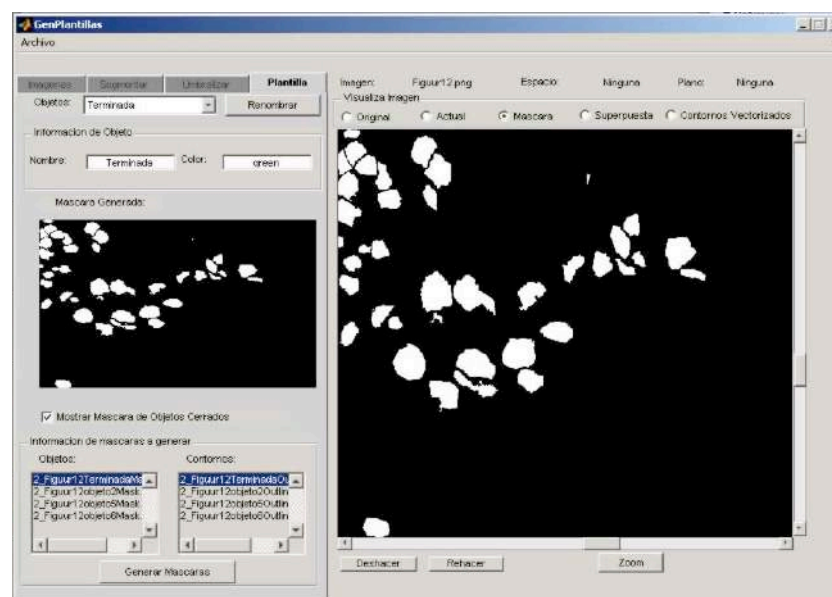


Figura 5.5. Módulo generación de plantillas.

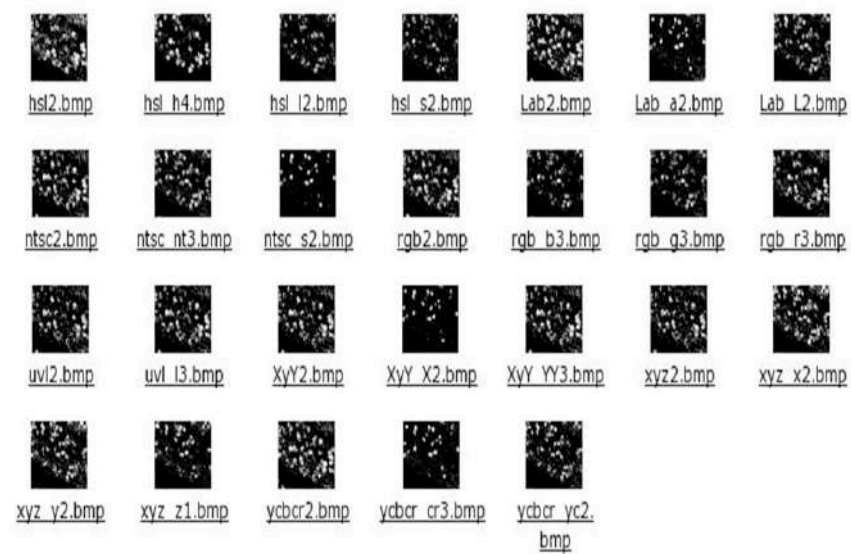
En la Figura 5.6. se muestra el módulo de generación semi-automática de plantillas. El cual muestra la estructura que debe tener el algoritmo de segmentación implementado (ver Figura 5.6 (a)) y las diferentes imágenes binarias obtenidas por la variación de parámetros (ver Figura 5.6 (b)).

```

1 function gen_parameter_template(Image, Template)
2 ParametroList={'RGB':'RGE_R':'RGE_G':'RGE_B':'Lab':...
3   'Lab_L':'Lab_a':'Lab_B':'HSL':'HSL_H':'HSL_S';...
4   'HSL_L':'XYZ':'XYZ_X':'XYZ_Y':'XYZ_Z':'NTSC':...
5   'NTSC_NT':'NTSC_S':'NTSC_C':'UVL':'UVL_U':'UVL_V';...
6   'UVL_L':'XyV':'XyV_X':'XyV_y':'XyV_VV':'ycbcr':...
7   'ycbcr_gc':'ycbcr_b':'ycbcr_cr'};
8
9 path=pwd;
10 path=[path, filesep];
11 name_folder='GradosParam';
12 carpeta=[path, name_folder];
13 if exist(carpeta, 'dir')==0
14     create_folders(path, name_folder);
15 end
16 path=[path, name_folder, filesep];
17 tope=length(ParametroList);
18 objMarker=getCenterMarkers(Template);
19 bgMarker=getbgMarkers(Template);

```

(a)



(b)

Figura 5.6. Módulo generación de plantillas. (a) Algoritmo generado (b) Resultados obtenidos.

En la Figura. 5.7 se muestra el módulo de evaluación, donde se muestra la gráfica ROC y la tabla con los resultados obtenidos del proceso de evaluación.

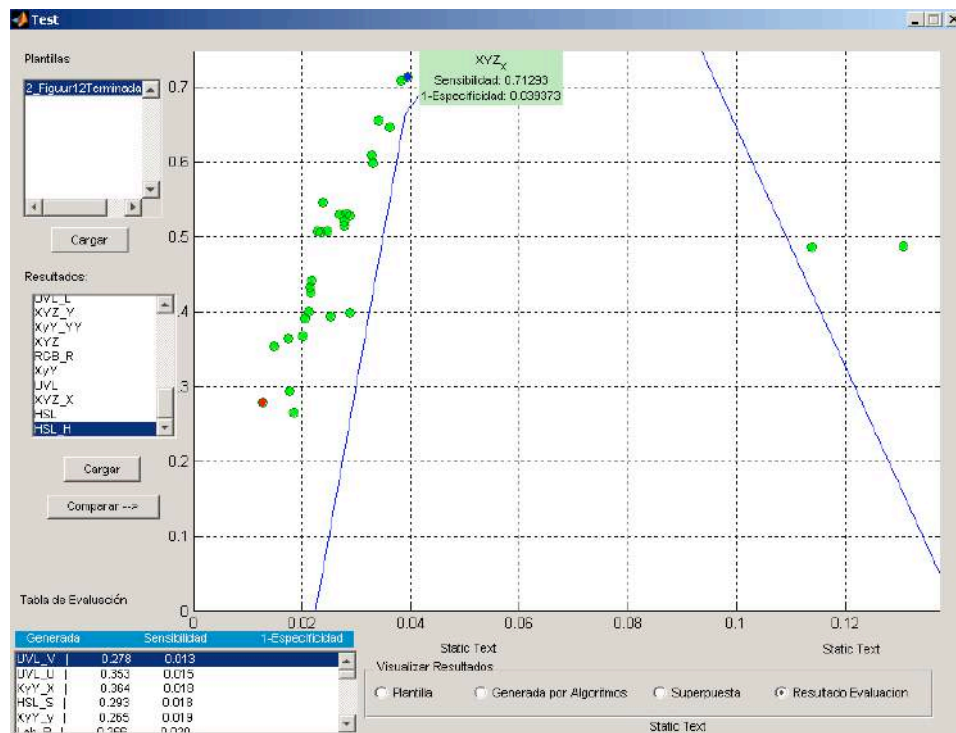


Figura 5.7. Módulo de evaluación.

Problemas presentados en la implementación.

Se detallan los diversos problemas presentados durante la etapa de implementación.

Uno de los principales problemas fue la creación de una interfase gráfica sencilla que permita realizar todas las funcionalidades propuestas usando Matlab, el cual posee controles básicos para su creación. La idea de la herramienta era presentar una sola interfaz que tenga varias funcionalidades, pero que no recargue la interfaz de

controles. Para esto se uso un control que permite la generación de tabs, código que fue descargado del sitio de intercambio de archivos de Matlab. Además se tuvo que mejorar la interfaz para visualizar la imagen a la cual se le añadió barras de desplazamiento y se tuvo que modificar la herramienta de magnificar con la finalidad de que se pueda trabajar fácilmente sobre los píxeles de la imagen.

Debido a las limitaciones que tiene Matlab para la creación de interfaces gráficas se tuvo que implementar funciones que trabajen con eventos del mouse, ya que este cuenta con rutinas básicas, las cuales no pueden ser modificadas.

Al momento de clasificar los píxeles usando herramientas de selección manual de objetos, no se tomo en cuenta, que las coordenadas obtenidas al momento de graficar contornos sobre la imagen eran coordenadas geométricas las cuales contenía decimales, estas coordenadas no coincidían con la ubicación de los píxeles de la imagen, los cuales poseen coordenadas enteras. Este problema fue solucionado usando el algoritmo de Bresenham [38] que permite a partir de dos puntos de la imagen generar una serie de puntos intermedios los cuales coinciden con los píxeles de la imagen.

Para evitar errores de conversión de un espacio de color a otro, hay que considerar los píxeles de la imagen en donde la aplicación de las fórmulas utilizadas para este propósito tomen valores indeterminados.

Otra dificultad encontrada fue poder implementar algoritmos de umbralización que se ejecuten sobre diferentes componentes de color. Para esto se normalizó los valores de cada píxel de la componente de color a su respectivo nivel de gris, de manera que la forma del histograma obtenida por esa componente no se altere y se puede aplicar el algoritmo de umbralización deseado.

Pruebas realizadas.

Con el objetivo de probar la herramienta implementada se definió un experimento donde se analiza la influencia que tiene el parámetro de color durante el proceso de segmentación. Se seleccionó una imagen de células cervicales con diagnóstico normal (que no presente deformaciones), la cual es representativa del grupo de muestras de su mismo tipo y fue adquirida bajo las mismas condiciones de luz. Esta imagen permite diferenciar los colores de los objetos contenidos en la muestra y facilitan la comprensión del análisis. A continuación se detalla la metodología seguida durante el proceso de evaluación. Los

resultados preliminares son analizados y es posible caracterizar el comportamiento de los algoritmos evaluados. Además se puede concluir que espacio o parámetro de color brinda mejores resultados durante la detección de objetos.

Metodología

Una vez implementada la herramienta la estrategia a seguir se detalla.

Se escogieron dos tipos de algoritmos de segmentación de uso generalizado en la comunidad de análisis de bio-imágenes. El primero Watershed, que se basa en los principios de morfología matemática para el crecimiento de regiones y MeanShift que es un algoritmo de agrupamiento de píxeles basado en la búsqueda de modas en histogramas, generados a partir de la descomposición de valores singulares de imágenes.

La metodología utilizada fue el análisis de curvas ROC (Receiving Operating Curves) que miden términos de sensibilidad y especificidad el rendimiento de los algoritmos utilizados. Estas métricas son definidas como la proporción de píxeles detectados correctamente e incorrectamente que coincidan con su estado real.

El parámetro de evaluación que se escogió para las pruebas del sistema fue la componente de color, en el que se busca que espacio o componente de color obtiene mejores resultados durante el proceso de segmentación. Se probaron 12 tipos diferentes de componentes de color para la representación de los píxeles de la imagen. Estos componentes se derivan de espacios de color, cada uno diseñado para realzar algún aspecto (cromaticidad, intensidad, saturación) de la imagen. Además se establecieron parámetros fijos en los algoritmos implementados, estos valores fueron extraídos de la plantilla segmentada manualmente; con la finalidad de que se evalúe sólo la influencia del color durante el proceso de segmentación.

Las imágenes para las pruebas del sistema corresponden a muestras de células cérvico uterinas (provistas por APROFE). De las cuales se escogió una imagen representativa que posea diagnóstico normal para ser analizada durante el experimento. Estas imágenes fueron capturadas con un microscopio Axioskop2 plus equipado con una cámara digital de alta definición AxioCam Mrc5.

Algoritmos utilizados

La implementación de los algoritmos Watershed y Meanshift, es detallada abajo.

El algoritmo de Watershed supone que una imagen de gradiente (ver sección 1.3.2) puede ser interpretada como un perfil topográfico, donde los valores altos de gradiente corresponden a zonas elevadas de terreno y los valores mínimos a valles. Basados en esta analogía, se puede empezar a inundar zonas en donde se encuentren estos mínimos locales (puntos que pueden ser estratégicamente seleccionados y forman parte de los objetos a detectar), hasta que se alcance su tope en las zonas elevadas. A partir de este punto se comienza a identificar todas las zonas inundadas y el punto de origen de cada inundación, los cuales corresponderían a los objetos que se pretenden detectar dentro de la escena, en donde los contornos de estos coinciden con las zonas de alto gradiente (más información se puede encontrar en [20]).

Uno de los puntos claves para la segmentación usando la transformada de Watershed es la elección de puntos que pertenecen a los objetos a detectar, a estos se los conoce comúnmente como marcadores o semillas. Por lo general antes de aplicar la transformada

de Watershed se suele modificar la imagen gradiente, de modo que sus mínimos locales coincidan con los marcadores identificados en la etapa previa. Además se le aplica un preprocesamiento a la imagen de entrada antes de que se procese la función del gradiente. Frecuentemente se usa la imagen gradiente como imagen de entrada en la transformada de watershed, puesto que el principal criterio de la segmentación es la homogeneidad de los valores de gris presentes en la imagen.

Para obtener la segmentación usando la transformada de Watershed se ha aplicado el siguiente algoritmo:

1. Aplicar la magnitud del gradiente sobre la imagen.
2. Procesar los marcadores de los objetos (Estos son los puntos en los cuales comienza el proceso de inundación).
3. Procesar marcadores del fondo. Estos son los píxeles que no forman parte de ningún objeto. Idealmente se quiere que los marcadores del fondo no estén cerca de los contornos de los objetos (lo que ocurre al obtener la imagen binaria del fondo) que nosotros estamos intentando segmentar. Es por esto que se obtienen los marcadores del fondo de manera lineal, aplicándole un método conocido como "esqueletonización por zonas de influencia", o SKiz

sobre la imagen binaria del fondo (objetos tienen valor 1 y el fondo valor 0). Esto se logra utilizando Watershed sobre la imagen distancia [33] de la imagen binaria, e identificando las líneas de contorno detectadas por Watershed. Estas líneas corresponden a los marcadores del fondo.

4. Modificar la imagen gradiente, de modo que ésta tenga sus mínimos locales en la ubicación donde se encuentren los marcadores del fondo y los objetos.
5. Procesar el algoritmo de Watershed sobre la imagen gradiente modificada.
6. Visualizar el resultado de la segmentación. Hay que notar que el algoritmo de Watershed etiqueta cada objeto detectado con un valor numérico. La cantidad de objetos detectados dependen particularmente de la ubicación de los marcadores seleccionados.

En la Figura. 5.8 se muestra la transformada de Watershed. En el primer paso se extrae la imagen gradiente y se seleccionan los marcadores ver Figura.5.8(a). En el segundo paso se inicia el proceso de inundación de Watershed desde los marcadores ver Figura.5.8(b). Finalmente se detecta todos los objetos segmentados Figura.5.8(c).

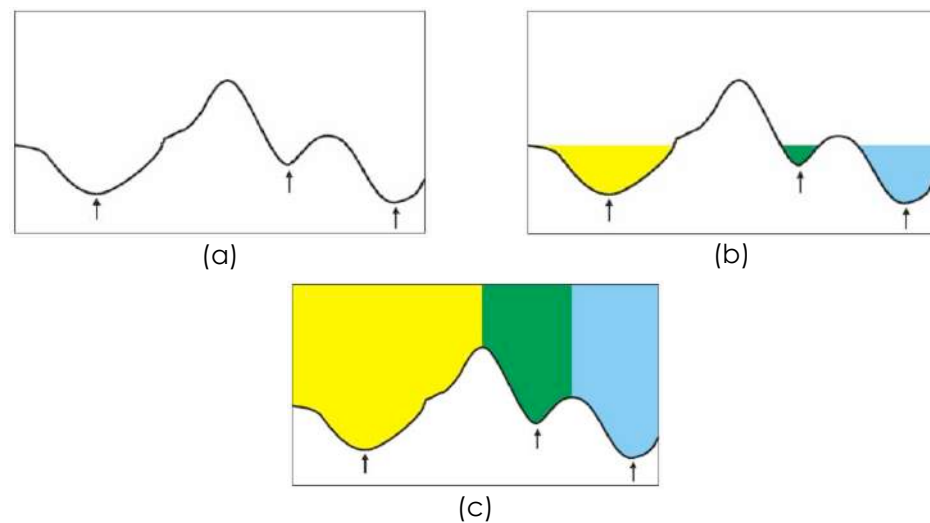


Figura. 5.8. Proceso de segmentación usando el algoritmo de Watershed: (a) Extracción de perfil y selección de marcadores. (b) Proceso de inundación desde los marcadores. (c) Watershed segmentación finalizada.

El otro algoritmo implementado durante la fase pruebas es el algoritmo de Meanshift [32], que es un algoritmo de segmentación de regiones diseñado para encontrar modas (regiones con alta concentración de píxeles) de datos que están representados como vectores de n dimensiones sobre el histograma. Este algoritmo se describe a continuación:

- 1.- Se define el ancho de una ventana de búsqueda.
- 2.-Se escoge un punto de partida en donde se ubicará el centro de la ventana de búsqueda con la cual se iniciará el algoritmo.
- 3.-Se obtiene la media de todos los puntos que se encuentren dentro de la ventana y se traslada el centro de la ventana en esta ubicación.

4.- Se repite el paso 3 hasta que la distancia entre los centros de las ventanas (punto inicial y punto final) sea menor que un umbral determinado.

5.- Cuando termina el algoritmo, el centro de la ventana de búsqueda queda localizada en un pico del histograma (ver Fig. 5.9), este valor es el resultado del algoritmo y los píxeles que se encuentren en el rango de la ventana serán los objetos segmentados.

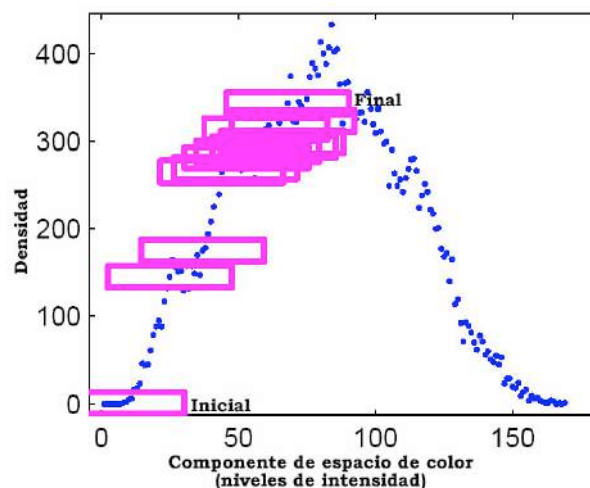


Figura. 5.9. Proceso de segmentación usando el algoritmo de Meanshift.

En este trabajo de tesis, el algoritmo implementado fue diseñado de tal manera que permita encontrar todas las modas posibles dentro del histograma (aplicando Meanshift en cada iteración). Esto se lo logro de la siguiente manera:

1. Se define la variable `min_group_size`. Este valor especifica el mínimo número de píxeles que deben estar contenidos dentro de la ventana de búsqueda con centro en la moda.
2. Se define la variable `min_image_size` como criterio de parada. Este valor indica que la suma acumulada de los valores de las frecuencias del histograma no debe ser menor a este valor.
3. Se construyó el histograma con la distribución de los niveles de gris.
4. A los niveles de gris con valor 0 se les asignó el número de frecuencia 0, esto significa que este nivel de gris no participará en la búsqueda de moda en el histograma.
5. Siempre que la suma acumulada de las frecuencias del histograma sea mayor a `min_image_size`. Ejecuto el paso 6, caso contrario termina la ejecución.
6. Busco el nivel de gris que tenga una densidad de píxeles diferente de cero. Este será el valor inicial en la ventana de búsqueda de Meanshift .
7. Ejecuto Meanshift. Este obtiene la ubicación de la moda y clasifica el número de píxeles contenidos dentro de la ventana de búsqueda como objetos segmentados.

8. Si el número de píxeles contenidos dentro de la ventana de búsqueda (obtenido en el paso 7) es mayor que `min_group_size`, entonces guardo la moda en una variable y establezco en cero el valor de densidad de cada nivel de gris contenido dentro de la ventana de búsqueda. Disminuyendo así el tamaño del histograma y descartando los niveles de gris ya seleccionados.
9. Si no se cumple el paso 8 establezco en cero todos los valores de densidad de los niveles de gris contenidos dentro de la ventana con centro en la moda. Pero no almaceno información de la moda. Esto implica que la cantidad de píxeles detectados como objetos no cumplen con el tamaño de grupo para ser clasificados como tal.
10. Repito el paso cinco hasta que se encuentre todas las modas.
11. Terminado el algoritmo se obtienen todas las modas y en consecuencia todos los objetos detectados que sean mayores `min_group_size`. Estos son etiquetados de tal manera que para todos los niveles de gris que se encuentren dentro de la ventana con centro en la moda se les asigna un valor numérico, el cual es el índice del arreglo que corresponda a cada moda encontrada.

En la Figura. 5.10 se muestra la ejecución del algoritmo. Podemos notar que en la Figura. 5.10 (a) el algoritmo comienza ubicando el centro de la ventana de búsqueda en el nivel de gris que tenga un valor de densidad diferente de cero. Meanshift es ejecutado, el resultado es la moda, la cual corresponde al centro de la ventana de búsqueda y los píxeles contenidos en ella son los objetos detectados. En la Figura. 5.10 (b) notamos que los niveles de gris ubicados dentro de esta ventana de búsqueda se les coloca el valor de cero, por tal motivo, estos píxeles no participarán en la próxima ejecución de Meanshift. Se ejecuta nuevamente el algoritmo de Meanshift ubicando el centro de la ventana de búsqueda en el nivel de gris cuya densidad sea diferente de cero, por lo general es el nivel de gris ubicado más a la izquierda del histograma. El algoritmo de Meanshift es ejecutado iterativamente como vemos en la Figura. 5.10 (c) y 5.10 (d) hasta que se encuentren todas las modas posibles en el histograma y todos los niveles de gris hayan sido clasificados.

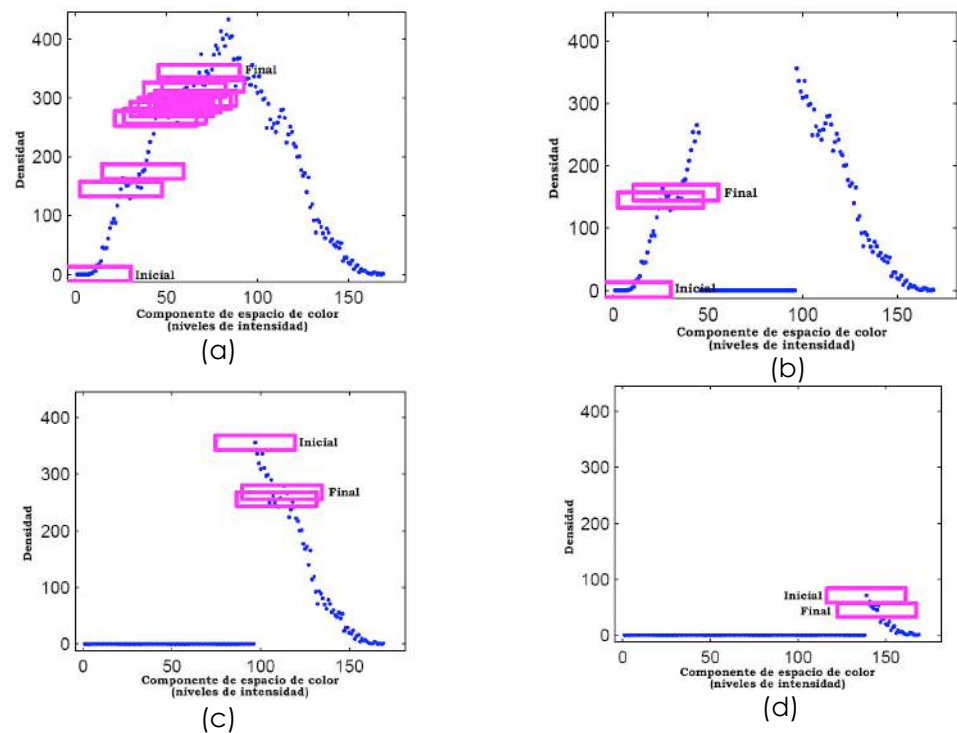


Figura. 5.10. Búsqueda de modas en el histograma aplicando el algoritmo de MeanShift. (a) primera iteración se ejecuta MeanShift y se obtiene la moda. (b) se descarta valores contenidos dentro de la ventana de búsqueda con centro en la moda y se ejecuta nuevamente MeanShift. (c) y (d) MeanShift es ejecutado hasta encontrar todas las modas.

Tipos de imágenes analizadas.

En la etapa de adquisición se digitalizaron muestras con dos tipos de magnificación: 20x y 63x. Las muestras fueron facilitadas por APROFE y el equipo utilizado durante la adquisición (detallado en la sección 3.5) fue facilitado por el Centro de Visión y Robótica – CVR. Se tomaron fotos de cinco muestras que tenían diferentes grados de malignidad y se digitalizaron ochenta fotos que fueron almacenadas en un CD. En la Figura. 5.11 se presenta un ejemplo de las imágenes adquiridas. Hay

que tomar en cuenta, que de una misma muestra se pueden obtener varias imágenes, esto se debe, a que el microscopio enfoca en una sección de la muestra en particular. Las diferentes imágenes tomadas de la misma muestra poseen propiedades similares y fueron adquiridas bajo las mismas condiciones de luz.

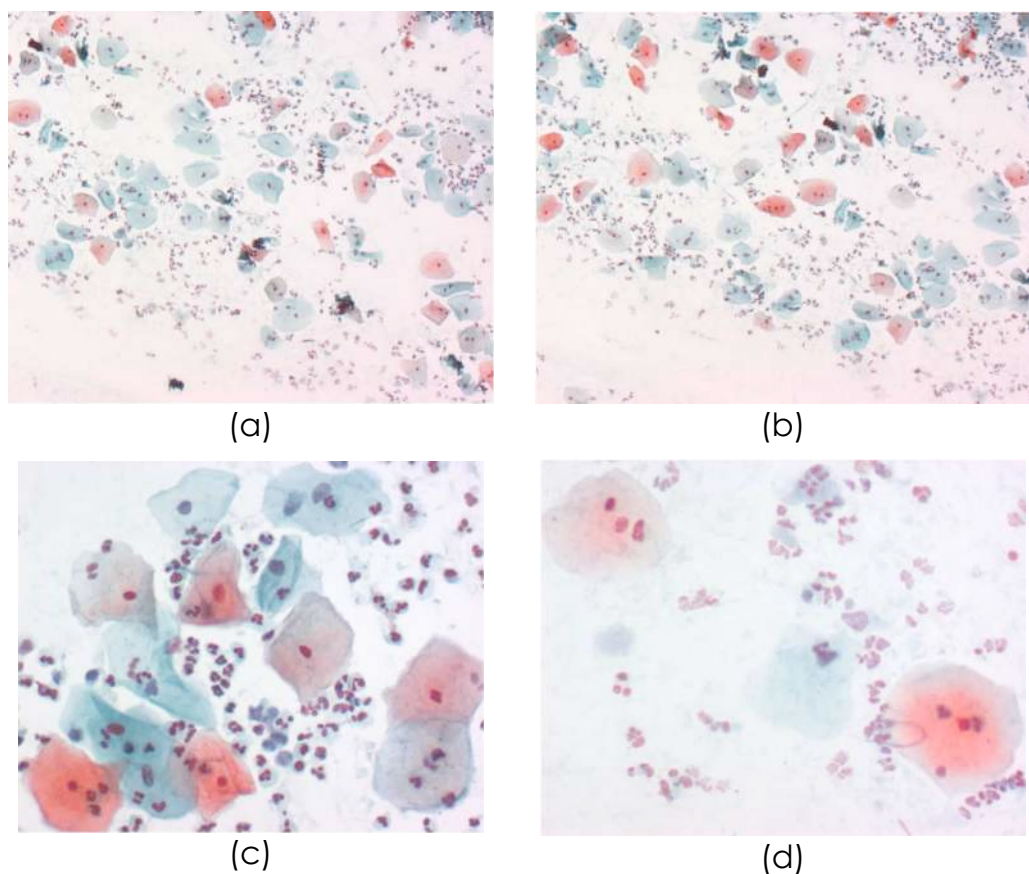


Figura. 5.11. Algunos ejemplos de imágenes cervico-uterinas. En (a) y (b), muestras obtenidas a 20x. En (c) y (d) muestras obtenidas en 63x

Resultados de los experimentos.

Sobre la imagen analizada se evaluaron 12 espacios de color diferentes. Cada espacio de color se analizó usando sus componentes de forma individual o combinada. En la Figura. 5.12 y 5.13 se muestran las curvas ROC obtenidas durante la evaluación tanto para el algoritmo de Watershed como el de Meanshift con sus respectivas tablas de resultados (5.1 y 5.2). De los resultados iniciales, se puede anotar que MeanShift obtiene de manera general una mejor segmentación que Watershed. Además, el MeanShift reporta buenos resultados para un número mayor de espacios de color que Watershed, en particular aquellos derivados del espacio XYZ donde la cromaticidad esta mejor representada. Es notable, que espacios de color RGB/HSL ampliamente usados en investigaciones biológicas no son los mejores para segmentación de células.

Podemos notar que en la componente X del espacio XYZ, el algoritmo de Watershed detecta la mayor cantidad de células, esto se debe, a que esta componente distribuye mejor la distribución del color entre

los objetos y el fondo produciendo un alto contraste que resalta los contornos.

En ambos algoritmos el espacio LUV obtiene buenos resultados durante la segmentación, este espacio permite discriminar el color rojo, verde colores característicos de la muestra analizada. Además la uniformidad que tiene este espacio permite agrupar mejor estos colores.

Los espacios usados para la transmisión de señales como el YUV, NTSC, Ycbcr no obtienen buenos resultados durante la segmentación de células.

La distancia ROC es el criterio escogido para realizar este análisis tal como se estudio en la sección 2.3.6., el valor que posee la menor distancia al punto óptimo es el parámetro que obtiene mejores resultados.

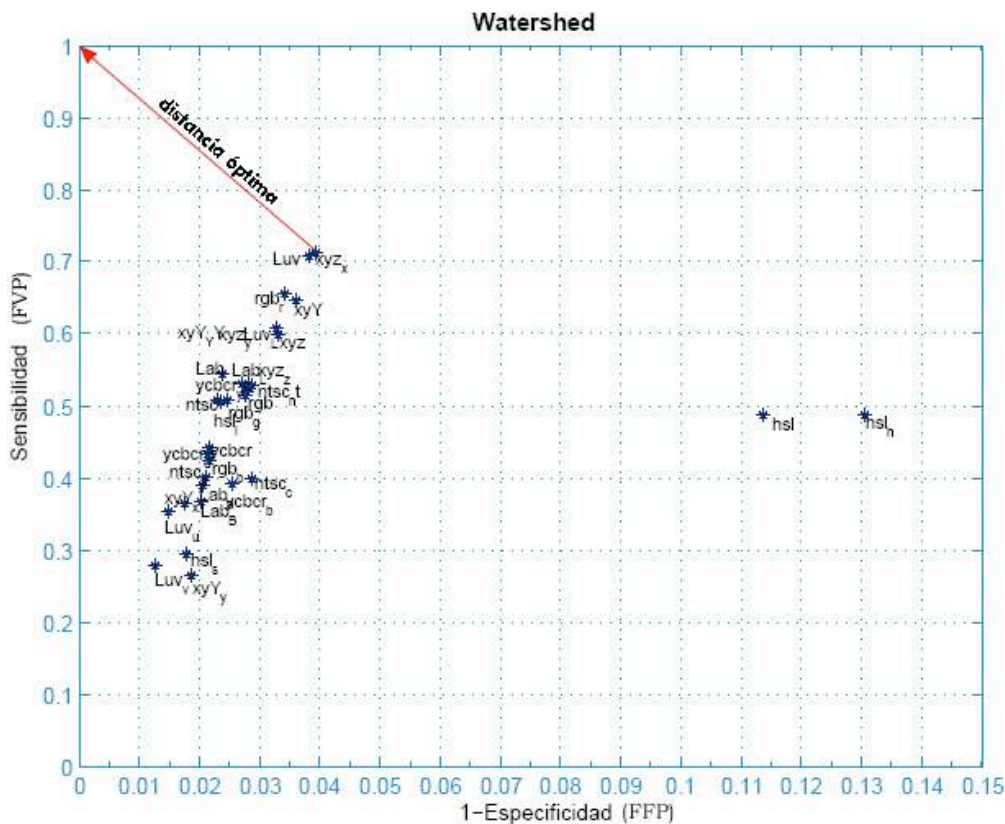


Figura 5.12. Curva ROC generada por el algoritmo de Watershed.

Distancia ROC Watershed					
xyz_x	Luv	rgb_r	XyY	xyY_YY	Luv_L
0.2898	0.2937	0.3466	0.3555	0.3931	0.3931
xyz_y	xyz	Lab	ycbcr_yc	Lab_L	xyz_z
0.3931	0.4032	0.4552	0.4696	0.4707	0.4721
Ntsc_nt	rgb	rgb_g	ntsc	hsl_l	hsl
0.4805	0.4860	0.4930	0.4935	0.4952	0.5258
Hsl_h	ycbcr	ycbcr_cr	rgb_b	ntsc_s	ntsc_c
0.5285	0.5584	0.5676	0.5757	0.6003	0.6028
ycbcr_b	Lab_a	Lab_B	XyY_X	Luv_u	hsl_s
0.6075	0.6101	0.6338	0.6361	0.6474	0.7070
Luv_v	XyY_y				
0.7219	0.7355				

Tabla 5.1. Distancia ROC algoritmo de Watershed.

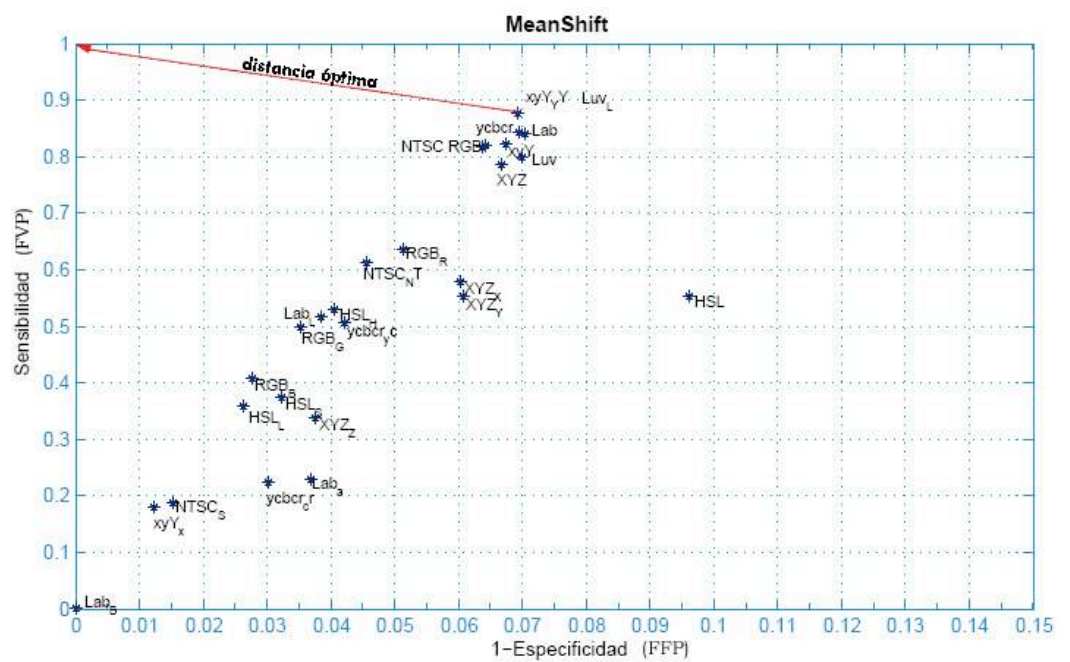


Figura. 5.13. Curva ROC generada por el algoritmo de MeanShift.

Distancia ROC MeanShift					
Luv_L	XyY_YY	Lab	ycbcr	XyY	RGB
0.1411	0.1411	0.1713	0.1736	0.1902	0.1905
NTSC	Luv	XYZ	RGB_R	NTSC_NT	XYZ_X
0.1937	0.2128	0.2249	0.3680	0.3908	0.4243
XYZ_Y	HSL	HSL_H	Lab_L	ycbcr_yc	RGB_G
0.4517	0.4574	0.4724	0.4851	0.4948	0.5032
RGB_B	HSL_S	HSL_L	XYZ_Z	Lab_a	ycbcr_cr
0.5920	0.6260	0.6419	0.6635	0.7709	0.7754
NTSC_S	XyY_X	Lab_B			
0.8112	0.8206	0.9998			

Tabla 5.2. Distancia ROC algoritmo de Meanshift.

Análisis de costos.

Los costos de los equipos y materiales que se utilizaron en la implementación del proyecto se describen en la Tabla 5.3. En el Capítulo III se estudió detalladamente las distintas opciones que se pudieron utilizar para la adquisición imágenes microscópicas.

Ítem	Descripción	Cant.	P.U.	Total
1	Microscopio Carl Zeiss modelo Axioskop2 plus configurado campo claro con objetivos 10x – 40x	1	10823.68	10823.68
2	Computador Pentium 4	1	800	800
3	Cámara AxioCam Mrc5 con software de adquisición.	1	3850	3850
TOTAL				15473.68
P.U.= Precio Unitario, Cant. = Cantidad				

Tabla 5.3. Costos de equipos requeridos para implementación del proyecto.

Los equipos utilizados para el análisis de muestras tienen costos altos, la principal motivación del centro de Visión por computador y Robótica es incentivar la creación de proyectos que involucren el análisis de imágenes microscópicas que a futuro generen ingresos para la ESPOL. Además de aportar a la comunidad científica de servicios de capacitación y digitalización de muestras

Resultados esperados de los algoritmos de segmentación evaluados

Se ha demostrado que el trabajo realizado permite la evaluación cuantitativa de parámetros de segmentación. El parámetro escogido, el color, es una elección natural al segmentar imágenes celulares más

solo la evaluación cuantitativa nos ha permitido descubrir los mejores espacios/componentes de color. Aquellos espacios que representan mejor los cambios en la cromaticidad se ajustan mejor a la segmentación de células en imágenes de campo claro.

La herramienta para la generación de plantillas permite de manera rápida crear y guardar información de la segmentación, además posee un módulo que permite estudiar la distribución del color mediante la definición de umbrales. Esta puede mantener información de varios objetos segmentados dentro de una sola plantilla, y a partir de ella generar la máscara binaria para todos los posibles objetos segmentados.

La información cuantitativa que se puede extraer de los objetos segmentados manualmente, provee una descripción completa de los objetos de interés previo a realizar la clasificación y caracterización de la muestra.

Dada la arquitectura del software desarrollado es posible medir en el futuro parámetros intrínsecos a los algoritmos de segmentación o lograr combinaciones de espacios de color que permitan evaluar si los resultados obtenidos pueden ser mejorados.

Dada la estructura del software desarrollado, este puede ser usado para la evaluación de cualquier tipo algoritmo implementado sobre cualquier tipo de imagen.

Recomendaciones a considerar durante la segmentación de imágenes microscópicas

- El color es un parámetro importante al momento de segmentar imágenes microscópicas, particularmente las imágenes de células usando técnicas de microscopía en campo claro. De la buena selección de los espacios de color depende una correcta segmentación.
- Es posible implementar los algoritmos usados para la segmentación de escala de grises sobre diferentes espacios de color ya sea en componentes individuales o combinados a fin de resaltar criterios como similitud, discontinuidad y conectividad.
- La mayor parte de los espacios de color son construidos con la finalidad de separar el color de la luminosidad, debido a que, al momento de adquirir una imagen la información de reflectancia (color) contenida en los objetos de la escena es constante, puesto

que esta depende de sus propiedades físicas, mientras que la información de luminancia es variante y depende de las condiciones de luz con los cuales la imagen digital fue adquirida. Por tanto es importante encontrar una componente de color que permita la segmentación de los objetos de interés y sea independiente de las variaciones de luz que se producen durante la etapa de adquisición.

- Antes de implementar una solución en visión por computador o publicar resultados de una técnica de segmentación es importante que los resultados obtenidos estén respaldados por pruebas que midan el rendimiento de estas técnicas. Idealmente estas pruebas deberían incluir: la creación de bancos de prueba de imágenes segmentadas, la estandarización de los mecanismos de evaluación y la codificación de los algoritmos de segmentación de interés.
- La evaluación basada en curvas ROC nos permite discriminar el comportamiento de cualquier técnica de segmentación de distinta naturaleza, ya que este estudia principalmente el porcentaje de aciertos y fallos que se obtienen respecto a su estado real.
- Como se revisó en la sección 3.4.1. No es posible todavía automatizar completamente al análisis de células cervicales, puesto

que existen casos en donde las células no se pueden caracterizar por medio de un computador, pues existen propiedades cognitivas que posee el ser humano que no han podido ser modeladas todavía.

- El análisis de muestras usando técnicas de visión por computador disminuyen la subjetividad de la prueba y aumenta el número de muestras que pueden ser evaluadas.
- Es importante capturar la experiencia del experto en una plantilla, ya que se puede extraer información cuantitativa de los objetos segmentados y nos permite describir características comunes de estos.
- Se implementó completamente esta herramienta usando Matlab, ya que se quería aprovechar la variedad de librerías bien documentadas que este posee. Además usa un lenguaje sencillo de desarrollo, la documentación que posee es completa y fácil de entender, también el código implementado es multiplataforma.
- Debido a que esta herramienta es de carácter investigativo no es necesario poner excesivo énfasis en el diseño de la GUI, lo importante es que los algoritmos de transformación de espacios de

color, segmentación y el módulo de evaluación sean eficientes y ayuden al investigador a escoger el algoritmo junto con el espacio de color que brinde mejores resultados al ser aplicados sobre un banco común de imágenes.

- Dada la arquitectura del software desarrollado es posible medir en el futuro parámetros intrínsecos a los algoritmos de segmentación y lograr combinaciones que permitan obtener resultados óptimos.
- Consideramos que el trabajo realizado puede ser utilizado como base para desarrollo o prueba de técnicas de segmentación ya existentes.
- Continuar con el análisis de otro tipo de imágenes microscópicas, y documentar los resultados obtenidos. Para que quede constancia de los trabajos iniciales que se están realizando en la ESPOLE en esta área de investigación. En esta línea ya se adelantan trabajos con el CIBE en proyectos de análisis foliar.
- Implementar más técnicas de segmentación y añadirlas a las dos existentes para en el futuro hacer pruebas con más algoritmos.
- Añadir más técnicas de segmentación que las existentes al módulo semiautomático de creación de plantillas, para que en

posteriores versiones de la herramienta el usuario agilice la segmentación manual.

- Utilizar el módulo de generación de plantilla como material didáctico que le sirva a los estudiantes a comprender la importancia del color durante el proceso de segmentación de imágenes.
- Implementar el módulo de evaluación como parte de un sistema de reconocimiento de objetos, que participe de manera automática en la elección de los mejores parámetros. Esto plantea que el sistema tenga un módulo de aprendizaje y use técnicas más complejas.

BIBLIOGRAFÍA

- [1] Sankur and Sezgin, "Survey over image thresholding techniques and quantitative performance evaluation", *Journal of Electronic Imaging* Vol. 13 (1), 146-165, Enero 2004.

- [2] Arturo de la Escalera, "Visión por Computador Fundamentos y Métodos", Editorial Prentice Hall ,Universidad Carlos III de Madrid, 2001.

- [3] Antti Niemisto, "A Comparison of Nonparametric Histogram-Based Thresholding Algorithms, 8002202 *Digital Image Processing III*, Tampere University of Technology ,Octubre 27 2004.

- [4] Y. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recognition* 29(8), pp. 1335-1346, 1996.

- [5] Hui Zhang, "A Co-evaluation Framework for Improving Segmentation Evaluation", In *Proceedings of IS&T/SPIE's Deense and Security Symposium Signal Processing, Sensor Fusion and Target Recognition XIV*, SPIE Vol. 5809, March 2005.

- [6] Heath, M., Sarkar, S., Sanocki, T. and Bowyer, K., (1996) 'Comparison of Edge Detectors: a Methodology and Initial Study', *Computer Vision and Pattern Recognition*, San Francisco.

- [7] R.M. Haralick. Propagating covariance in computer vision. *Int. J. of Patt. Recog. and Art. Intell.*, 10:561--572, 1996.

- [8] P. van Osta, J.M. Geusebroek, K. Ver Donck, L. Bols, J. Geysen, and B. M. ter Haar Romeny. "The principles of scale space applied to structure and colour in light microscopy.", *Proc. R. Microsc. Soc.*, 37(3):161-166, 2002."

- [9] Kyujin Cho, Peter Meer, Javier Cabrera, "Performance Assessment through Bootstrap," Proc. IEEE, Vol. 19, No. 11, pp. 1185-1198, November 1997.
- [10] GREEN, D.M. & SWETS, J.A., "Signal detection theory and psychophysics." New York: John Wiley & Sons, 1966.
- [11] Min C. Shin, Dmitry Goldgof, and Kevin W. Bowyer, "Comparison of Edge Detectors Using an Object Recognition Task", Computer Vision and Pattern Recognition IEEE pp. 365 Vol. 1, June 1999.
- [12] Minoru Kawahara, Hiroyuki Kawano, "Mining Association Algorithm with Threshold Based on ROC Analysis", Proceedings of the 34th Hawaii International Conference on System Sciences, 2001.
- [13] Claremont Graduate University and the Claremont Colleges Mellon Project, "An Introduction to Signal Detection Theory", <http://wise.cgu.edu/sdt/overview.html>, November 2005.
- [14] DR. H. G. Kapitza , "Microscopy from the very beginning" , [http://www.zeiss.de/C1256B5E0051569F/EmbedTitelIntern/Microscopy_from_the_very_beginning/\\$File/Microscopy_from_the_very_beginning.pdf](http://www.zeiss.de/C1256B5E0051569F/EmbedTitelIntern/Microscopy_from_the_very_beginning/$File/Microscopy_from_the_very_beginning.pdf), Carl Zeiss Jena GmbH, 1997.
- [15] VayTek, Inc., "Compare Cameras: Definitions", <http://www.vaytek.com/camdef.htm> , VayTek's offices in Fairfield,, October 2005.
- [16] Julio Ignacio Ruiz, "Optimización del tamaño de los archivos", <http://www2.canalaudiovisual.com/ezine/books/jirimag/1Imag63.htm>.
- [17] Jähne, B.. "Digital Image Processing: Concepts, Algorithms and Scientific Applications. Second Edition". Springer-Verlag Berlin Heidelberg, 1993.

- [18] Sheila Timp, Nico Karssemeijer, "A new 2D segmentation method based on dynamic programming applied to computer aided detection in mammography.", http://www.xs4all.nl/~stimp/sheila/Timp04_segmentatie.pdf, pp. 6 - 8, September 2006.
- [19] Bouman, C.A., Shapiro, M.. "A Multiscale Random Field Model for Bayesian Image Segmentation". IEEE Transactions on Image Processing. Vol. 3, No. 2, pp. 162-177, May 1989.
- [20] Beucher-Meyer. "The Morphological Approach to Segmentation: the Watershed Transformation" en "Mathematical Morphology in Image Processing", E. Dougherty De., Marcel Dekker: New York, 1992.
- [21] Fernando Vázquez Núñez, "Estado del arte de la segmentación de imágenes", <http://www.aisa.uvigo.es/thesis/fvazquez/cap2.pdf>, Departamento de Ingeniería de Sistemas y Automática, Vigo, June 1996.
- [22] Marshal. S., "Parallel Edge Detection and Related Algorithms". "Parallel Algorithms for Digital Image Processing, Computer Vision and Neural Networks" ed. Pitas, I., John Wiley & Sons, pp. 91-122. 1993.
- [23] Lezoray Oliver, "Segmentation of cytological Images using color and Mathematical Morphology", Abstract Book of the 7th European Congress of Stereology , pp. 52 , Amsterdam, Hollande, April 1998.
- [24] Lezoray Oliver, "Segmentation of color images by clustering 2D histogram and merging regions", Moroccan Journal of Control Computer Science and Signal Processing, 1999.
- [25] H.D. Cheng., "Color image segmentation: advances and prospects", Pattern Recognition 34 pp. 2259-2281, December 2001.

- [26] V. Meas-Yedid, "Automatic color space selection for biological image segmentation", Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04), IEEE, pp. 1051-4651, 2004.
- [27] Andrew Queisser, "Color Spaces for Inspection of Natural Objects", IEEE Proceedings of the 1997 International Conference on Image Processing ICIP, 1997.
- [28] Wikipedia, "CIE 1931 color space", http://en.wikipedia.org/wiki/Chromaticity_diagram, November 21 2006.
- [29] MacAdam, D.L., "Color Measurement - Theme and Variations", Springer-Verlag, 1985.
- [30] Yu Jin Zhang, "A review of recent evaluation methods for image segmentation", Dept. of Electron. Eng., Tsinghua Univ., Beijing, Signal Processing and it's Applications, Sixth International, Symposium on. 2001, pp. 148-151 vol.1, August 2001.
- [31] Yu Jin Zhang, Huitao Luo, "Optimal selection of segmentation algorithms based on performance evaluation", The International Society for Optical Engineering, Volume 39, Issue 6, pp. 1450-1456, June 2000.
- [32] Keselman, Y., Micheli-Tzanakou, E., "Extraction and characterization of regions of interest in biomedical images", Rutgers Univ, USA, Information Technology Applications in Biomedicine ITAB, pp. 87-90, May 1998.
- [33] Wikipedia, "Distance transform", http://en.wikipedia.org/wiki/Distance_transform, December 2006.
- [34] Roboreal, "OpenSource/Freeware vision software", http://roboreal.com/links/vision_software.php, May 2006.

- [35] Intel®, "Intel Integrated Performance Primitives 5.1 for Windows, Linux and Mac OS", <http://www.intel.com/cd/software/products/asmo-a/eng/perflib/ipp/302910.htm>, 07-Aug-2006.
- [36] Intel®, "Open Source Computer Vision Library", <http://www.intel.com/technology/computing/opencv/index.htm>, August 2006.
- [37] Java World, "Image processing with Java 2D", <http://www.javaworld.com/javaworld/jw-09-1998/jw-09-media.html>, February 2006.
- [38] Wikipedia, "Bresenham's line algorithm", http://en.wikipedia.org/wiki/Bresenham's_line_algorithm, January 6 2007.

Christopher Gabriel Crespo León, Master en Sistemas de Información Gerencial. investigador en áreas de visión y robótica con publicación de artículos científicos, organización de eventos de difusión. Líder de arquitectura en Multinacional de Telecomunicaciones. Docente, investigador y coordinador del Área de software de la Facultad de Ciencias Matemáticas y Físicas en la Universidad de Guayaquil. Conocimientos en Sistemas Linux, Matlab, herramientas open sources, nodejs, servidores de aplicaciones. Cursos de CCNA (CISCO), SOA (ARCITURA), Administración e implementación de software para base de datos, SOA suite, weblogic y bus empresarial (Oracle), Administración de plataformas de facturación para TELCOS (Huawei y Tecnotree). Diplomado en gestión de Proyectos (Tecnológico de Monterey).

Jeniffer Denisse Bonilla Bermeo, Magíster en Economía y Dirección de Empresas (Escuela Superior Politécnica del Litoral). Ingeniera en Ciencias Empresariales, concentración en Finanzas, mención en Dirección y Planeación comercial. Contadora Pública Autorizada; mención en Economía Empresarial, Finanzas Internacionales y Gestión Empresarial (Universidad de Especialidades Espíritu Santo). Jefe de Análisis de Riesgos en Institución Financiera, amplia experiencia en Crédito y Finanzas; capacitadora indoor a Nivel Nacional de Riesgos y Análisis Financiero, conocimientos en Estadística y Economía. Docente de la Facultad de Ciencias Matemáticas y Físicas de la Universidad de Guayaquil. Cursos en metodologías ágiles y proyectos. Artículos científicos, Revista indexada a Scopus.

Yuliana Yohany León Bazan, Magíster en Sistemas de Información Gerencial (Escuela Superior Politécnica del Litoral). Ingeniera en Sistemas Computacionales (Universidad de Guayaquil). Experiencia profesional en el Área de Desarrollo y Arquitectura de Software, con conocimientos en varias plataformas de desarrollo en ambiente web, bases de datos y metodologías de administración de negocios. Posee Certificaciones Internacionales y cursos en Lenguajes de Programación, Metodologías Ágiles de Desarrollo de Software, Planificación y Desarrollo de Proyectos (Programa de certificación, PMP). Miembro fundador del PMI Capítulo Guayas. Docente de la Facultad de Ciencias Matemáticas y Físicas de la Universidad de Guayaquil.

ISBN: 978-9942-770-34-9



compAS