

**Universidad Central “Marta Abreu” de Las Villas**

**Facultad de Ingeniería Eléctrica**

**Centro de Estudios de Electrónica y Tecnologías de la Información**



## **TRABAJO DE DIPLOMA**

**Título: Aplicación de algoritmos clasificadores para  
la detección de agregados celulares en imágenes de  
microscopía celular.**

**Autor: Jorge Luis Lisca Del Sol**

**Tutor: Ing. Lyanett China Valdés**

**Cotutor: Dr. C. Juan V. Lorenzo Ginori**

**Santa Clara**

**2015**

**"Año 57 de la Revolución"**

**Universidad Central “Marta Abreu” de Las Villas**

**Facultad de Ingeniería Eléctrica**

**Centro de Estudios de Electrónica y Tecnologías de la Información**



## **TRABAJO DE DIPLOMA**

**Título: Aplicación de algoritmos clasificadores para la detección de agregados celulares en imágenes de microscopía celular.**

**Autor: Jorge Luis Lisca Del Sol**

jlisca@uclv.edu.cu

**Tutor: Ing. Lyanett China Valdés**

Instructor: Centro de Estudios de Electrónica y Tecnología de la Información

Facultad de Ing. Eléctrica. UCLV.

e-mail: lyanett@uclv.cu

**Cotutor: Dr. C. Juan V. Lorenzo Ginori**

Profesor titular Consultante: Instructor: Centro de Estudios de Electrónica y Tecnología de la Información

Facultad de Ing. Eléctrica. UCLV.

e-mail: juanl@uclv.edu.cu

**Santa Clara**

**2015**

**"Año 57 de la Revolución"**



Hago constar que el presente trabajo de diploma fue realizado en la Universidad Central “Marta Abreu” de Las Villas como parte de la culminación de estudios de la especialidad de Ingeniería Biomédica, autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

---

Firma del Autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

---

Firma del Tutor

---

Firma del Jefe de Departamento  
donde se defiende el trabajo

---

Firma del Responsable de  
Información Científico-Técnica

## **PENSAMIENTO**

*“Si he logrado ver más lejos, ha sido porque he subido a hombros de gigantes”*

*Isaac Newton*

*“La única manera de hacer un trabajo genial es amar lo que haces”*

*Steve Jobs*

## **DEDICATORIA**

A mi madre en especial, para que cada día viva orgullosa de este regalo

A mi abuelo y abuela, por cada enseñanza que me ha dado en la vida.

A mi hermanita por apoyarme y darme siempre su confianza.

A mi padre, para que cada día sienta más orgullo de mí.

## **AGRADECIMIENTOS**

A toda mi familia, en especial a mi madre, abuelo, abuela y hermana que estuvieron el día a día dándome aliento y apoyándome para dar cada paso en mi carrera.

A mi padre, aun estando lejos de mí, me apoyaba y sentía con gran orgullo en el profesional que me convertía.

A mi hermana Yoana por siempre darme sus mejores consejos.

A mi novia, por tener esa paciencia tan grande conmigo para enfrentar cada situación o dificultad, al igual que su familia.

A todos mis amigos que me guiaban siempre por el buen camino y compartían en las buenas y malas.

A mi tutora Lyanett China Valdés, por toda la paciencia e incansable lucha que tuvo junto a mí para lograr terminar todo satisfactoriamente y ver un sueño hecho realidad.

A mi tutor Juan V. Lorenzo Ginori por esa maravillosa pedagogía que siempre supo emplear conmigo, haciendo de mí un mejor profesional.

A todos los profesores del CEETI que participaron en mi formación como estudiante.

A mi gran amigo y casi como un hermano Marino Adrian Godoy Arcias por brindarme su ayuda y apoyo incondicional en todos y cada uno de los momentos de mis 5 años de carrera.

A mi gran amigo y compañero de aula Jorge Enrique Alfonso Dulzaides por estar siempre apoyándome cuando pensaba que no lograba las cosas y demostrándome que era capaz de todo.

A mi buen amigo José Carlos que tanto me ayudo con el inglés, una tarea difícil de vencer.

## **TAREA TÉCNICA**

1. Revisión bibliográfica sobre los métodos de detección de objetos que se tocan o se solapan, sintetizando la información obtenida y haciendo un análisis crítico.
2. Implementación de algoritmos de acondicionamiento y segmentación de imágenes en Matlab.
3. Obtención de rasgos para la clasificación de imágenes de eritrocitos aislados y aglomerados.
4. Evaluación de los diferentes algoritmos clasificadores empleados, determinando los índices de efectividad.
5. Elaboración de recomendaciones sobre el uso de los diferentes rasgos obtenidos en las tareas de clasificación de las imágenes celulares analizadas.

---

Firma del Autor

---

Firma del Tutor

## RESUMEN

La microscopía celular es hoy en día uno de los métodos de diagnósticos más importantes en el campo de la salud, ya que comprende el estudio de muchas enfermedades y patologías presentes en los humanos. Se hace necesario el estudio de grandes volúmenes de imágenes de células (en este caso específico de glóbulos rojos o hematíes), por lo que se hace necesaria la automatización de estos procesos, ya que la realización manual puede provocar errores en los resultados por el agotamiento físico y mental de los especialistas y lentitud en su desempeño. El desarrollo de los procesos automáticos es hoy todo un reto para los expertos y está en constante desarrollo siendo un método muy novedoso aún. En las imágenes de microscopía celular se encuentran células que se tocan o se superponen, una primera segmentación con el objetivo de extraer y clasificar dichos componentes, no los separa, por lo que es necesario primero detectarlos. Este trabajo presenta un método de detección de agregados celulares por medio de la implementación de rasgos descriptores. Se realizan estos procesos sobre una primera segmentación por el método de Otsu, Se implementaron y evaluaron 3 algoritmos clasificadores: el clasificador lineal, el método de  $k$  vecinos más cercanos y máquinas de vectores soporte. Se clasifican las células en, independientes y aglomeradas, siendo el clasificador máquinas de vectores soporte el que mejor índices de desempeño presentó.



## TABLA DE CONTENIDOS

PENSAMIENTO .....	i
DEDICATORIA .....	ii
AGRADECIMIENTOS .....	iii
TAREA TÉCNICA .....	iv
RESUMEN .....	v
INTRODUCCIÓN .....	1
CAPÍTULO 1. FUNDAMENTOS TEÓRICOS .....	3
1.1 Introducción .....	3
1.2 Importancia de la detección de agregados celulares para el procesamiento digital de imágenes de eritrocitos.....	3
1.3 Pre-procesamiento y segmentación de imágenes .....	5
1.3.1 Técnicas de pre-procesamiento de imágenes digitales .....	5
1.3.2 Algoritmos de segmentación .....	10
1.3.2.1 Segmentación de Otsu Adaptativo .....	11
1.4 Extracción de rasgos .....	14
1.5 Algoritmos clasificadores.....	15
1.6 Evaluación del desempeño de clasificadores .....	18
CAPÍTULO 2. MATERIALES Y MÉTODOS.....	21
2.1 Introducción .....	21

2.2	Acondicionamiento de las imágenes digitales utilizadas .....	21
2.3	Segmentación de las imágenes y creación de una base de datos anotada. ....	24
2.4	Extracción de rasgos y clasificación de las imágenes. ....	26
2.5	Evaluación del clasificador .....	29
CAPÍTULO 3. RESULTADOS Y DISCUSIÓN .....		30
3.1	Introducción .....	30
3.2	Base de datos de imágenes de células aglomeradas e independientes .....	30
3.3	Rasgos descriptores empleados en la clasificación .....	31
3.4	Análisis del índice de desempeño de los clasificadores .....	33
3.5	Análisis estadístico de los resultados .....	34
CONCLUSIONES Y RECOMENDACIONES .....		41
Conclusiones .....		41
Recomendaciones .....		41
REFERENCIAS BIBLIOGRÁFICAS .....		42
ANEXOS .....		46
Anexo I Códigos de los algoritmos programados .....		46

## INTRODUCCIÓN

La información digital para el campo de la medicina ha adquirido una gran relevancia en las dos últimas décadas, debido al gran contenido de información que se encuentra en las imágenes médicas para diagnóstico y tratamiento de muchas enfermedades. Esto ha despertado el interés de muchos expertos e investigadores por desarrollar programas y algoritmos informáticos para el análisis de dichas imágenes. Entre los diferentes métodos de adquisición de estas imágenes se pueden mencionar la microscopía celular, la resonancia magnética nuclear (RMN), la tomografía computarizada (TC), el ultrasonido, las cámaras gamma (CG), entre otras. Este trabajo responde a la necesidad de analizar grandes cantidades de imágenes de microscopía celular (en particular eritrocitos humanos), en la cuales se presentan aglomeraciones o agregados de estas células. Este tema es de suma importancia ya que existen diversas enfermedades como el paludismo o malaria, la anemia u otras patologías que pueden ser diagnosticadas mediante alteraciones morfológicas que les ocurren a estas células. Con el crecimiento de estos estudios y el aumento considerable de grandes volúmenes de información presentes en las imágenes, se hace necesaria la automatización de estos procesos de diagnósticos, ya que estos trabajos manualmente pueden ser lentos y tediosos para los especialistas, y están sujetos a errores inter e intra-analistas. Estos análisis automáticos permiten el conteo y clasificación de estas células, que posibilita por ejemplo, los exámenes de hemoglobina, ya que los eritrocitos contienen hemoglobina, la cual transporta oxígeno, por lo que la cantidad de oxígeno que los tejidos corporales reciben depende de cuantos hematíes tenga la persona y de cómo funcionen, entre otras aplicaciones. Entre las técnicas de Procesamiento Digital de Imágenes (PDI) como método de automatización, se encuentra la segmentación de estas imágenes, que como procesos de extracción de componentes y división de las células aglomeradas o

solapadas, no logra en una primera segmentación separa las células aglomeradas, por lo que es necesaria la detección de estos agregados, para su posterior segmentación, para de esta forma contribuir a mejorar los resultados. Este trabajo trata sobre la implementación de un método de detección de agregados celulares precedido de una primera segmentación inicial, mediante la implementación de rasgos descriptores y algoritmos clasificadores seleccionados por presentar buenos resultados en la literatura.

Esta investigación se realiza tomando en cuenta los lineamientos 131 y 132 del VI Congreso del PCC que plantean:

131. Sostener y desarrollar los resultados alcanzados en el campo de la biotecnología, la producción médico-farmacéutica, la industria del software y el proceso de informatización de la sociedad, las ciencias básicas, las ciencias naturales, los estudios y el empleo de las fuentes de energía renovables, las tecnologías sociales y educativas, la transferencia de tecnología industrial, la producción de equipos de tecnología avanzada, la nanotecnología y los servicios científicos y tecnológicos de alto valor agregado.

132. Perfeccionar las condiciones organizativas, jurídicas e institucionales para establecer tipos de organización económica que garanticen la combinación de investigación científica e innovación tecnológica, desarrollo rápido y eficaz de nuevos productos y servicios, su producción eficiente con estándares de calidad apropiados y la gestión comercializadora interna y exportadora, que se revierta en un aporte a la sociedad y en estimular la reproducción del ciclo. Extender estos conceptos a la actividad científica de las universidades

Con lo planteado anteriormente se proponen los objetivos siguientes:

**Objetivo General:**

Implementar funciones sobre imágenes de microscopía celular de muestras de sangre, para la clasificación en agregados celulares o células aisladas, a partir del empleo de rasgos distintivos que faciliten esta tarea, utilizando el software Matlab.

**Objetivos Específicos:**

1. Obtener una base de imágenes, anotada por especialistas, que contengan eritrocitos aislados y aglomerados.
2. Seleccionar rasgos adecuados y programar las funciones más apropiadas para el cálculo de estos.
3. Realizar aplicaciones específicas de reconocimiento de patrones y clasificación entre eritrocitos aislados y aglomerados.

Se plantea la siguiente tarea técnica para lograr la solución del problema planteado:

1. Revisión bibliográfica sobre los métodos de detección de objetos que se tocan o se solapan, sintetizando la información obtenida y haciendo un análisis crítico.
2. Implementación de algoritmos de acondicionamiento y segmentación de imágenes en Matlab.
3. Obtención de rasgos para la clasificación de imágenes de eritrocitos aislados y aglomerados.
4. Evaluación de los diferentes algoritmos clasificadores empleados, determinando los índices de efectividad.
5. Elaboración de recomendaciones sobre el uso de los diferentes rasgos obtenidos en las tareas de clasificación de las imágenes celulares analizadas.

**Organización del informe**

La estructura de este trabajo se compone de la siguiente manera: introducción, capítulos, conclusiones y recomendaciones, referencias bibliográficas y anexos. La introducción breve contiene los aspectos fundamentales del trabajo. El capítulo 1 aborda los

fundamentos teóricos sobre la detección de agregados celulares, técnicas de pre-procesamiento, rasgos descriptores, algoritmos de segmentación propuestos, algoritmos de clasificación y la evaluación del desempeño de los algoritmos clasificadores. El capítulo 2 trata sobre el acondicionamiento de las imágenes, la segmentación de las imágenes utilizando las funciones del software Matlab, la creación de una base de datos anotada por especialistas, la extracción de los rasgos junto con la clasificación de las imágenes y la evaluación de los clasificadores. El capítulo 3 presenta imágenes de células independientes y aglomeradas de la base de datos, evaluación de los rasgos descriptores empleados, análisis del índice de desempeño de los clasificadores y un análisis estadístico de los resultados y en los anexos se encuentran los pasos de la programación.

## **CAPÍTULO 1. FUNDAMENTOS TEÓRICOS**

### **1.1 Introducción**

En este capítulo se presentan los fundamentos teóricos acerca de la detección de agregados en imágenes de microscopía celular, particularmente de eritrocitos sanguíneos. Se tratan además temas de pre-procesamiento y segmentación digital de imágenes. El capítulo está constituido por 6 epígrafes, el epígrafe 1.1 es introductorio y el epígrafe 1.2 trata sobre la importancia de la detección de agregados celulares. El epígrafe 1.3 expone fundamentos teóricos acerca del pre-procesamiento y segmentación de imágenes digitales. El epígrafe 1.4 trata sobre la extracción de rasgos y el epígrafe 1.5 acerca de los algoritmos clasificadores. Finalmente en el epígrafe 1.6 se presenta la evaluación del desempeño de los algoritmos clasificadores.

### **1.2 Importancia de la detección de agregados celulares para el procesamiento digital de imágenes de eritrocitos**

El PDI introducido en el campo de la microscopía celular es una de las herramientas principales para muchos investigadores y expertos en todo el mundo. Está destinado principalmente para la automatización de los procesos de diagnóstico y tratamiento en el ámbito de la medicina. La automatización de estos procesos surge por la necesidad de mejorar los resultados de los análisis, realizados por expertos humanos, desde el punto de vista de la rapidez, disminución de errores, contradicciones de resultados entre analistas, entre otros, ya que dichos procesos al ser efectuados por ellos, con grandes volúmenes de imágenes, pueden estar expuestos a errores producto del agotamiento físico natural y de la subjetividad. Entre los análisis de microscopía celular está la clasificación y conteo diferencial que se les realiza a las imágenes, en las cuales se puede encontrar células

aisladas (independientes) y células que se tocan o se superponen (aglomeradas). Esto lleva a que los procesos de segmentación de imágenes como elementos importantes en la extracción y división de objetos (en este caso células), produzcan falsos resultados o inservibles en una segmentación inicial, de ahí la gran importancia de la detección de células agregadas o aglomeradas previa, logrando con esto obtener mejores resultados en la posterior segmentación. En el artículo [1] se expone una alternativa de segmentación basada en la transformada de distancia para objetos que se tocan o se solapan, siendo este un gran problema para cuando se tienen muchas imágenes, por lo que se hace más vigente la importancia de la detección previa a estos procesos. Se han realizado trabajos sobre la detección de agregados como por ejemplo, en [2] se expone la construcción de un vector de rasgos que identifica a cada objeto (célula o agregado) en la imagen. Ese vector se construye a partir de parámetros relacionados con la forma de cada objeto (para lo cual se obtiene una curva que lo identifica) y además con la localización del contorno externo del objeto. Se toman como conjunto de entrenamiento de un clasificador los vectores correspondientes a células aisladas y a agregados, correspondientes a objetos previamente clasificados por analistas humanos y se realiza la clasificación de los objetos desconocidos, obteniendo así una cierta tasa de clasificación correcta. En la Fig. 1 se muestra una imagen de frotis de sangre con agregados celulares de dos y más células.

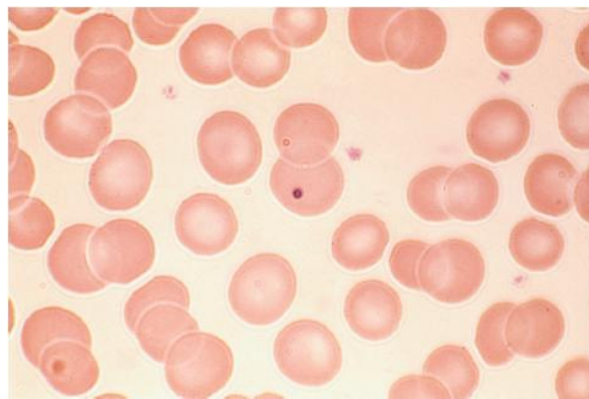


Fig. 1. Imagen de con diversos agregados celulares.<sup>1</sup>

---

<sup>1</sup> Imagen obtenida de [3]



### 1.3 Pre-procesamiento y segmentación de imágenes

En el procesamiento de imágenes existen tres tipos de procesos fundamentales, estos son:

- Los procesos de nivel bajo: se caracterizan por el hecho de que su entrada y salida son imágenes.
- Los procesos de nivel medio: se caracterizan por el hecho de que sus entradas son por lo general imágenes, pero sus salidas son objetos o formas extraídos de tales imágenes por ejemplo: bordes, contornos, líneas.
- Los procesos de nivel alto: comprenden la interpretación del sentido de los objetos que se conocen, desarrollando funciones cognitivas normalmente asociadas con la visión [4].

Los procesos de nivel bajo incluyen operaciones primitivas como es el pre-procesamiento de imágenes para la reducción de ruido, mejoramiento del contraste y resaltar más la imagen mejorando las definiciones de la misma. Los procesos de nivel medio incluyen métodos como la segmentación (la partición de una imagen en regiones u objetos), la descripción de esos objetos para reducirlos a una forma idónea para su procesamiento digital y la clasificación (reconocimiento) de objetos individuales [4-6].

Como etapa inicial en el procesamiento de imágenes recomendado según la literatura consultada [6-8] se realiza un pre-procesamiento de imagen. Este proceso incluye filtrado, ecualización del contraste, eliminar la corrección no uniforme, entre otras. Este paso es importante, dado que contribuye a mejorar los resultados de la posterior segmentación de la imagen.

#### 1.3.1 Técnicas de pre-procesamiento de imágenes digitales

##### Procesamiento morfológico de imágenes

El procesamiento morfológico como una rama del PDI, toma su nombre a partir de la morfología matemática, para extraer y procesar elementos de la imagen que pueden resultar útiles en la representación y descripción de formas como son: límites, esqueletos y cascos convexos [4, 9]. Este procesamiento comprende procesos que se realizan a las imágenes, desde el punto de vista de su estructura geométrica, en forma que concuerda con la intuición y percepción humana [4-6]. En la morfología matemática, se definen filtros no

lineales que son de gran utilidad en el PDI, estos están basados en las transformaciones morfológicas de las funciones, las cuales son imágenes en este caso.

Las operaciones elementales de la morfología matemática son: erosión, dilatación, apertura y cierre.

Estas operaciones morfológicas se definen de la siguiente manera:

- Erosión:

Para los conjuntos  $A$  y  $B$  en  $Z^2$  la erosión de  $A$  por  $B$ , está definida como:

$$A \ominus B = \{z | (B)_z \subseteq A\} \quad (1)$$

Por tanto la erosión de  $A$  por  $B$  es el conjunto de todos los puntos  $z$  tales que  $B$ , trasladado por  $z$ , está contenida en  $A$ .

- Dilatación:

Con  $A$  y  $B$  como conjuntos en  $Z^2$ , la dilatación de  $A$  por  $B$ , es denotada  $A \oplus B$  y es definida como:

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\} \quad (2)$$

Esta ecuación está basada en la obtención de la reflexión de  $B$  respecto al origen y en el corrimiento por  $z$ . La dilatación de  $A$  por  $B$  es el conjunto de todos los desplazamientos,  $z$ , tal que  $\hat{B}$  y  $A$  se superponen al menos por un elemento.

Basado en esta interpretación, la ecuación se puede escribir como:

$$A \oplus B = \{z | [(\hat{B})_z \cap A] \subseteq A\} \quad (3)$$

El conjunto  $B$  es comúnmente referido como el elemento estructurante (EE) de la dilatación, al igual que en las demás operaciones morfológicas.

- Apertura:

$$A \circ B = (A \ominus B) \oplus B \quad (4)$$

Expresándose como una operación de erosión de  $A$  por  $B$ , seguida de una dilatación del resultado por  $B$ .

A partir de la interpretación geométrica, la apertura se puede definir como la unión de todas las traslaciones de  $B$  que están comprendidas en  $A$ :

$$\bigcup \{(B)_z | (B)_z \subseteq A\} \quad (5)$$

La apertura generalmente suaviza los contornos de un objeto, rompe regiones de unión, y elimina “islas” y “picos”.

- Cierre:

El cierre de un conjunto  $A$  por un elemento estructurante  $B$ , es denotado  $A \bullet B$ , y se define como:

$$A \bullet B = (A \oplus B) \ominus B \quad (6)$$

Y se puede expresar, como la operación de dilatación de  $A$  por  $B$  seguida de una erosión del resultado por  $B$ .

El cierre también tiende a suavizar los contornos, pero opuesto a la apertura, tiende a fusionar “canales” angostos, eliminar “agujeros” pequeños y rellenar las “ensenadas” en los contornos.

La implementación de dichas operaciones sucesivamente posibilita la supresión de ruido en imágenes en escala de grises, la eliminación de detalles irrelevantes sin que exista una distorsión de los elementos no eliminados, el suavizado de los bordes o contornos, el adelgazamiento, engrosamiento, esqueletización, la reconstrucción morfológica, entre otros [6, 10-12].

### **Corrección de la iluminación no uniforme**

Entre los factores que afectan la calidad de la segmentación está la no uniformidad de la iluminación, ya sea debido a la fuente luminosa o la reflectancia de la superficie. La iluminación no uniforme puede ser corregida mediante técnicas de procesamiento morfológico [4, 5, 9], entre las que se encuentra la transformación “top hat”. Esta transformación es definida de la siguiente manera:

$$f \hat{o} g = f - (f \circ g) \quad (7)$$

Donde  $f$  es la imagen en escala de grises y  $g$  es el elemento estructurante apropiadamente elegido. Debido a que la apertura es una operación anti-extensiva, por lo que la apertura se encuentra debajo de la imagen original, así que  $f \hat{\circ} g$  es siempre positiva [13].

Este efecto de la iluminación de fondo no uniforme, ocurre con frecuencia en aplicaciones como la microscopía óptica, donde el campo luminoso puede no presentar una intensidad uniforme.

En la Fig. 2. se muestra un ejemplo de corrección de iluminación no uniforme mediante la transformación “top hat” tomada de [14].

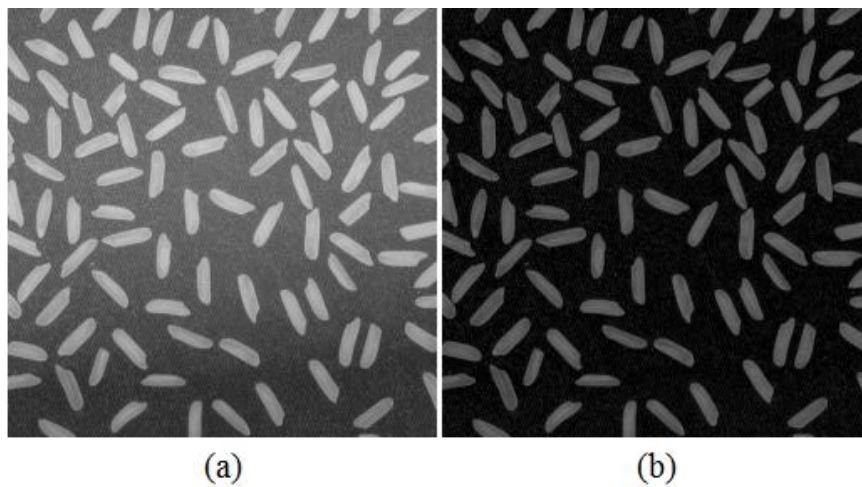


Fig. 2. (a) Transformación “top hat”.

### Realce de imágenes

El contraste en una imagen se define como la diferencia relativa en intensidad entre un punto de esta y sus alrededores o la diferencia entre los niveles de intensidad máximos y mínimos en una imagen en escala de grises [4, 5]. En una imagen con poco contraste un gran número de píxeles ocupan solamente una pequeña porción del rango disponible de intensidades. A través de la modificación del histograma, se reasigna a cada píxel un nuevo valor de intensidad para aumentar el rango dinámico de los niveles de gris. El principio, es estirar el rango dinámico de los valores de los píxeles, de tal forma que los píxeles claros puedan tornarse aún más claros, mientras que los relativamente oscuros puedan ser oscurecidos aún más. Es claro que a través de el estiramiento idóneo de los valores de los píxeles, mejorará el contraste de la imagen [15].

El realce de la imagen se puede mejorar mediante técnicas como la ecualización o especificación del histograma y ajustes de intensidad Fig. 3. El basado en histogramas realiza una re-distribución de los niveles de gris de la imagen original en escala de grises [4, 6]. En los métodos de detección de bordes, resalta el contraste de la imagen. Detectando el contraste se enfatizan los bordes de los rasgos dentro de una imagen, es aquí donde ocurre la diferencia en intensidad de la misma. Así es naturalmente, como la visión humana puede percibir el perímetro de un objeto, siendo este de diferente intensidad con respecto a su medio [16]. Este es un proceso que permite obtener una imagen con mejores condiciones (más apropiada) para aplicaciones determinadas. Las modificaciones que se le realizan a la imagen bajo estudio, pretenden tener un efecto mayormente sobre la percepción subjetiva, este proceso no presenta soluciones generales ni definidas puesto que se necesita de la valoración de un especialista.

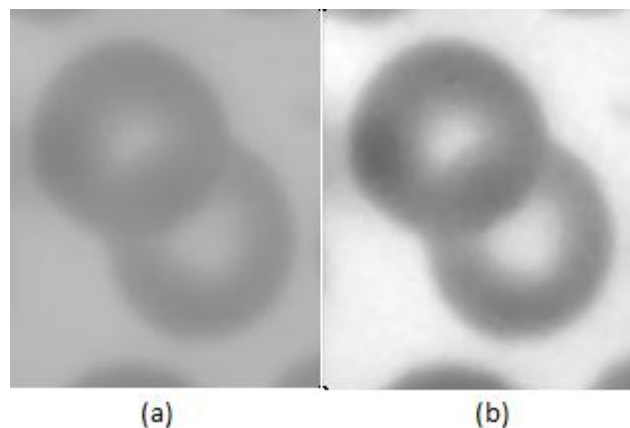


Fig. 3. (a) Imagen original en escala de grises, (b) Realce de la imagen.

### **Suavizado**

El suavizado de imágenes es un paso de pre-procesamiento que caracteriza y mejora los elementos constituyentes fundamentales de estas, como los bordes sobresalientes, mientras que disminuye los detalles menos significantes. Generalmente este método es utilizado para eliminar características irrelevantes, reduciendo la complejidad computacional y eliminando el ruido que pueda estar presente; posibilitando así, la obtención de mejores resultados en la segmentación [5, 6].

El suavizado consiste en realiza un filtrado paso-bajo que atenúa las componentes de alta frecuencia, dejando pasar las de baja frecuencia, es de suma importancia cuando se quiere

extraer grandes elementos en una imagen y esto proporciona la eliminación de pequeños detalles de alta frecuencia [17]. Los filtros de suavizado se utilizan para opacar (borrosidad) y atenuar ruido. En la Fig. 4 se muestra un ejemplo de imagen a la cual se le realizó un proceso de suavizado.

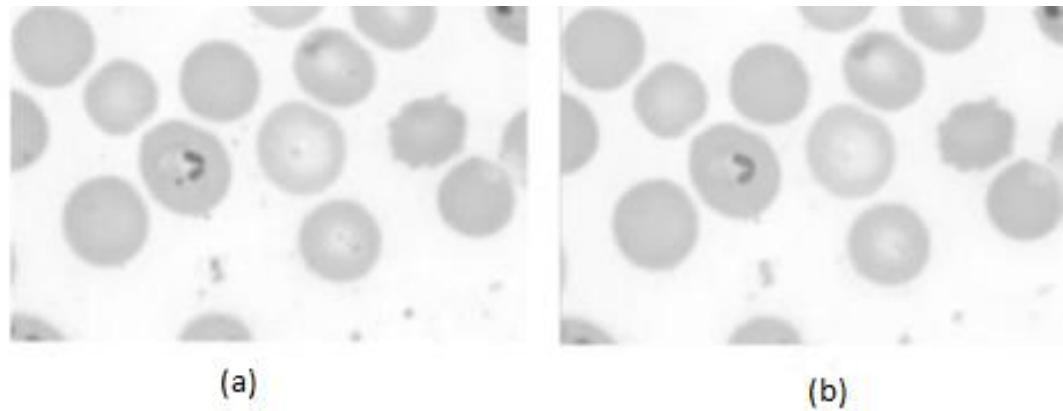


Fig. 4. (a) Imagen en escala de gris, (b) Suavizado realizado a la imagen.

### 1.3.2 Algoritmos de segmentación

La segmentación de imágenes subdivide las mismas en regiones o partes que la conforman; la segmentación terminaría cuando el objeto de interés sea separado o aislado. Este proceso permite identificar y extraer las diferentes características de un objeto en una imagen determinada [6, 18, 19].

Existen diversos algoritmos de segmentación, estos se basan en dos principios:

1. La discontinuidad de los niveles de gris (en una imagen en escala de grises): Esto se fundamenta en segmentar la imagen a partir de grandes cambios en las intensidades de gris entre los píxeles. Existen técnicas que utilizan las discontinuidades como son la detección de líneas, bordes o puntos aislados.
2. La similitud entre los niveles de gris (en una imagen en escala de grises): Que es todo lo contrario a la discontinuidad, donde las fracciones de la imagen se crean agrupando los píxeles que tienen características similares. Uno de los métodos basados en este tipo de algoritmo es el umbralado.

Los algoritmos de segmentación que se han creado con el propósito general para la segmentación en imágenes, y dado que no hay en existencia un método de segmentación

que pueda resolver todos los problemas que se presentan, a menudo se combinan varias técnicas para obtener resultados óptimos, ya sea desde el punto cualitativo como cuantitativo, siendo el primero un factor muy importante en los analistas [5, 6].

En la literatura existe una gran diversidad de métodos de segmentación como se muestran en [6, 18, 19]. De manera general, los métodos clásicos de segmentación se pueden clasificar como:

- Métodos heurísticos de segmentación, los cuales se basan en el conocimiento previo de la imagen a segmentar y en la experiencia del observador, ellos incluyen en muchas ocasiones los métodos supervisados de segmentación.
- Métodos basados en la detección de discontinuidades, éstos dividen la imagen a partir de grandes cambios en los niveles de grises.
- Métodos basados en la propiedad de similitud de los valores de los niveles de grises, estos permiten la agrupación de puntos a partir de ciertos criterios de similitud u homogeneidad.
- Métodos basados en el umbralado del histograma, los cuales obtienen un umbral de comparación para el agrupamiento de los píxeles.

### **1.3.2.1 Segmentación de Otsu Adaptativo**

El algoritmo de segmentación de Otsu fue publicado en el año 1979 por Nobuyuki Otsu [20]. Esta técnica ha sido una de las más utilizadas para la obtención automática de umbrales, lo que hace que este algoritmo no necesite supervisión por el usuario ni información previa de la imagen antes de su procesamiento [21]. El método de Otsu se formula a partir de la decisión estadística, donde el objetivo es minimizar el error promedio, el que se comete al asignar píxeles a uno o más grupos (clases). En el mismo se determina el valor umbral de forma que la dispersión dentro de cada segmento sea lo más pequeña posible, pero al mismo tiempo la dispersión sea lo más alta posible entre segmentos diferentes, por lo que se necesita de una compensación de la iluminación para poder definir de manera eficaz el o los umbrales [5, 20].

La segmentación por Otsu está basada en cálculos que se realizan a partir del histograma de la imagen, con los siguientes pasos:

1. Se realiza el cálculo del histograma normalizado de la imagen bajo estudio. Se denotan las componentes del histograma como  $P_i$ ,  $i=0, 1, \dots, L-1$  donde  $L$  son los diferentes niveles de intensidad presentes en la imagen.

$$MN = n_0 + n_1 + n_2 + \dots + n_{L-1} \quad (8)$$

Donde  $MN$  Son el número de pixeles contenidos en la imagen y  $n_i$  es el número de pixeles que posee el nivel de intensidad correspondiente a su subíndice.

$$P_i = \frac{n_i}{MN} \quad (9)$$

$$\sum_{i=0}^{L-1} P_i = 1, P_i \geq 0 \quad (10)$$

Se selecciona un umbral  $T_{(k)} = k, 0 < k < L - 1$  y se emplea para dividir los pixeles de la imagen  $C_1$  y  $C_2$  (es decir, el objeto y el fondo).

2. Se calculan las propiedades acumulativas  $P_i(k)$ :

$$P_1(k) = \sum_{i=0}^k P_i \quad (11)$$

$$P_2(k) = \sum_{i=k+1}^{L-1} P_i = 1 - P_1(k) \quad (12)$$

- Donde  $P_1(k)$  es la probabilidad de que un pixel sea asignado a la clase  $C_1$ .
- Donde  $P_2(k)$  es la probabilidad de que un pixel sea asignado a la clase  $C_2$ .

Siendo  $C_1$  los pixeles de la imagen en el rango de  $[0, k]$  y  $C_2$  los pixeles de la imagen en el rango de  $[k + 1, L - 1]$ .

3. Se calculan los valores medios acumulativos  $m_i(k)$ .

Se determinan los valores medios de las intensidad correspondientes a las calces  $C_1$  y  $C_2$ .

$$m_1(k) = \frac{1}{P_1(k)} \sum_{i=0}^k iP_i \quad (13)$$



$$m_2(k) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} iP_i \quad (14)$$

4. Se calcula la media global  $m_G$ .

La intensidad promedio  $k$  para toda la imagen:

$$m(k) = \sum_{i=0}^k iP_i \quad (15)$$

La intensidad media global para toda la imagen es:

$$m_G = \sum_{i=0}^{L-1} iP_i \quad (16)$$

5. Se calcula la varianza entre clases.

Ya con las definiciones planteadas se cumplen las relaciones siguientes:

$$P_1m_1 + P_2m_2 = m_G \quad (17)$$

$$P_1 + P_2 = 1 \quad (18)$$

Se define la varianza global como:

$$\sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 P_i \quad (19)$$

La varianza entre clases es:

$$\sigma_B^2(k) = P_1(m_1 - m_G)^2 + P_2(m_2 - m_G)^2 \quad (20)$$

6. Se determina el umbral de Otsu,  $k^*$ , que maximiza la varianza entre clases, si no es único, se promedian los valores obtenidos y se redondean.

Para determinar el umbral en el método de Otsu, se calcula el valor de  $k$  (es el umbral buscado) para el cual es máxima la varianza entre clases.

Según el algoritmo, el umbral óptimo será el valor de  $k=k^*$  tal que:

$$\sigma_B^2(k^*) = \max_{0 \leq k \leq L-1} \sigma_B^2(k) \quad (21)$$

Para hallar el valor de  $k^*$  se evalúa la expresión anterior para todos los posibles valores enteros de  $k$ , (tratando que se cumpla que  $0 < P_1(k) < 1$ ). Si se encuentra más de un valor de  $k$  que satisfaga la condición de máximo de la varianza entre clases, se promedian estos valores.

7. Se calcula la medida de separabilidad  $n$ :

$$n(k) = \frac{\sigma_B^2(k)}{\sigma_G^2} \quad (22)$$

Se procede a segmentar la imagen con el umbral calculado.

Este algoritmo a pesar de que en su mayoría se utiliza para imágenes en escala de grises, también se implementa en imágenes a color aplicándolo a cada una de sus componentes.

#### 1.4 Extracción de rasgos

En las imágenes de microscopía, como también en una gran variedad de imágenes médicas, existen regiones o elementos en ellas que son de interés, las cuales se diferencian por sus formas irregulares; por lo cual resulta de dificultad para su determinación. Ante este conflicto se han identificado una serie de rasgos que pueden ser utilizados para la determinación de dichos objetos o regiones presentes en las imágenes, una óptima selección de estos hace que sus resultados sean los deseados.

Estas regiones presentan dos tipos de representación:

1. Representación por características externas: llamado criterio de formas por sus fronteras.
2. Representación por características internas: criterio de texturas utilizando los píxeles incluidos en su interior [4].

La selección de la representación depende de la aplicación o tarea a resolver, en algunos casos el uso de los dos tipos de representación resulta la mejor alternativa.

En este trabajo se presentan rasgos geométricos, los cuales proporcionan información sobre el tamaño y la forma de las regiones segmentadas. Algunas de estas características son el área, el diámetro, el perímetro, entre otras.

Los rasgos geométricos se pueden clasificar en dos categorías:

- Los que presentan propiedades de posición y orientación como son: el área, el perímetro, el centro de masa, el eje de elongación.
- Los que presentan propiedades de formas como son: la relación de aspecto, la relación de delgadez, las proyecciones, el diámetro de Feret de los objetos [22].

De la correcta elección de los rasgos implementados depende de la efectividad del clasificador.

### 1.5 Algoritmos clasificadores

Los algoritmos clasificadores son mecanismos a partir de los cuales un conjunto de características tomadas inicialmente como entrada, devuelven una clase o agrupaciones etiquetadas a la salida. Existen dos tipos de técnicas de aprendizaje:

1. Aprendizaje supervisado: tiene un conjunto de patrones  $n$  denominados conjunto de entrenamiento que están asociados a una clase determinada.
2. Aprendizaje no supervisado: tiene un conjunto de patrones  $n$  que no están etiquetados y se pretende descubrir agrupaciones o clases en el conjunto de datos, con el objetivo de etiquetar dicho conjunto en diferentes clases [23].

Existen diversos tipos de clasificadores de aprendizaje supervisado y no supervisados, algunos de estos son:

- Entre los algoritmos de aprendizaje supervisado están las redes neuronales (RNA) [24, 25], las máquinas de vectores soporte (SVM) [26-28],  $k$  vecinos más cercanos (KNN) [29], el clasificador lineal, así como el clasificador basado en la distancia de Mahalanobis [4, 30].
- Entre los algoritmos de aprendizaje no supervisado esta *fuzzy C-means* [15], *Spectral Clustering* [31], Algoritmo de Lloyd generalizado (*k-medias*) [32], entre otros.

En este trabajo se usaron algoritmos clasificadores de aprendizaje supervisado estos son de gran importancia puesto que pueden minimizar los errores entre la salida real y la salida deseada, algunos de estos clasificadores son definidos a continuación.

### 1. Clasificador lineal

Este clasificador toma una decisión basada en el valor de una combinación lineal de rasgos o características de los elementos a clasificar. Las características de un objeto son típicamente presentadas en un vector llamado “*vector de rasgos*”. Si la entrada del clasificador es un vector de características reales  $x$  entonces el resultado de la salida es:

$$y = f\left(\sum_j W_j X_j\right) \quad (23)$$

Donde  $W_j$  es un vector real de pesos y  $f$  es una función que convierte el producto punto a punto de los vectores en la salida deseada. El vector de pesos  $W_j$  aprende de un conjunto de muestras de entrenamiento. Normalmente  $f$  es una función simple que mapea todos los valores por encima de un cierto umbral a la primera clase y el resto a la segunda. Este clasificador al no presentar gran carga computacional, es utilizado donde la velocidad de clasificación es un factor esencial [33].

### 2. El clasificador basado en la distancia Mahalanobis

Este clasificador tiene su base en el cálculo de la distancia de Mahalanobis, entre dos variables aleatorias con la misma distribución de probabilidad  $x$  y  $y$  con la matriz de covarianza  $\Sigma$  se define como:

$$dm(x, y) = \sqrt{(x - y)^T \Sigma^{-1} (x - y)} \quad (24)$$

Su forma útil radica en que determina la similitud entre dos variables aleatorias multidimensionales, se diferencia de la distancia euclidiana ya que tiene en cuenta la correlación entre dichas variables. Es de gran utilidad cuando las variables no son independientes [4, 30].

### 3. El método de los $k$ vecinos más cercanos (KNN)

El método de los  $k$  vecinos más cercanos asigna la muestra  $x$  a la clase más frecuente de entre sus  $k$  vecinos más cercanos, según una cierta medida de similitud o distancia [23]. El

algoritmo presenta un entrenamiento que se basa en almacenar los vectores característicos y las etiquetas de las clases de los ejemplos de entrenamiento. En la etapa de clasificación, la evaluación de un ejemplo del cual no se conoce su clase, es representada por un vector en el espacio de rasgos. Se calcula la distancia entre los vectores almacenados y el nuevo vector, se seleccionan los  $k$  ejemplos más cercanos, una medida de distancia elevada entre individuos indica que son muy diferentes y una baja que son muy similares. El nuevo ejemplo se clasifica con la clase que más se repite en los vectores seleccionados. Existen varias distancias como son: euclidiana, cityblock, coseno, hamming, entre otras [23].

La distancia usada en este trabajo fue la métrica City Block, también conocida como métrica Manhattan. Esta se define como:

$$dist(p, q) = \sum_{i=1}^n |p_i - q_i| \quad (25)$$

El algoritmo de KNN es uno de los más utilizados en la literatura por su sencillez. La dificultad que presenta este clasificador, es la de encontrar un valor apropiado para  $k$ , puesto que si el mismo es muy pequeño puede ocurrir una imprecisión en la clasificación a causa de los pocos datos seleccionados como petición de comparación, y si es muy grande puede ocurrir que la clasificación sea de acuerdo a la mayoría y no por su similitud.

#### 4. Máquinas de vectores soporte (SVM)

Las máquinas de vectores soporte tienen una fundamentación matemática pura dentro de la teoría estadística de aprendizaje, a pesar de esto, la implementación básica cuenta con algunas limitaciones puesto que están diseñadas originalmente para problemas de clasificación binarios (dos clases), lo que lo hace para este trabajo muy efectivo, y además tienen el problema que su algoritmo básico de entrenamiento genera gran cantidad de vectores soporte lo que ocasiona lentitud y sobreajuste en la clasificación [34].

Este algoritmo se basa en mapear los puntos de entrenamiento a un espacio vectorial de una dimensión mayor, construir hiperplanos en un espacio multidimensional para separar los puntos en sus clases respectivas y clasificar un punto nuevo de acuerdo a su ubicación con respecto al hiperplano de separación. La SVM encuentra el hiperplano de separación óptimo, o sea, que maximice la distancia (margen) entre las clases en este espacio,

utilizando un *kernel* que calcule el producto escalar de los puntos de entrada en el espacio de características. La solución del hiperplano óptimo puede ser escrita como la combinación de unos pocos puntos de entrada que son llamados vectores de soporte [35].

Dentro de los *kernels* que se pueden emplear están, el *kernel* polinomial, el lineal, el rbf (función de base radial) gaussiano, el cuadrático, entre otros [4]. Esta función de núcleo es la que utiliza el algoritmo clasificador para separar los datos y crear el hiperplano entre ellos. Los datos son evaluados y de acuerdo a su resultado es que se asignan a una clase.

### 1.6 Evaluación del desempeño de clasificadores

Para la evaluación de los clasificadores es necesario disponer de una referencia apropiada (“estándar dorado”), la cual fue la clasificación de las células (eritrocitos) clasificadas a mano por especialistas, en células aisladas (independientes) y agrupadas (aglomeradas), esta evaluación definirá cual es el algoritmo más apropiada para esta investigación. La valoración de los algoritmos clasificadores requiere la especificación de medidas de similitud para la comparación de los datos de la referencia con los resultados obtenidos por los clasificadores [33, 36]. A continuación se muestran mediadas de desempeño que resultan de gran utilidad en la comparación. En la Tabla 1 se muestran las definiciones que serán comentadas a continuación.

Tabla 1. Matriz de confusión

		Clasificación realizada por analistas humanos	
		Aglomeradas $X_i = 1$	Independientes $X_i = 0$
Clasificadas por el algoritmo	Aglomeradas $y_i = 1(+)$	VP	FP
	Independientes $y_i = 0(-)$	FN	VN

- $VP$  = cantidad de ocurrencias de  $y_i = 1 / x_i = 1$  (células aglomeradas detectadas como aglomeradas), conocidas como verdaderos positivos.
- $FP$  = cantidad de ocurrencias de  $y_i = 1 / x_i = 0$  (células independientes detectadas como aglomeradas), conocidas como falsos positivos.

- $FN$  = cantidad de ocurrencias de  $y_i = 0 / x_i = 1$  (células aglomeradas detectadas como no- aglomeradas), conocidas como falsos negativos.
- $VN$  = cantidad de ocurrencias de  $y_i = 0 / x_i = 0$  (células no- aglomeradas detectadas como no- aglomeradas), conocidas como verdaderos negativos.

Por medio de estas definiciones, se calculan diferentes medidas de similitud o de diferencias como indicadores de desempeño, como son: la sensibilidad, la especificidad, la predictividad positivas, la predictividad negativa y la tasa de clasificación.

1. La sensibilidad ( $Se$ ) o tasa de detección correcta, es la probabilidad de que las anomalías sean detectadas correctamente. Es definida como:

$$Se = p(Y = 1|X = 1) = 1 - p(Y = 0|X = 1) = \frac{VP}{VP + FN} = 1 - \frac{FN}{VP + FN} \quad (26)$$

2. La especificidad ( $Sp$ ) es la probabilidad de detectar correctamente las regiones normales, se calcula como:

$$Sp = p(Y = 0|X = 0) = 1 - p(Y = 1|X = 0) = \frac{VN}{FP + VN} = 1 - \frac{FP}{FP + VN} \quad (27)$$

3. La predictividad positiva ( $Pp$ ) es la posibilidad de que los positivos hayan sido correctamente detectados, y se calcula como:

$$Pp = p(X = 1|Y = 1) = 1 - p(X = 0|Y = 1) = \frac{VP}{VP + FP} = 1 - \frac{FP}{VP + FP} \quad (28)$$

4. La predictividad negativa ( $Pn$ ) o tasa bayesiana negativa, es la probabilidad de que los negativos hayan sido correctamente detectados. Se define como:

$$Pn = p(X = 0|Y = 0) = 1 - p(X = 1|Y = 0) = \frac{VN}{FN + VN} = 1 - \frac{FN}{FN + VN} \quad (29)$$

5. La tasa de clasificación correcta ( $Cr$ ) es la proporción del número total de casos que fueron clasificados correctamente.

$$\begin{aligned} Cr &= p(X = Y) = 1 - p(X \neq Y) = \frac{VP + VN}{VP + FP + FN + VN} \\ &= 1 - \frac{FP + FN}{VP + FP + FN + VN} \end{aligned} \quad (30)$$

6. La tasa de error ( $Er$ ) es el número total de casos que fueron clasificados incorrectamente.

$$Er = \frac{FP + VN}{VP + FP + FN + VN} = 1 - \frac{VP + VN}{VP + FP + FN + VN} \quad (31)$$

7. La distancia Czekanowski, Dice, Sorensen o medida  $F$  ( $F_{measure}$ ) combina sensibilidad y predictividad positiva.

$$D_c = F_{measure} = \frac{2S_e P_p}{S_e + P_p} = \frac{2VP}{2VP + FP + FN} \quad (32)$$

Para mejorar los resultados de los clasificadores, se entiende que debe existir una mayor correlación entre los datos de la referencia y los obtenidos de la clasificación. Lo cual implica que exista una mayor cantidad de  $VP$  y menor cantidad de  $FN$ , aspecto importante dado que en las aplicaciones médicas, la presencia de  $FN$  conlleva a errores en el diagnóstico médico.



## **CAPÍTULO 2. MATERIALES Y MÉTODOS**

### **2.1 Introducción**

En este capítulo se abordan los procesos realizados a las imágenes y los métodos realizados para esto. El capítulo cuenta con 5 epígrafes, el epígrafe 2.1 es introductorio, el epígrafe 2.2 trata sobre todo el proceso de adquisición y acondicionamiento de las imágenes utilizadas para el trabajo. El epígrafe 2.3 trata sobre la segmentación de las imágenes y la creación de la base de datos anotada. En el epígrafe 2.4 se explica la metodología utilizada para la extracción de los rasgos y la clasificación de las imágenes. Y el epígrafe 2.5 trata sobre la evaluación de los clasificadores utilizados.

### **2.2 Acondicionamiento de las imágenes digitales utilizadas**

Las imágenes presentes en este trabajo fueron adquiridas en el laboratorio de histopatología del Centro de Bioactivos Químicos (CBQ) de la Universidad Central “Marta Abreu” de Las Villas. Dichas imágenes son obtenidas de frotis de sangre humana contaminada con el parásito *Plasmodium Berghei*, causante de la malaria, enfermedad infecciosa y perjudicial para los seres humanos, con un alta mortalidad asociada; según la Organización Mundial de la Salud (OMS), esta ocasiona anualmente más de un millón de muertes y entre 198 millones de personas infestadas a nivel mundial [37].

Para la preparación del frotis se empleó el método convencional, el cual consiste en extender una gota de sangre sobre la superficie de un portaobjetos o un cubreobjetos. El frotis de sangre se compone de 3 partes: cabeza, cuerpo y cola, los estudios de la misma son realizados en las 2 últimas partes, puesto que es aquí donde se obtiene mejor visualización de cada uno de los elementos que componen la sangre [38], un buen resultado depende del especialista en este tipo de técnica. La sangre en las muestras, es fijada con metanol y

teñida con *Giemsa*, esta técnica es utilizada principalmente para teñir los frotis de sangre en estudios de organismos microscópicos, su principal función es poner de manifiesto parásitos en las células huéspedes. Estos trabajos con la sangre, son realizados por personal calificado y entrenado para estos rigurosos procedimientos.

Las tomas digitales de las imágenes fueron realizadas con un microscopio del centro, el cual tiene acoplada una cámara de fotomicrografía para el estudio de microorganismos, células u otras estructuras microscópicas. Las imágenes son visualizadas por los sistemas de lentes, que establecen la calidad de la imagen en cuanto a su nitidez y resolución. La imagen es enfocada con un aumento de 100X en la escala de su nomenclatura. El instrumento emplea un sistema de iluminación de Kohler, que brinda una iluminación uniforme en el campo objeto de examen. La cámara presente en el microscopio está conectada a una computadora, la cual trabaja con el software *ToupView* para procesar las imágenes [39]. En la Fig. 5 se muestra una de las imágenes que fue obtenida mediante el sistema antes descrito.

Las imágenes utilizadas en este trabajo, tienen formato tif, resolución de 2048 x 1536 píxeles y se encuentran en el espacio de color RGB. El procesamiento de las imágenes se utilizó empleando el software *Matlab versión 7.10.0.499 (R2010a)* con mayor apoyo en su *Toolbox* de procesamiento digital de imágenes [14]. Siendo, *imread*, *im2double*, *adapthisteq*, *imtophat*, *graythresh*, *im2bw*, *imfill*, *imclearborder* y *regionprops* las principales funciones empleadas. La evaluación estadística se realizó utilizando el programa SPSS versión 21 [40].

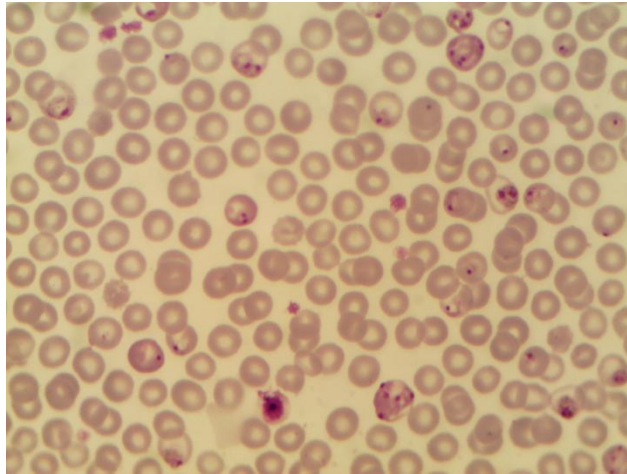


Fig. 5. Imagen de eritrocitos en frotis de sangre.

El acondicionamiento de las imágenes se realizó siguiendo los siguientes pasos:

1. Se lleva la imagen a doble precisión normalizada con la función *im2double*.
2. Todas las imágenes son convertidas al plano de color HSI (hue, saturación e intensidad), mediante la función *rgb2hsi* usada por González & Woods [4]. Luego se toma la componente de intensidad, debido a que la segmentación sobre este plano brinda buenos resultados, respecto a las propiedades de discontinuidad y similitud que esta presenta.
3. Se realiza un filtrado de mediana por la función *medfilt2* para eliminar el ruido aditivo y atenuar las altas frecuencias.
4. Se procesa el contraste de la imagen en escala de gris por una transformación de valores usando la ecualización del histograma, esta operación es realizada por la función *adapthisteq*.
5. Para la corrección de la iluminación de fondo no uniforme se utiliza la función *imtophat* sobre la imagen de intensidad invertida. El elemento estructurante (EE) utilizado es el disco de radio 40 ya que es lo suficientemente grande para que con la apertura quede solamente el fondo no uniforme, que al restarse de la imagen original devuelva el mapa de intensidades con la distribución deseada, este EE se crea utilizando la función del Matlab *strel*.

### 2.3 Segmentación de las imágenes y creación de una base de datos anotada.

Para la segmentación de imágenes, se usó el algoritmo de Otsu, utilizado para este tipo de imágenes en varios trabajos citados en la bibliografía [41-46] y que fue descrito en detalle en el epígrafe 1.3.2.1. Este umbral es calculado mediante la función *graythresh* y luego con la función *im2bw* se binariza la imagen. Posteriormente se procede, con la función *imfill* a rellenar los “huecos” presentes en el centro de las células, debidos a la forma que presentan los eritrocitos, de ser células que no tienen núcleo cuando maduran y con una depresión en el centro. La función *imclearborder* se utiliza luego para eliminar las regiones de los bordes que no son de interés para los especialistas dado que las células aparecen cortadas por estos. En la Fig. 6 se muestran resultados obtenidos al realizar la segmentación por el método propuesto.

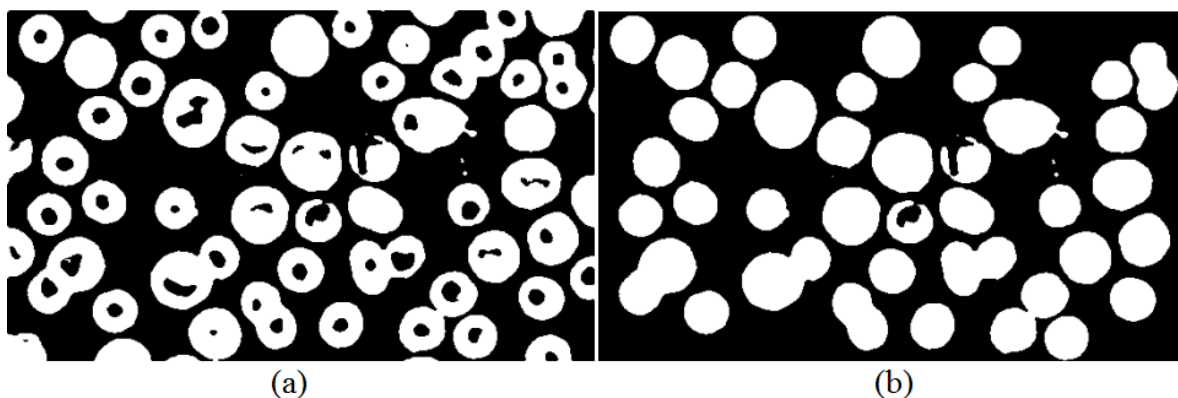


Fig. 6. (a) Máscara binaria obtenida luego de segmentación por el método de Otsu, (b) máscara binaria después del relleno de huecos.

Para eliminar artefactos y zonas espurias presentes luego de la segmentación, se realiza un etiquetado de la imagen con la función *bwlabel* y con la función *regionprops* se calcula toda el área de las regiones etiquetadas. Se crea un vector donde quedan almacenados los valores de las áreas de todas las regiones etiquetadas en la imagen. Se calcula la mediana de ese vector y se realiza una apertura por área con el valor de la mediana/2 como elemento estructurante, logrando eliminar elementos que tengan un área más pequeña que este parámetro, dichos elementos son llamados artefactos (ver Anexo 1). La imagen resultante de estas operaciones se muestra en la Fig. 7.

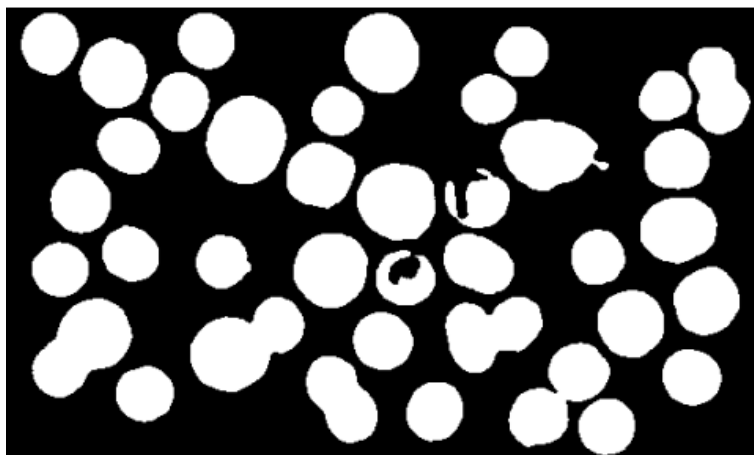


Fig. 7 Máscara binaria.

Una vez obtenidas las máscaras de las imágenes, se recortan todas las regiones dentro de la imagen por medio de la función *regionprops*, con el parámetro *BoundingBox* el cual encierra la región etiquetada en el polígono más pequeño en el cual esta pueda ser contenida. Las regiones recortadas se guardan en un *cell array* para su posterior análisis (ver Anexo 1).

La base de datos está compuesta por 240 células dividida en dos grupos, 120 células independientes y 120 células aglomeradas anotadas por especialistas del CBQ, con iguales cantidades en ambos grupos se garantiza un balance entre clases a la hora de evaluar los clasificadores. En la Fig. 8 se muestra un ejemplo de una célula independiente y un agregado, estas imágenes recortadas presentan una resolución que varía de 138 x 88 hasta 244 x 339 píxeles aproximadamente. La creación de la base de datos se realiza con el objetivo de clasificar las células, en independientes y aglomeradas mediante la evaluación de los clasificadores en un análisis estadístico. En la Fig. 9 se muestran las máscaras de las imágenes mostradas en la Fig. 8.

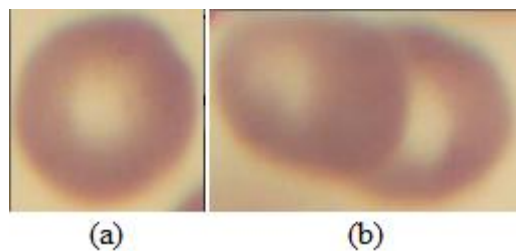


Fig. 8. (a) Célula independiente, (b) Células aglomeradas.

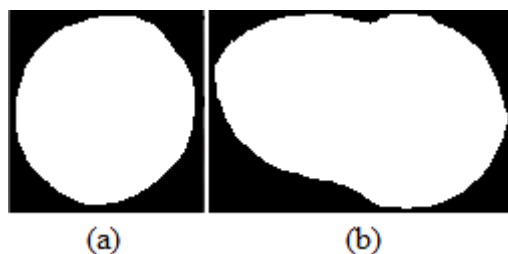


Fig. 9. (a) Máscara binaria de célula independiente, (b) Máscaras binaria de células aglomeradas.

## 2.4 Extracción de rasgos y clasificación de las imágenes.

Para la clasificación de cada una de estas imágenes, se utilizan rasgos geométricos que están implementados en la función *regionprops*, definida en el software Matlab y de esta manera se prueba la efectividad que tiene estos rasgos para la clasificación de las imágenes. Los rasgos utilizados en la clasificación son:

1. Área: define el número de píxeles en una región.
2. Área convexa: es el número de píxeles en la imagen convexa, formada por el mínimo polígono convexo que contiene la región.
3. Excentricidad: escalar que especifica la excentricidad de la elipse que tiene el mismo momento de segundo orden que la región. Es la relación de la distancia entre los focos de la elipse y la longitud de su eje principal.
4. Longitud del eje mayor: longitud del eje mayor de la elipse que se ajusta a la región.
5. Longitud del eje menor: longitud del eje menor de la elipse que se ajusta a la región.
6. Solidez: relación entre área y área convexa de la región.
7. Diámetro equivalente: diámetro de un círculo que tiene igual área que la región.
8. Extensión: relación entre el área y el polígono más pequeño donde quepa la región.

Con los rasgos descritos se crea una matriz de rasgos, donde cada columna representa cada rasgo implementado y por las filas los valores de todos los rasgos para las 120 células aglomeradas y las 120 células independientes, por lo que tiene una dimensión de  $M \times N$  donde  $M$  son las imágenes (240) y  $N$  son los rasgos (8) (ver Anexo 1).

Para la clasificación de las imágenes se crean 12 grupos de células, cada grupo tiene una cantidad de 20 células, de estas 10 son células aglomeradas y 10 células independientes. Para la evaluación del primer grupo se tiene que, dentro del hay 5 agrupaciones de 4 imágenes cada una, donde se toman 16 imágenes (que serían 4 agrupaciones) para el entrenamiento y 4 imágenes (que sería la quinta agrupación) para la evaluación, y así sucesivamente se va barriendo con todas las agrupaciones hasta terminar la evaluación del primer grupo, y luego seguiría con el siguiente grupo. Esta es una manera muy apropiada para la evaluación de los clasificadores ya que siempre hay una gran parte de los datos en el entrenamiento de una iteración, que fueron usados en el entrenamiento de la iteración anterior, por lo que hay una correlación grande entre los resultados de cada iteración, por estar usando una gran parte de la base de entrenamiento en todos los casos (ver Anexo 1).

Los clasificadores implementados fueron:

1. El clasificador lineal.
2. El método de los k vecinos más cercanos (KNN).
3. Máquinas de vectores soporte (SVM)

Se tomaron para este trabajo estos clasificadores, puesto que son de uso bastante generalizado teniendo en cuenta la rapidez con la que se desempeñan, su simplicidad, son buenos clasificadores binarios y se han utilizado en otras investigaciones en imágenes de eritrocitos infestados con malaria con muy buenos resultado en la clasificación [47-50].

El primer clasificador fue implementado mediante la función *classify*, esta función presenta los siguientes parámetros:

*class = classify (sample, training, group, 'type')*

La función clasifica todas las filas de la muestra (*sample*), en el grupo usado para el entrenamiento (*training*). El parámetro *group* contiene en sus filas la clasificación de la matriz de datos de entrenamiento (*training*), estos deben tener la misma cantidad de filas. El *group* es una cadena de caracteres con la misma longitud que la del (*training*). En este trabajo se emplea un *group* de cadena de caracteres en el cual se encuentra la clase a la que pertenece cada célula (independiente o aglomerada). El parámetro '*type*' especifica el tipo

de clasificador, que pueden ser: lineal, cuadrático, basado en la distancia mahalanobis, y otros. El seleccionado en este trabajo fue el lineal.

El segundo clasificador se implementó por la función *knnclassify*, con el parámetro de distancia *cityblock*, con  $k=1$ . El valor de  $k$  se determina realizando un análisis estadístico de las medidas  $F$  obtenidas al implementar el clasificador KNN para los valores de  $k$ : 1, 3 y 5 [47].

La función es implementada de la siguiente forma:

```
class = knnclassify (sample, training, group, k, distance)
```

Donde el *sample* es la matriz de muestra, que será clasificada por el parámetro *group*. El *training* debe tener el mismo número de columnas que el *sample* y contiene en sus filas la clasificación de parámetro *group*. El *group* es un vector cuyos valores definen la agrupación de las filas del *training*. La  $k$  define el número de vecinos más cercanos, que por defecto se toma como 1. El parámetro *distance* contiene una serie de distancias las cuales pueden ser: euclidiana, coseno, hamming, correlación, *cityblock*, entre otras.

El tercer clasificador (SVM) se implementó de la forma siguiente:

```
svmstruct = svmtrain (... , 'Kernel_Function', Kfun)
```

El parámetro *Kfun* usa el mapa de datos de entrenamiento dentro del espacio *Kernel* el cual puede ser: lineal, cuadrático, gaussiano de base radial, polinomial, perceptron de capa múltiple, entre otros. El *Kernel* utilizado fue el lineal el cual ha resultado ser muy efectivo en este tipo de clasificaciones [47]. La función *svmtrain* devuelve la información del clasificador entrenado, junto con los vectores de soporte. Estos valores son usados posteriormente en la función que realiza la clasificación de las células, la misma se implementa de la siguiente manera:

```
classsvm = svmclassify(svmstruct, sample)
```

Donde (*sample*) es la muestra a clasificar y *svmstruct* son los resultados obtenidos de la función *svmtrain*.

En *classsvm* quedan almacenados los resultados de la clasificación (ver Anexo 1).



## 2.5 Evaluación del clasificador

Para la evaluación de los clasificadores se utilizó la función de Matlab *classperf*. Esta función evalúa la calidad, el desempeño o el funcionamiento de los clasificadores en términos de *VP*, *FP*, *VN* y *FN* dichas definiciones se muestran en el epígrafe 1.6.

La función *classperf* presentan los siguientes parámetros:

$cp = \text{classperf}(\text{truelabels})$

$\text{classperf}(cp, \text{class}, \text{test})$

Donde *cp* es la variable que se inicializa, que contiene la evaluación realizada por el clasificador. El parámetro *truelabels* son las etiquetas de las clases para cada observación, donde las mismas pueden ser vectores numéricos o cadena de caracteres.

El clasificador clasifica un conjunto de datos etiquetados llamados conjunto de prueba (*test*), por medio de un conjunto de entrenamiento o conjunto de aprendizaje (*class*).

Cuando se tiene una gran cantidad de datos es muy factible usar el tipo de validación cruzada de *K* iteraciones [23]. Se implementa la función *crossvalind* para cada uno de los 12 grupos creados. La función *crossvalind* tiene como argumento los parámetros siguientes:

$\text{indices} = \text{crossvalind}('K\text{fold}', N, K)$

Con este método de cruzamiento se divide las *N* muestras para la clasificación, en *K* grupos ordenados aleatoriamente, donde se utilizan *K*-1 grupos para entrenamiento y el restante para el ser clasificado. De esta forma siempre se toman subconjuntos diferentes para la evaluación hasta clasificar todo los grupos. Para este trabajo se utilizó una validación cruzada de (*5-Fold Cross-validation*).

Luego se determina la medida *F* ( $F_{\text{measure}}$ ) para la evaluar el desempeño de los clasificadores. Esta medida se calcula mediante la Sensibilidad (*Se*) y la Predictividad positiva (*Pp*).

Se obtienen también otros índices como son: la Tasa de error (*Er*), la Predictividad negativa (*Pn*), la Clasificación correcta (*Cr*), la Especificidad (*Sp*).

## CAPÍTULO 3. RESULTADOS Y DISCUSIÓN

### 3.1 Introducción

En este capítulo se exponen los resultados obtenidos de los experimentos realizados. Se muestran ejemplos de imágenes de la base de datos creada. Se realiza un análisis estadístico de los índices de desempeño obtenidos por el clasificador KNN para determinar el valor de  $k$  más adecuado para la investigación. Se realiza un análisis del desempeño de los clasificadores utilizando para esto la medida  $F$  y se muestran los índices de efectividad obtenidos de los clasificadores.

### 3.2 Base de datos de imágenes de células aglomeradas e independientes

Para la creación de la base de datos se recortaron 15 imágenes como la que se muestra en la Fig. 10, de las cuales se obtuvieron 120 células independientes (Cell-Ind) y 120 células aglomeradas (Cell-Agl) como las que se muestran en la Fig. 11. La base de datos fue anotada por un especialista designado por el CBQ.

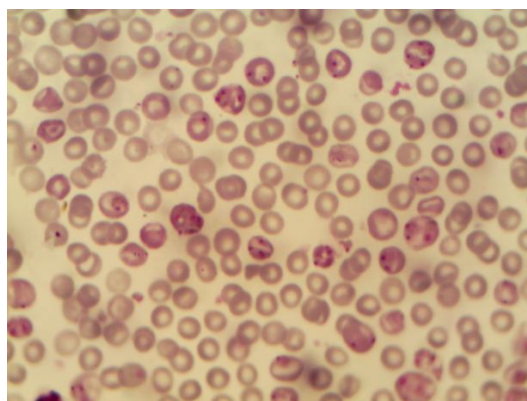


Fig. 10. Imágenes obtenidas en el CBQ

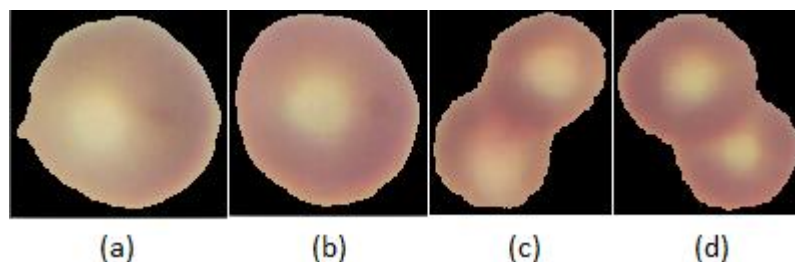


Fig. 11. (a) y (b) Imágenes de células independientes. (c) y (d) Imágenes de células aglomeradas.

Las imágenes de eritrocitos aglomerados pueden tener dos o más células agregadas como se muestra en la Fig. 12

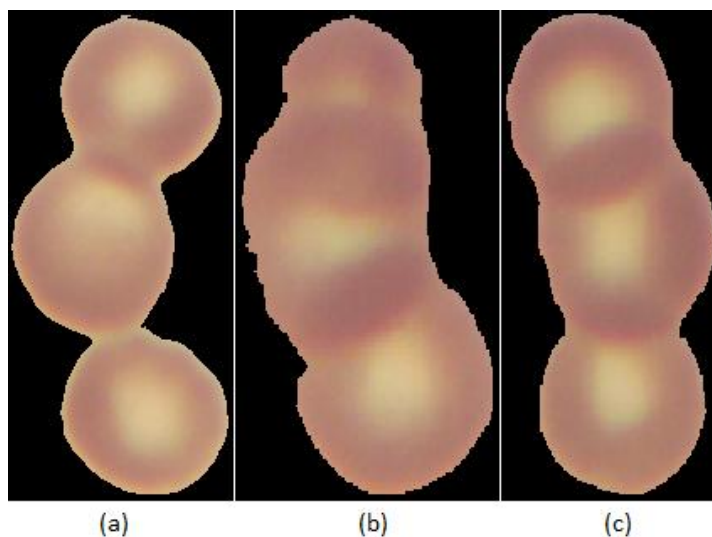


Fig. 12. (a), (b), (c) Aglomeraciones de tres eritrocitos.

### 3.3 Rasgos descriptores empleados en la clasificación

Estas células se caracterizan por sus irregularidades de ahí su dificultad para clasificarlas. Con las imágenes ya recortadas se obtuvieron sus máscaras mediante la segmentación por el método de Otsu como se muestra en la Fig. 13, el cual brinda muy buenos resultados tanto de manera visual como estadístico como se define en la Tabla 2. En la Tabla 2 se exponen el valor medio y la desviación estándar para cada uno de los 8 rasgos (con una numeración que se corresponde con la del epígrafe 2.4) utilizados para las Cell-Ind y Cell-Agl.

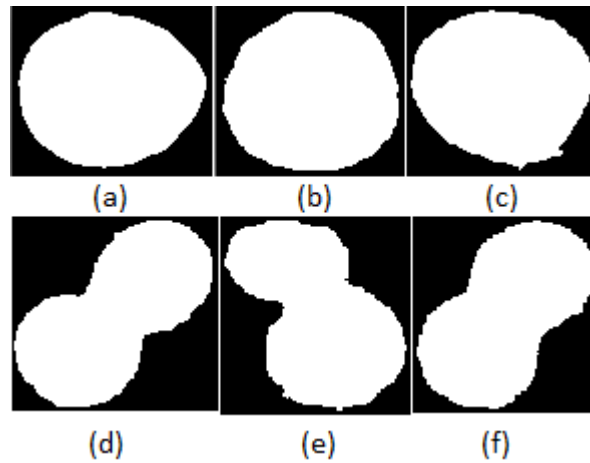


Fig. 13. (a), (b) y (c) Diferentes células independientes. (d), (e) y (f) Diferentes células aglomeradas.

Tabla 2. Resultados de las medias y las desviaciones estándar de los rasgos en células independientes y aglomeradas.

Cell-Ind	Media							
Rasgos	1	2	3	4	5	6	7	8
	6,6548	6,7464	0,0004	0,0958	0,0878	0,0010	0,0916	0,0008
	Desviación Estándar							
	1,3428	1,3567	0,0001	0,0097	0,0091	0,0000	0,0090	0,0000
Cell-Agl	Media							
Rasgos	1	2	3	4	5	6	7	8
	1,5538	1,6956	0,0001	0,0194	0,0104	0,0001	0,0138	0,0001
	Desviación Estándar							
	7,0070	8,3639	0,0001	0,0499	0,0239	0,0000	0,0284	0,0001

### 3.4 Análisis del índice de desempeño de los clasificadores

Para el análisis de los clasificadores se muestra la medida  $F$  (para los 12 grupos en cada uno de los clasificadores) como evaluador del desempeño de los clasificadores, junto con el cálculo de la media donde se demuestra que el clasificador SVM es el que mayor media presenta (ver Tabla 3).

Tabla 3. Medida  $F$  de los clasificadores en los 12 grupos.

	Lineal	KNN(1)	KNN(3)	KNN(5)	SVM
1	0,952	0,842	0,909	0,909	0,952
2	0,976	0,923	0,884	0,810	0,976
3	0,968	0,933	0,906	0,857	0,984
4	0,952	0,938	0,918	0,892	0,988
5	0,943	0,941	0,933	0,913	0,990
6	0,952	0,951	0,944	0,927	0,992
7	0,946	0,936	0,917	0,895	0,986
8	0,952	0,938	0,920	0,901	0,981
9	0,957	0,944	0,929	0,912	0,983
10	0,962	0,950	0,936	0,921	0,985
11	0,965	0,955	0,942	0,928	0,986
12	0,956	0,950	0,943	0,930	0,983
Media					
	0,957	0,933	0,923	0,899	<b>0,982</b>

En la Tabla 4 se observan los valores de los índices de efectividad (antes descritos en el epígrafe 1.6) para cada clasificador.

Tabla 4. Indicadores de desempeño

	<i>Se</i>	<i>Sp</i>	<i>Pp</i>	<i>Pn</i>	<i>Cr</i>	<i>Er</i>
Lineal	0,997	0,917	0,922	0,991	0,954	0,046
KNN(1)	0,950	0,950	0,950	0,950	0,950	0,050
KNN(3)	0,958	0,925	0,927	0,957	0,942	0,058
KNN(5)	0,947	0,917	0,919	0,940	0,929	0,071
SVM	0,992	0,975	0,975	0,992	0,983	0,017

### 3.5 Análisis estadístico de los resultados

A continuación se realiza un análisis estadístico para determinar en el clasificador KNN, el valor más apropiado de  $k$  entre las variantes implementadas ( $k=1, 3$  y  $5$ ).

Para determinar la normalidad de las poblaciones se utiliza la prueba de Kolmogorov-Smirnov que se le realiza a las medidas  $F$  de cada grupo, para cada tipo de  $k$  (ver Tabla 5). Se toma la hipótesis nula ( $H_0$ ) de que las poblaciones analizadas siguen una distribución normal, y como hipótesis alternativa ( $H_1$ ) de que no sigan una distribución normal. Como se puede observar en la tabla para los tres valores de  $k$  la significación asintótica bilateral es mayor que 0,05 (nivel de significación que se tomó), por tanto, se acepta  $H_0$ . A pesar de esto es importante destacar, que cuando se analiza una reducida cantidad de casos, se utilizan generalmente en los análisis estadísticos pruebas no paramétricas, por lo que en este caso se optó por emplear la prueba de Friedman.

Tabla 5. Prueba de Kolmogorov-Smirnov

	KNN(1)	KNN(3)	KNN(5)
N	12	12	12
Z de Kolmogorov-Smirnov	1,149	0,439	0,843
Sig. asintót. (bilateral)	0,142	0,991	0,477

Luego se aplica la prueba de Friedman para verificar la existencia de diferencias significativas entre las muestras. En la Tabla 6 se muestran los rangos promedios obtenidos al aplicar la prueba de Friedman, mediante los cuales se indica si las variables difieren entre sí de forma estadísticamente significativa. Los rangos de valores más altos indican un mejor comportamiento durante la clasificación de los 12 grupos analizados. Como se puede observar el valor de  $k$  que mayor rango aportó fue  $k=1$ .

Tabla 6. Rangos promedios

	Rango promedio
KNN(1)	2,83
KNN(3)	1,50
KNN(5)	1,67

En la Tabla 7 se muestran los resultados de la prueba de Friedman en del clasificador KNN para cada valor de  $k$  seleccionado, usando la medida  $F$  como índice de efectividad. Esta contiene el número de casos válidos, el estadístico *Chi-cuadrado* de Friedman, sus grados de libertad (gl) y el nivel crítico (Sig. asintót. bilateral).

Tabla 7. Prueba de Friedman

N	12
Chi-cuadrado	12,667
gl	2
Sig. asintót. (bilateral)	0,002

Debido a que el valor del nivel crítico es menor que 0,05, se rechaza la  $H_0$  de igualdad de promedios poblacionales, afirmando que existen diferencias entre las muestras (ver Tabla 7).

Estos resultados dieron lugar a la realización de una prueba *post hoc* para comparar dos muestras relacionadas y determinar si existen diferencias entre ellas. Se determinaron los estadísticos descriptivos (ver Tabla 8) y la prueba de rangos con signos de Wilcoxon, cuyos resultados se muestran en la Tabla 9.

Tabla 8. Estadísticos descriptivos de la medida  $F$ 

	N	Media	Desviación típica	Mínimo	Máximo
KNN(1)	12	0,947	0,015	0,900	0,959
KNN(3)	12	0,910	0,027	0,837	0,935
KNN(5)	12	0,913	0,033	0,829	0,952



Tabla 9. Prueba de rangos con signos de Wilcoxon

	KNN(3) – KNN(1)	KNN(5) – KNN(1)	KNN(5) – KNN(3)
Z	-2,981	-2,275	-0,941
Sig. asintót. (bilateral)	0,003	0,023	0,347

Es necesario realizar la corrección de Bonferroni de los resultados obtenidos con la prueba de Wilcoxon para de esta manera controlar la probabilidad de cometer error Tipo I. El nuevo nivel de significación que se toma es de 0,017 ( $0,05/3=0,017$ ). Con el nuevo nivel de significación solo hay una diferencia significativa para valores entre  $k=3$  y  $k=1$ , y entre  $k=5$  y  $k=1$ . Siendo la diferencia entre  $k=5$  y  $k=1$  un caso marginal puesto que el valor de la probabilidad es muy cercano al nivel de significación establecido (0.017).

Se puede observar que  $k=1$  es el que mayor media presenta (ver Tabla 8), siendo este el que se escoge para el clasificador KNN, lo cual además contribuye a tener menor costo computacional.

### **Evaluación de los clasificadores Lineal, KNN y SVM**

Para la evaluación de los clasificadores se realiza la prueba estadística de normalidad, mediante la prueba de Kolmogorov-Smirnov que se les realiza a las medidas  $F$  de cada grupo, para cada clasificador (ver Tabla 10). En los resultados de esta prueba se acepta la  $H_0$  (las poblaciones analizadas siguen una distribución normal) y se rechaza la  $H_1$  de que no sigan una distribución normal, puesto que la significación asintótica bilateral es mayor que 0,05.

Tabla 10. Prueba de Kolmogorov-Smirnov

	Lineal	KNN(1)	SVM
N	12	12	12
Z de Kolmogorov-Smirnov	0,796	1,111	1,039
Sig. asintót. (bilateral)	0,550	0,169	0,231

A continuación se realiza la prueba de Friedman para verificar la existencia de diferencias significativas entre las muestras (ver Tabla 11 y 12). En la Tabla 11 se exponen, para cada clasificador, los rangos medios resultantes del proceso de asignación de rangos. La prueba de Friedman compara estos rangos e indica si las variables difieren entre sí. Los rangos con valor más alto indican un mejor comportamiento durante la clasificación, siendo para este caso el clasificador SVM el que mayor rango presenta.

Tabla 11. Rangos promedios

	Rango promedio
LINEAL	1,63
KNN(1)	1,42
SVM	2,96

Para la prueba de Friedman la significación asintótica es menor que 0,05 por lo que se rechaza la hipótesis de igualdad promedios de los grupos relacionados entre sí, y se concluye que existen diferencias estadísticas significativas entre los clasificadores (ver Tabla 12).

Tabla 12. Prueba de Friedman

N	12
Chi-cuadrado	17,149
Gl	2
Sig. asintót. (bilateral)	0,000

La prueba *post hoc* para determinar qué clasificadores difieren entre sí, se hace mediante la prueba de rangos con signo de Wilcoxon (ver Tabla 13). La  $H_0$  es que no hay diferencias estadísticamente significativas. En este caso es necesario realizar nuevamente la corrección de Bonferroni para controlar la posibilidad de cometer error Tipo I. El nivel de significación que se toma por tanto es de 0,017, por lo que no hay diferencias significativas entre el clasificador Lineal y el KNN, en cambio hay diferencias entre SVM y Lineal, y SVM y KNN. Del análisis de la Tabla 14 se concluye que el clasificador SVM tiene un mejor desempeño que los clasificadores Lineal y KNN, lo cual coincide con que sea el de mayor media.

Tabla 13. Prueba de rangos con signos de Wilcoxon

	KNN(1) – Lineal	SVM – Lineal	SVM – KNN(1)
Z	-4,71	-2,934	-3,059
Sig. asintót. (bilateral)	0,638	0,003	0,002

Tabla 14. Rangos

		N	Rango promedio	Suma de rangos
Lineal - SVM	Rangos negativos	11 <sup>a</sup>	6,00	66,00
	Rangos positivos	0 <sup>b</sup>	0,00	0,00
	Empates	1 <sup>c</sup>		
	Total	12		
KNN - SVM	Rangos negativos	12 <sup>d</sup>	6,50	78,00
	Rangos positivos	0 <sup>e</sup>	0,00	0,00
	Empates	0 <sup>f</sup>		
	Total	12		
Lineal - KNN	Rangos negativos	5 <sup>g</sup>	9,00	45,00
	Rangos positivos	7 <sup>h</sup>	4,71	33,00
	Empates	0 <sup>i</sup>		
	Total	12		

a. Lineal &lt; SVM

d. KNN &lt; SVM

g. Lineal &lt; KNN

b. Lineal &gt; SVM

e. KNN &gt; SVM

h. Lineal &gt; KNN

c. Lineal = SVM

f. KNN = SVM

i. Lineal = KNN

## **CONCLUSIONES Y RECOMENDACIONES**

### **Conclusiones**

- 1 La obtención de una base de imágenes de microscopía celular de muestras de sangre de eritrocitos aislados y aglomerados, anotada por especialistas, permitió realizar el análisis comparativo entre diferentes algoritmos clasificadores implementados.
- 2 Los rasgos utilizados en la clasificación de las imágenes celulares de eritrocitos aislados e independientes, demostraron ser representativos en el reconocimiento de patrones.
- 3 Se obtuvieron resultados satisfactorios en la clasificación entre eritrocitos aislados y aglomerados empleando diferentes algoritmos, siendo el clasificador de máquinas de vectores soporte el que mejor índices de desempeño presentó.

### **Recomendaciones**

- 1 Ampliar la cantidad de rasgos a utilizar en la clasificación de los eritrocitos aislados y aglomerados con el objetivo de mejorar los resultados en la clasificación.
- 2 Implementar y aplicar otros algoritmos clasificadores citados en bibliografía y evaluar su desempeño respecto a los utilizados en el presente trabajo.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] L. Chinea-Valdés and J. V. Lorenzo-Ginori, "Evaluation of distance transform based alternatives for image segmentation of overlapping objects.," presented at the Scientific Conference on Computer Science and Informatics, Informática 2011, Havana, Cuba, 2011.
- [2] D. Jie, L. Jing-feng, and Y. Jing-yu, "Combined Technologies in Analysis of Overlapping Cells," in *E-Business and E-Government (ICEE), 2010 International Conference on*, 2010, pp. 1608-1612.
- [3] D. L. Longo, *Atlas de hematología y análisis de frotis de sangre periférica vol. 2: The McGraw-Hill Companies, Inc*, 2012.
- [4] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing using Matlab*: Pearson Prentice Hall, 2004.
- [5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2da ed.: Pearson Prentice Hall, 2002.
- [6] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3ra ed.: Pearson Prentice Hall, 2008.
- [7] C. Di Ruberto, A. Dempster, S. Khan, and B. Jarra, "Analysis of infected blood cell images using morphological operators," *Image and Vision Computing*, vol. 20, pp. 133-146, 2002.
- [8] E. Espinoza, G. Martinez, J. G. Frerichs, and T. Scheper, "Cell cluster segmentation based on global and local thresholding for in-situ microscopy," in *Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on*, 2006, pp. 542-545.
- [9] P. Soile, *Morphological Image Analysis: Principles and Applications*: Springer Verlag New York, Inc, 2003.
- [10] J. R. Parker, *Algorithms For Image Processing and Computer Vision*: John Wiley & Sons, Inc, 1997.
- [11] A. Koschan and M. Abidi, *Digital Color Image Processing*: Jhon Wiley & Sons, Inc, 2008.
- [12] W. Burger and M. J. Burge, *Digital Image Processing: An Algorithmic Introduction using Java*, 1ra ed.: Springer, 2010.

- [13] E. R. Dougherty and R. A. Lotufo, *Hands-on Morphological Image Processing*: Spie Press, 2003.
- [14] "MATLAB and Image Processing Toolbox Release 2010a," ed. Massachusetts, United States: The MathWorks, Inc., Natick.
- [15] T. Acharya and A. K. Ray, *Image Processing. Principles and Applications* John Wiley & Sons, Inc. , 2005.
- [16] M. S. Nixon and A. S. Aguado, *Feature Extraction and Image Processing*, 1ra ed.: Newnes, 2002.
- [17] R. Llobregat, "Procesado digital de imágenes de video para la detección de humo," Departamento de Comunicaciones, Gandia, Universidad Politécnica de Valencia, 2011.
- [18] B. Jahne, *Digital Image Processing*, 5ta ed.: Springer, 2002.
- [19] J. A. Leehan, V. M. Bañuelos, and J.-F. Lerallut, *Procesamiento de imágenes biomédicas*, 2000.
- [20] N. Otsu, *A threshold selection method from gray-level histograms* vol. 19: IEEE Transactions on Systems, Man and Cybernetics, 1979.
- [21] R. R. Morales, *Fundamentos de la Vision por Computadora*: Instituto de Cibernetica, Matematica y Fisica (ICIMAF) 2004.
- [22] E. U. Scott, "Computer vision and image processing," *PH*, New Jersey, 1998.
- [23] F. M. Seco, "Clasificadores eficaces basados en algoritmos rápidos de búsqueda del vecino más cercano," Departamento de Lenguajes y Sistemas Informáticos, Universidad de Alicante, 2004.
- [24] N. A. M. Isa, M. Y. Mashor, and N. H. Othman, *An automated cervical precancerous diagnostic system* vol. Vol. 42: Artif. Intell. Med, 2008.
- [25] I. A. Yusoff, N. A. Mat Isa, N. H. Othman, S. N. Sulaiman, and Y. Jusman, "Performance of neural network architectures: Cascaded MLP versus extreme learning machine on cervical cell image classification," in *Information Sciences Signal Processing and their Applications (ISSPA), 2010 10th International Conference on*, 2010, pp. 308-311.
- [26] S. Osowski, R. Siroic, T. Markiewicz, and K. Siwek, "Application of Support Vector Machine and Genetic Algorithm for Improved Blood Cell Recognition," *Instrumentation and Measurement, IEEE Transactions on*, vol. 58, pp. 2159-2168, 2009.
- [27] C. Y.-K. H. Po-Chi, C. Po-Chou, C. Yung-Fu, C. Rung-Ching, y H. Y.-R. Huang, *Quantitative Assessment of Pap Smear Cells by PC-Based Cytopathologic Image Analysis System and Support Vector Machine*. ICMB 2008, LNCS, D. Zhang: Springer-Verlag Berlin Heidelberg, 2007.
- [28] R. D. N. Dehak, P. Kenny, N. Brammer, P. Ouellet, y P. Dumouchel, *Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification*: LNCS, 2009.

- [29] Y. Marinakis, G. Dounias, and J. Jantzen, "Pap smear diagnosis using a hybrid intelligent scheme focusing on genetic algorithm based feature selection and nearest neighbor classification," *Computers in Biology and Medicine*, vol. 39, pp. 69-78, 2009.
- [30] M. T. Escobedo and J. A. Salas, *P. CH. Mahalanobis y las aplicaciones de su distancia estadística* vol. 5: Culcyt, 2008.
- [31] M. I. Jordan and Y. Weiss, *Analysis and an algorithm on spectral clustering* vol. 14. Adv. Neural Inf. Process. Syst., 2002.
- [32] A. Gersho and R. M. Gray, *Vector quantization and signal compression* vol. 159: Springer Science & Business Media, 2012.
- [33] J. Jantzen, J. Norup, G. Dounias, and B. Bjerregaard, "Pap-smear Benchmark Data For Pattern Classification," ed. Herlev, 2005, p. 9.
- [34] M. Wernick, Y. Yongyi, J. Brankov, G. Yourganov, and S. Strother, "Machine Learning in Medical Imaging," *Signal Processing Magazine, IEEE*, vol. 27, pp. 25-38, 2010.
- [35] G. A. Betancourt, *Las máquinas de soporte vectorial (svms)* vol. 1: Scientia et Technica, 2005.
- [36] A. Taboada-Crispi, *Anomaly Detection in Medical Image Analysis. Handbook of Research on Advanced Techniques in Diagnostic Imaging and Biomedical Applications*: Medical Info Science Reference, 2009.
- [37] W. H. Organization, "World malaria report: 2014," World Health Organization (WHO), 2014.
- [38] H. p. e. Residente. ((2013), accedido el 11/5/2015). *Frotis de Sangre Periférica*. Available: <http://hematologiaparaelresidente.blogspot.com/2013/07/frotis-de-sangre-periferica.html>
- [39] D. Frances, ""Adquisición de imágenes de microscopía celular en estudios de malaria," Trabajo de Diploma," Centro de Estudios de Electrónica y Tecnologías de la Información (CEETI), Universidad Central "Marta Abreu" de Las Villas, 2013.
- [40] SPSS and IBM, "IBM SPSS Statistics 21 Core System User's Guide," 2012.
- [41] N. Abbas, D. Muhamad, and A. H. Abdullah, "Semi-Automatic Clustered Red Blood Cells Splitting and Counting in Thin Blood Smear Images," 2014.
- [42] N. Abbas and D. Muhamad, "Accurate red blood cells automatic counting in microscopic thin blood smear digital images," *Science International Journal*, vol. 26, pp. 1119-24, 2014.
- [43] J. Arco, J. Górriz, J. Ramírez, I. Álvarez, and C. Puntonet, "Digital image analysis for automatic enumeration of malaria parasites using morphological operations," *Expert Systems with Applications*, vol. 42, pp. 3041-3047, 2015.
- [44] I. Baghli, A. Nakib, E. Sellam, M. Benazzouz, A. Chikh, and E. Petit, "Hybrid framework based on evidence theory for blood cell image segmentation," in *SPIE Medical Imaging*, 2014, pp. 903815-903815-8.



- [45] J. Kaur and K. Kaur, "Review of Content Based Retrieval of Malarial Positive Images from Clinical Database," *Journal of Emerging Technologies in Web Intelligence*, vol. 6, pp. 381-387, 2014.
- [46] Z. May, S. Azreen Mohd Aziz, and R. Salamat, "Automated quantification and classification of malaria parasites in thin blood smears," in *Signal and Image Processing Applications (ICSIPA), 2013 IEEE International Conference on*, 2013, pp. 369-373.
- [47] V. M. Ramos, "Implementación y aplicación de funciones de granulometría morfológica para la clasificación de imágenes de microscopía celular," Facultad de Ingeniería Eléctrica, Universidad Central "Marta Abreu" de Las Villas, Centro de Estudios de Electrónica y Tecnologías de la Información, 2014.
- [48] J. A. Alkrimi, H. A. Jalab, L. E. George, A. R. Ahmad, A. Suliman, and K. Al-Jashamy, "Comparative Study Using Weka for Red Blood Cells Classification," 2015.
- [49] S. Savkare and S. Narote, "Automated system for malaria parasite identification," in *Communication, Information & Computing Technology (ICCICT), 2015 International Conference on*, 2015, pp. 1-4.
- [50] S. Widodo, "Texture analysis to detect malaria tropica in blood smears image using support vector machine," 2014.

## ANEXOS

### Anexo I      Códigos de los algoritmos programados

```
name='name'; % Cargo las imágenes de una en una solo cambiando el 'name'
ImOrg=imread (name,'tif'); % Se carga desde la variable name
Imagen_orig=im2double(ImOrg); % Se convierte la intensidad de la imagen a doble
%precisión
% Segmentación por Otsu
% Obtener la imagen HSI para utilizar el plano de la intensidad.
bb=rgb2hsi(Imagen_orig);% convierto la imagen al espacio HSI
bbint=bb(:, :,3); %Tomo la tercera componente
bbint= medfilt2(bbint,[5 5]);% filtrado de mediana
bbint = adapthisteq(bbint,'ClipLimit' ,0.03,'Range' , 'Original');
% APLICAR TOP HAT
% Realizando la corrección de la iluminación de fondo no uniforme.
% A partir de aquí se trabajará con la imagen de intensidades invertidas.
strg=strel('disk',40);
bbintn=1-bbint; % Hallando el "negativo" de la imagen.
bbintnc = imtophat(bbintn,strg);
% Segmentación por Otsu
level = graythresh(bbintnc);
imagsegm= im2bw(bbintnc,level);
huecoscell=imfill (imagsegm,'holes');%Eliminado de huecos
huecoscell = imclearborder(huecoscell,8); %Eliminar las células de los bordes
```

Se eliminan los artefactos y regiones espurias

%Se etiqueta la imagen binaria.

[L cantidad] = bwlabel(huecoscell);

% Con la función regionprops se calcular todas las áreas de la regiones

% etiquetadas.

stats = regionprops(L,'Area');

allArea = [stats.Area];

%Se realiza una apertura por área para eliminar los elementos pequeños.

Mascaracell= bwareaopen(huecoscell,round( mediana/2));

### **Recortado de las células**

[L, num] = bwlabel(Mascaracell);

box = regionprops(L, 'BoundingBox');

coord= cat(1, box.BoundingBox);

[frgb crgb]=size(Mascaracell);% Buscar tamaño

%submasc=zeros([frgb crgb num]);

subregion=cell([1 num]); % salva espacio en memoria

subblackw=cell([1 num]); % salva espacio en memoria

imacolor=cell([1 num]); % salva espacio en memoria

for i=1:num

celula=L==i;

% Multiplica la máscara y la imagen en cada plano, para obtener la célula en colores.

red=immultiply(celula,Imagen\_orig(:, :,1));

green=immultiply(celula,Imagen\_orig(:, :,2));

blue=immultiply(celula,Imagen\_orig(:, :,3));

% RECORTAR

C=coord(i,:);

C=round(C);

ext\_red=red(C(2)-3:(C(2)+C(4))+3,C(1)-3:(C(1)+C(3))+3);

ext\_green=green(C(2)-3:(C(2)+C(4))+3,C(1)-3:(C(1)+C(3))+3);

ext\_blue=blue(C(2)-3:(C(2)+C(4))+3,C(1)-3:(C(1)+C(3))+3);

subblackw1=celula(C(2)-3:(C(2)+C(4))+3,C(1)-3:(C(1)+C(3))+3);

```

imarecort=cat(3, ext_red,ext_green,ext_blue);
imacolor1=Imagen_orig(:,:,1);
imacolor1=imacolor1(C(2)-3:(C(2)+C(4))+3,C(1)-3:(C(1)+C(3))+3);
imacolor2=Imagen_orig(:,:,2);
imacolor2=imacolor2(C(2)-3:(C(2)+C(4))+3,C(1)-3:(C(1)+C(3))+3);
imacolor3=Imagen_orig(:,:,3);
imacolor3=imacolor3(C(2)-3:(C(2)+C(4))+3,C(1)-3:(C(1)+C(3))+3);
imacolorg=cat(3,imacolor1,imacolor2,imacolor3);
imacolor(i)={imacolorg};% Cell Array con las células en colores con el fondo
subblackw(i)={subblackw1}; % Máscara de las células
subregion(i)={imarecort};% Cell array con las imágenes de las células recortadas con el
fondo negro.
end

```

### **Se implementan los rasgos para la clasificación**

```

for g=1:120
imamascara=imread(['BlackW',num2str(g),'.tif']);% se cargan las imágenes de máscaras
[MATRIZNUM2 NUM2] = bwlabel(imamascara);
% Los rasgos son los siguientes
stats2 = regionprops(MATRIZNUM2, 'Area');
allArea = [stats2.Area];
stats3 = regionprops(MATRIZNUM2,'ConvexArea');
allConvexArea = [stats3.ConvexArea];
stats4 = regionprops(MATRIZNUM2, 'Eccentricity');
allEccentricity = [stats4.Eccentricity];
stats5 = regionprops(MATRIZNUM2,'MajorAxisLength');
allMajorAxisLength = [stats5.MajorAxisLength];
stats6 = regionprops(MATRIZNUM2,'MinorAxisLength');
allMinorAxisLength = [stats6.MinorAxisLength];
stats7 = regionprops(MATRIZNUM2, 'Solidity');
allSolidity = [stats7.Solidity];
stats8 = regionprops(MATRIZNUM2, 'EquivDiameter' );

```

```

allEquivDiameter = [stats8.EquivDiameter];
stats9 = regionprops(MATRIZNUM2, 'Extent');
allExtent = [stats9.Extent];
% Se crea un vector con los rasgos.
matrasgos=[allArea',allConvexArea',allEccentricity',allMajorAxisLength',allMinorAxisLength',allSolidity',allEquivDiameter',allExtent'];
matriz(g,:)=matrasgos(1,:);% esto (1,:) significa que de la variable matrasgos
% solo tomo la primera fila con todas las
% columnas
end

```

### **Conformación de grupos para pruebas de clasificadores**

```

load 'Cell_Agl.mat' % se cargan la matriz de rasgos de las células aglomeradas
rasgosaglome=matriz; % debido a que la matriz de rasgos se le puso otro nombre para
%diferenciarlos
load 'Cell_Ind.mat' % se cargan la matriz de rasgos de las células independientes
rasgosindep=matriz;% debido a que la matriz de rasgos se le puso otro nombre para
%diferenciarlos
%rasgosindep (120 filas x 8 rasgos columnas)
%rasgosaglome(120 filas x 8 rasgos columnas)
numgrup=12; % 12 grupos
numfeats=8; % 8 rasgos
numfilgrup=20; % 20 imágenes por grupo
%rasgos3d(#filas,rasgos,grupo)
%clases3d(clasificación, grupo)
rasgos3d=zeros(numfilgrup,numfeats,numgrup);
clases3d=cell(numfilgrup, numgrup);
for j=1:numgrup
pos=(j-1)*10+1
rasgosin=rasgosindep(pos:pos+9,:);
rasgosagl=rasgosaglome(pos:pos+9,:);
rasgos3d(:,j)=[rasgosin;rasgosagl];

```

```

claind10= cell(10,1);
claaglom10= cell(10,1);
claind10(:,1)={'independiente'};
claaglom10(:,1)={'aglomerado'};
clases3d(:,j)=[claind10;claaglom10]; % Cada columna: clases de una matriz en el arreglo
%rasgosg3.
end

```

### **Implementación de los clasificadores y su evaluación**

```

%Cargar Rasgos 3D
%Cargar Clases 3D
%Clasificadores Agregados
cp1 = classperf(clases3d(:,1));% inicialización de CP
cp2 = classperf(clases3d(:,1));
cp3 = classperf(clases3d(:,1));
f_mean=zeros(12,3);
for j=1:12
    indices = crossvalind('Kfold',clases3d(:,j),5); % la validación cruzada
    for i = 1:5
        test = (indices == i); train = ~test;
        %caso 'lineal'
        classlineal = classify(rasgos3d(test,:,j),rasgos3d(train,:,j),clases3d(train,1,:), 'linear');
        %caso 'KNN'
        classknn = knnclassify(rasgos3d(test,:,j),rasgos3d(train,:,j),clases3d(train,1,:),1,'cityblock');
        % se probó para  $k = 1, 3$  y  $5$ 
        %caso 'SVM'
        svmStruct1=svmtrain(rasgos3d(train,:,j),clases3d(train,1:), 'KERNEL_FUNCTION','rbf','RBF_Sigma',2);
        classsvm = svmclassify(svmStruct1,rasgos3d(test,:,j));
        classperf(cp1,classlineal,test);
        classperf(cp2,classknn,test);
        classperf(cp3,classsvm,test);
    end
end

```

```
end
for z=1:3
eval( ['er(z,:)=cp' int2str(z) '.ErrorRate;'] );
eval( ['se(z,:)=cp' int2str(z) '.Sensitivity;'] );
eval( ['cr(z,:)=cp' int2str(z) '.CorrectRate;'] );
eval( ['sp(z,:)=cp' int2str(z) '.Specificity;'] );
eval( ['pp(z,:)=cp' int2str(z) '.PositivePredictiveValue;'] );
eval( ['pn(z,:)=cp' int2str(z) '.NegativePredictiveValue;'] );
f_mean(j,z)=(2*se(z)*pp(z))/(se(z)+pp(z));
end
end
```