



Universidad Autónoma de Querétaro
Facultad de Informática
Maestría en Ciencias de la Computación

Segmentación Automática de Células Mediante Grafos


Opción de titulación
Tesis

Que como parte de los requisitos para obtener el Grado de
Maestría en Ciencias de la Computación

Presenta:
Ing. Miguel Angel Razo Salas

Dirigido por:
Dra. Ana Marcela Herrera Navarro

Dra. Ana Marcela Herrera Navarro
Presidente


Firma

Dr. Iván Ramón Terol Villalobos
Secretario


Firma

Dra. Sandra Luz Canchola Magdaleno
Vocal


Firma

M.I.S.D Erika Del Rio Magaña
Suplente


Firma

Dra. Diana Margarita Córdova Esparza
Suplente


Firma



M.I.S.D. Juan Salvador Hernández
Valerio
Director de la Facultad



Dra. Ma. Guadalupe Flavia Loarca
Director de Investigación y Posgrado

RESUMEN

El calcio intracelular es parte fundamental del cuerpo humano ya que actúa como segundo mensajero intracelular. Dependiendo de la concentración que presentan las células foliculares es posible diagnosticar ciertos tipos de enfermedades como la hipertensión, arritmia cardíaca, problemas hematológicos, enfermedades musculares, trastornos hormonales, y otras más específicas, como la lesión y muerte celular. Por lo tanto, el estudio del calcio intracelular despertó mucho interés en ramas como la bio-química y bio-medicina donde por medio de un microscopio de fluorescencia capta la luz que emiten las células en la región del citoplasma. La cantidad de calcio en la célula está directamente relacionada con la suma de las intensidades que presenta la imagen en cada célula y el proceso que comúnmente es utilizado para esto se llama segmentación. La segmentación es una parte de la visión por computadora donde por medio de algún algoritmo de segmentación se obtienen las coordenadas espaciales en la imagen de ciertos objetos de interés que en este caso son las células. Este proceso generalmente lo hacen manualmente pero el proceso tiene a ser muy tardado y propenso a error humano cuando son demasiadas las células a encontrar. En esta investigación se propone un método de segmentación completamente automático donde la imagen a segmentar pasa por dos procesos importantes, el primero es el de normalización y eliminación de ruido de la imagen aplicando el algoritmo de umbral adaptativo local para definir los bordes de las células de una mejor manera ya que la fluorescencia tiende a difuminarlos mezclándolos con el fondo. El segundo proceso es el de localizar las células automáticamente por medio de grafos utilizando el algoritmo de Grafos llamado Max tree que trabaja en base a regiones facilitando así el procesamiento de las imágenes.

(Palabras clave: segmentación, grafos, células)

SUMMARY

The intracellular calcium is important for the human body cause it act as a second intracellular messenger. Depending the concentration of calcium in follicular cells it's possible detect some types of sickness as hypertension, hormonal disorders, cardiac arrhythmia, hematological problems, muscle diseases and other more specific as injury and cell death. Thus, the study of intracellular calcium aroused much interest in biochemistry and biomedecine where a fluorescence microscope takes the light produced by the cytoplasm of the cells. The amount of calcium in the cells is strictly corelated with the sum of all the intensities that the image present in each cell and segmentation is the most common process to do this work. The segmentation is part of computer vision in which the segmentation algorithms obtain spatial coordinates in the images of certain interest objects that in this case are the cells. Typically, this process is manual, and it tends to be slow and susceptible to a human error when there is a large set of cells to found. This investigation proposes a segmentation method fully automatic where the image to segment pass through two important process, the first one consist in normalize and noise remove whit the algorithm Adaptive Local Thresholding to define better the limits of the cells because the fluorescence tends to degrade the border mixing it whit the background. The second process is to locate the cells automatically through graphs using the Max tree algorithm that work in base of regions making easier the image processing.

(Key words: segmentation, graphs, cells)

AGRADECIMIENTOS

Principalmente quiero agradecer a mi familia. A mi madre Ma. de Lourdes Salas, a mi padre Eduardo Daniel Jareño y hermana Camila Jareño Salas por el apoyo incondicional de manera económica, intelectual y sobre todo moral que recibí de su parte a lo largo de estos dos años en mi maestría.

A mis asesores Dra. Ana Marcela Herrera Navarro y Dr. Ivan Terol Villalobos por el tiempo invertido en revisiones y correcciones a mi trabajo de investigación.

A Tzitzí Oliva por el apoyo otorgado durante dos largos años de estudios, consejos y revisión en términos médicos.

A cada uno de mis amigos de Querétaro y Zacatecas por aguantarme durante situaciones de estrés.

Por último, también quiero agradecer al Consejo Nacional de Ciencia y Tecnología (CONACYT) por todo el apoyo financiero recibido, con el cual se obtuvieron diversos productos como la tesis, 2 artículos publicados y el código del algoritmo.

TABLA DE CONTENIDOS

1. INTRODUCCIÓN.....	7
1.1 OBJETIVO GENERAL.....	10
1.2 OBJETIVO PARTICULAR	10
2. ANTECEDENTES Y FUNDAMENTOS TEORICOS	12
2.1 GRAFOS.....	14
2.2 ÁRBOLES	15
2.3 ESTADO DEL ARTE	16
2.3.1 <i>Redes neuronales</i>	16
2.3.2 <i>Fuzzy c-mean</i>	17
2.3.3 <i>Umbral bimodal</i>	17
2.3.4 <i>Patrones de membrana</i>	18
2.4 ANTECEDENTES.....	20
2.4.1 <i>Umbral Adaptativo Local</i>	20
2.4.2 <i>Watershed</i>	21
2.4.3 <i>Max-tree</i>	23
3. METODOLOGÍA	26
3.1 NORMALIZACIÓN DE IMAGEN	26
3.2 CONSTRUCCIÓN DE MAX-TREE.....	28
3.3 SIMPLIFICACIÓN Y FILTRADO DEL ÁRBOL MAX-TREE.....	29
3.4 RESTITUCIÓN DE IMAGEN.....	30
3.5 CONTEO DE CALCIO INTRACELULAR	31
4. RESULTADOS Y DISCUSIÓN	32
5. REFERENCIAS.....	35
6. APÉNDICE.....	37
6.1 ARTÍCULO ACEPTADO	43
6.2 CONSTANCIAS Y RECONOCIMIENTOS	49

ÍNDICE DE TABLAS

Figura 1.2 Histograma de la imagen original.	9
Figura 1.3 Aplicación de umbral a la imagen a) de la Figura 1.1a umbral con valor de 5, b) umbral con valor de 40 aplicado a la Figura 1.1b.	10
Figura 2.1 Representación gráfica de la abstracción a un relieve topográfico.	14
Figura 2.1.1 Grafo dirigido con pesos en las aristas.	15
Figura 2.2.1 Representación gráfica de un árbol.	16
Figura 2.3.1.1 a) Imagen original, b) obtención de marcadores con LLM y c) obtención de contornos con CML.	17
Figura 2.3.3.2 Proceso de restauración de imagen en citoplasma y núcleo.	18
Figura 2.3.4.1 Reconocimiento inicial de patrones en membranas.	19
Figura 2.4.1.1 Localidad ideal y dinámica con radio diferente las 3 imágenes.	20
Figura 3.1 Diagrama de flujo de la metodología a seguir para segmentar las células.	26
Figura 4.1 (a) Imágenes originales, (b) imágenes después de aplicar el Algoritmo Umbral Adaptativo Local.	33
Figura 4.2 Etiquetamiento de células en imagen normalizada.	33

1. INTRODUCCIÓN

Muchos de los procesos que realizan las células están altamente relacionados con los niveles de Calcio (Ca^{2+}) que tienen, donde algunas veces el calcio actúa como un mensajero intracelular alterno, produciendo alteraciones fisiopatológicas, tales como; la hipertensión, arritmia cardíaca, problemas hematológicos, enfermedades musculares, trastornos hormonales, y otras más específicas, como la lesión y muerte celular (Berridge, Bootman, & Roderick, 2003).

Debido a que es el único metal que ejerce la función de segundo mensajero y a su importancia, ha despertado un gran interés por su estudio dentro de varias ramas de la investigación, como en la bioquímica y la biomedicina (Doadrio, n.d.). Al estimular las células con distintas sustancias generan movimiento y activación celular, el cual se manifiesta con un efecto fluorescente en la parte del citoplasma como lo muestra la Figura 1.1 (Stricker, Centonze, Paddock, & Schatten, 1992).

Se han desarrollado distintas técnicas para analizar el comportamiento y su concentración, como la microscopia de fluorescencia, pero los procesos biológicos son demasiado rápidos, creando la necesidad de segmentar las células para observar dicho comportamiento. En un sistema de microscopio, la señal de una muestra existe primero como fotones emitidos, que siguen una trayectoria de luz a un fotodetector para convertirse en una señal digital. Durante este proceso, varias fuentes de error pueden oscurecer la señal verdadera y causar una pérdida de resolución espacial en imágenes sin procesar (Arce, Wu, & Tseng, 2013). Esta gran cantidad de parámetros que pueden o no influenciar la salida biológica, experimentos sobre imágenes de células seguidas son replicados y/o emparejados simultáneamente y resultan en grandes conjuntos de imágenes (Zhang & Li, 2017). El contraste que existe entre el fondo y las células se ve afectado de manera significativa debido a estos errores y a la alta oclusión de células, haciendo la detección de las células más difícil. Usualmente termina en la múltiple identificación de células como una sola entidad o que una célula la identifique como varias entidades. El bajo contraste y pobre iluminación conlleva a una mala segmentación, por lo tanto, es de gran importancia desarrollar algoritmos de segmentación robustos para resolver problemas con condiciones no ideales.

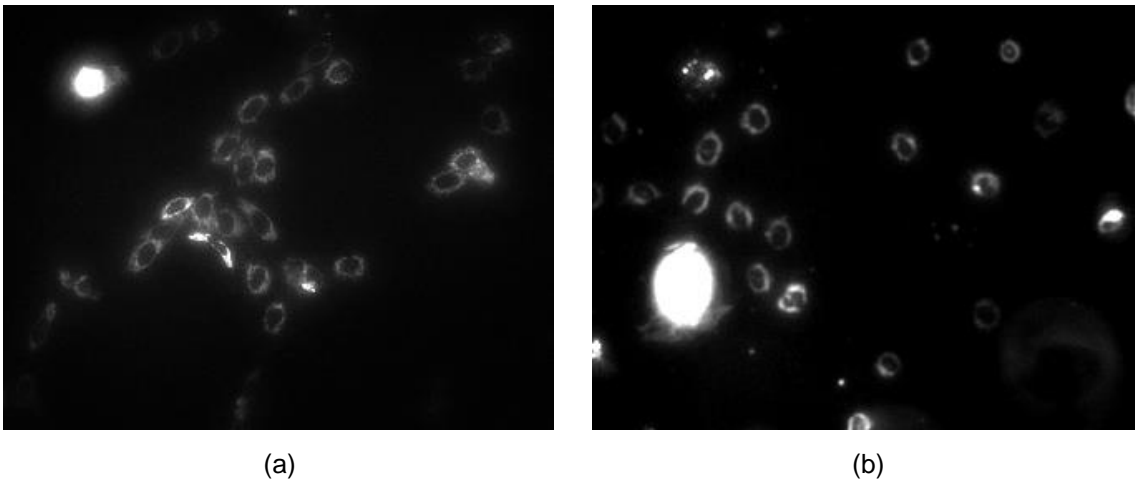


Figura 1.1 Imágenes originales de células tomadas con un microscopio de fluorescencia.

Fuente: Instituto Neurobiología UNAM

La segmentación manual es subjetiva y consume mucho tiempo al ser tan grande la cantidad de imágenes, decrementando la eficiencia de inspección, dejando como la única opción viable el análisis y procesamiento automático.

La segmentación de células es un análisis muy común que provee características de los objetos identificados como el espacio donde se encuentran, siendo un proceso previo a operaciones para cuantificar parámetros como la fluorescencia intracelular (Dima et al., 2011).

Esta segmentación de células en imágenes de microscopía fluorescente se torna difícil usando métodos convencionales. Algunos de los métodos que actualmente se utilizan en la segmentación de células son watershed, agrupación basada en histogramas, umbral y detectores de límites.

Detectar orillas es una de las tareas primarias en la visión por computadora. La manera más sencilla de detectar bordes es el uso de la derivada o diferencial, dentro de un vecindario pequeño. Esta operación permite calcular las variaciones entre un punto y su localidad. Los bordes no son más que una discontinuidad en dicha imagen (Sucar & Gomez, 2002). Los métodos para la detección de contornos solo son eficientes si están cerca de la solución final, ya que, cuando se estudia una estructura celular compleja, éstos modelos generalmente fallan si su primer acercamiento está lejos de la solución (Ujihara, Nakamura, Miyazaki, & Wada, 2013).

La segmentación por histograma se basa en la suposición de que existen dos regiones en la imagen, un objeto sobre el fondo homogéneo y bien definido. Entonces, si existen solo dos

secciones se puede asumir que en el histograma habrá dos picos conformados uno por el fondo y el otro por los píxeles del objeto en sí, siendo el valle que se forma entre estos dos picos el valor óptimo para diferenciar y segmentar (Sucar & Gomez, 2002). La segmentación por una sola intensidad como umbral ayuda a distinguir la célula del fondo particionando la imagen en esas dos clases antes mencionadas, pero como lo muestra el histograma de la imagen original, en la Figura 1.2, no se tiene un valle, del que se pueda hacer una selección del umbral óptimo para dividir estas dos secciones, esto se debe a que en la imagen original se tiene un importante degradado haciendo que el histograma sea una especie de pendiente.

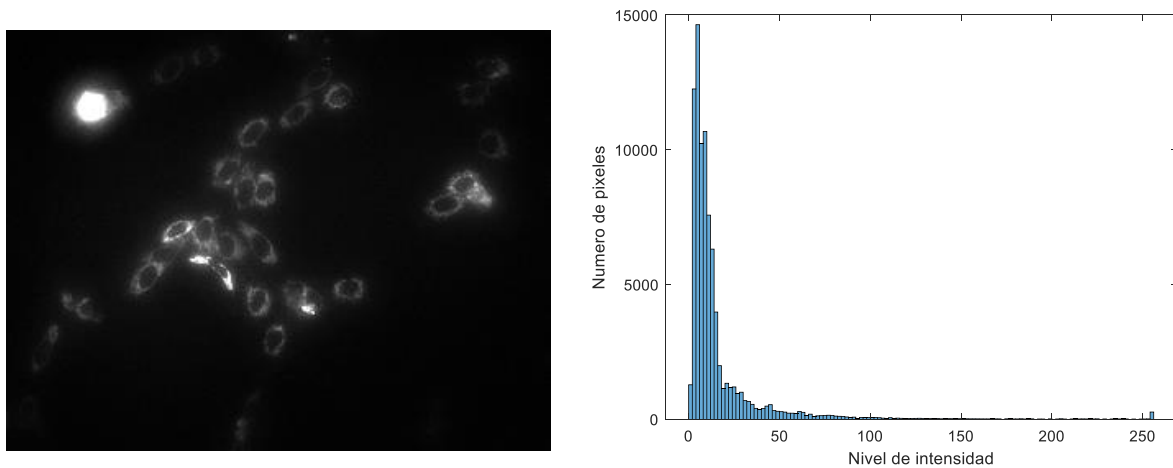


Figura 1.1.1 Histograma de la imagen original.

Fuente: Elaboración propia.

Por consiguiente, configurar el umbral cerca del pico del histograma puede omitir gran parte del fondo y garantizar una buena segmentación para células con intensidades bajas (Arce et al., 2013). Sin embargo, también incluirá algunos píxeles del fondo y ruido que se encuentre afectado por brillo de células fluorescentes cercanas. Alternativamente, configurar el umbral lejos del pico del histograma omitirá píxeles del fondo cercanos a las células, pero también omitirá regiones de bajas intensidades en las células véase Figura 1.3.

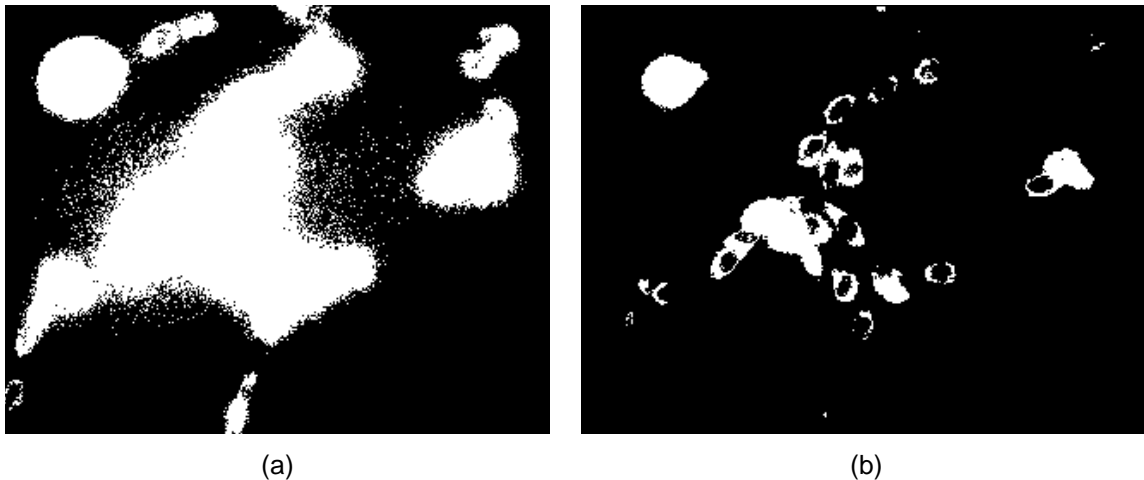


Figura 1.1.2 Aplicación de umbral a la imagen a) de la Figura 1.1a umbral con valor de 5, b) umbral con valor de 40 aplicado a la Figura 1.1b.

Fuente: Elaboración propia.

Las imágenes de células fluorescentes deben ser sometidas a cierto nivel de procesamiento para evitar los errores y obtener un buen resultado en la segmentación. Muchos algoritmos han sido propuestos para la segmentación de células, Watershed es uno de los más populares en estos casos, aunque no es tan bueno lidiando con células solapadas y es muy fácil que caiga en un problema de sobre-segmentación. Watershed con marcadores controlados reduce esta sobre-segmentación, pero la detección de los marcadores aun no es tan precisa (Zhang & Li, 2017).

Estos tipos de estrategias para segmentar solo se aproximan a los límites celulares, y por consecuencia produce un gran error agregando o excluyendo píxeles que no son propiamente de la célula. Se han desarrollado métodos de procesamiento de imágenes más sofisticados que ofrecen una mejor segmentación, pero generalmente requieren de una gran cantidad de tiempo computacional, interacción del usuario o entrenamiento especializado para que realice su trabajo adecuadamente (Arce et al., 2013).

1.1 Objetivo General

Proponer un algoritmo de segmentación automática de células mediante grafos para el estudio de la concentración del Ca^{2+} intracelular a partir de la variación de la fluorescencia en la célula.

1.2 Objetivo Particular

1. Aplicar el Algoritmo de Umbral Adaptativo Local para la normalización de la imagen.

- Definir la desviación estándar de la imagen completa.
 - Automáticamente establecer la vecindad de cada uno de los píxeles en la imagen.
 - Calcular el promedio y desviación estándar de cada vecindario en especial.
 - Normalizar la imagen pixel por pixel.
2. Detectar automáticamente las células.
- Etiquetar cada pixel de las imágenes.
 - Construir un árbol (Max-tree) con las distintas intensidades de grises.
 - Aplicar filtros de área al árbol construido.
 - Reconstruir la imagen con los píxeles obtenidos.
3. Estudiar el comportamiento de la variabilidad del calcio intracelular mediante el análisis temporal de la intensidad de los píxeles.
- Recorrer cada una de las ramas construidas en el Max-tree para sumar intensidades de los píxeles que cada uno de los nodos almacena siendo estos el citoplasma de las células.

2. ANTECEDENTES Y FUNDAMENTOS TEORICOS

La segmentación de una imagen está compuesta por todas las técnicas y métodos de los cuales se hacen uso para extraer la información de una imagen. El principal objetivo de está es localizar objetos o límites dentro de la imagen descomponiéndola en sus partes constituyentes, basándose en características locales, permitiendo diferenciar el fondo del objeto y los objetos entre si con el fin de pasar una simple representación gráfica a información significativa. Por lo general las imágenes están conformadas por secciones o regiones que tienen características similares (nivel de gris, textura, forma, etc.), zonas homogéneas; sin embargo, en las imágenes de las células, la fluorescencia hace que exista un contraste casi nulo entre las células y el fondo.

Las imágenes de células tienen un degradado importante debido a la luz que emite la fluorescencia dándole un mayor grado de dificultad a la segmentación. La segmentación está dividida en dos importantes procesos, 1) el tratamiento de la imagen y remoción de ruido para evitar la sobre-segmentación, y 2) extraer la información de la imagen detectando límites y objetos de interés para el usuario. Una vez concluidos estos dos procesos se hace una toma de decisión en base a la información obtenida. Por lo tanto, para lograr una buena segmentación es necesario aplicar una serie de técnicas hasta tener un fondo homogenizado y bien definido, es decir remover partículas sueltas en la imagen que no cumplen con características particulares de los objetos de interés, agregando estas al fondo de la imagen y así poder aislar el fondo de todo lo demás.

Existen diversas imposiciones o problemas con la segmentación automática de células; para empezar, es necesario segmentar o extraer múltiples células de una misma imagen y no solamente un objeto de interés. La forma de las células está sujeta a cambios dinámicos y por lo tanto no tienen una forma homogénea, entonces proponer un modelado matemático es casi imposible. Por otra parte, la segmentación celular induce por naturaleza una distribución de marcadores no homogéneos provocando así una sobre-segmentación o una segmentación incorrecta. Adicionalmente existen diversas variables que también hacen de la segmentación de células sea un proceso difícil de hacer como los siguientes:

- Ruido en la etapa de adquisición de imágenes.
- Bajo Contraste en las imágenes.
- Los procesos metabólicos son rápidos.

- Oclusión de células.

Antes de empezar a localizar las células en las imágenes es conveniente tratar de mitigar los problemas anteriormente descritos. Algunas herramientas serán utilizadas como lo es la desviación estándar (1) muy utilizada en el procesamiento de imágenes ya que nos da información estadística sobre la variación que existe entre la intensidad de píxeles o alguna determinada localidad en la imagen.

$$\sigma = \sqrt{\frac{\sum(x_i - m)^2}{n - 1}} \quad (1)$$

Otra herramienta muy utilizada para la eliminación de objetos e incluso para la misma segmentación son los umbrales (2), no solo en la práctica, si no en la formalización y caracterización de transformaciones. Este se define como el umbral inferior U_i y el superior U_s :

$$f(x) = \begin{cases} 255, & U_i \leq I(x, y) \leq U_s \\ 0, & \text{otros} \end{cases} \quad (2)$$

Algunos otros procesos como la discriminación de objetos por área no son considerados filtros o transformaciones, por el hecho de no contar con ciertas propiedades, aunque también son altamente utilizados en la etapa de mitigación de ruido, ya que ofrecen una clasificación importante en casos donde los objetos de interés son similares en cuanto a sus dimensiones.

Una imagen se puede representar como un arreglo bidimensional o matriz en donde sus valores son el nivel de intensidad ya sea en escala de grises o colores.

Para esta investigación las imágenes están en escala de grises con valores que van de 0 a 255, de los cuales se puede hacer una abstracción y ver la imagen como un relieve topográfico donde los valores más cercanos a 255 son montañas y los cercanos a 0 valles como muestra la Figura 2.1

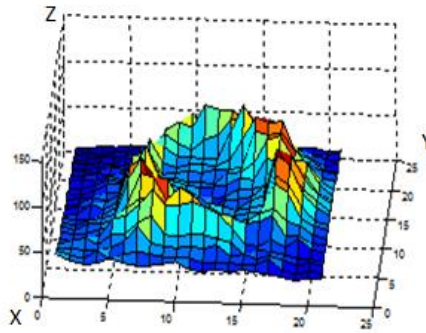


Figura 1.2.1 Representación gráfica de la abstracción a un relieve topográfico.

Aunque el vector bidimensional es una de las formas más comunes de representar a la imagen tiene ciertos inconvenientes. Primero el pixel provee información totalmente local y segundo los algoritmos que trabajan a nivel de pixel necesitan ser eficientes debido a la gran cantidad de pixeles que necesitan procesar. La representación de imagen basada en regiones tiene la solución a estos problemas. Grafo de vecindad de regiones (RAG por sus siglas en ingles) y Max Tree son dos técnicas que dan una representación de la imagen mediante grafos.

2.1 Grafos

Se plantea realizar una segmentación en base a grafos (G), estos son estructuras de datos lineales o no lineales donde cada componente puede tener uno o más predecesores y sucesores. Un grafo G consta de dos conjuntos: $V(G)$ y $A(G)$. El primero lo integran elementos llamados nodos o vértices; el segundo, arcos o aristas. Denotando una gráfica como:

$$G = (V, A)$$

Donde V representa el conjunto de vértices de G y A el conjunto de aristas de G . Si no se hace ninguna especificación, los conjuntos V y A son finitos.

Un camino P de longitud n se define como la secuencia de n vértices que se debe seguir para llegar del vértice v_1 (origen) al vértice v_n (destino).

$$P = (V_1, \dots, V_n)$$

Camino cerrado: El camino P es cerrado si el primer y último vértice es igual; es decir, si $v_1 = v_n$.

Camino simple: El camino es simple si todos sus nodos son distintos, con excepción del primero y del último, que pueden ser iguales.

Grafo conexo: Un grafo conexo se considera así solo si existe un camino simple entre cualesquiera dos de sus nodos

Grafo árbol: Se dice que es un grafo del tipo árbol o árbol libre si G es conexo entre sus nodos

Grafo etiquetado: Se dice que un grafo es etiquetado si sus aristas tienen asignado un valor o un peso

Otra de las propiedades importantes de los grafos es que sus aristas pueden o no ser dirigidas, un grafo es dirigido si la arista a tiene una dirección asignada, es decir cada arista está asociada a un par ordenado (u, v) de vértices de G y sirven para denotar que del vértice u al vértice v solo existe un sentido o dos dependiendo de la dirección se le dé a la arista en sí (Cairó & Guardati, 2010) véase Figura 2.1.1.

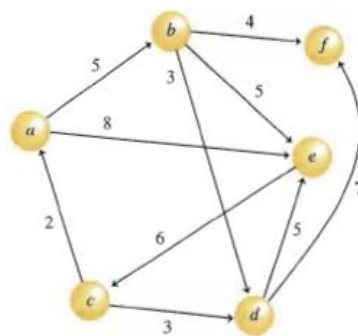


Figura 2.1.1 Grafo dirigido con pesos en las aristas.

Fuente: (Cairó & Guardati, 2010).

2.2 Árboles

Se define árbol como una estructura jerárquica aplicada sobre una colección de elementos u objetos llamados nodos, donde uno de ellos es conocido como raíz. En la estructura se distinguen los nodos para almacenar información y las aristas utilizadas para establecer las relaciones entre los nodos.

Para los árboles, es común utilizar las notaciones padre, hijo, hermano, antecesor, sucesor, etc. A continuación, se explicarán algunas características que tienen los árboles:

Un nodo n es descendiente directo de un nodo x , si existe una arista que apunte del nodo x al nodo n , es decir, n es hijo de x . Se dice que todos los nodos descendientes directos de otro son

hermanos y aquellos que no tiene ramificaciones o hijos se les conoce como nodos terminal u hojas.

Un nodo tiene un grado relacionado directamente de la suma de sus descendientes directos, así mismo el grado del árbol completo es igual al máximo de todos los nodos.

El nivel se define como el número de arcos que deben ser recorridos para llegar a un determinado nodo, y la altura del árbol es el máximo número de niveles de todos los nodos Figura 2.2.1

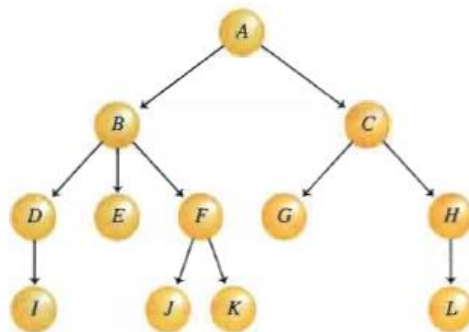


Figura 2.2.1 Representación gráfica de un árbol.

Fuente: (Cairó & Guardati, 2010).

Existen tres tipos de recorridos de árboles 1) Pre-orden (raíz, izquierdo, derecho) visita primero raíz, atraviesa sub-árbol izquierdo y después atraviesa sub-árbol derecho, 2) In-orden (izquierdo, raíz, derecho) atraviesa sub-árbol izquierdo, visita raíz y después atraviesa sub-árbol derecho, y por último 3) Post-orden (izquierdo, derecho, raíz) atraviesa sub-árbol izquierdo, atraviesa sub-árbol derecho y después visita la raíz (Cairó & Guardati, 2010).

2.3 Estado del Arte

A través del tiempo se han propuesto y desarrollado distintos algoritmos para la segmentación de células (Elter, Daum, & Wittenberg, 2006), entre éstos, se puede hacer mención de:

2.3.1 Redes neuronales

Este método hace uso de redes neuronales para la segmentación de células y se descompone en dos grandes módulos con dos propósitos diferentes. Antes de empezar a segmentar células es necesario hacer un previo entrenamiento de la red neuronal. El experto define manualmente el conjunto de entrenamiento el cual son células en parches de 15x15 pixeles. Una vez entrenada

la red neuronal LLM (mapa local linear) es capaz de detectar el núcleo de cada una de las células colocando un punto marcador en el centro de ellas. El segundo módulo se le llama modelo de capa competitivo (CML) y se basa en la obtención de bordes a partir de los marcadores generados por el LLM. El CML toma los vecinos de 20x20 píxeles entorno al marcador asegurándose así que el total de la célula sea extraída, para después delimitar los contornos con operadores morfológicos el proceso completo se representa en la Figura 2.3.1.1

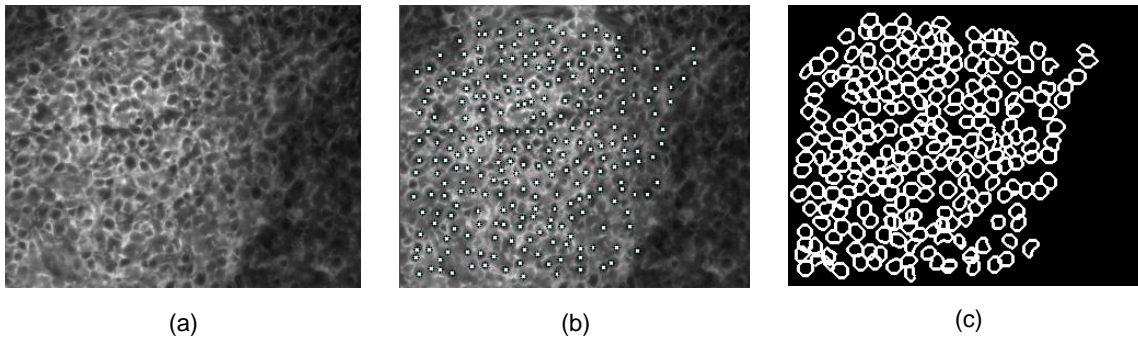


Figura 2.3.1.1 a) Imagen original, b) obtención de marcadores con LLM y c) obtención de contornos con CML.

Fuente: (Nattkemper, Wersing, Schubert, & Ritter, 2002).

No obstante, las redes neuronales al ser un método que necesita una previa enseñanza o capacitación para detectar las células limita la robustez del mismo y sus posibles aplicaciones (Nattkemper et al., 2002).

2.3.2 Fuzzy c-mean

Se basa en el algoritmo fuzzy c-mean para la segmentación de células utilizando 3 clases o clusters como los denomina el algoritmo, spot o célula, el fondo y la sombra que genera la fluorescencia. Este método divide la imagen de entrada en 4 cuadrantes acelerando el proceso, y generando un umbral óptimo diferente para cada uno de los cuadrantes y así obtener mejores resultados (Pham, Crane, Tran, & Nguyen, 2004).

2.3.3 Umbral bimodal

Existen 4 tipos de células de leucemia en la sangre y las características para reconocer el tipo están localizadas en el citoplasma o en el núcleo. Por esto, es apropiado extraer cada una de las partes de la célula por separado. El citoplasma y el núcleo son en términos de intensidad de gris son muy parecidos casi uniformes. Este método realiza la segmentación de células de sangre con leucemia, en una primera etapa del método se aplican dos tipos de umbrales. El primer umbral

es escogido mediante el promedio de la distribución de píxeles entre las intensidades de 65% a 75% descartando los de intensidad 0 para separar el núcleo del citoplasma. El segundo umbral se obtiene con la intensidad del píxel con mayor valor entre el 75% y el 98% de la distribución para separar el citoplasma del fondo véase Figura 2.3.3.1

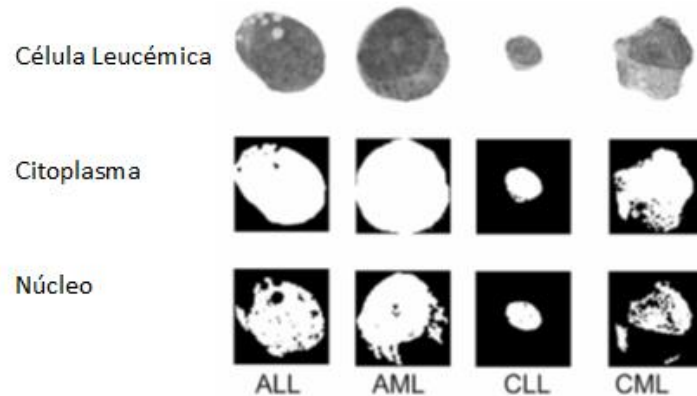


Figura 2.3.3.1 Separación entre núcleo, citoplasma y fondo.

Fuente: (Khashman & Al-zgoul, 2010).

Una vez separado el núcleo del citoplasma se aplica una dilatación para hacer un puente entre las células y píxeles aislados. Después se realiza un llenado de regiones iterándola hasta que no haya cambio. Después es necesario filtrar y eliminar partículas que no son requeridas o no son parte del citoplasma o núcleo, esto se logra obteniendo los elementos conectados con mayor área. Por último, se realiza una restauración imponiendo el núcleo sobre el citoplasma y rellenando lo demás con ceros (Khashman & Al-zgoul, 2010), como lo muestra la Figura 2.3.3.2.

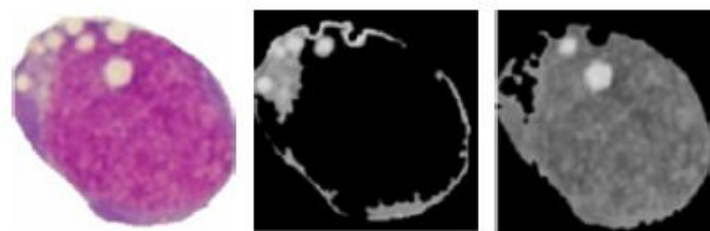


Figura 2.3.2.2 Proceso de restauración de imagen en citoplasma y núcleo.

Fuente: (Khashman & Al-zgoul, 2010).

2.3.4 Patrones de membrana

En esta parte se propuso una técnica para segmentar distintos tipos de células en base a diferentes patrones de membranas de las mismas, no es un método estrictamente relacionado con un solo tipo de célula, ya que su fuerte es la previa enseñanza de patrones del tipo de célula a segmentar. El proceso de segmentado comienza con la configuración manual o automática supervisada de la segmentación de células, para esto se obtienen el radio de la célula más pequeña y la más grande, se determina el largo de la célula más grande, se estima el patrón de la membrana mediante el promedio de sus intensidades y por último se etiqueta cada célula véase Figura 2.3.4.1

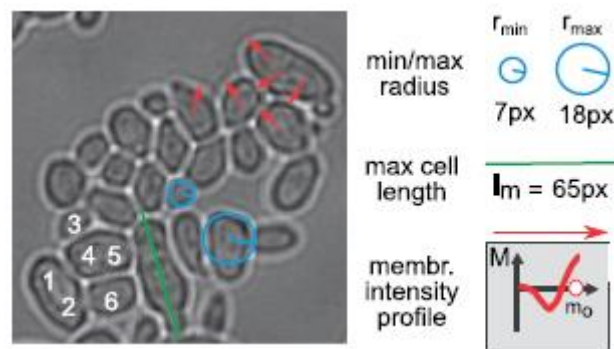


Figura 2.3.3.1 Reconocimiento inicial de patrones en membranas.

Fuente: (Dimopoulos, Mayer, Rudolf, & Stelling, 2014).

Una vez la enseñanza previa haya acabado se inicia colocando las semillas o marcadores iniciales al centro de la célula y se calcula un radio de la misma, este deberá estar entre los valores de radio mínimo y el radio mayor mencionados en el párrafo anterior. Después, se analiza cada una de las células donde se plantaron las semillas y se comparan con los patrones obtenidos en la primer etapa, si son similares se procede a realizar cortes de grafos para separar la célula de posibles solapamientos con otras células y del fondo en sí como lo muestra la figura 2.3.4.2 (Dimopoulos et al., 2014) .



Figura 2.3.4.2 Proceso de segmentación mediante cortes en grafos y delimitación en bordes.

Fuente: (Dimopoulos et al., 2014).

2.4 Antecedentes

2.4.1 Umbral Adaptativo Local

Particularmente las imágenes con las que se trabajó en esta investigación tienen un fondo heterogéneo debido a la luminosidad que despiden la fluorescencia en las células, existe un degradado importante en la imagen y por esta razón un umbral global no daría un buen resultado. En función con la luminosidad de un fondo cambiante y bajo contraste a causa de la fluorescencia, se necesita generar un umbral apropiado en base a las propiedades estadísticas de cada localidad en la imagen. El umbral global se puede adaptar para trabajar en píxeles por separado, es decir en dependencia de una localidad por píxel obtener un umbral diferente para cada uno de ellos. El método tradicional de umbral local determina el valor de este umbral de cada píxel examinando una ventana con forma y tamaño fijo, los más comunes están entre un vecindario de 4 a 8 píxeles. El elemento estructural de forma y tamaño fijo recorre la imagen de manera que se aplica un umbral específico para cada región. Pero como se explicó anteriormente dadas las condiciones de las imágenes, hay casos en los que regiones como el fondo contienen porciones de objetos de interés o que un objeto contenga algo del fondo (Peng & Hsu, 2009). Debido a esto, sería más apropiado dejar que el tamaño de la ventana cambie dinámicamente acorde a las características estadísticas de cada píxel y su vecindario. Este vecindario debe ser suficientemente grande como para contener estructuras locales relevantes, pero no tanto como para contener porciones de fondo como lo muestra la Figura 2.4.1.1

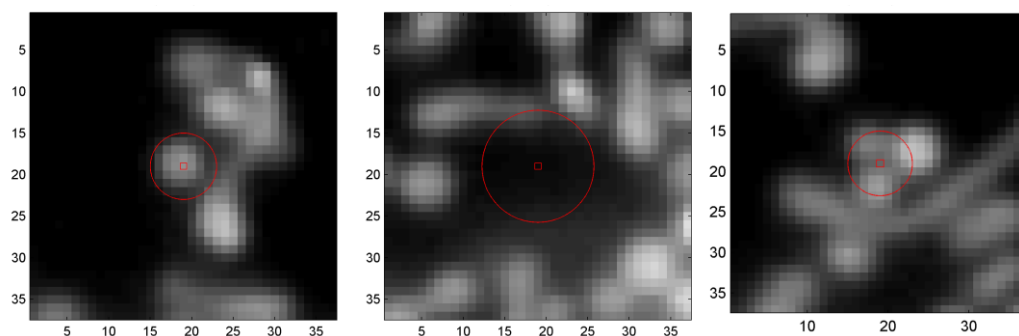


Figura 2.4.1.1 Localidad ideal y dinámica con radio diferente las 3 imágenes.

Fuente: (Peng & Hsu, 2009).

Si el vecindario se deja con un tamaño específico es posible que objetos espurios aparezcan en la imagen, ya que al ser fijo puede que el cálculo lo haga con píxeles del fondo y objetos a la vez siendo esto incorrecto porque no contienen las mismas características, véase Figura 2.4.1.2.

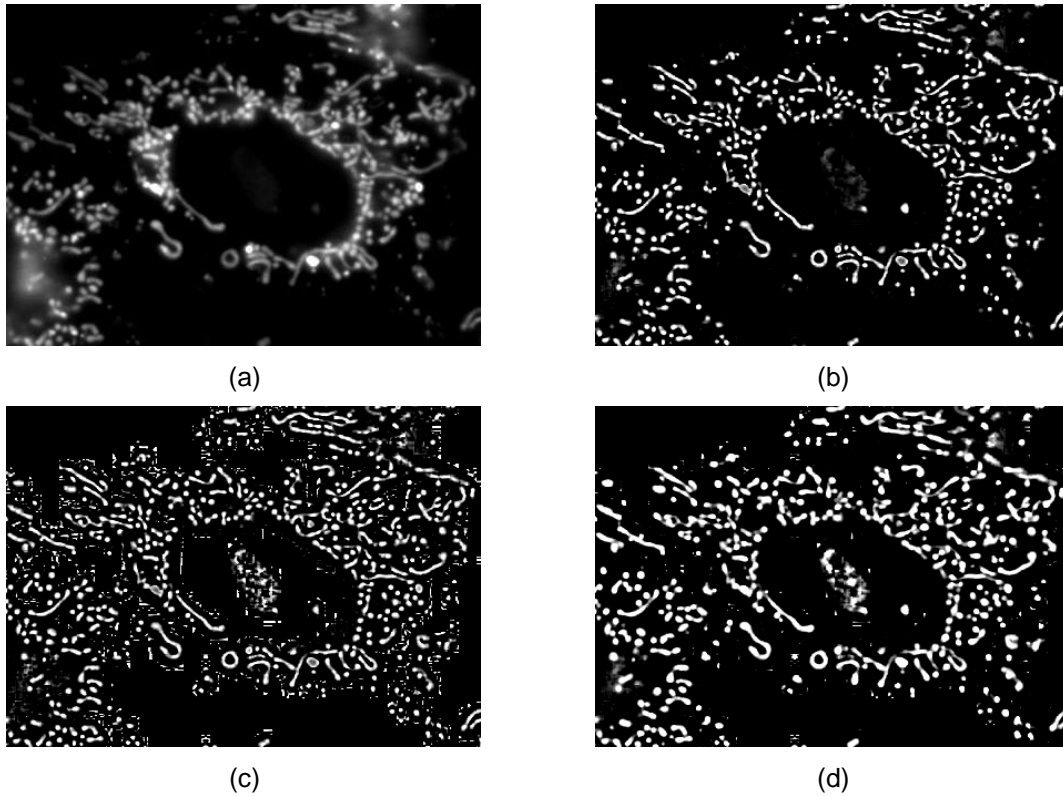


Figura 2.4.1.2 a) Imagen original, b) Radio Adaptativo $T = 0.35 \text{ StD}$, c) $R = 4$ píxeles y d) $R = 8$ píxeles.

Fuente: (Peng & Hsu, 2009).

2.4.2 Watershed

Watershed o línea divisoria de agua es uno de los algoritmos más utilizados en la actualidad basado en morfología matemática, permitiendo extraer fronteras de las regiones que hay en una imagen. Esta técnica clasifica píxeles según su proximidad espacial, el gradiente de sus niveles de gris, y la homogeneidad de sus texturas. La técnica utiliza la analogía de una superficie topográfica donde la altura del relieve está dada por el nivel de gris de la imagen es decir que para niveles más cercanos a 255 pueden verse una especie de “montañas” y los cercanos a ceros como “cuencas de inundación” como lo muestra la Figura 2.4.2.1.

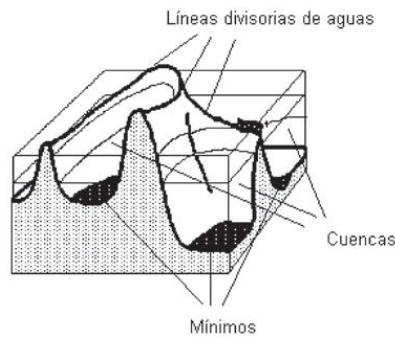
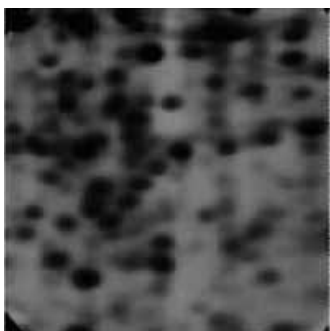


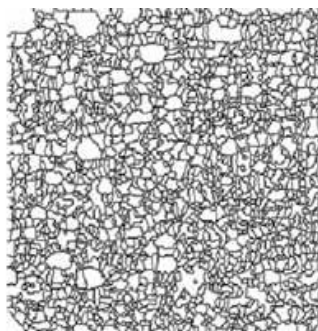
Figura 2.4.2.1 Representación del algoritmo Watershed como relieve terrestre.

Fuente: (La & Palomino, 2010).

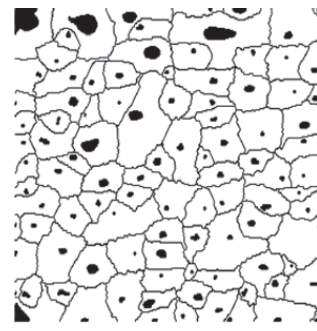
El algoritmo Watershed es ampliamente recomendado para imágenes médicas. Hace uso de las llamadas FIFO Jerárquicas (First in First out). El proceso comienza con la selección de marcadores o mínimos (regiones con el nivel de intensidad más baja) ya sea manual o automáticamente. Una vez definidos los mínimos, se agregan los contornos de cada uno de los mínimos a la FIFO jerárquica para ser procesados después. Es aquí cuando comienza el proceso de inundación de los valles, se van sacando elementos de la FIFO para empezar a inundar verificando que sus vecinos estén desocupados, si es el caso se etiqueta dicho pixel y se agrega ese vecino a la FIFO, el proceso continúa normal con el siguiente elemento de la lista. Las líneas divisorias se crean cuando dos de los mínimos han crecido de tal manera que chocan y se provoca lo que normalmente se conoce como un dique siendo este la frontera entre un objeto y otro. Sin embargo, a pesar de ser una técnica actualmente muy utilizada tiene deficiencias importantes, como la sobre-segmentación debido a los mínimos locales que se generan cuando se produce algún ruido en la imagen, existiendo la posibilidad de que segmente más partes que las de interés (La & Palomino, 2010) véase Figura 2.4.2.2



(a)



(b)



(c)

Figura 2.4.2.2 a) imagen original, b) sobre-segmentación de imagen debido a mínimos y c)

Imagen segmentada con mínimos seleccionados manualmente.

Fuente: (La & Palomino, 2010).

2.4.3 Max-tree

Max-tree es una representación de la imagen a multi-escala formado a partir de conjuntos de píxeles acomodados por nivel de intensidad, es decir, llevan un orden jerárquico propuesto por Salembier et al. (1998). Aunque el algoritmo es muy efectivo existen procesos de que deben ser mejorados para que su implementación se ejecute con mayor eficiencia. Se deben de construir nodos de forma dinámica e ir generando el enlace entre padres e hijos. Siendo que el árbol se puede construir en tiempo casi lineal, el problema es resuelto mediante los tres siguientes algoritmos de creación recursivos: algoritmos de inmersión (Berger et al., 2007; Najman y Couprie, 2006), algoritmos de inundación (Salembier et al., 1998; Wilkinson, 2011) y algoritmos basados en combinación (Wilkinson et al., 2008; Matas et al., 2008). Existen diversas formas de implementar Max-tree, en la primera implementación se crea un vector llamado parent con el mismo número de píxeles que la imagen en el cual se llevaba el control de los padres de cada píxel. Carlinet y Geraud (2014) argumentan, que solo tener la matriz principal es una representación incompleta, ya que no es suficiente para realizar fácilmente recorridos de árboles. Prefieren representar el Max-tree usando un par de arreglos parent y S, cada uno con la misma cantidad de elementos que la imagen. El arreglo S es un vector ordenador, de modo que muestra el camino de como navegar por sus elementos permitiendo hacer recorridos transversales en el árbol ya sea de raíz a hojas o viceversa como lo muestra la Figura 2.4.3.1. Estas estructuras de datos utilizadas para la representación de un Max-tree orientado a píxeles ya que no es posible acceder a los nodos directamente.

0:6	1:4	2:3	3:3
4:3	5:5	6:2	7:8
8:7	9:7	10:3	11:3

(a)

5	2	6	2	2	1	6	2	5	8	2	2	Parent
6	2	11	10	4	3	7	1	5	8	0	9	S

(b)

Figura 2.4.3.1 a) Imagen original (Id_pixel : nivel de gris) y b) vectores con el mismo número de elementos que la imagen donde uno denota los padres de cada uno de los pixeles y el otro el orden de recorrido del árbol.

Fuente: (Souza, 2015)

Posteriormente Souza et al. (2015) propuso una forma similar a la anterior pero ahora son dos vectores bidimensionales, el primero llamado Node array donde su primera columna indica el nodo padre y la segunda de que intensidad es el hijo. La segunda matriz es Node index de mismas dimensiones que la imagen en la cual denota el número de nodo en el que se encuentra ese pixel véase Figura 2.4.3.2.

Node array		Node index			
0	2	6	3	1	1
0	3	1	4	0	2
1	8	5	5	1	1
1	4				
3	5				
4	7				
4	6				

Figura 2.4.3.2 Vectores bidimensionales para representar la imagen en Maxtree.

Fuente: (Souza, 2015)

Cada una de las implementaciones anteriores dieron resultado, pero normalmente se hace una implementación con arreglos cuando se tiene conocimiento de cuál es el uso exacto de memoria, y por naturaleza del problema una mejor opción es manejar y almacenar memoria dinámicamente. Huang et al. (2003) propuso una forma de representar el árbol máximo usando una lista vinculada y una tabla hash. De esta forma aseguramos que el tamaño del árbol pueda ser expandible y que los nodos son accesibles directamente.

Cada nodo está constituido por varios atributos que almacena al momento de su creación:

1. ID: identificador del nodo.
2. ID_padre
3. Lista de hijos
4. Lista de pixeles

Para el filtrado de una imagen es imprescindible tener el árbol construido sin importar el método que se utilice. Una vez construido el árbol se pueden aplicar filtros para remover nodos del árbol. A este proceso se le conoce comúnmente como podar el árbol (pruning) Figura 2.4.3.3. Los filtros en arboles están basados en atributos que son extraídos de sus nodos como tamaño, contraste y forma. Dichos filtros se consideran anti-extensivos porque la manera de trabajar es contrayendo nodos o eliminándolos.

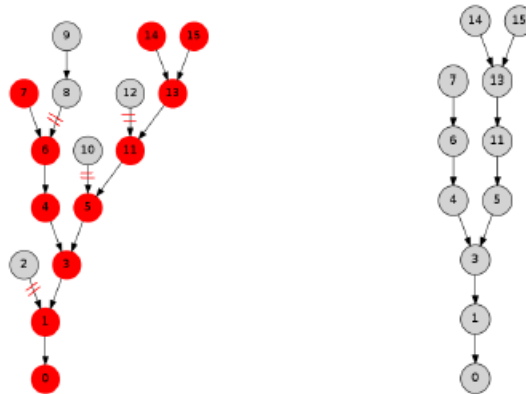


Figura 2.4.3.3 Representación gráfica de aplicar filtros sobre un árbol Max-tree.

Fuente: (Salembier, Oliveras, & Garrido, 1998)

Existen dos reglas en el filtrado de árboles, la regla directa y la sustractiva. La directa consiste en conservar el valor de nivel de gris en los nodos descendientes de que fue eliminado, mientras que la regla sustractiva decrementa el nivel de gris de los nodos descendientes conservando el contraste en la imagen. La ventaja de la aplicación de filtros en Max-tree es que no es necesario el recorrer la imagen completa para ir aplicando dicha condición a cada pixel como comúnmente se conoce. Como la imagen ya está dividida en regiones planas (nodos) solo es cuestión de contraer nodos que no cumplen con cierto criterio como lo es área, forma, volumen o peso del nodo.

Una vez aplicados los filtros al árbol y poder visualizar el resultado es necesario restituir la imagen. La restitución de la imagen consta en recorrer todos los nodos del árbol o incluso si solo se desea recuperar una parte de la imagen solo reconstruir algunos nodos he ir plasmando los pixeles que almacena cada uno de ellos. Al recorrer el árbol completo o solo ciertos nodos muestra la imagen de los pixeles que se reconstruyeron y el resultado del filtrado donde muchas de las ocasiones es una imagen con mejor contraste, con menos ruido, sin objetos no deseados, etc.

3. METODOLOGÍA

Se procesaron distintas secuencias de imágenes en las que se percibe un cambio en la fluorescencia a través del tiempo. Cada secuencia contiene entre 800 y 1000 imágenes aproximadamente, en las cuales la vista humana no distingue cambios significativos en el nivel de fluorescencia entre una y otra. Sin embargo, si existe un cambio, se hará uso de la visión por computadora para segmentarlas y después calcular el nivel de calcio intracelular como lo muestra el diagrama de flujo en la Figura 3.1

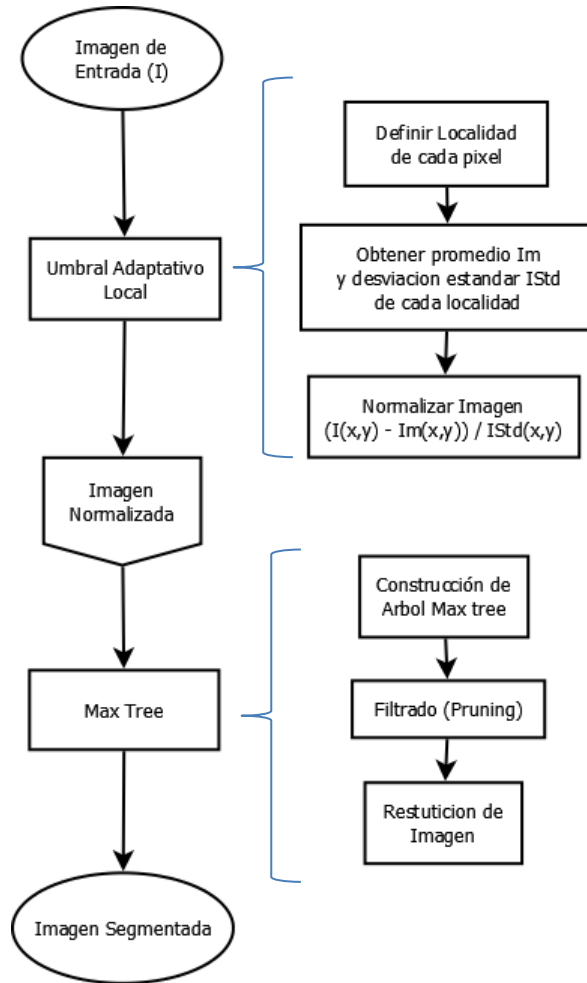


Figura 3.1 Diagrama de flujo de la metodología a seguir para segmentar las células.

Fuente: Elaboración propia.

3.1 Normalización de imagen

Dentro de las 800 imágenes que se tienen en la secuencia se seleccionó una para ser segmentada y a partir de la locación espacial de las células obtenida se procesa el resto. Esta imagen seleccionada es la imagen de entrada. Se comenzó el proceso de normalización y eliminación de ruido calculando el 0.3 de la desviación estándar de la imagen completa T_{Std} . Se definió un radio mínimo y uno máximo, 3 y 15 píxeles respectivamente con el fin de reducir el tiempo de procesamiento. Se inicializó el proceso de definición de vecindario haciendo crecer un círculo con origen en el punto $I_{(x,y)}$ calculado en cada iteración la desviación estándar de los píxeles dentro del círculo con radio n y limite hasta el radio máximo o en su caso que la desviación estándar a un determinado radio entre el mínimo y el máximo posible sea mayor o igual que T_{Std} cualquiera que pase primero (3).

$$r(x, y) = \min\{R_{max}, \min\{r > 0 \mid StD[R_{r(x,y)}(x, y)] \geq T_{Std}\}\} \quad (3)$$

donde $r = (x, y)$ es el radio para el pixel en x, y . Después de establecer ese vecindario para cada uno de los píxeles se calculó el promedio I_M y la desviación estándar I_{Std} de cada localidad. En caso de que la localidad hubiera sobrepasado el límite de radio máximo la desviación estándar de esa localidad será igual a T_{Std} .

$$I_M = mean[R_{r(x,y)}(x, y)] \quad (4)$$

$$I_{Std} = \max\{T_{Std}, StD[R_{r(x,y)}(x, y)]\} \quad (5)$$

Una vez calculados estos dos valores para cada pixel en particular se normalizó cada uno de ellos I_N con (6).

$$I_N = \frac{I(x, y) - I_M(x, y)}{I_{Std}(x, y)} \quad (6)$$

Se tienen dos imágenes, la imagen original (I_{mo}) y la imagen normalizada (I_{mn}). I_{mn} es una imagen marcador del mismo tamaño que I_{mo} con valores de 0 en todos sus píxeles en la cual se irá actualizando la intensidad de cada pixel una vez normalizado. Ya calculado el valor normalizado del pixel se aplicó un umbral, si I_N es negativo significa que el promedio de su vecindario fue mayor que la intensidad del pixel y por consecuente asumimos que el pixel es parte fondo, de manera contraria si es mayor que 0 indica que la localidad contiene suficiente

información para ser parte de la célula, actualizando la intensidad del pixel en I_{mn} con el valor de esa misma posición en I_{mo} .

3.2 Construcción de Max-tree

Esta imagen normalizada será la entrada de algoritmo de Max-tree. El árbol se creó mediante el método recursivo de inundación (flooding). Cada nodo que conformará el árbol contiene tres estructuras tipo Nodo padre, anterior y siguiente, una bandera para saber si es nodo hoja, el nivel de intensidad de sus pixeles, una etiqueta para identificar a que objeto pertenece y la lista de pixeles que almacena. Se manejaron objetos derivados de la clase Pixel constituida por cuatro atributos que son "x", "y", "valor" (nivel de gris) y el "status". Para la construcción del árbol se utilizaron las siguientes estructuras de datos:

1. FIFO Jerárquica de Pixeles (LP): Lista de tamaño dinámico para llevar control del proceso de inundación.
2. Arreglo de Pixeles (AP): Arreglo con tamaño igual al de los elementos de la imagen guardado un objeto de tipo Pixel.
3. FIFO Jerárquica de Nodos (sort): Lista de tamaño dinámico para el control de recorridos transversales en el árbol.
4. Arreglo de Nodos (AN): Arreglo con tamaño de 256 para llevar un control de nodos creados e ir generando las conexiones entre padres e hijos.
5. Un vector de enteros donde se almacenaran las etiquetas de los objetos encontrados junto con sus áreas.

El algoritmo inicia creando un objeto Pixel por cada uno de los pixeles en la imagen con sus atributos espaciales (x, y), la intensidad y el status por default que es "no analizado". El principio de creación consta de dos procesos en un mismo método. El primer proceso se encarga de recorrer la imagen desde el pixel de menor intensidad (flood[pixel->valor]) etiquetando los pixeles entre "analizado", "no analizado" y "en LP". Se agrega el primer pixel a LP y partir de ese punto empezar la inundación. Inmediatamente el primer pixel es retirado de la lista y se recorren los vecinos, si ese pixel no ha sido analizado se va agregando a LP[pixel->valor], se actualiza su status a "en LP" y se verifica si actualmente existe un nodo en AN[pixel->valor], si este es el caso se agrega el pixel a la lista de pixeles del nodo y si no, se crea el nodo en esa posición y se agrega de igual manera el pixel a la lista de ese mismo nodo. Si alguno de los pixeles del vecindario es

de una intensidad mayor entonces se asume que se ha encontrado un hijo. Es aquí cuando se empieza la recursividad llamando de nuevo el método flood pero con el nuevo valor de intensidad. Esto se ira repitiendo hasta que alguno de los vecinos tenga un valor igual al de intensidad anterior, de esta manera se va recorriendo la imagen. El segundo proceso del método consiste en encontrar los padres de cada hijo. Cada vez que se termina de recorrer por completo una zona plana con nivel h de intensidad de gris se agrega a la lista jerárquica $sort[h]$ y es momento de buscar un padre para ese nodo. Para esto en el arreglo de nodos si se tiene un nodo completo en $AP[h]$ se busca el primer nodo declarado en el arreglo con $AP[0 \leq m < h]$ y es aquí cuando al nodo $AP[h]$ en su apuntador al padre se actualiza a $AP[m]$. El proceso de construcción termina una vez que la imagen es recorrida por completo y todos los pixeles pertenecen a algún nodo dentro de la lista sort.

3.3 Simplificación y filtrado del árbol Max-tree

Se asume que cada una de las células debido a su naturaleza están representadas en diferentes niveles de intensidad de gris, por lo tanto, ninguna de ellas estará contenida en su totalidad en un solo nodo. Se aplicó un filtro de contracción de nodos mediante regla directa eliminándolos si y solo si el nodo es hijo de la raíz y hoja. Esto implica que los pixeles removidos son movidos al padre de cada nodo. En este caso será la raíz.

Después se aplicó un umbral directamente al árbol donde los nodos con una intensidad menor o igual a 15 serán eliminados, es decir, los pixeles que contiene cada nodo menor a esta intensidad se agregaran al nodo raíz. Como los nodos están ordenados estrictamente por intensidad, el filtro no se aplica a toda la imagen, solo es cuestión de contraer los nodos que cumplan con la condición.

Así como existen partículas muy pequeñas sueltas en la imagen creando un nodo inservible por cada una de ellas, también puede haber objetos grandes almacenados en un solo nodo. Se implementó un filtro de área a cada uno de los nodos que constituyen el árbol a excepción del nodo raíz. Un área mayor a 100 pixeles es definida por nodo, de otra manera si el nodo "x" contiene más, se puede catalogar como ruido en la imagen y este es contraído directamente al nodo raíz sin importar su nivel de intensidad.

El proceso de construcción del árbol va creando una rama por cada uno de los objetos encontrados en la imagen, entonces es posible etiquetar cada nodo con un valor agrupando los

nodos de cada rama con el mismo valor en su atributo etiqueta. Se recorrieron todas las hojas del árbol y por cada una de ellas se inicia un camino hasta la raíz. Si algún camino de las hojas se une con otro se considera que esa subrama es parte de un mismo objeto y se etiqueta con el mismo valor que tienen los nodos del camino con el que se cruzó. De esta manera creamos conjuntos de nodos unidos por una etiqueta en común conformando los objetos de la imagen.

Aunque ya se tiene una separación dentro del árbol de los objetos, los nodos de cada rama en árbol pueden o no almacenar tanto los valores espaciales y de intensidad de una sola célula o de varias de ellas solapadas. Por lo tanto, es conveniente aplicar otro filtro por área, pero ahora a cada rama del árbol. Las células tienen un área en promedio entre 100 y 250 píxeles. Estos serán los parámetros de umbral menor y mayor respectivamente catalogando las ramas con menor área al umbral menor como ruido en la imagen y a las de mayor área que el umbral como células solapadas. Si se tratan de varias células solapadas el proceso es trabajar solo con esta región de la imagen creando un árbol nuevo por cada conjunto de células solapadas. A la imagen con el conjunto de células solapadas se le aplicó una dilatación con el elemento estructural básico 5x5 para cerrar bien los citoplasmas y tener un núcleo bien definido. Se invierte la imagen de tal manera que cada núcleo de las células y el fondo queden almacenados en las hojas de este nuevo árbol. Sin embargo, en ciertas ocasiones algunos núcleos pueden quedar divididos en dos generando la separación de una simple célula en dos. Por esto, se volvió a aplicar un dilatado sobre los núcleos para juntarlos. El criterio para separar las células solapadas está dado por la distancia menor entre todos los píxeles de la rama y cada uno de los píxeles de los núcleos obtenidos en el paso anterior. Se dice que si la distancia entre el píxel "p" y todos los píxeles del núcleo " N_1 " fue menor que la distancia entre "p" y " N_2 " hasta " N_n " entonces el píxel "p" pertenece a " N_1 ".

Para terminar cada célula es descompuesta en Nodos según el nivel de intensidad y creando conexiones desde el nivel más alto con el primer nodo con un nivel inferior a este, este proceso se repite hasta ligar el último nodo con la raíz.

3.4 Restitución de Imagen

La restitución se hace mediante el recorrido transversal del árbol Max-tree desde la raíz hasta las hojas o viceversa. Para estos recorridos se hará uso del arreglo sort. Se puede recorrer de manera secuencial y debido a su estructura jerárquica se puede acceder a reconstruir cualquier

nivel además de tener acceso directo a cualquier nodo del árbol. Se toma el primer nodo de cada nivel en la lista sort, se obtiene el primer pixel de la lista de pixeles de dicho nodo y se va construyendo la imagen según los atributos del pixel, después se toma el siguiente pixel de esta lista y así hasta terminar de recorrerla por completa y pasar al siguiente nodo, una vez terminado de recorrer sort en cierto nivel se pasa al siguiente realizando los mismos pasos.

3.5 Conteo de calcio intracelular

Para terminar, como el nivel del calcio que tiene cada célula el cual está directamente correlacionado con el nivel de intensidad de gris en la imagen. Entre más alto sea el nivel de gris más alto será el nivel de calcio en la célula, por lo tanto, el nivel total de calcio en la célula esta dado por la sumatoria de las intensidades de cada una de las células (7). Se hicieron recorridos completos del árbol reconstruyendo cada una de las etiquetas y se sumaron los niveles de intensidad de los pixeles almacenados en cada uno de los nodos de las ramas con posiciones en pixel->x, pixel->y.

$$S = \sum_{x_i}^{x_f} \sum_{y_i}^{y_f} f(x, y) \Delta x \Delta y \quad (7)$$

4. RESULTADOS Y DISCUSIÓN

Se aplicó el algoritmo a siete distintas imágenes de células foliculares fluorescentes obtenidas del Instituto de Neurobiología de la UNAM campus Juriquilla con tamaño de 332x256 píxeles donde se normalizó, segmentó y calculó el calcio de intracelular de cada célula encontrada. El proceso de normalización de imagen del algoritmo propuesto logra eliminar de manera propicia mucha de la luz que provoca la fluorescencia de las células agregando un mejor contraste entre ellas y el fondo como lo muestra la Figura 4.1.

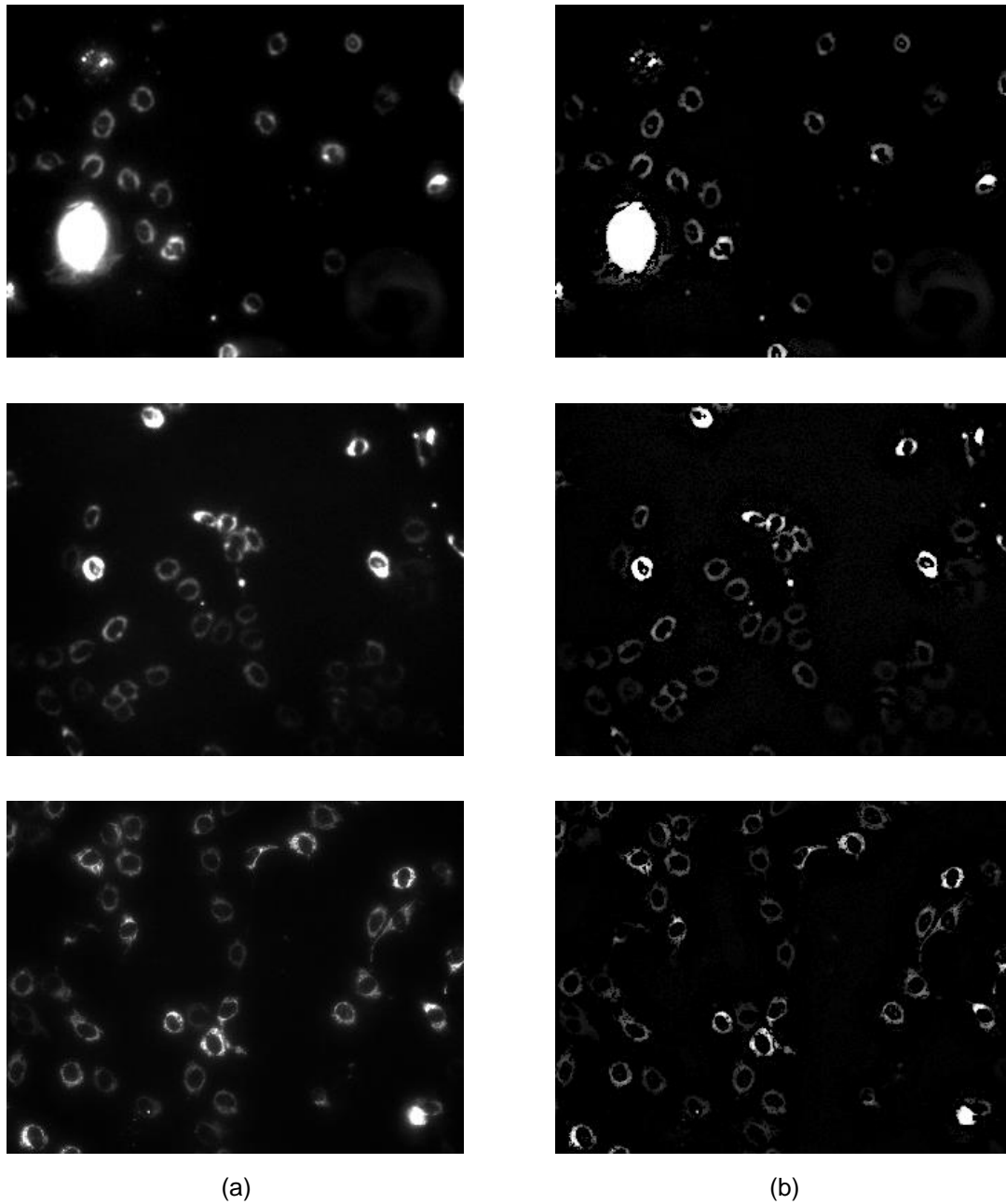


Figura 4.1 (a) Imágenes originales, (b) imágenes después de aplicar el Algoritmo Umbral Adaptativo Local.

Fuente: Elaboración propia.

La normalización fue un proceso importante para obtener una buena segmentación, una vez aplicado, el contorno de las células queda delimitado con mayor precisión quitando la difuminación sobre el fondo causada por la fluorescencia del citoplasma.

La segmentación de células tuvo resultados favorables, logrando separarlas del fondo y en caso de las células solapadas dividiendo el conjunto completo en sus componentes individuales Figura 4.2. Al final se despliega el área de las células y la suma de las intensidades de cada una de ellas siendo este último el calcio intracelular.

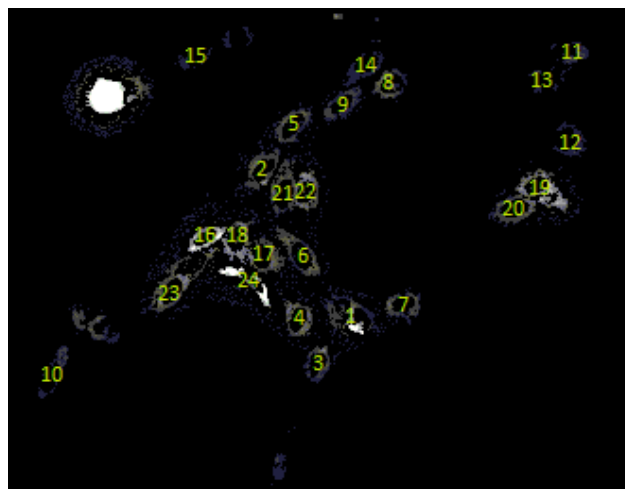


Figura 4.2 Etiquetamiento de células en imagen normalizada.

Elaboración propia.

En la Tabla 1 se muestran los resultados obtenidos de cada una de las células en la Figura 4.2.

Tabla 4.1 Resultados obtenidos de la imagen original 1a donde se muestra el área y el calcio de cada una de las células encontradas.

Célula	Área	Calcio Intracelular
1	226	17805
2	145	9299
3	108	5496
4	131	9964
5	144	7882

6	169	10789
7	123	6928
8	122	7129
9	141	6336
10	149	4676
11	105	3641
12	154	4205
13	143	3157
14	123	3774
15	147	3142
16	87	156345
17	148	78729
18	134	113489
19	213	214172
20	143	42248
21	143	36337
22	114	70970
23	280	113521
24	230	102253

La normalización de la imagen, aunque no es un proceso como tal de la segmentación por medio del algoritmo Max-tree es un método que elimina ruido, agrega contraste a la imagen y limita mejor los contornos. Esto disminuye la posibilidad de una sobre-segmentación o de procesar pixeles que no son propiamente parte de alguna célula dando como resultado una mejor segmentación.

Trabajar con regiones en la imagen disminuye el problema ya que hace posible el modificar solo ciertas zonas de interés sin afectar la imagen por completo y cada zona provee más información que trabajar los pixeles por separado. Además, desde un principio al dividir la imagen en regiones se tiene una previa segmentación y solo hace falta catalogar en conjuntos las regiones que conformen las células.

El proceso fue completamente autónomo desde su inicio, el algoritmo obtuvo las células en la imagen de manera automática, creo el árbol en base a la misma imagen, clasifíco los nodos agrupándolos en ramas y separo células solapadas en base a la distancia de los pixeles a los núcleos encontrados.

5. REFERENCIAS

- Arce, S. H., Wu, P. H., & Tseng, Y. (2013). Fast and accurate automated cell boundary determination for fluorescence microscopy. *Scientific Reports*, 3. <https://doi.org/10.1038/srep02266>
- Berridge, M. J., Bootman, M. D., & Roderick, H. L. (2003). Calcium signalling: dynamics, homeostasis and remodelling. *Nature Reviews. Molecular Cell Biology*, 4(7), 517–29. <https://doi.org/10.1038/nrm1155>
- Cairó, O., & Guardati, S. (2010). Estructuras de datos (3era ed., pp. 177–183, 277–282). Mc Graw Hill.
- Dima, A. A., Elliott, J. T., Filliben, J. J., Halter, M., Peskin, A., Bernal, J., ... Plant, A. L. (2011). Comparison of segmentation algorithms for fluorescence microscopy images of cells. *Cytometry Part A*, 79 A(7), 545–559. <https://doi.org/10.1002/cyto.a.21079>
- Dimopoulos, S., Mayer, C. E., Rudolf, F., & Stelling, J. (2014). Accurate cell segmentation in microscopy images using membrane patterns. *Bioinformatics*, 30(18), 2644–2651. <https://doi.org/10.1093/bioinformatics/btu302>
- Doadrio, L. A. (n.d.). Las redes de señales intracelulares del calcio. <https://doi.org/http://dx.doi.org/ES/monoranf.v0i0.810>
- Elter, M., Daum, V., & Wittenberg, T. (2006). Maximum-Intensity-Linking for Segmentation of Fluorescence-Stained Cells, 1–4.
- Khashman, A., & Al-zgoul, E. (2010). Image Segmentation of Blood Cells in Leukemia Patients. *Recent Advances in Computer Engineering and Applications*, 104–109.
- La, N., & Palomino, S. (2010). Watershed : un algoritmo eficiente y flexible para segmentación de imágenes de geles 2-DE, 7(2), 35–41.
- Nattkemper, T. W., Wersing, H., Schubert, W., & Ritter, H. (2002). A neural network architecture for automatic segmentation of fluorescence micrographs. *Neurocomputing*, 48, 357–367. [https://doi.org/10.1016/S0925-2312\(01\)00642-7](https://doi.org/10.1016/S0925-2312(01)00642-7)
- Peng, J., & Hsu, C. (2009). Adaptive Local Thresholding for Fluorescence Cell Micrographs,

(Technical Report No. TR-IIS-09-008).

- Pham, T. D., Crane, D. I., Tran, T. H., & Nguyen, T. H. (2004). Extraction of fluorescent cell puncta by adaptive fuzzy segmentation. *Bioinformatics*, 20(14), 2189–2196. <https://doi.org/10.1093/bioinformatics/bth213>
- Salembier, P., Oliveras, A., & Garrido, L. (1998). Antiextensive connected operators for image and sequence processing. *IEEE Transactions on Image Processing*, 7(4), 555–570. <https://doi.org/10.1109/83.663500>
- Souza, R. (2015). An overview of max-tree principles , algorithms and applications.
- Stricker, S. A., Centonze, V. E., Paddock, S. W., & Schatten, G. (1992). Confocal microscopy of fertilization-induced calcium dynamics in sea urchin eggs. *Developmental Biology*, 149(2), 370–380.
- Sucar, L. E., & Gomez, G. (2002). Procesamiento de imágenes y visión computacional. *Spanish*.
- Ujihara, Y., Nakamura, M., Miyazaki, H., & Wada, S. (2013). Segmentation and Morphometric Analysis of Cells from Fluorescence Microscopy Images of Cytoskeletons. ... *Mathematical Methods in ...*, 1111(ex 27508), 1–44. <https://doi.org/10.1155/2013/381356>
- Zhang, W., & Li, H. (2017). Automated segmentation of overlapped nuclei using concave point detection and segment grouping. *Pattern Recognition*, 71, 349–360. <https://doi.org/10.1016/j.patcog.2017.06.021>

6. APÉNDICE

En esta sección se presentan los experimentos con otras imágenes de células foliculares mostrando la imagen original seguido de la imagen normalizada y una etiqueta en las células encontradas.

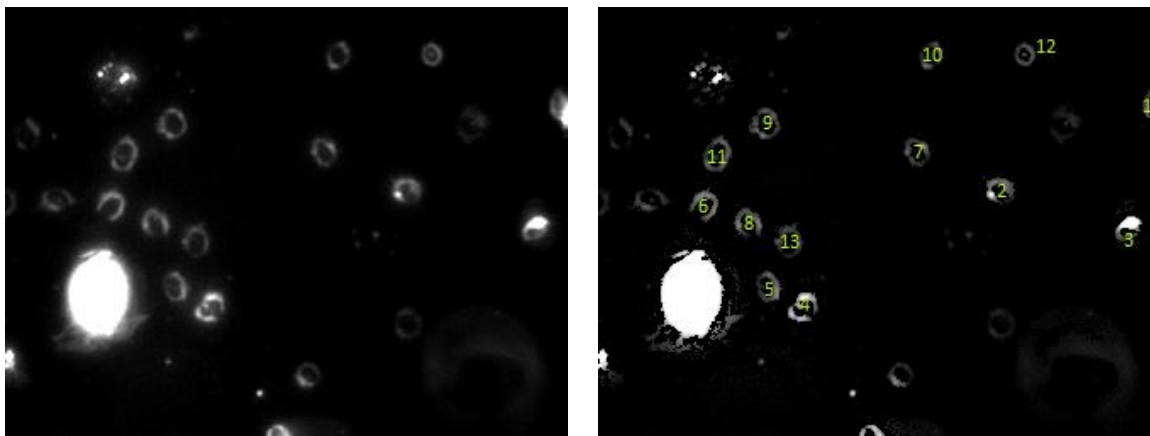


Figura 6.1 Imagen original e imagen normalizada con etiquetas en las células encontradas.

Tabla 6.1 Área y calcio intracelular de la imagen etiquetada en la Figura 6.1.

Célula	Área	Calcio Intracelular
1	134	17965
2	151	13594
3	133	22024
4	146	20503
5	141	9998
6	136	12225
7	139	8035
8	150	11010
9	161	10020
10	125	5837
11	172	10980
12	106	6279
13	167	7616

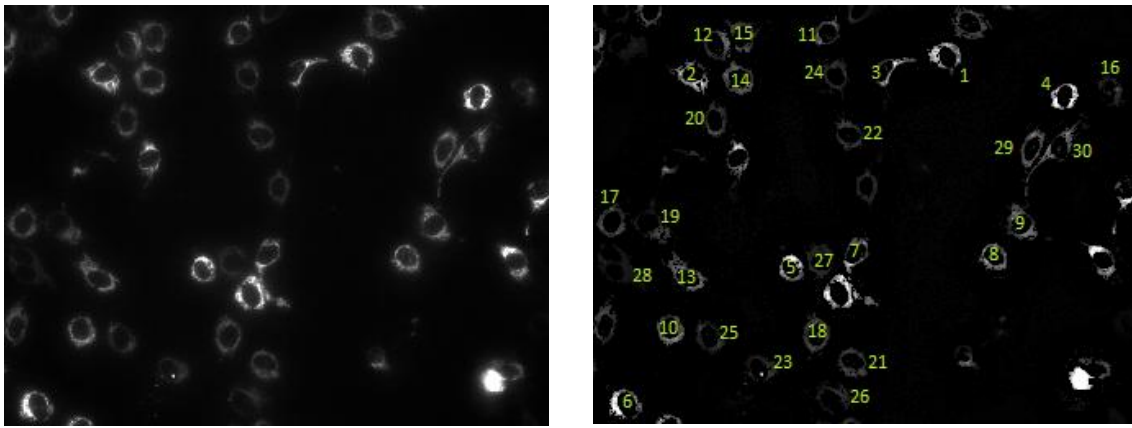


Figura 6.2 Imagen original e imagen normalizada con etiquetas en las células encontradas.

Tabla 6.2 Área y calcio intracelular de la imagen etiquetada en la Figura 6.2.

Célula	Área	Calcio Intracelular
1	147	17967
2	141	14674
3	108	9192
4	111	18193
5	108	13607
6	135	18179
7	111	8907
8	121	10202
9	175	11269
10	135	10010
11	111	6008
12	112	6560
13	167	11199
14	145	9300
15	130	6232
16	109	3383
17	144	6983
18	164	7659
19	143	5380
20	136	5568
21	150	6809
22	132	5153
23	129	3972
24	143	4624

25	146	4624
26	148	4123
27	136	4466
28	133	2889
29	119	35870
30	166	60012

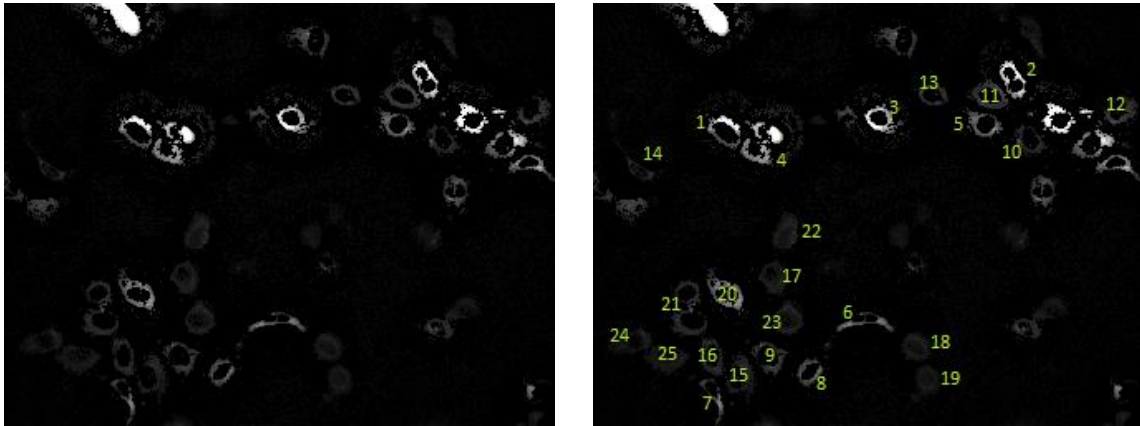


Figura 6.3 Imagen original e imagen normalizada con etiquetas en las células encontradas.

Tabla 6.3 Área y calcio intracelular de la imagen etiquetada en la Figura 6.3.

Célula	Área	Calcio Intracelular
1	142	19480
2	132	22495
3	146	22621
4	123	15009
5	137	10824
6	127	9094
7	98	5703
8	116	7191
9	148	6293
10	131	4968
11	163	8225
12	147	6327
13	114	4540
14	103	2589
15	208	5693
16	163	5160

17	236	5452
18	220	4767
19	166	3362
20	205	110309
21	256	28417
22	279	20678
23	259	15360
24	260	12892
25	240	12757

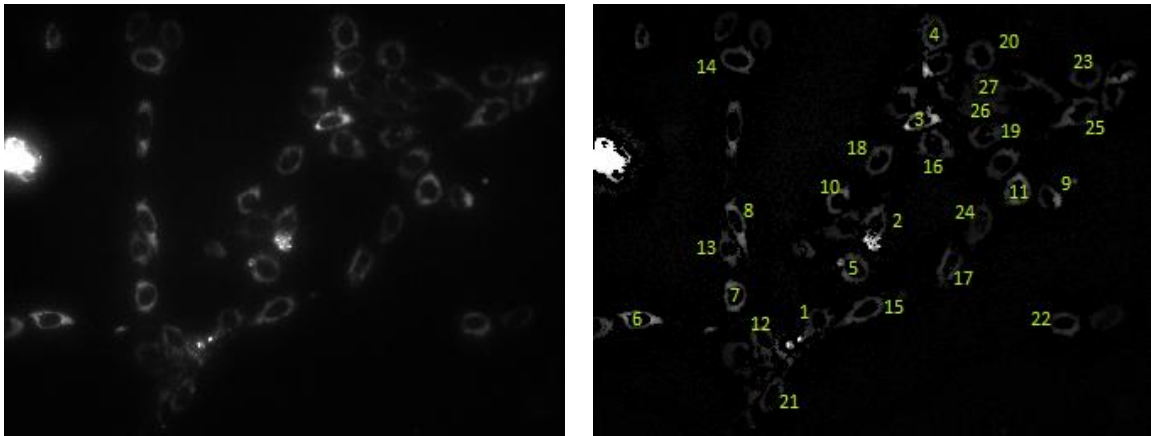


Figura 6.4 Imagen original e imagen normalizada con etiquetas en las células encontradas.

Tabla 6.4 Área y calcio intracelular de la imagen etiquetada en la Figura 6.4.

Célula	Área	Calcio Intracelular
1	146	5728
2	176	15389
3	121	11469
4	242	11224
5	142	7014
6	113	8482
7	120	6909
8	149	8917
9	99	4466
10	95	4950
11	150	7512
12	205	7939
13	106	4353

14	150	5990
15	181	6992
16	151	5488
17	145	4706
18	142	5041
19	138	3916
20	160	4187
21	154	3920
22	139	4186
23	110	2485
24	158	3338
25	356	35402
26	129	6705
27	133	8462

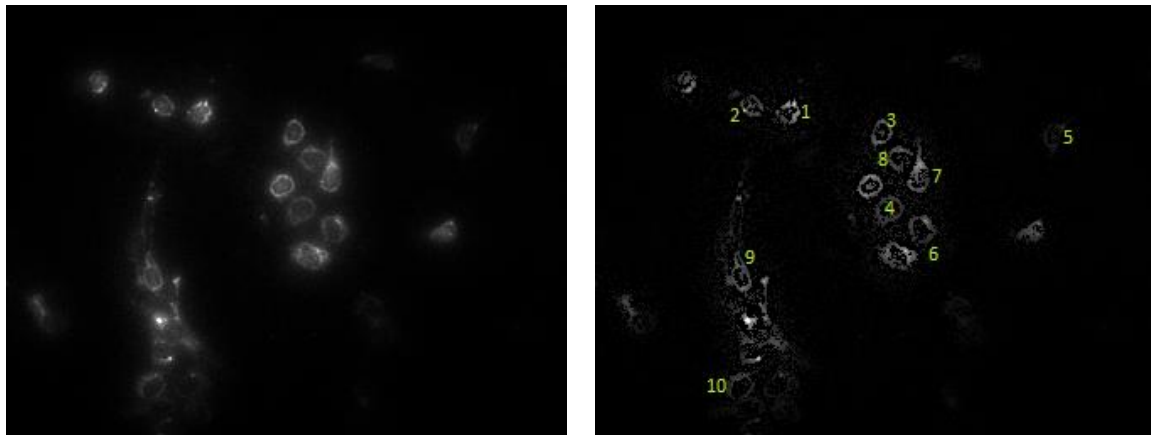


Figura 6.5 Imagen original e imagen normalizada con etiquetas en las células encontradas.

Tabla 6.5 Área y calcio intracelular de la imagen etiquetada en la Figura 6.5.

Célula	Área	Calcio Intracelular
1	107	9327
2	98	6238
3	101	6859
4	171	7852
5	106	2258
6	356	150649

7	155	96719
8	127	42274
9	447	95386
10	173	30925

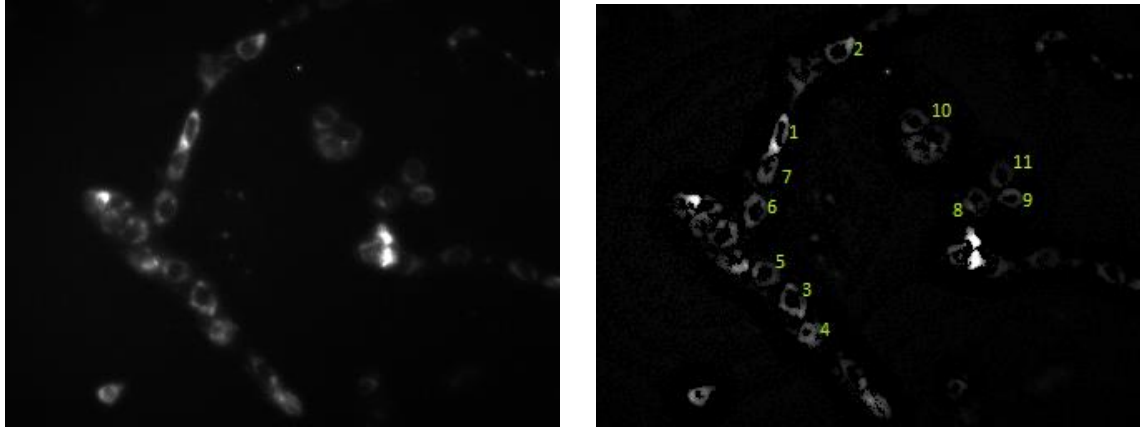


Figura 6.6 Imagen original e imagen normalizada con etiquetas en las células encontradas.

Tabla 6.6 Área y calcio intracelular de la imagen etiquetada en la Figura 6.6.

Célula	Área	Calcio Intracelular
1	154	15633
2	128	7735
3	154	9232
4	170	9357
5	139	5569
6	149	7398
7	105	5625
8	127	3430
9	104	3602
10	112	3550
11	100	2228



International Journal of Advanced and Innovative
research (IJAIR)

ISSN: 2278-7844

Date: **20/11/2017**

Paper ID: **T171114**

“Optimal Image Selection for Segmentation with Particle Swarm Optimization.”


Dear Author,

Congratulations! The manuscript listed above has been approved for **Publication in IJAIR Volume 06 Issue 11(November 2017)**, after having completed peer-review and editorial consideration. Once the article is published, it will be freely accessible for unlimited download to all readers without subscription or other costs.

For further process, you need to submit one copy of duly filled copyright form (attached herewith this mail) with online maintenance fee to IJAIR (send scanned copy of payslip) and final formatted paper according to the attached IJAIR format.

Please send all the required documents to editor.ijair@gmail.com

Again, congratulations on publication of this research contribution. We hope to have the opportunity to consider additional manuscripts from your side going forward. Thank you for submitting a paper to IJAIR.

IJAIR Journal is indexed in Google, Scribd, Advance research library, I2OR , Researcher ID, Thomas Reuters EndNote, ISI - International Society of Indexing. Please communicate to us for any further queries.

Regards,
Priyanka Gupta
Managing Editor
IJAIR
Website: ijair.jctjournals.com
Email ID: editor.ijair@gmail.com

Optimal Image Selection for Segmentation with Particle Swarm Optimization

¹Miguel Angel Razo Salas^{#1}, Ivan Terol Villalobos^{#2}, Hugo Jiménez Hernández^{#3}, Carlos Alberto Olmos Trejo^{#4}, Ana Marcela Herrera Navarro^{#5}

¹ *STUDENT, Computer Science, ¹Universidad Autónoma de Querétaro, Facultad de Informática, Querétaro, México*
mrzo21@alumnos.uaq.mx

² *Assist. Professor, Centro de investigación y Desarrollo Tecnológico en Electroquímica, Querétaro, México*
iterol@cideteq.mx

³ *Assist. Professor, Computer Science, Centro de Ingeniería y Desarrollo Industrial, Querétaro, México*
hugojh@gmail.com

⁴ *Assist. Professor, Computer Science, Universidad Autónoma de Querétaro, Facultad de Informática, Querétaro, México*
mherrera@uaq.mx

Abstract—Calcium is an ion that acts as messenger in the cells. An important step to analyse and to study the Calcium (Ca^{2+}) using images to consist on localization of the cells inside the images. Sometimes this process can be difficult because of environmental factors. In this work, the PSO (Particle Swarm Optimization) algorithm is introduced in order to obtain the best image to segment, reducing the time of processing .

Keywords— Optimization, segmentation, PSO.

INTRODUCTION

Many process performed by cells are highly related with the levels of calcium that contains, where the calcium act as an altern messenger [1]. Producing, haematological problems as muscular diseases, hormonal disorders and another more specific like injury and cell death [2]. The Ca^{2+} can be analysed through substances that generates the movement or activation of calcium [3]. In the literature, different authors have developed different methods to watch stimulating cells cause that calcium has a fluorescence effect as shown in Fig. 1.

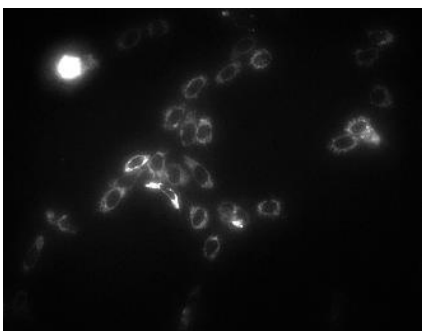


Fig. 1: Images with fluorescence of calcium.

The segmentation allows to locate the cells in the images to evaluate the fluorescence of calcium. Hence, it is possible to analyse the variations.

Image segmentation allows automatically locate objects of interest, hence it is possible graphically represent the significant information, partitioning it into a region of interest. There are many algorithms and it have been developed different techniques to solve this kind of problems [4]. In order to get an image segmentation is necessary execute two important process:

- 1) Pre-processing. This stage consists on removing noise to avoid the over-segmentation,
- 2) Segmentation. This stage consists on extracting the information of interest, Particularly, the objects are localized with the edges detection.

To obtain a good segmentation different techniques can have been applied Particularly, in a first stage is necessary to homogenize the background and to remove loss particles that does not comply with the particular characteristics of the objects to analyse, adding them to the background of the images.

Frequently, images can have minimal changes across the time. Hence, it is important select one image to segment it to analyse the rest of images. This image represents the global optimum.

The paper is organized as follows. The next section presents a review of the foundations of Bio inspired algorithms. In Section 3, the PSO method is presented to detect the cells. Finally, in section 4 we present and discuss the results and the conclusions.

THEORETICAL FUNDAMENTATION

Sequential Search

Sequential search consists in verify element by element until find the information. Basically the search space is analysed

from the beginning to end until to find the item or the data set is over, whichever occurs first. Normally when the algorithm comes to the end, the position of the element is what we want to know in case of being found [5].

Then for this case every pixel from the first image is evaluated, to verify the level of noise that every one of it contains so we can take the best, but this requires processing and time to do so. This method offers some advantages in this kind of problems, always find the optimum global, never get stock on optimum locals of search space, it is trustworthy, among others. Its disadvantage is that in some process is necessary get the solution satisfying certain requirements reducing the processing level. The algorithm starts evaluating all the pixels of every image in the bank of images, keeping just the one who has less noise than the others. If an image i is the actual solution, and when you compare it with $i+1$ exist an improvement the solution now will be this one, finishing only with the optimum global. This require evaluate every possible solution.

Bio Inspired Algorithms

The bioinspired computer systems emerged as a set of models based on the social and cooperative behaviour of some biological species to solve their needs on the searching place. The computer solutions (algorithms) bio inspired reside in the field of optimizations, artificial intelligence, data mining, among others. Particle Swarm Optimization by its initials PSO is one of many bioinspired algorithms that has been used for optimization problems [6][7]). The algorithm act with certain guidelines among particles, like personal and global experience [7][8].

PSO is a practice tool where each particle moves aleatory through velocity and inertia. Each iteration of the algorithm makes one personal solution updating the global if this came out with better than actual one. The erratic movements obtain solutions that may not be the best but they can be a possible optimum local case[9][10]. Reducing the execution time and processing level. The PSO process is described in Fig. 2.

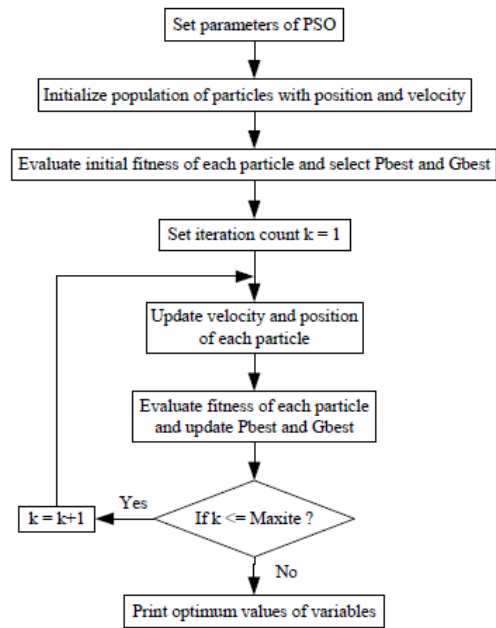


Fig. 2. PSO algorithm flowchart.

Bio Inspired Algorithms

MATERIALS AND METHODS

All the tests were made in a computer with an Intel(R) Core(TM) i7-6500U 2.5GHz processor.

The natural selection consists in evaluate every image, in order to get an optimum global. Particularly 892 images of 256x332 pixels were analysed, with intensities from 0 to 255.

The bio inspired algorithm was applied to optimize and to reduce the time of processing, making easier future work and then use that time on another task like noise removal and image segmentation. PSO was codified on Matlab.

To calculate the noise level in the image without known the objects of interest we propose a noise counting method through threshold. A threshold is applied, where it, makes possible split background from the rest, so we can group all the pixels that do not belong to the background, leaving the variations only in the noise, because the objects of interest remain almost invariant. The image $i+2$ is the best choice, it has less noise than the others two as it is shown in Fig. 3.

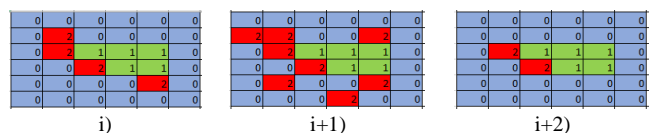


Fig. 3. Matrices 2 value is noise, 1 object of interest and 0 the background. i) image with 9 pixels higher than 0 and therefore they don't belong to the background, i+1) Matrix with 13 pixels that do not belong to the background, i+2) Matrix with 7 pixels that do not belong to the background.

First, 20 periods (iterations) with 25 particles are defined, the velocity of 1.5, inertia of 0.5 and the actual position of the particles we could obtain those solutions presented previously. Each particle evaluates an image selected from the set randomly doing a noise counting, therefore, having 25 particles working together with 20 periods we only verify 500 images instead the 892 that sequential search does.

Six tests were carried out with sequential search, covering the 892 with a total of 343 noise pixels in 3.799458167 seconds. Then another 30 test were done but with PSO method to verify if it reduce the time on the same 892 images trying to get the same solution.

RESULTS

As it can be seen, the 892 images show a very similar behaviour in different threshold levels (See Figs. 4-7). Therefore, optimal solutions are found.

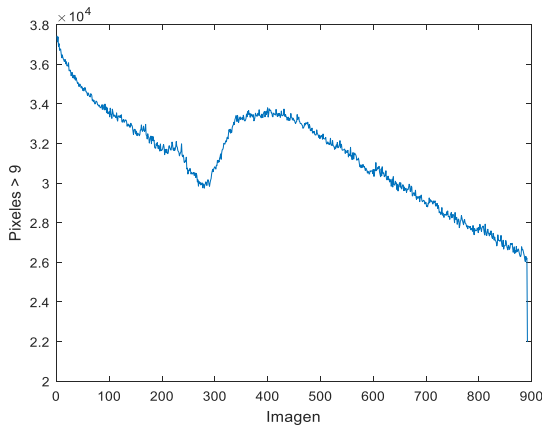


Fig. 4: Pixels higher than a 9 threshold.

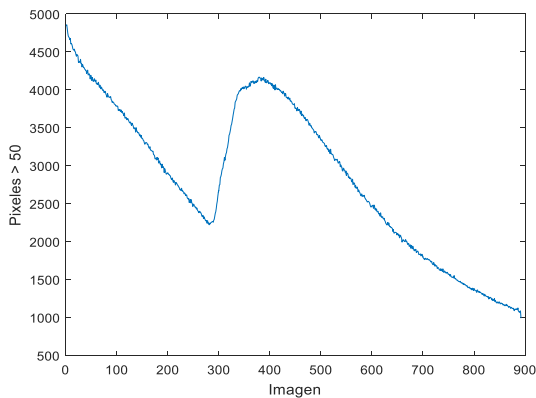


Fig. 5: Pixels higher than a 50 threshold.

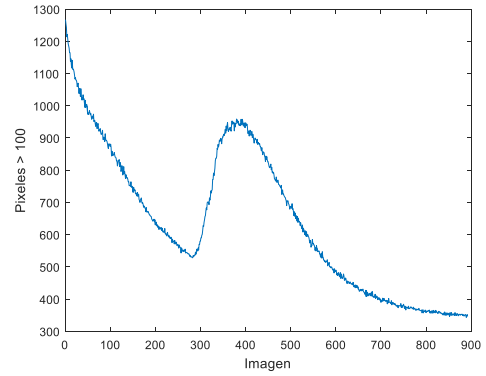


Fig. 6: Pixels higher than a 100 threshold.

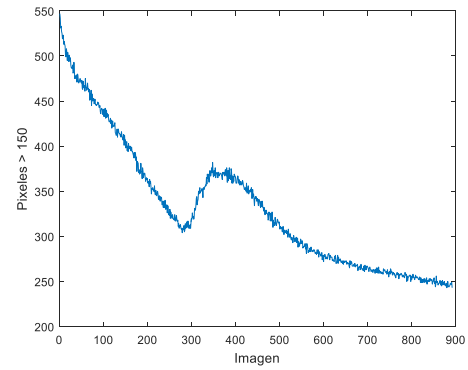


Fig. 7: Pixels higher than a 150 threshold.

In both methods, the time was measured In order to have a reference of it as shown in Fig. 8, 9, 10, 11. We make zoom at the part of the solutions on figures 4,5,6,7

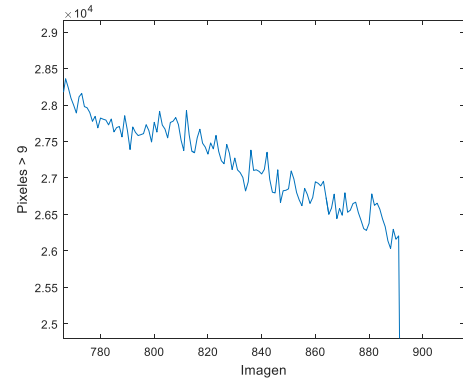


Fig. 8. solution part zoom of Fig. 4

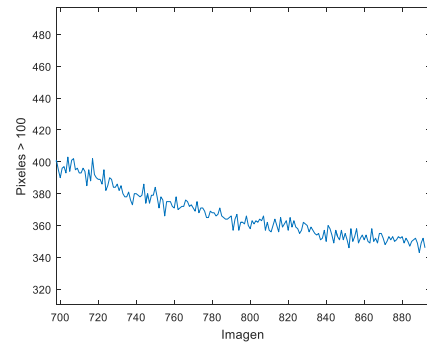


Fig. 9: solution part zoom of Fig. 5

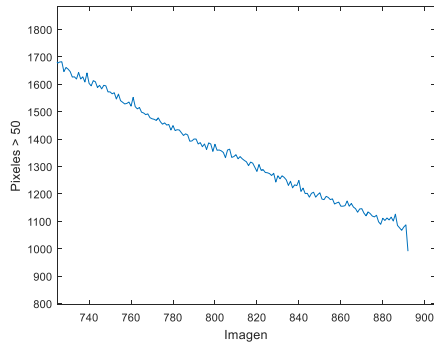


Fig. 10: solution part zoom of Fig. 6

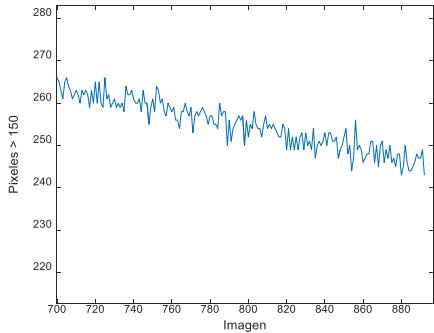


Fig. 11: solution part zoom of Fig. 7

Favorable results were obtained with 8 of 30 tests show the global solution. The others 22 obtained a local solution in the images 852 and 892 with 346 noise pixels which is a very close approximation to the global one. All this, in 2.072606133 seconds as shown in table 1.

TABLE I
COMPARATIVE BETWEEN SEQUENTIAL SEARCH
AND PSO WITH A 100 THRESHOLD.

Sequential Search			
Number of Tests	Image	Noise	Average Time
6	889	343	3.799458167
Particle Swarm Optimization			
8	889	343	2.072606133
5	852	346	
17	892	346	

There are some changes among the images that the human eye cannot distinguish, but with a simple threshold we can see that there are many difference in the intensities of their pixels as shown in the Fig 12.

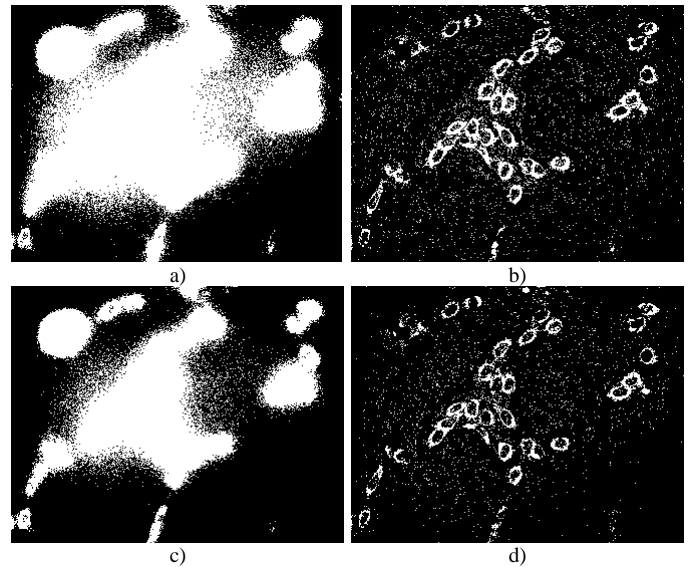


Fig. 6. a) Image 1 from the set with a 9 threshold, b) Image 1 with with the tophat transformation, structural element of 2 and a threshold with 1976 object inside of the image, c) Image 892 of the set with a 9 threshold, d) Image 892 with with the tophat transformation, structural element of 2 and a threshold with 1473 object inside of the image..

CONCLUSIONS

Bio- inspired are algorithms that imitate simple and practical biological behaviors that can make numeric predictions or optimize search processes. For this investigation PSO was used in the optimization field, doing the image selection search for the segmentation. The time was optimized by reducing the total iterations and evaluations of the images. It is a good choice dedicate some time to tasks who make easier the future work, like select an image from the set that contain less noise than the others. Depending on the selection, the application of filters and transformations we get a filtered image with more or less noise making easier or harder the segmentation. A total of 1.726832034 seconds was reduced equivalents to almost 1450555708.56 operations that can be used in others tasks like noise removal or the segmentation.

ACKNOWLEDGMENT

The author, Miguel Razo thank the government agency CONACyT for the financial support. The author Ana Herrera would like to thank Ana Paula for their great encouragement.

REFERENCES

- [1] M. J. Berridge, M. D. Bootman, and H. L. Roderick, "Calcium signalling: dynamics, homeostasis and remodelling.," *Nat. Rev. Mol. Cell Biol.*, vol. 4, no. 7, pp. 517–29, 2003.
- [2] M. J. Berridge, P. Lipp, and M. D. Bootman, "The versatility and universality of calcium signalling," *Nat Rev Mol Cell Biol*, vol. 1, no. 1, pp. 11–21, Oct. 2000.
- [3] S. A. Stricker, V. E. Centonze, S. W. Paddock, and G.

Schatten, "Confocal microscopy of fertilization-induced calcium dynamics in sea urchin eggs.," *Dev. Biol.*, vol. 149, no. 2, pp. 370–380, Feb. 1992.

- [4] P. G. Ho, Ed., *Image Segmentation*. InTech, 2011.
- [5] O. Cairó and S. Guardati, "Estructuras de datos," 3era ed., Mc Graw Hill, 2010, pp. 177–183, 277–282.
- [6] J. A. Ruiz-Vanoye and D. O. Diaz-Parra, *Bio-innovacion*, Primera. Cuernavaca, Morelos, Mexico: Editada, 2013.
- [7] T. Roorkee, "Particle Swarm Optimization : Algorithm and its Codes in MATLAB Particle Swarm Optimization : Algorithm and its Codes in MATLAB," no. March, 2016.
- [8] J. J. Dongarra, "Performance of various computers using standard linear equations software in a Fortran environment," *SIGARCH Comput. Arch. News*, vol. 20, no. 3, pp. 22–44, 1992.
- [9] E. Aarts and J K Lenstra. *Local Search in Combinatorial Optimization*. Wiley. 2003.
- [10] Xin-She Yang. *Nature-Inspired. Metaheuristic Algorithms*, 2nd Edition. Luniver Press, 2010.
- [11] A Mozaffari, A. Fathi, and S. Behzadipour. The great salmon run: a novel bio-inspired algorithm for artificial system design and optimisation. *International Journal of BioInspired Computation*, 4(5):286–301, 2012.

6.2 Constancias y Reconocimientos



**FOPER
2017**

La Universidad Autónoma de Querétaro
y La Secretaría de la Rectoría



otorgan el presente

RECONOCIMIENTO

a

Miguel Ángel Razo Salas

Por su participación como

Responsable de Proyecto

FOPER 2017

"Fondo de Proyectos Especiales de Rectoría"

Santiago de Querétaro, Qro, noviembre 2017.

Dr. Gilberto Herrera Ruiz
RECTOR

Dra. Rebeca del Rocío Peniche Vera
TITULAR DE LA SECRETARÍA DE LA RECTORÍA