

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO
FACULTAD DE MEDICINA
LICENCIATURA EN BIOINGENIERÍA MÉDICA
DEPARTAMENTO DE EVALUACIÓN PROFESIONAL



**DISEÑO DE UN ALGORITMO BASADO EN SUPERPIXELES APLICADO A
IMÁGENES DE HISTOPATOLOGÍA PARA LA DETECCIÓN DE NIDOS DE
AMASTIGOTES**

TESIS

QUE PARA OBTENER EL TÍTULO DE
LICENCIADO EN BIOINGENIERÍA MÉDICA

PRESENTA:

P.L.B.M. PEDRO VÁZQUEZ OSORIO

DIRECTOR:

DR. JOSÉ JAVIER REYES LAGOS

ASESOR EXTERNO:

DR. JORGE LUIS PÉREZ GONZÁLEZ

REVISORES:

DRA. ADRIANA CRISTINA PLIEGO CARRILLO

MTRA. CLAUDIA IVETTE LEDESMA RAMÍREZ

TOLUCA, ESTADO DE MÉXICO; MAYO, 2023.

Resumen

El objetivo del estudio fue la implementación de un algoritmo basado en superpíxeles para la detección de nidos de amastigotes en imágenes de histopatología cardíaca, considerando la importancia de un diagnóstico temprano de la enfermedad de Chagas para evitar las consecuencias de una enfermedad en etapa crónica.

La investigación de este trabajo fue de tipo aplicada y se trabajó con 22 microfotografías de cortes histopatológicos del corazón de ratones infectados con *Trypanosoma Cruzi*. Para la segmentación con superpíxeles, se utilizó el método *Simple Linear Iterative Clustering* (SLIC) debido al buen desempeño del mismo y para reducir el trabajo computacional.

Con el presente trabajo, se pretendió obtener imágenes con segmentos bien definidos, con una buena agrupación por similitud de color y proximidad espacial. El algoritmo desarrollado tuvo un *accuracy* de 98.20% en la etapa de validación cruzada, y un *accuracy* promedio de 94.34% para las 6 pruebas finales realizadas. Comparando los resultados obtenidos con trabajos previos de detección de nidos de amastigotes, en los cuales se utilizaron técnicas de aprendizaje profundo con una cantidad mayor de imágenes se observaron métricas similares a las obtenidas en esta investigación. Con esto, se pudo concluir que la detección de nidos de amastigotes en imágenes de histopatología chagásica mediante algoritmos basados en superpíxeles y aprendizaje computacional podría ser una alternativa al diagnóstico clásico de la enfermedad de Chagas, con la ventaja de una detección de la enfermedad en una etapa temprana.

Índice general

1. Marco teórico	6
1.1. Enfermedad de Chagas	6
1.1.1. Diagnóstico	7
1.1.2. Estudio de imágenes histopatológicas	8
1.1.3. Tratamiento	9
1.2. Procesamiento digital de imágenes	9
1.3. Segmentación de superpíxeles	11
1.3.1. Algoritmos de generación de superpíxeles	12
1.4. Aprendizaje computacional	17
1.4.1. Aplicaciones del aprendizaje computacional en la medicina	18
1.5. Métodos de aprendizaje automático para el diagnóstico de enfermedades	20
1.5.1. Diagnóstico de enfermedad de Chagas	21
2. Planteamiento del problema	23
2.1. Pregunta de investigación	24
3. Justificación	25
4. Hipótesis	27
5. Objetivos	28
5.1. Objetivo general	28
5.2. Objetivos específicos	28
6. Metodología	29
6.1. Base de datos y preprocesamiento	29
6.2. Implementación del algoritmo SLIC	31
6.3. Extracción de características	31

6.4. Clasificación y validación	33
7. Implicaciones Éticas	36
8. Resultados y Discusión	37
8.1. Implementación del algoritmo SLIC	37
8.2. Extracción de características	38
8.3. Clasificación y validación	39
9. Conclusiones	43
10.Recomendaciones	44

Índice de figuras

1.1. Ciclo de la infección del <i>T. cruzi</i> . Obtenida de [1].	7
1.2. Distribución mundial de casos de enfermedad de Chagas. Obtenida de [2].	8
1.3. Digitalización de una imagen. Obtenida de [3].	10
1.4. Etapas del procesamiento digital de imágenes. Obtenida de [4].	11
6.1. Metodología.	29
6.2. A la izquierda, imagen de histopatología de tejido cardiaco de un ratón infectado con <i>T. cruzi</i> . A la derecha; segmentación manual de los nidos de amastigotes, realizada por un experto en histopatología. Elaboración propia.	30
6.3. Diagrama de flujo de las tareas realizadas en el código de Python. Elaboración propia.	35
8.1. Imagen original con el nido de amastigotes señalado con una flecha roja (izquierda). Segmentación SLIC de superpíxeles, con los segmentos de distinto color (derecha).	37
8.2. Resultados generales de las pruebas.	38
8.3. Resultados de las pruebas con imágenes 1 y 2.	40
8.4. Resultados de las pruebas con imágenes 3 y 4.	40
8.5. Resultados de las pruebas con imágenes 5 y 6.	41
8.6. Tabla de descriptores.	42

Capítulo 1

Marco teórico

1.1. Enfermedad de Chagas

La enfermedad de Chagas, identificada en la primera década del siglo XX, es una enfermedad causada por el protozooario *Trypanosoma cruzi* (*T. cruzi*) [5]. La infección se produce mediante el contacto entre cualquier ruptura de la piel, ojos o boca con heces contaminadas de un insecto triatomino, insecto que vive en grietas de las paredes y techos de las casas mal construidas; otras formas de transmisión son la transfusión de sangre infectada, transmisión oral a través de alimentos contaminados, transmisión vertical (de madre a feto), trasplante de órganos [6], accidentes de laboratorio, uso compartido de agujas intravenosas. La transmisión sexual se ha observado en experimentos con ratones; sin embargo, actualmente no se reportan casos de este tipo en humanos [7].

El *T. cruzi* presenta dos fases diferenciadas (figura 1.1), una en los hospedadores vertebrados y otra en los insectos transmisores. En el vector, el ciclo de vida del parásito tiene lugar en el intestino, donde pasa de tripomastigote (forma extracelular, no replicativa e infectiva, que se encuentra en la sangre de los mamíferos, en el intestino posterior de los vectores y en sus deyecciones), a tripomastigote metacíclico, forma que surge con sus heces y que ya es infectiva para el hombre y los animales [8].

El ciclo de vida del *T. cruzi* en mamíferos comienza con la mordedura para alimentarse de sangre, liberando tripomastigotes en sus heces cerca del sitio de la lesión; los tripomastigotes entran al huésped a través de la herida o a través de membranas mucosas. Dentro del huésped, los tripomastigotes invaden las células cercanas al lugar de inoculación, donde se diferencian en amastigotes intracelulares; estos amastigotes se multiplican por fisión binaria y se diferencian en tripomastigotes para posteriormente liberarse a la circulación como tripomastigotes del torrente sanguíneo [9]. Los tripomastigotes del torrente sanguíneo

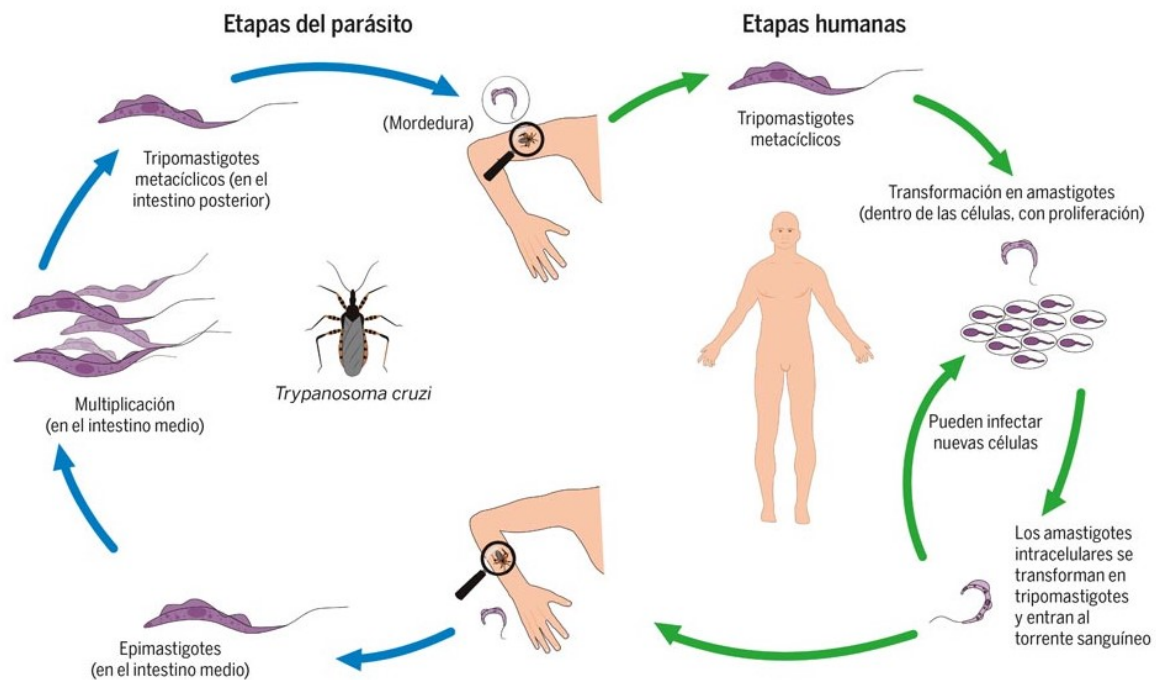


Figura 1.1: Ciclo de la infección del *T. cruzi*. Obtenida de [1].

pueden entonces invadir nuevas células e iniciar nuevos ciclos de replicación y de esta forma son capaces de infectar vectores que se alimenten del huésped [10].

Tras la infección por *T. cruzi* los pacientes entran en la etapa aguda de la enfermedad de Chagas, etapa que toma entre 40 y 60 días y posteriormente pasan a la fase crónica, la cual puede durar toda su vida. La fase aguda se caracteriza por ser normalmente asintomática o por presentar un cuadro de fiebre [11]. En algunos casos de transmisión vertical se presenta un nódulo en la piel del bebé llamado chagoma o un edema en el párpado llamado el signo de Romaña, esto puede ser un indicador de la inoculación del parásito [12].

Si tras la fase aguda no se tuvo algún tratamiento, cerca del 30% de los pacientes infectados entran a la fase crónica [7], la cual tiene un tiempo indeterminado y puede durar algunos años o toda la vida. Esta fase esta caracterizada por la ausencia de síntomas relevantes y parasitemia intermitente o inexistente [12].

1.1.1. Diagnóstico

El diagnóstico durante la fase aguda, se puede obtener mediante estudios de microscopía convencional como el test de Strout o técnicas de tinción con hematoxilina-eosina [13], esto debido principalmente a los niveles de parasitemia, sin embargo, los estudios directos como

el microhematocrito o el test de Strout, presentan limitaciones de sensibilidad y requieren ser analizados por personal altamente capacitado. Durante la etapa crónica, el diagnóstico requiere de métodos de amplificación biológica como el hemocultivo y el xenodiagnóstico, métodos que además de ser difíciles de realizar, son costosos y consumen mucho tiempo [12]. La prevalencia de la enfermedad de Chagas en el mundo se muestra en la (figura 1.2).

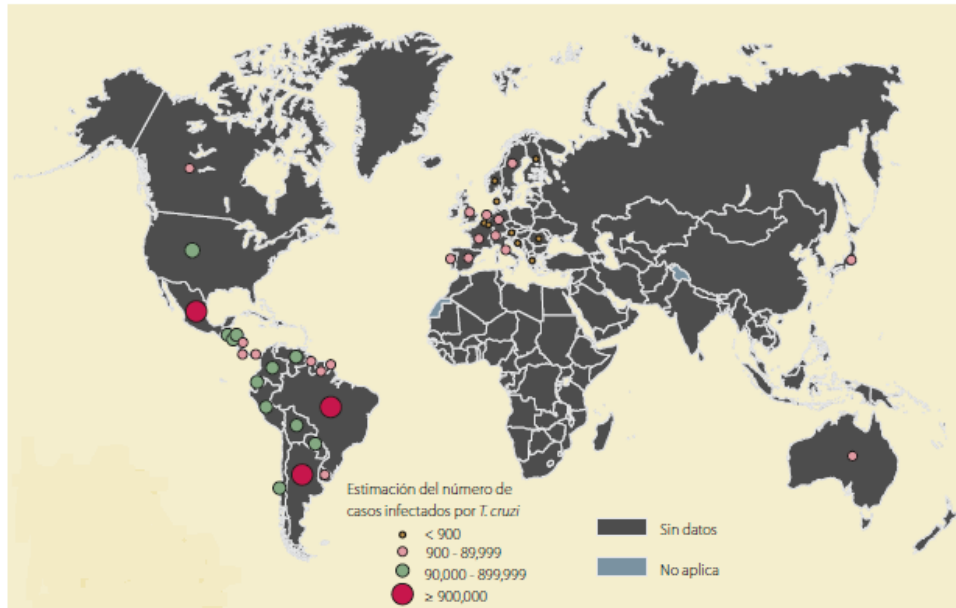


Figura 1.2: Distribución mundial de casos de enfermedad de Chagas. Obtenida de [2].

Durante los 10-20 años posteriores a la infección, se provoca una miocarditis derivada de la presencia de parásitos en las miofibrillas del corazón y cerca del 25% de los infectados desarrollan cardiopatía chagásica crónica (CCC), esta es la manifestación más importante de la infección, la CCC puede traer al paciente arritmias cardiacas, trombo embolias o un paro cardiaco [14].

1.1.2. Estudio de imágenes histopatológicas

La identificación protozoos se realiza frecuentemente con técnicas de tinción hematoxilina-eosina, para el diagnóstico del *T. cruzi* se utilizan específicamente tres [13]:

- **Tinción de Giemsa:** Este método de tinción fue diseñado principalmente para la demostración de parásitos en la malaria [15]. En esta tinción se observan protozoos y otros microorganismos en color azul oscuro, el fondo rosa/azul pálido y los núcleos en color azul [16].

- **Tinción de Romanowsky:** es la tinción tras la aplicación de colorantes que contienen derivados del azul de metileno más un rojo-naranja halogenado colorante de fluoresceína. En esta tinción el color característico de los núcleos celulares y de los gránulos de polimorfos de neutrófilos es color púrpura, otras estructuras celulares pueden presentarse en color rojo anaranjado, rojo ladrillo, rosa, rojo, azul, azul claro, negro, violeta-negro, violeta-rojo, etc [17].
- **Tinción de Wright:** es un tipo de tinción de Romanowsky usada comúnmente en hematología para la tinción de frotis de sangre periférica. Los resultados que se observan son, eritrocitos en color rojo amarillento, núcleo de neutrófilos color púrpura oscuro, núcleo de eosinófilos en color azul, núcleo de basófilos color púrpura a azul oscuro, núcleo de linfocitos en púrpura oscuro, núcleo de monocitos púrpura oscuro [18].

1.1.3. Tratamiento

El tratamiento de la enfermedad de Chagas se basa en dos medicamentos, el benznidazol y el nifurtimox. Las tasas de respuesta terapéutica van a depender de la fase en la que se encuentre el paciente y de la zona geográfica. En la fase aguda ambos fármacos presentan entre 65 y 80 % en la tasa de curación, mientras que en la etapa crónica su tasa de curación es de entre 15 y 40 %.

A pesar de que estos dos medicamentos presentan tasas de curación aceptables, estos tienen como limitante la alta tasa de efectos adversos, cosa que lleva a alrededor del 6 y 40 % de los pacientes a abandonar el tratamiento con nifurtimox y aproximadamente entre el 7 y 30 % de los que reciben benznidazol, por esta razón el benznidazol es el tratamiento estándar actual [19].

1.2. Procesamiento digital de imágenes

Una imagen es la representación visual de un objeto, persona o paisaje [20]. Una imagen se puede definir como una función $f(x, y)$ donde (x, y) representan coordenadas espaciales y la amplitud de f para cualquier par de coordenadas (x, y) es la intensidad o el nivel de gris de la imagen en ese punto. Cuando (x, y) y los valores de intensidad de la función son todos cantidades discretas, finitas, se le llama imagen digital [21].

La imagen digital de cualquier objeto puede formularse como resultado de la reflexión de la iluminación que cae sobre un objeto (figura 1.3). La luz reflejada pasa a un sistema de imágenes (parecido a un convertidor analógico a digital) y este aplica un muestreo para hacer

la señal discreta en el tiempo, una cuantificación para hacer la señal discreta en amplitud y por ultimo una codificación para mapear los valores cuantificados en bits para formar la señal digital [3].

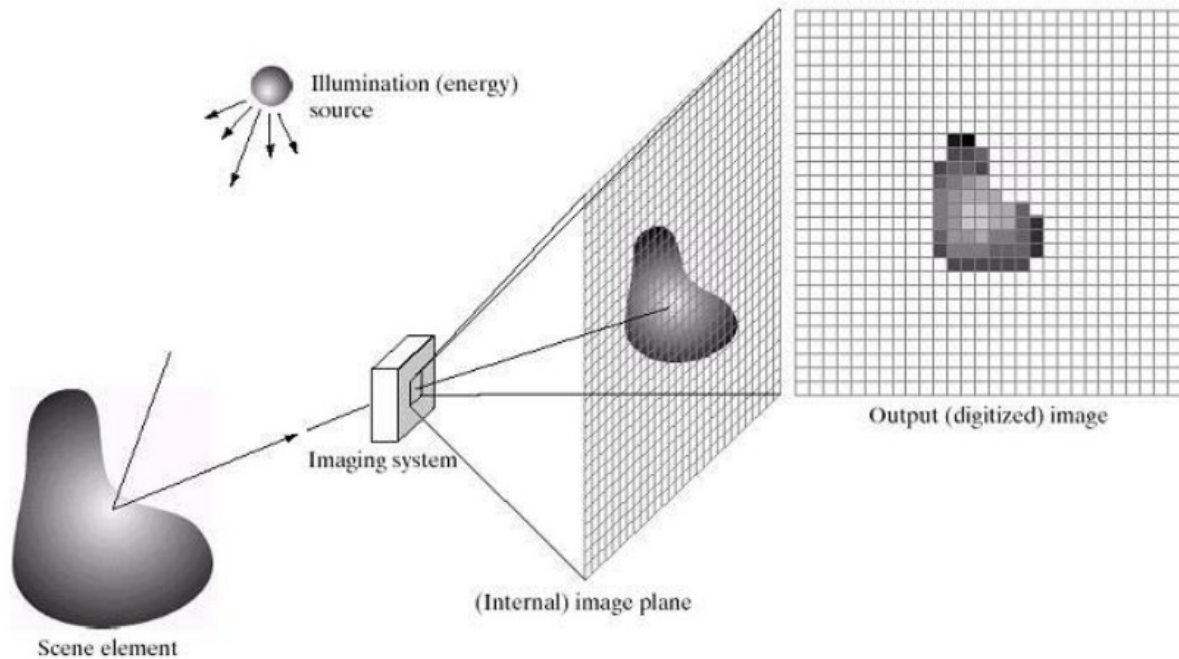


Figura 1.3: Digitalización de una imagen. Obtenida de [3].

Las imágenes digitales, se componen de píxeles (*picture elements*), cada píxel representa el color o nivel de gris en cierto par de coordenadas en la imagen. Una imagen digital es entonces, un arreglo rectangular de píxeles [22]. Cada píxel es representado por valores numéricos; para imágenes en escala de grises un solo valor es suficiente para representar su intensidad, este valor generalmente se encuentra en el rango de $[0, 255]$ donde el 0 es el valor mas oscuro y 255 el mas claro. Para las imágenes a color existen distintos modelos, el más común es el *red, green, blue* (RGB), en este modelo se requiere de 3 valores para cada píxel, el primer valor representa la intensidad de color rojo, el segundo la intensidad de color verde y el tercer valor la intensidad de azul. Si la imagen tiene solo dos intensidades se conoce como una imagen binaria [20].

En la (figura 1.4) se muestran las etapas del procesamiento de imágenes. El proceso inicia con la adquisición de imágenes, en esta etapa se requiere de un sensor de imágenes, cuyas señales producidas deben ser digitalizadas. La naturaleza del sensor dependerá del tipo de aplicación que se quiera estudiar, se utilizan por ejemplo, la luz para la fotografía; rayos X para la radiografía, ultrasonido para la ecografía, etc. El siguiente paso es el preprocesamiento, el objetivo de esta etapa es detectar y eliminar las fallas que puedan existir en la imagen

para mejorarla, para esto, las tres técnicas más utilizadas son la mejora del contraste la eliminación del ruido, y la restauración. La siguiente etapa es la segmentación, en esta, la imagen se divide en objetos con el fin de separar las partes necesarias del resto de la imagen, para la etapa de segmentación, existen técnicas básicas y están orientadas a los píxeles, a los bordes, y a las regiones. Algo muy importante de estas técnicas es que no son excluyentes sino que se combinan de acuerdo a su aplicación. La siguiente etapa es la descripción o extracción de características, esta consiste en extraer características con información de interés o que sea fundamental para diferenciar una clase de objetos de otra. La etapa siguiente es el reconocimiento, este proceso consiste en asignar etiquetas a los objetos basándose en la información proporcionada por los descriptores. La interpretación implica asignar significado a un conjunto de objetos reconocidos. Por último, la etapa base de conocimiento, va a almacenar el dominio del problema para guiar la operación de cada módulo de procesamiento, esta etapa también controla la interacción entre dichos módulos [4].

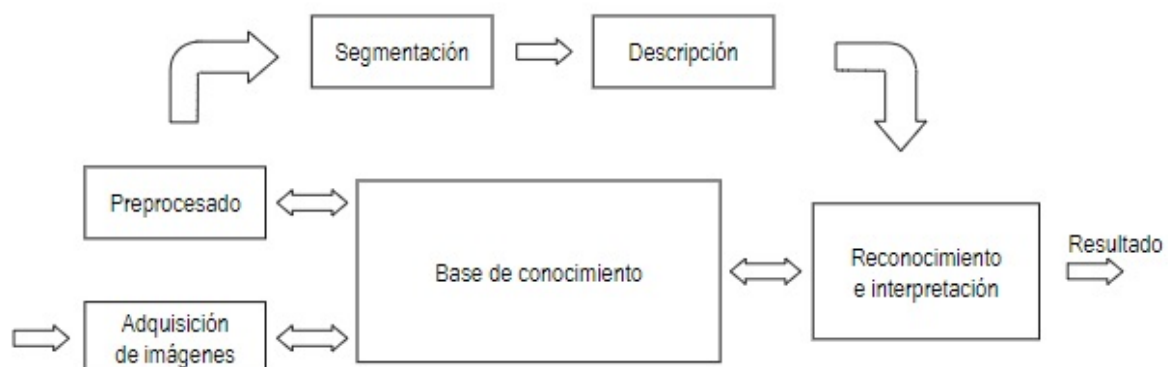


Figura 1.4: Etapas del procesamiento digital de imágenes. Obtenida de [4].

1.3. Segmentación de superpíxeles

El término *superpixel* se refiere a la agrupación de píxeles similares en color y algunas otras propiedades de bajo nivel, es decir los superpíxeles son el resultado de una discretización de píxeles [23].

Hoy en día, existen diversos métodos para generar superpíxeles, cada uno presenta distintas ventajas y desventajas, las cuales se tienen que evaluar dependiendo de la aplicación que se les quiera dar. En general, las propiedades deseables para la obtención de superpíxeles son:

- Buena adherencia a los límites de la imagen.
- Cuando se van a utilizar con la finalidad de reducir la complejidad computacional como un paso de preprocesamiento, los superpíxeles deben ser rápidos de calcular, eficientes en memoria y fáciles de usar y;
- Cuando se van a utilizar con fines de segmentación, los superpíxeles deben aumentar la velocidad y mejorar la calidad de los resultados [24].

1.3.1. Algoritmos de generación de superpíxeles

Los algoritmos para generar superpíxeles se pueden clasificar en métodos basados en gráficos o basados en agrupamiento [25].

Métodos basados en gráficos: Estos métodos tratan cada píxel como un nodo en un gráfico y las similitudes existentes entre píxeles vecinos se denominan pesos de borde. Los superpíxeles son generados mediante una minimización de la función de costo definido sobre el gráfico y ofrecen mejoras en la calidad visual y eficiencia computacional, pero pueden presentar desventajas como la pérdida de resolución y problemas de contorno [25]. Dentro de esta clasificación encontramos métodos tales como:

- ***N-cut:*** Este algoritmo divide el gráfico de todos los píxeles de la imagen utilizando el contorno y la textura, su adherencia a los límites es relativamente pobre [25].
- ***Felzenszwalb and Huttenlocher:*** Este método aglomera píxeles definiendo una predicción para medir la evidencia de un límite entre las dos regiones utilizando una representación gráfica de la imagen. Los superpíxeles se generan encontrando el árbol de expansión mínimo de los píxeles constituyentes. En comparación con la segmentación con corte-N, este método se adhiere bien a los límites de la imagen en la práctica, pero los superpíxeles tienen formas y tamaños irregulares, además de que el método Felzenszwalb and Huttenlocher no puede controlar directamente el número de superpíxeles [25].
- ***Homogeneous Superpixels:*** Este algoritmo calcula superpíxeles mediante el uso de agrupamiento de Markov, el cual es un algoritmo basado en gráficos que emplea la circulación de flujo estocástico. Los nodos del gráfico se corresponden con los píxeles de la imagen de entrada y una matriz de adyacencia se inicializa usando una medida de similitud simple para definir el peso del borde [25].

- ***Pseudo-Boolean:*** Este algoritmo trabaja asignando múltiples etiquetas. Inicialmente, la imagen de entrada está cubierta por tiras horizontales superpuestas a la mitad y cada píxel tiene la posibilidad de ser asignado a una de dos tiras latentes alternativas, luego, la tarea es decidir a qué tiras pertenecen los píxeles. Genera superpíxeles que son regulares tanto en tamaño como en forma y la velocidad de este algoritmo es independiente del número de superpíxeles [25].
- ***Topology Preserved regular Superpixel:*** Este método consta de tres pasos para lograr la segmentación del superpíxel. Primero, las semillas originales se colocan en una cuadrícula regular. Después, cada semilla se reubica en el límite apropiado siguiendo las restricciones de magnitud máxima del borde. Por último, el algoritmo genera la ruta óptima local conectando la vecindad de cada semilla reubicada, vertical y horizontalmente. En este algoritmo cuando el número de superpíxeles es inferior a 300, se convierte en el algoritmo de superpíxeles más rápido [25].
- ***Lazy Random Walk:*** Con este algoritmo la imagen de entrada se traduce en un gráfico. El vértice del gráfico es el píxel de la imagen y el borde se define en la función de ponderación gaussiana. Este método inicializa semillas en el gráfico, después los superpíxeles iniciales se van optimizando iterativamente mediante la nueva función de energía, la cual se encuentra relacionada con el tiempo iterativo y la medición de la textura. Con este método la segmentación se adhiere bien a los límites del objeto [25].
- ***Entropy rate superpixel:*** Este algoritmo trabaja la segmentación como un problema de maximización en el gráfico y presenta una nueva función objetivo en la topología de grafos. Se asigna la imagen a un gráfico con vértices que indican los píxeles y el borde pesos que denotan las similitudes por pares. El objetivo consiste en seleccionar un subconjunto de aristas del gráfico resultante con un número exacto de superpíxeles subgráficos conectados [25].

Métodos basados en agrupamiento Estos algoritmos agrupan y refinan superpíxeles iterativamente hasta que se satisfagan algunos criterios de convergencia. Estos algoritmos ofrecen ventajas como la segmentación coherente, preservación de bordes y flexibilidad en forma y tamaño de los superpíxeles. Sin embargo, pueden presentar desventajas relacionadas con la sensibilidad a la inicialización, inconsistencia en el tamaño, ruido y necesidad de ajuste de parámetros. [25]. Algunos algoritmos de superpíxeles basados en agrupamiento son:

- **SLIC (Simple Linear Iterative Clustering):** Este método puede dividir la imagen en un número predeterminado de superpíxeles. De acuerdo con el número establecido de superpíxeles, el algoritmo primero convierte la imagen del espacio RGB al espacio *Lab* (*L* de luminosidad, este canal es el mapa de contraste en escala de grises. La letra *a* corresponde al balance de color entre verde y magenta, y la *b* es el balance de color entre azul y amarillo), divide la imagen en áreas pequeñas de manera uniforme y selecciona el centro del grupo en cada área pequeña de acuerdo con el principio de gradiente mínimo. La generación de superpíxeles la realiza calculando la similitud de color y el espacio de distancia entre píxeles en la imagen. Cada píxel de la imagen se descompone en un vector de cinco dimensiones (l, a, b, x, y) , donde *l*, *a*, *b* son los componentes en el espacio de color CIELab y *x*, *y* son las coordenadas del píxel. Las ventajas del algoritmo SLIC son la simplicidad, la eficiencia, el número controlable de superpíxeles, una cobertura de bordes más precisa y superpíxeles generados de manera más uniforme. El algoritmo SLIC distribuye los centros de agrupación uniformemente en la imagen en la etapa inicial y no toma en cuenta algunas regiones locales con información rica, lo que puede causar la pérdida de muchos detalles e información de textura de borde en las regiones locales de destino [26].
- **Watershed:** Este algoritmo parte del mínimo local para producir líneas que separan las cuencas hidrográficas. Los superpíxeles resultantes son muy irregulares en tamaño y forma, y no exhiben una buena adherencia a los límites, el método es relativamente rápido con la complejidad, pero no ofrece control sobre la cantidad de superpíxeles o su compacidad. Existe un algoritmo derivado, Watershed compact el cual es una modificación que considera las restricciones de compacidad de los superpíxeles y puede controlar el número de superpíxeles directamente [25].
- **Meanshift:** Este algoritmo cambia continuamente de manera estadística el centro del grupo. El algoritmo Meanshift genera superpíxeles al agrupar píxeles con el mismo punto de modo en la misma área. En este algoritmo la cantidad de superpíxeles no se puede controlar de manera efectiva y el efecto de segmentación no es ideal [26].
- **Quickshift:** Es un método de segmentación de búsqueda de modo. Mueve cada punto en el espacio de características al vecino más cercano que aumenta la estimación de densidad de Parzen. Este método tiene una adherencia a los límites relativamente buena, pero no puede controlar el tamaño ni el número de superpíxeles [25].
- **Turbopixel:** Este es un algoritmo rápido para generar superpíxeles de flujos geomé-

tricos. El método establece el punto semilla inicial, este se expande para formar una curva cerrada y luego realizar la evolución de la curvatura para formar un superpixel. El superpixel generado tiene una mejor uniformidad, la velocidad de evolución de los puntos límite se ralentiza o se detiene en el lugar donde el gradiente de píxeles es grande, lo que hace que la imagen mantenga su límite local [26].

- **Manifold SLIC:** A diferencia del SLIC convencional, Manifold SLIC asigna la imagen a un colector bidimensional para calcular la densidad del contenido. De esta forma se ejecuta 10 veces más rápido que los algoritmos de superpixel sensibles al contenido [25].
- **Contour Relaxed Superpixel:** Este algoritmo produce superpíxeles bajo la restricción de obtener la máxima homogeneidad de la textura dentro de cada parche y la máxima conformidad de los contornos tanto con el contenido de la imagen como con un modelo de campo aleatorio de Gibbs-Markov quién formuló el problema de segmentación de superpíxeles como una tarea de estimación [25].
- **Linear Spectral Clustering:** Este algoritmo además de generar superpíxeles compactos y uniformes con bajo costo computacional también puede capturar los atributos de imagen globales. El algoritmo se basa en la medición de la similitud de color y la proximidad espacial entre los píxeles de la imagen, utilizando la función kernel para aproximar la medida de similitud y asignando los valores y coordenadas de los píxeles al espacio de características de alta dimensión. El rendimiento de este algoritmo es equivalente al algoritmo de segmentación de superpíxeles más avanzado [26].
- **Depth-adaptive superpixels:** Este algoritmo es específico para imágenes $RGB - D$, imágenes que tienen información de profundidad. El algoritmo utiliza la información de profundidad para simplificar la tarea de segmentación. El algoritmo consta de 3 pasos, primero, a partir de la imagen de profundidad se calcula la densidad de grupos de superpíxeles en el espacio de la imagen; el segundo paso consiste en el método de muestreo multiescala que se utiliza para muestrear puntos que garantizarán la propiedad del espectro de ruido azul, por último, el agrupamiento de las medias se utiliza para asignar puntos a los superpíxeles y mejorar su centro de los, este método puede generar superpíxeles en tiempo real [25].
- **Compact Superpixels:** Este algoritmo asume que la imagen de entrada está cubierta intensamente por parches cuadrados correspondientes a una etiqueta superpuestos a

la mitad con el mismo tamaño. Después se asigna a cada píxel a un parche único minimizando una función de energía compuesta de términos de datos los cuales deciden que cada píxel pertenece a qué parche y términos de suavizado que van a controlar el límite del parche. En este método, los tamaños de los superpíxeles usualmente se igualan y la función de energía fomenta que los límites sean compactos [25].

- ***Constant intensity superpixels:*** Este método primero asigna a cada parche el color del píxel en el centro y requiere que el superpíxel tenga una intensidad constante al agregar restricciones a la función del objeto. Los superpíxeles resultantes de este algoritmo se vuelven coherentes en color [25].
- ***Seeds:*** Este algoritmo parte de una partición inicial de superpíxeles, luego, los superpíxeles se refinan modificando los límites de acuerdo con la función de energía, la cual se define en los límites de superpíxeles y el histograma de color y contiene dos términos: término de color y término de límite. El tiempo de ejecución de SEEDS está relacionado con el tiempo de iteración, sin embargo los superpíxeles sufren de una gran irregularidad de forma y la cantidad es difícil de controlar [25].
- ***VCells:*** VCells tiene dos pasos, en el primero, la imagen se divide en segmentos con tamaño y forma uniformes, y luego se usó EWCVTLLN [25] un algoritmo de agrupamiento clásico, menos costoso que los métodos de segmentación basados en ecuaciones diferenciales parciales [27]. para ajustar el límite de estos segmentos para que los superpíxeles cumplan con la restricción de compacidad [25].
- ***VCCS Voxel Cloud Connectivity Segmentation:*** Este es el primer algoritmo que utiliza relaciones de vóxeles para producir superpíxeles. Primero se construyó el gráfico de adyacencia para el vóxel, luego se eligieron varios puntos semilla como centros de conglomerados en el espacio 3D y después de la inicialización, se genera VCCS ajustando iterativamente los centros de conglomerados hasta que los centros del racimo se estabilicen [25].
- ***Superpixel Lattices:*** Este es un algoritmo de superpíxel que genera superpíxeles al detectar franjas verticales u horizontales. Inicialmente, la imagen se divide en un segmento vertical y uno horizontal, después se agrega una ruta vertical u horizontal adicional en cada paso subsiguiente. El objetivo de esta segmentación es encontrar rutas verticales y horizontales ponderadas mínimas cuyo costo límite sea el más bajo. El método de Superpixel Lattices no se adhiere bien al límite de la imagen [25].

- ***Lattice cut:*** Este método comienza con una cuadrícula regular y luego combina superpíxeles de forma iterativa, de esta forma asocia una etiqueta a cada píxel para indicar a qué superpíxel pertenece. Este algoritmo puede construir una solución óptima global en dirección horizontal o vertical utilizando un solo corte de gráfico [25].
- ***Scene Shape Superpixel:*** Este algoritmo es adecuado para imágenes donde los objetos en el centro de la imagen tienden a estar más distantes y más pequeños que los del borde. El método primero aprende la distribución de límites antes ser utilizado para mejorar la segmentación y luego utiliza el algoritmo de ruta de costo mínimo no uniforme para encontrar la ruta más corta. El tiempo de ejecución de este método se relaciona con los tiempos de iteración en el paso de aprendizaje [25].
- ***Co-superpixel:*** Se generan los superpíxeles a través de la coincidencia de gráficos. Co-superpíxel genera superpíxeles para un par de imágenes, las cuales contienen objetos similares. Primero genera superpíxeles para cada imagen y después los superpíxeles se fusionan. [25].
- ***Saliency-based superpixel:*** Este algoritmo obtiene superpíxeles mediante una estrategia de fusión, primero, los superpíxeles son generados por otra segmentación de superpíxeles, luego los superpíxeles adyacentes se fusionan de acuerdo con la prominencia de la superpíxeles [25].
- ***Structure Sensitive Superpixels:*** Este método genera superpíxeles en dos pasos, primero coloca algunas semillas con una estructura similar al lattice, estas semillas se usan como estimaciones iniciales de centros de superpíxel los cuales en el segundo paso van a ser movidos por criterios de tamaño y forma [25].

1.4. Aprendizaje computacional

El aprendizaje computacional estudia la construcción de sistemas capaces de aprender a partir de datos [28]. Existen 3 tipos de aprendizaje computacional, los modelos de aprendizaje supervisado, los modelos de aprendizaje no supervisado y el aprendizaje por refuerzo [29].

El aprendizaje supervisado busca inducir modelos que puedan predecir el valor de ciertas variables dependientes a partir de variables independientes. En los modelos de aprendizaje no supervisado no existe distinción entre variables dependientes y no dependientes, en este tipo de modelos se busca la estructura subyacente que explique la estructura de los datos. [28]. El aprendizaje por refuerzo se introduce al sistema en un ambiente dinámico,

se proporcionan datos con etiquetas y sin etiquetas, el sistema interactúa con el entorno y recibe retroalimentación de acuerdo con sus acciones, esto va a hacer que se perfeccione, desarrollando mejores caracterizaciones y clasificaciones, se puede decir que la computadora aprende sin recibir instrucciones explícitas [29].

1.4.1. Aplicaciones del aprendizaje computacional en la medicina

Desde el inicio de los años 60, los científicos usaron una computadora como medio de diagnóstico de enfermedades en la sangre. Hoy en día el aprendizaje computacional enseña a los investigadores y médicos nuevas formas de estudiar las enfermedades, crear medicinas y dar tratamiento a los pacientes [30]. Actualmente la presencia de estos métodos de aprendizaje computacional en las distintas áreas de la medicina ha tenido tal impacto que podemos ver aplicaciones de esta en:

- **Reconstrucción de enfermedades.** De acuerdo a Colin Hill, CEO y cofundador de *GNS Healthcare*, la combinación del aprendizaje automático con conjuntos de datos multi-modales permite que los investigadores clínicos puedan reconstruir los mecanismos subyacentes de alguna enfermedad [30].
- **Prueba de hipótesis.** En todas las investigaciones médicas, probar una tesis resulta muy difícil. Sin embargo con ayuda de los modelos estadísticos se pueden revelar ciertas estructuras y predecir resultados [30].
- **Reclutamiento de pacientes.** Los ensayos clínicos son clave en la investigación médica, pero presentan el obstáculo del reclutamiento de pacientes. Un grupo de científicos de la *Commonwealth Scientific and Industrial Research Organisation* en Australia, desarrollaron una técnica de aprendizaje automático que explora registros del paciente con la finalidad de encontrar personas que podrían entrar en determinados ensayos clínicos [30].
- **Big Data.** "En el pasado, un gran conjunto de datos solo significaba muchos pacientes", dice Pearse Keane quien actualmente estudia la degeneración macular relacionada con la edad, causa común de ceguera en Europa, Reino Unido, Estados Unidos entre otros países. Keane menciona, "En la próxima década anticipo que haremos estudios clínicos usando imágenes de cada paciente diagnosticado con degeneración macular relacionada con la edad, tal vez encontrándose con los cientos de miles de pacientes al año [30].

- **Desarrollo de diagnósticos.** Expertos clínicos y en imágenes del Hospital Académico Maxeke de Johannesburgo se asociaron para describir el impacto del aprendizaje automático en la interpretación de imágenes médicas, como radiografías de tórax. Subtipos de aprendizaje computacional como redes neuronales pueden identificar cambios sutiles en radiografías de tórax y, en algunos casos, la precisión es equivalente o superior a la de los médicos, los científicos mencionan que “A diferencia de los métodos estadísticos tradicionales, con el aprendizaje automático los algoritmos imitan la capacidad cognitiva humana en procesos de toma de decisiones”. En abril de 2018, la FDA aprobó el primer sistema de diagnóstico basado en inteligencia artificial, IDx-DR, que detecta retinopatía diabética en personas con diabetes por análisis de imágenes retinianas. El aprendizaje automático puede a corto plazo aplicarse a muchas otras condiciones médicas, desde cardiología hasta enfermedades neurodegenerativas y demás [30].
- **Mejorar los pronósticos.** Además de usarlo para el diagnóstico, los médicos pueden usar el aprendizaje automático para predecir el pronóstico de un paciente. Un ejemplo de la mejora de pronósticos es el que un equipo internacional de científicos creó una herramienta basada en aprendizaje automático para analizar el pronóstico de pacientes con cáncer de colon en etapa III, el grupo informó que los resultados podrían ayudar brindando información para planificar el tratamiento de los pacientes. El presidente de *Mayo Clinic Platform*, John Halamka y sus colegas mencionan que el aprendizaje automático podría mejorar al médico la capacidad para determinar el resultado probable de un paciente con COVID19.
- **Monitoreo de pacientes** Usualmente, los médicos se ponen en contacto con los pacientes una vez que aparecen los síntomas, incluso a veces hasta que la enfermedad crea un problema grave de salud, cosa que va cambiando con el desarrollo y uso creciente de análisis predictivo dice Ali Rezai, el jefe de neurociencia del instituto John D. Rockefeller IV. El aprendizaje automático y los dispositivos *wearables* nos ayudan a monitorear continuamente la salud de una persona, incluso como menciona Rezai, “Dos de los sistemas de IA disponibles comercialmente están incorporados en dispositivos como el *Apple Watch* o los dispositivos *Kardia Alivecor*, dispositivos que pueden detectar arritmias y enviar alertas a los pacientes a través una aplicación en su *smartphone*”. De igual forma, menciona que “Si bien esto no está completamente integrado en la clínica actual, la IA probablemente tendrá un gran impacto en cardiología, cáncer y neurociencias al ayudar a perfilar a los pacientes, permitiendo un cuidado y una

gestión más pro activa" [30].

- **Para colaboraciones.** Quizás más que cualquier otra cosa en medicina, el aprendizaje automático, promete impulsar la colaboración. Maria Littmann, doctora en bioinformática de la Universidad de Munich y sus colegas científicos, descubrieron que de 250 artículos, en el 73 % existen aplicaciones del aprendizaje automático. Las colaboraciones entre investigadores de distintas áreas, se construyen con conjuntos de datos muy grandes. y como dice Hill. "El impacto es claro: el aprendizaje automático impulsado por los datos correctos tiene el poder de transformar el desarrollo de avances, nuevos medicamentos y optimizar su uso en atención al paciente" [30].

1.5. Métodos de aprendizaje automático para el diagnóstico de enfermedades

El trabajo realizado por (Montero, J. et al. 2019) muestra un trabajo que utilizó el software Weka para diagnosticar patologías de la columna vertebral, combinándolo con cinco algoritmos de clasificación diferentes (Logistic Model Tree, Redes Bayesianas, Naive Bayes, J48 y Naive Bayes Simple). Los resultados demostraron que el algoritmo LMT fue el más efectivo, clasificando correctamente 265 de las 310 instancias con un porcentaje de éxito del 85.483 % y un error absoluto de 0.2168. Los otros algoritmos tuvieron porcentajes de éxito y errores absolutos similares (Redes Bayesianas porcentaje de éxito del 80 % y un error absoluto de 0.249, Naive Bayes clasificó correctamente 248 instancias y mostró un error absoluto de 0.2, J48 clasificó correctamente 253 instancias y el algoritmo Naive Bayes Simple clasificó correctamente 241 instancias), pero el LMT fue el que obtuvo los mejores resultados [31].

En (Araujo, A. et al. 2018) se compararon diferentes algoritmos de aprendizaje automático para clasificar quistes mamarios en imágenes termográficas. La base de datos de imágenes fue obtenida en un hospital universitario y está compuesta por imágenes termográficas que fueron clínicamente clasificadas en quistes benignos o malignos. Se utilizaron atributos como los momentos de Zernike y Haralick y los algoritmos evaluados incluyeron redes neuronales artificiales, árboles de decisión, clasificadores bayesianos. Las máquinas de aprendizaje extremo (ELM) y las redes de Perceptrón multicapa (MLP) demostraron ser eficientes clasificadores. Al utilizar el 75 % de la base de datos para entrenar el sistema, se obtuvo una precisión máxima del 73.38 % con un índice Kappa de 0.6007, una sensibilidad del 78 % y una especificidad del 88 %. La eficiencia global del sistema fue del 83 %. En conclusión, ELM demostró ser un clasificador prometedor para la diferenciación de lesiones de mama en

imágenes termográficas debido a su bajo costo computacional y robustez [32].

1.5.1. Diagnóstico de enfermedad de Chagas

El trabajo presentado por Uc-Cetina et al. (2013), utiliza un algoritmo basado en el análisis de discriminante de Gauss, mientras que el trabajo de Soberanis-Mukul et al. (2013) utiliza técnicas de segmentación y clasificación [32]. Ambos trabajos tienen una base de datos con 120 imágenes, 60 de las cuales tienen presencia de parásitos. En ambos algoritmos se preprocesan las imágenes para extraer información relevante, se implementan técnicas de segmentación para separar el objeto de interés del fondo y se implementan clasificadores para determinar si la muestra es un parásito o no. El trabajo de Uc-Cetina utiliza el análisis de componentes principales y reporta tasas de rendimiento de 0.0167 en falsos negativos, 0.1563 en falsos positivos, 0.8437 en verdaderos negativos y 0.9833 en verdaderos positivos. El trabajo realizado por Soberanis-Mukul utiliza un clasificador gaussiano y da como resultado regiones de píxeles de interés que pueden ser un parásito con un 98 % de sensibilidad y 85 % de especificidad. En un trabajo posterior, Soberanis-Mukul et al. (2014) se enfoca en comparar tres clasificadores (Support-Vector Machine (SVM), AdaBoost y redes neuronales artificiales) para detectar y segmentar el parásito *T. cruzi* en imágenes. La metodología consta de tres etapas:

1. Cálculo de superpíxeles,
2. Extracción de características óptimas y
3. Entrenamiento de los clasificadores.

La base de datos utilizada es de 900 imágenes positivas y 900 negativas. Los clasificadores reciben como entrada un vector de características basado en el espacio de color de los superpíxeles. Al final del estudio, se reporta que el clasificador gaussiano tuvo el menor error cuadrático medio, seguido del clasificador SVM y finalmente el clasificador de redes neuronales. El autor concluye que la técnica de superpíxeles mejora la detección del objeto de interés y la separación del fondo [33]. El uso de superpíxeles es de gran importancia ya que presentan características como:

- Ser computacionalmente eficientes, ya que reducen la complejidad de una imagen de cientos de miles de píxeles a solo unos cuantos cientos de superpíxeles.
- Realizar una representación eficiente de la imagen; los superpíxeles pueden relacionarse entre sí dentro de una sección más grande en la imagen.

- Los superpíxeles son significativos perceptualmente; todos los píxeles dentro de un superpíxel son uniformes y similares en características, por ejemplo, en color, textura y posición espacial.
- Debido a que los superpíxeles son resultado de una sobre-segmentación, muchas de las estructuras de la imagen se conservan [34].

Capítulo 2

Planteamiento del problema

El diagnóstico de la enfermedad de Chagas puede ser difícil debido a la falta de síntomas durante las primeras etapas de la infección, así como a la falta de herramientas diagnósticas confiables y accesibles en áreas endémicas, sin dejar de lado que la enfermedad puede ser confundida con otras enfermedades tropicales similares, lo que puede dificultar aún más el diagnóstico.

Por lo tanto, es necesario encontrar una manera precisa, accesible y económica de detectar la presencia del parásito *T. cruzi* en un paciente. Esto es fundamental para garantizar un tratamiento temprano y efectivo de la enfermedad.

El diagnóstico automático de la enfermedad de Chagas es un desafío importante en la medicina, debido a la necesidad de un diagnóstico preciso y temprano para prevenir complicaciones graves. Actualmente, el diagnóstico de la enfermedad de Chagas se basa en pruebas serológicas y/o moleculares, que requieren equipamiento y personal especializado. Además, en muchos casos, estos métodos son invasivos y costosos. Por esta razón, el desarrollo de un método automático y no invasivo para el diagnóstico de la enfermedad de Chagas sería de gran importancia para mejorar la detección temprana y el tratamiento de la enfermedad.

El diagnóstico automático de la enfermedad de Chagas podría ser posible a través de la utilización de técnicas de procesamiento de imágenes médicas para analizar imágenes microscópicas de sangre o tejido. Sin embargo, aún es necesario investigar y desarrollar estos métodos con el fin de lograr un alto nivel de precisión y confiabilidad en el diagnóstico de la enfermedad de Chagas.

2.1. Pregunta de investigación

¿La utilización de superpíxeles para la segmentación de nidos de amastigotes y la implementación de un algoritmo de aprendizaje automático, pueden ser una estrategia efectiva para realizar la detección de nidos de amastigotes y brindar una herramienta adicional para el diagnóstico temprano de la enfermedad de Chagas?

Capítulo 3

Justificación

De acuerdo con la Organización Mundial de la Salud (OMS), la enfermedad de Chagas afecta a un estimado de 6-7 millones de personas, con mayor presencia en América Latina [35]. En América, se registran 30.000 nuevos casos cada año, 12.000 muertes en promedio y aproximadamente 8.600 recién nacidos se infectan durante la gestación. En México, dos terceras partes del territorio pueden ser consideradas en riesgo de transmisión vectorial, es decir, 1.1 millones de personas podrían estar infectados con *Trypanosoma cruzi* y 29.5 millones en riesgo de contraer la infección [36].

La importancia del presente trabajo radica en que de acuerdo con la Organización Panamericana de la Salud (OPS), la enfermedad de Chagas es casi 100 % curable si se trata en sus etapas iniciales con los medicamentos: Benznidazol y Nifurtimox [37], esto hace que el diagnóstico pronto y acertado de los nidos de amastigotes, los cuales se presentan en la etapa temprana de la infección, tome importancia para evitar las consecuencias de una enfermedad de Chagas crónica que presenta cifras según la OMS de que alrededor del 30% de los enfermos crónicos presentan alteraciones cardíacas y hasta un 10% padecen alteraciones digestivas, neurológicas o combinadas [35].

Además de la detección temprana de la enfermedad, con un método de diagnóstico automático de la enfermedad de Chagas se puede tener acceso a comunidades remotas, lugares donde la enfermedad tiene mayor presencia y el acceso a servicios de salud puede ser limitado; y contar con una vigilancia epidemiológica para prevenir y controlar la enfermedad.

Al ser los métodos de microscopia algo que requiere de observación por parte de expertos, la comparación y entrenamiento de un método de aprendizaje computacional para la detección de los nidos de amastigotes, es una opción de bajo costo y buena confiabilidad para la detección temprana de la enfermedad.

Para la ingeniería biomédica, la importancia de contar con un método de diagnóstico

automático para la enfermedad de Chagas radica en la mejora de la precisión y confiabilidad, la automatización y eficiencia, el impulso a la innovación tecnológica y el impacto positivo en la atención médica global. De esta forma la ingeniería biomédica desarrolla soluciones tecnológicas que abordan desafíos de salud pública y se mejora la atención médica en todo el mundo.

Capítulo 4

Hipótesis

Es posible realizar la detección de nidos de amastigotes en imágenes de histopatología chagásica mediante algoritmos basados en superpíxeles y aprendizaje computacional.

Capítulo 5

Objetivos

5.1. Objetivo general

Implementar un algoritmo basado en superpíxeles para la detección de nidos de amastigotes en imágenes de histopatología cardíaca.

5.2. Objetivos específicos

- Implementar el algoritmo SLIC basado en superpíxeles para detectar nidos de amastigotes en imágenes de histopatología cardíaca.
- Efectuar pruebas del algoritmo SLIC en imágenes de histopatología.
- Realizar la detección de los nidos de amastigotes usando un algoritmo de Random Forest.

Capítulo 6

Metodología

La figura 6.1, muestra el diagrama de la metodología empleada para el presente trabajo, la cual está compuesta básicamente por 4 etapas descritas a detalle a continuación.



Figura 6.1: Metodología.

6.1. Base de datos y preprocesamiento

Las 22 microfotografías de cortes histopatológicos de corazón de ratones infectados con *T. Cruzi*, utilizadas en este trabajo, fueron obtenidas del Centro de Investigaciones Regionales Dr. Hideyo Noguchi. Estas imágenes fueron capturadas con un microscopio Olympus CX23 con un aumento de 40X y posteriormente fueron segmentadas manualmente por un experto en histopatología y procesamiento de imágenes, para identificar cada nido de amastigotes

(figura 6.2). La base de datos de imágenes histopatológicas y las segmentaciones expertas serán utilizadas para realizar evaluaciones y comparaciones en el presente estudio.

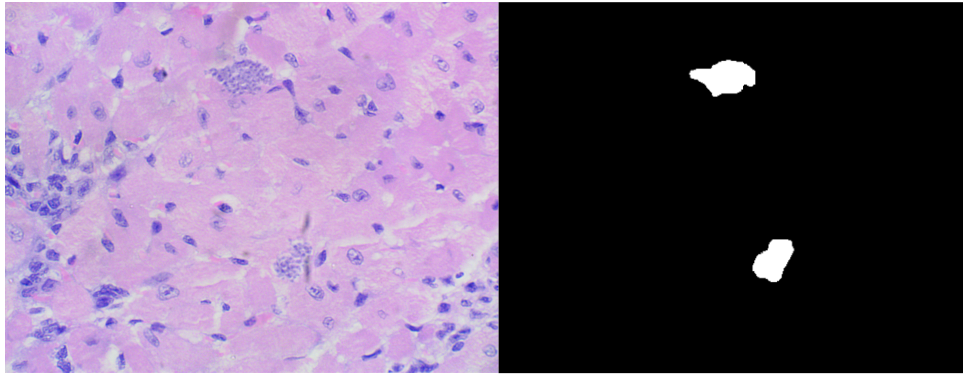


Figura 6.2: A la izquierda, imagen de histopatología de tejido cardíaco de un ratón infectado con *T. cruzi*. A la derecha; segmentación manual de los nidos de amastigotes, realizada por un experto en histopatología. Elaboración propia.

Las 22 microfotografías de cortes histopatológicos, presentaban originalmente las propiedades:

- Tipo de archivo .TIF;
- Dimensión de 2592 x 1944 píxeles;
- Profundidad en bits de 24.

Las máscaras de la segmentación de cada microfotografía, presentaban originalmente las propiedades:

- Tipo de archivo .PNG;
- Dimensión de 512 x 512 píxeles;
- Profundidad en bits de 1.

Para la aplicación de este estudio, se realizó un preprocesamiento de todas las imágenes, tanto las microfotografías como las máscaras fueron modificadas para cumplir con las siguientes características:

- Tipo de archivo .PNG.
- Dimensión de 500 x 375 píxeles.
- Profundidad en bits de 24 para las microfotografías y 8 para las máscaras.

La imagen 'image_histo_01_rs.png' correspondiente a la primer microfotografía, presentaba una profundidad en bits de 32, por lo que se codificó un pequeño algoritmo en Python para cargarlo con tipo de archivo .TIF y convertirlo a las mismas características de las demás imágenes.

La imagen 'image_histo_02_gt.png', correspondiente a la máscara de la segmentación de la segunda microfotografía, presentaba dimensiones de 499 x 374 píxeles, por lo cual, se redimensionó la imagen a 500 x 375 píxeles para tener las imágenes con las mismas dimensiones.

Tras revisar la teoría de los métodos de segmentación, se decidió evaluar cuatro métodos de segmentación de superpíxeles; Felzenszwalb, SLIC, Quickshift y Compact watershed y en base al tiempo de procesamiento, se decidió trabajar con el algoritmo SLIC. Se codificó en Python la segmentación de la 'image_histo_03_rs.png' con los 4 métodos antes mencionados y con base en un análisis del rendimiento y tiempo de procesamiento de cada uno de los métodos mencionados; se determinó que el método SLIC, es el más adecuado para el propósito de la investigación.

6.2. Implementación del algoritmo SLIC

Se cargó la base de datos a Google Drive y en Google Colaboratory, se implementó un algoritmo en lenguaje Python capaz de realizar una segmentación con el método SLIC. Mediante un método de ensayo y error, se determinaron los valores de número de segmentos, compacidad y sigma. El algoritmo debía mostrar el número de segmentos generados y la imagen original con los segmentos superpuestos con colores distintos para poder distinguirlos. Se ejecutó el algoritmo para verificar el correcto funcionamiento para cada una de las 22 microfotografías y se guardaron los resultados obtenidos.

6.3. Extracción de características

Luego de tener el código de segmentación SLIC, se codificó una comparación entre cada segmento obtenido y la máscara correspondiente a la microfotografía. El algoritmo diseñado debía primeramente cargar la imagen de la máscara correspondiente, después aislar y crear una nueva imagen para cada segmento, en esta nueva imagen se mantuvo la posición y color original del segmento, mientras que el resto de la imagen se mostraría de color negro; a continuación cada nueva imagen se convirtió a escala de grises y se multiplicó por la imagen de la máscara, el resultado de la multiplicación se dividió entre el producto de los renglones

por las columnas de la imagen y el resultado de la división se multiplicó por cien, esto para conocer el porcentaje del superpixel que podía contener nidos de amastigotes. Enseguida se calculó para cada superpixel, las métricas;

- Media

$$\bar{x} = \frac{x^1 + x^2 + x^3 + x^n}{n}; \quad (6.1)$$

- Mediana

$$\tilde{x} = x_{\frac{n+1}{2}}; \quad (6.2)$$

- Moda, que devuelve el valor más común de datos discretos o nominales;

- Curtosis

$$k = \frac{\sum (x_i - \bar{x})^4 * f_i}{ns^4} - 3; \quad (6.3)$$

- Oblicuidad '*skewness*'

$$Skewness = \frac{3(\bar{x} - \tilde{x})}{s}; \quad (6.4)$$

- Entropía

$$Entropy = k * \ln(\Omega) \quad (6.5)$$

- Porcentaje de contener un nido de amastigotes en el segmento.

$$\%na = \frac{n_{\neq 0}(S * M)}{renglones * columnas} * 100. \quad (6.6)$$

Donde:

- x = Dato;
- n = Número total de datos;
- f_i = Frecuencia absoluta;
- s = Desviación estándar;
- k = Constante de Boltzmann;
- Ω = Número de microestados que tienen la misma probabilidad de ocurrencia;
- $n_{\neq 0}$ = Número de segmentos distintos de 0;

- S = Matriz del segmento;
- M = Matriz de la máscara.

Además de la obtención de las métricas en el canal de escala de grises, se realizó el procedimiento anterior para los canales de color RGB y HSV y se omitió el cálculo del porcentaje de contener nido de amastigotes para estos 6 canales de color. Las métricas obtenidas de cada una de las 22 microfotografías se almacenaron en una tabla de Excel con sus respectivos identificadores de número de imagen y número de segmento, ordenándolas en orden ascendente para sus dos identificadores. Dicha tabla de Excel, finalmente se convirtió a un archivo de valores separados por comas .csv para facilitar su uso en el algoritmo RF.

6.4. Clasificación y validación

Para la clasificación se utilizó el algoritmo Random Forest (RF) el cual es una técnica de aprendizaje supervisado que se utiliza para generar múltiples árboles de decisión a partir de un conjunto de datos de entrenamiento. Cada árbol se crea a través de dos etapas: primero, se generan un gran número de árboles de decisión utilizando un subconjunto aleatorio de variables m (predictores) de forma que $m < M$, donde M es el número total de predictores. Luego, cada árbol se expande hasta su máxima extensión. El algoritmo RF se puede utilizar para clasificación o predicción. En la clasificación, cada árbol vota por una clase y el resultado final del modelo es la clase con el mayor número de votos de todos los árboles. En la predicción, el resultado final del modelo es el promedio de las salidas de todos los árboles. Además, el RF puede utilizarse para reducción de dimensionalidad ya que puede manejar cientos de predictores sin excluir ninguno, y es capaz de estimar cuáles son los predictores más importantes [38].

El algoritmo RF se alimentó con los descriptores previamente calculados para cada superpixel. Se partió el conjunto de imágenes en datos de entrenamiento $n=16$ imágenes y datos de prueba final $n=6$ imágenes.

A los datos de entrenamiento se les hizo una validación cruzada de 4 vías, la validación cruzada es un método para evaluar modelos en los que se utilizan muchos datos para asegurar la independencia de los resultados del entrenamiento y prueba final. La validación cruzada es ampliamente utilizada en machine learning, especialmente en algoritmos de aprendizaje supervisado y básicamente este método garantiza la independencia de los resultados de la partición inicial de los datos de entrada [39].

El resto de las imágenes se usaron para una prueba final con datos no vistos esto se realizó con la finalidad de evaluar el rendimiento del modelo después de haberlo entrenado. El conjunto de prueba final se utiliza para evaluar la precisión y la capacidad de generalización del modelo Random Forest, y por esta razón es importante que el conjunto de prueba sea independiente del conjunto de entrenamiento, de lo contrario se puede dar lugar a un ajuste excesivo y a estimaciones inexactas del rendimiento del modelo.

La métrica que se utilizó para evaluar el desempeño durante la validación cruzada fue el *accuracy* (ACC). Y durante la prueba final se uso el *accuracy*, *recall*, *precision*, *F1-score* metricas dadas por:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6.7)$$

$$Recall = \frac{TP}{TP + FN} \quad (6.8)$$

$$Precision = \frac{TP}{TP + FP} \quad (6.9)$$

$$F1 - score = 2 * \frac{precision * recall}{precision + recall} \quad (6.10)$$

Donde:

TP= Verdaderos Positivos

TF= Verdaderos Falsos

FP= Falsos Positivos

FN= Falsos Negativos

Además, para cada una de las 6 imágenes de prueba se generó un mapa de calor con las probabilidades post resultado de la clasificación. Estas probabilidades indican el porcentaje de que en ese segmento se encuentre un nido de amastigotes. Finalmente, se hizo un análisis de la importancia de las características mediante la métrica de impureza de Gini, medida que se utiliza para generar árboles de clasificación, esta medida proporciona más información acerca de la distribución de los datos.

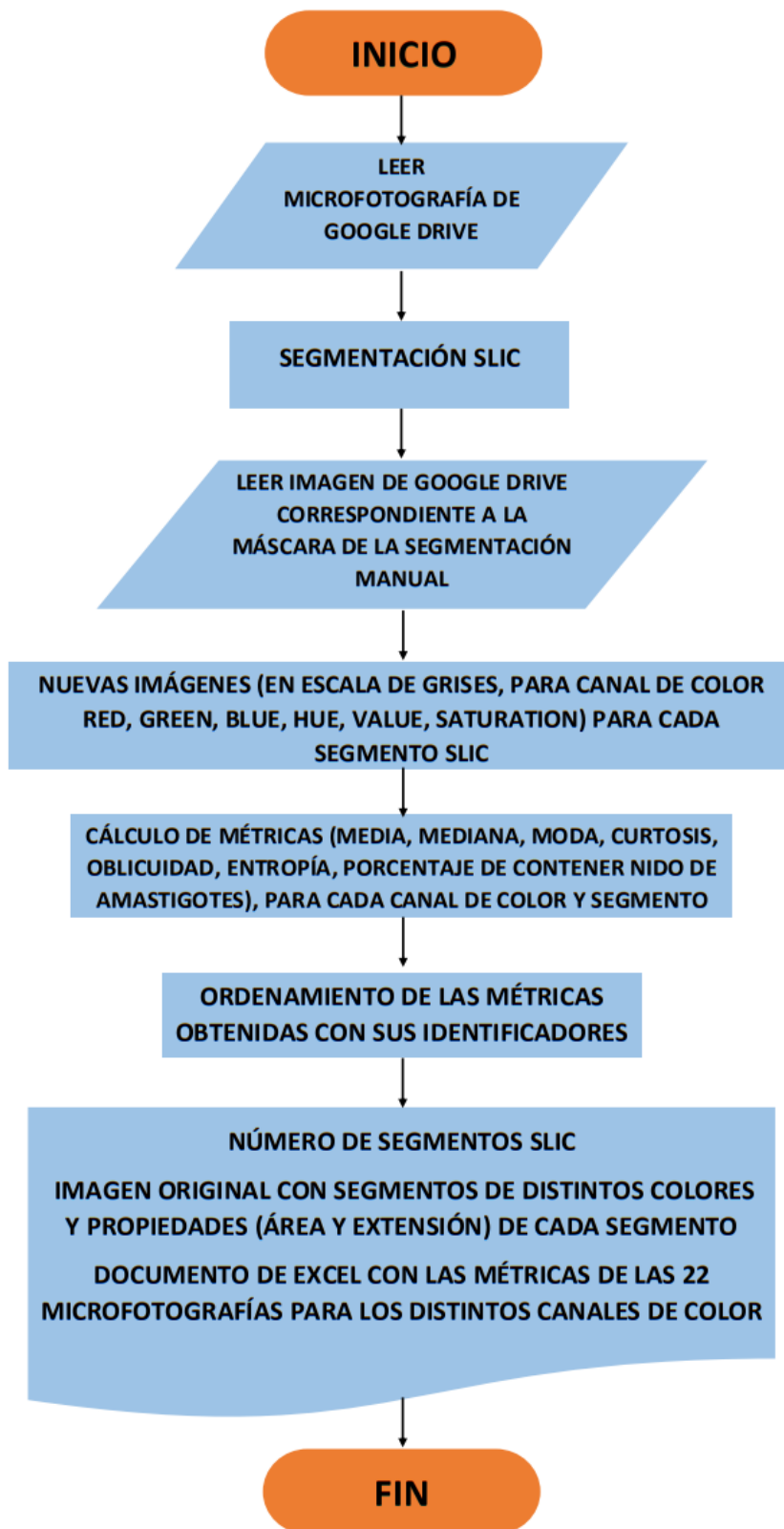


Figura 6.3: Diagrama de flujo de las tareas realizadas en el código de Python. Elaboración propia.

Capítulo 7

Implicaciones Éticas

Las imágenes proporcionadas del modelo experimental de la infección por *T. cruzi* en etapa aguda, así como el protocolo del cual se adquirieron las imágenes de los ratones fueron aprobados por el Comité de Ética del Centro Regional de Investigaciones Dr. Hideyo Noguchi de la Universidad Autónoma de Yucatán (CIR-UADY) con el número de registro CIRB-006-2017.

Capítulo 8

Resultados y Discusión

8.1. Implementación del algoritmo SLIC

La implementación de los algoritmos de segmentación SLIC y de clasificación y validación (Random Forest), se realizaron en Google Colaboratory que está conectado al backend de Google Compute Engine que utiliza Python 3.

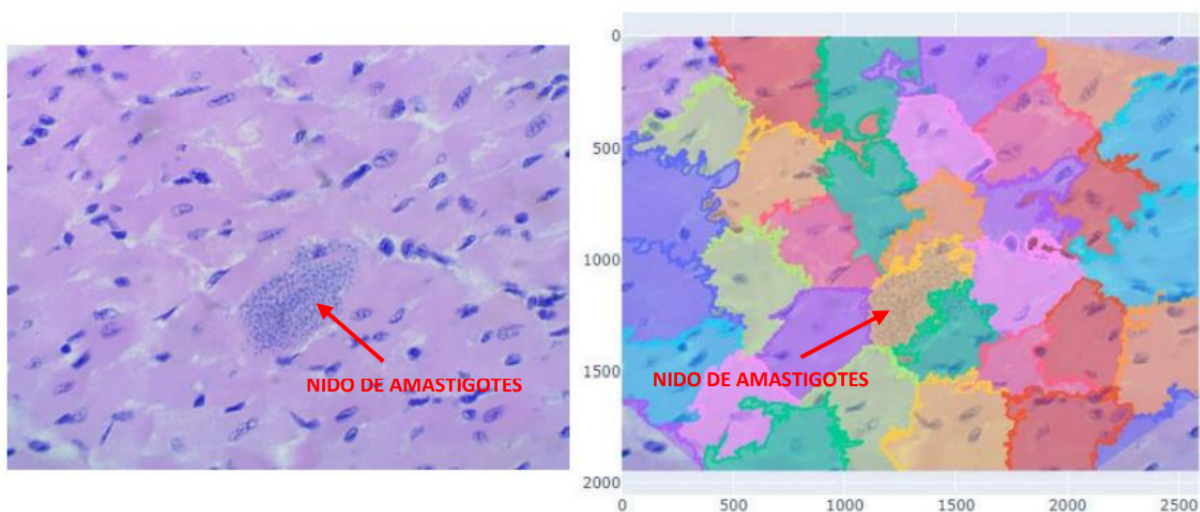


Figura 8.1: Imagen original con el nido de amastigotes señalado con una flecha roja (izquierda). Segmentación SLIC de superpíxeles, con los segmentos de distinto color (derecha).

En la figura 8.1, se observa del lado izquierdo una de las 22 imágenes de corte histopatológico con el nido de amastigotes señalado; mientras que del lado derecho se muestran los segmentos obtenidos con el método SLIC. Para la obtención de las segmentaciones con el método SLIC se ajustaron los siguientes parámetros: **número de segmentos=50, compacidad=10 y sigma=1.**

Los segmentos obtenidos se delimitaron con distintos colores. Del lado derecho de la figura 8.1, se observa un segmento de color naranja, el cual contiene la mayor parte del nido de amastigotes y el resto se encuentra en el segmento adyacente de color azul. En general, la segmentación para las 22 imágenes tuvo un desempeño similar al mostrado en la figura 8.1. Además la velocidad de procesamiento para la segmentación de las 22 imágenes tuvo un tiempo promedio de 10.5 segundos esto es consistente con el trabajo de Hernández Girón et al. (2017), el cual menciona que para la segmentación SLIC de una imagen donde el valor del número de segmentos es bajo, el resultado es una segmentación efectiva en término de tiempo de procesado [34].

8.2. Extracción de características

Para la extracción de características en los distintos canales de color, el tiempo promedio de procesamiento fue de 15.7 segundos para cada una de las 22 imágenes, sin embargo esta etapa tuvo que ser repetida ya que durante el transcurso, la librería de Python *skimage.segmentation*, tuvo una actualización con la cual se modificaron las operaciones que se realizaban para obtener las características, por lo cual se retardó la continuación del proyecto. Las características de los canales que se obtuvieron, se utilizaron en su totalidad para el desarrollo del proyecto.

IMAGEN DE PRUEBA	ACC	PRECISION	RECALL	F1-SCORE
1	92.85%	46.5%	50%	48%
2	93.33%	46.5%	50%	48.5%
3	89.28%	44.5%	50%	47%
4	96.77%	98.5%	75%	82.5%
5	97.43%	48.5%	50%	49.5%
6	96.42%	98%	75%	82.5%
PROMEDIO	94.34%	63.75%	58.33%	59.66%
DESVIACIÓN ESTÁNDAR	2.84	24.42	11.78	16.16

Figura 8.2: Resultados generales de las pruebas.

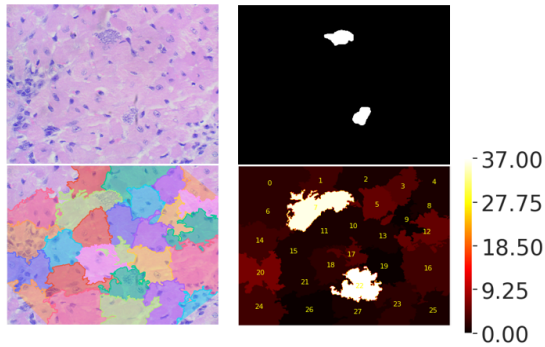
8.3. Clasificación y validación

El entrenamiento del algoritmo Random Forest en su etapa de validación cruzada, nos mostró una métrica *accuracy* de 98.20% y una desviación estándar de $\pm 0,35$. Para la etapa de pruebas finales figura 8.2, el algoritmo mostró un *ACC* promedio de **94.34%**, una *precision* promedio de **63.75%** esto debido principalmente al tamaño del conjunto de datos de entrenamiento, un *recall* promedio de **58.33%** y un *f1-score* promedio de **59.66%**; las métricas *recall* y *f1-score* mostraron valores bajos debido a que la función utilizada en el algoritmo de predicciones no utiliza un umbral de clasificación específico, esto causa que el comportamiento del modelo afecte estas dos métricas.

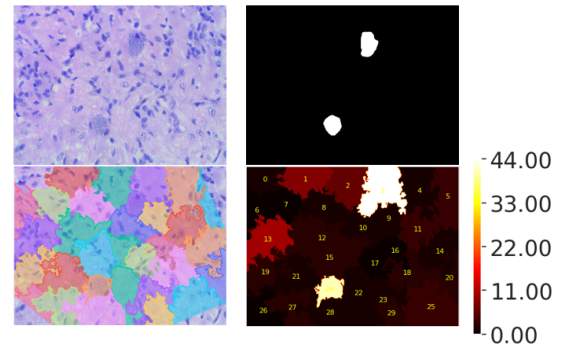
El trabajo de Cafundó et al. (2022) [40], en el cual se utilizaron imágenes de muestras de sangre para hacer detección de **tripomastigotes** mediante algoritmos de clasificación y en el cual en algoritmo RF obtuvo un *ACC* de 99.7% en la etapa de entrenamiento y de 89.5% para la etapa de prueba final; los resultados obtenidos en este trabajo son prometedores, ya que la detección de **nidos de amastigotes**, es una etapa aún más temprana a la de la aparición de tripomastigotes y las métricas obtenidas tanto en la validación cruzada, como en las pruebas finales muestran un gran desempeño.

Rosado Sánchez et al. (2021) [41] realizó detección de nidos de amastigotes haciendo uso de aprendizaje profundo a diferentes umbrales de pertenencia, en dicho estudio su mejor resultado fue con umbral de 50%, el cual mostró un *ACC* de 99.2% y *precisión* de 83.2%. En comparación con el clasificador RF utilizado en este trabajo, las métricas muestran diferencias a favor de la técnica de aprendizaje profundo, sin embargo, si analizamos que para la detección con aprendizaje profundo se utilizaron 11,508 imágenes, mientras que el RF solo 22 y que el algoritmo RF tiene una mayor velocidad de procesamiento que el aprendizaje profundo, podemos decir que los resultados obtenidos son relevantes.

En las figuras 8.3, 8.4 y 8.5, se muestran los resultados de la segmentación SLIC y de la detección de nidos de amastigotes con superpíxeles en las 6 imágenes que se utilizaron para las pruebas finales del algoritmo RF. La segmentación SLIC de la figura 8.4b fue la que tuvo un mejor desempeño, se observa un segmento con buena adherencia a los bordes del nido de amastigotes, observamos este mismo desempeño para el nido inferior de la figura 8.3b. Las segmentaciones de los nidos de amastigotes en las figuras 8.3a, 8.5a y el nido superior de la figura 8.3b, muestran una adherencia parcial al borde de los nidos de amastigotes presentes en ellas, sin embargo el superpíxel no solo abarca los nidos de amastigotes presentes, sino que también contiene píxeles que no pertenecen al nido, por este motivo podemos decir que su desempeño es aceptable. Por último, las segmentaciones de las figuras 8.4a y 8.5b,

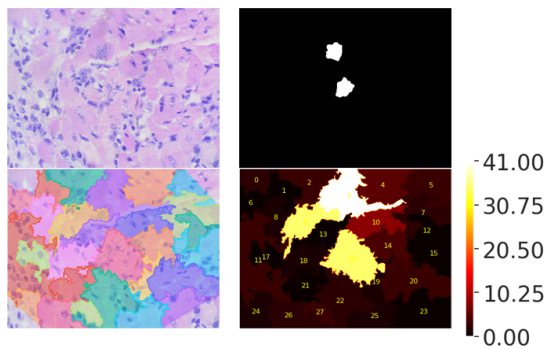


(a) Resultados de la prueba con imagen 1. Imagen original (arriba a la izquierda). Detección de nidos experta (arriba a la derecha). Segmentación SLIC (abajo a la izquierda). Mapa de calor de las probabilidades (abajo a la derecha). Elaboración propia.

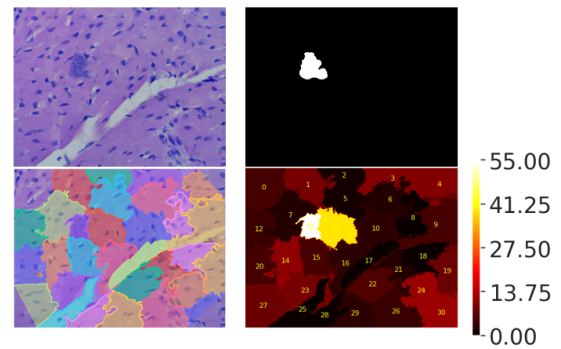


(b) Resultados de la prueba con imagen 2. Imagen original (arriba a la izquierda). Detección de nidos experta (arriba a la derecha). Segmentación SLIC (abajo a la izquierda). Mapa de calor de las probabilidades (abajo a la derecha). Elaboración propia.

Figura 8.3: Resultados de las pruebas con imágenes 1 y 2.

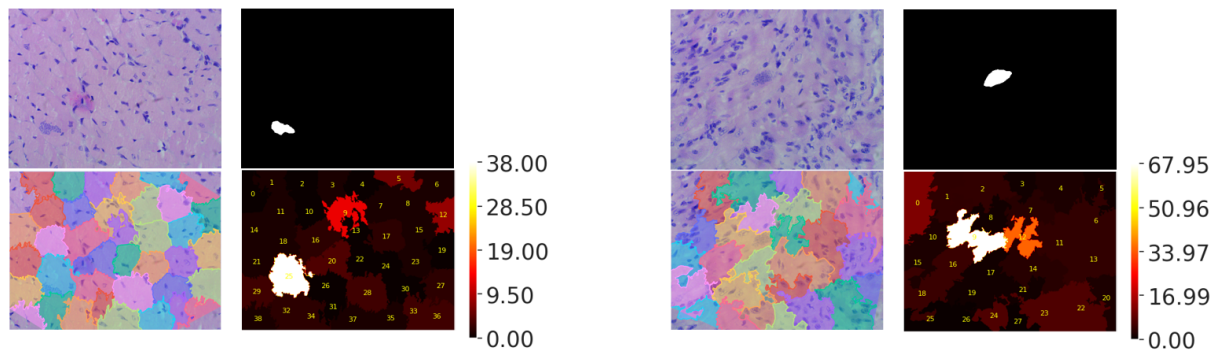


(a) Resultados de la prueba con imagen 3. Imagen original (arriba a la izquierda). Detección de nidos experta (arriba a la derecha). Segmentación SLIC (abajo a la izquierda). Mapa de calor de las probabilidades (abajo a la derecha). Elaboración propia.



(b) Resultados de la prueba con imagen 4. Imagen original (arriba a la izquierda). Detección de nidos experta (arriba a la derecha). Segmentación SLIC (abajo a la izquierda). Mapa de calor de las probabilidades (abajo a la derecha). Elaboración propia.

Figura 8.4: Resultados de las pruebas con imágenes 3 y 4.



(a) Resultados de la prueba con imagen 5. Imagen original (arriba a la izquierda). Detección de nidos experta (arriba a la derecha). Segmentación SLIC (abajo a la izquierda). Mapa de calor de las probabilidades (abajo a la derecha). Elaboración propia.

(b) Resultados de la prueba con imagen 6. Imagen original (arriba a la izquierda). Detección de nidos experta (arriba a la derecha). Segmentación SLIC (abajo a la izquierda). Mapa de calor de las probabilidades (abajo a la derecha). Elaboración propia.

Figura 8.5: Resultados de las pruebas con imágenes 5 y 6.

fueron las de menor calidad, ya que muestran nidos de amastigotes divididos en más de 1 superpixel, con una mala adherencia a los bordes. Las diferencias observadas en el desempeño de la segmentación SLIC concuerdan con lo descrito por Hernández Girón et al. (2017) quien menciona que todos los píxeles dentro de un superpixel son uniformes y similares en características [34].

Los mapas de calor mostrados en las figuras 8.3, 8.4 y 8.5 muestran los superpíxeles con su número de identificación correspondiente, mientras que la probabilidad de que en dicho segmento se encuentre un nido de amastigotes se representa visualmente con un color más oscuro correspondiente al valor 0 y en tonos más claros los superpíxeles en los cuales la probabilidad de observar un nido de amastigotes es mayor y su valor máximo de probabilidad se muestra en la tabla de valores a la derecha del mapa de calor. Se nota que dependiendo de la segmentación SLIC, previamente descrita, se muestran las probabilidades en los superpíxeles que contienen a los nidos de amastigotes y a pesar de que en algunos se distribuye la probabilidad en 2 segmentos para el mismo nido de amastigotes, la diferencia con el resto de segmentos es evidente.

Por último, la tabla de descriptores figura 8.6, muestra las 20 características más influyentes para la clasificación realizada con el algoritmo RF. Podemos observar que los descriptores del canal *grayscale* fueron los de mayor influencia y de estos, la media fue la base del clasificador, por lo cual para futuras investigaciones o trabajos con características de los canales de color se recomienda enfocarse en los descriptores del canal *grayscale*.

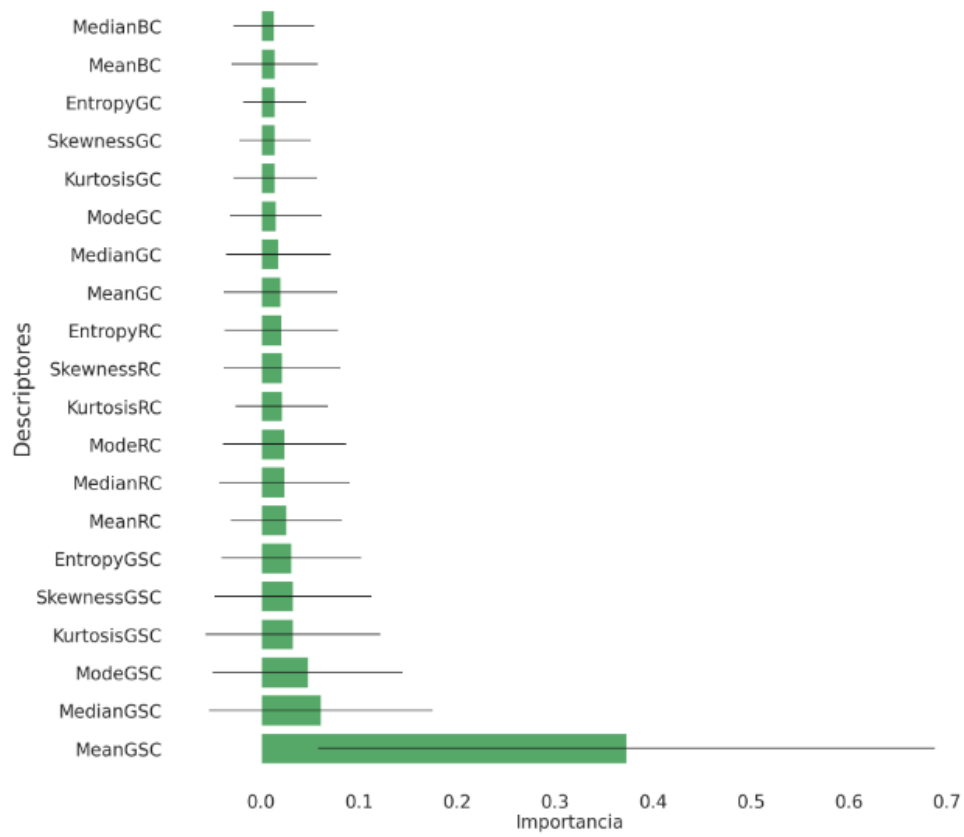


Figura 8.6: Tabla de descriptores.

Capítulo 9

Conclusiones

En el presente trabajo, se ha propuesto un método de detección de nidos de amastigotes basado en superpíxeles SLIC, para imágenes de histopatología de tejido cardíaco de ratones infectados con *T. cruzi*. La detección de nidos de amastigotes está basada en un clasificador *Random Forest*.

Los resultados obtenidos en este trabajo muestran que la implementación de un algoritmo de clasificación RF, es una alternativa para el diagnóstico temprano de la enfermedad de Chagas con bajo costo computacional, además de proporcionar información útil para futuras investigaciones de diagnóstico automático de enfermedades ocasionadas por parásitos.

A pesar de mostrar resultados prometedores, es necesario incluir un mayor número de imágenes, tanto en el entrenamiento, como en las pruebas finales e incluir en estas, imágenes de sujetos sanos para verificar el funcionamiento del método con estas muestras.

En conclusión, los resultados sugieren que la aplicación de métodos de aprendizaje computacional, es una alternativa confiable para la detección de anomalías en imágenes de histopatología que pueden ser útiles para el diagnóstico de enfermedades.

Capítulo 10

Recomendaciones

Con base en la metodología y los resultados del presente trabajo se recomienda:

- Tener un número mayor de imágenes e incluir imágenes de sujetos sanos para evaluar el desempeño del clasificador con estas muestras.
- Identificar los descriptores con mayor importancia, para evitar tener descriptores que no aportan mucho al clasificador.
- Conocer y evaluar las distintas opciones de segmentación de acuerdo a la aplicación.
- Tener presente las actualizaciones de las distintas librerías y entornos de ejecución de los lenguajes de programación.

Bibliografía

- [1] UNAM. La enfermedad de Chagas, silenciosa y potencialmente mortal. Gaceta UNAM. 2019 Sep. Available from: <https://www.gaceta.unam.mx/la-enfermedad-de-chagas-silenciosa-y-potencialmente-mortal/#:~:text=EscausadaporelparÃasito,adecuadoydemaneratemprana.>
- [2] María SSP. Enfermedad de Chagas en México. Revista de la Facultad de Medicina (México). 2016.
- [3] Alqadi Z. introduction to digital images. 2021 01. Available from: [https://www.researchgate.net/publication/348151080_introduction_to_digital_images.](https://www.researchgate.net/publication/348151080_introduction_to_digital_images)
- [4] Nora La Serna Palomino URC. Técnicas de Segmentación en Procesamiento Digital de Imágenes. Revista de ingeniería de sistemas e informática. 2009 Dec;6(2).
- [5] Pérez-Molina JA, Molina I. Chagas disease. The Lancet. 2018 jan;391(10115):82-94.
- [6] Organization WH. Dengue Bulletin. WHO; 2020. Available from: [http://whqlibdoc.who.int/publications/2010/9789241564090_eng.pdf.](http://whqlibdoc.who.int/publications/2010/9789241564090_eng.pdf)
- [7] Echeverria LE, Morillo CA. American Trypanosomiasis (Chagas Disease). Infectious Disease Clinics of North America. 2019 mar;33(1):119-34.
- [8] Pereira Á PM. Tripanosomosis. Enfermedad de Chagas y enfermedad del sueño. ELSEVIER. 2003 Feb;22(2):104-11. Available from: <https://www.elsevier.es/es-revista-offarm-4-articulo-tripanosomosis-enfermedad-chagas-enfermedad-del-130>
- [9] for DC PC, CDC. Chagas Disease - Epidemiology Risk Factors. CDC. 2019. Available from: https://www.cdc.gov/parasites/chagas/gen_info/detailed.html#:~:text=Chagasdiseaseiscausedby,referredtoasAmericantrypanosomiasis.

- [10] Bern C, Kjos S, Yabsley MJ, Montgomery SP. Trypanosoma cruzi and Chagas' Disease in the United States. *Clinical Microbiology Reviews*. 2011 oct;24(4):655-81.
- [11] Rassi A, Rassi A, Marin-Neto JA. Chagas disease. *The Lancet*. 2010 apr;375(9723):1388-402.
- [12] Balouz V, Agüero F, Buscaglia CA. Chagas Disease Diagnostic Applications. In: *Advances in Parasitology*. Elsevier; 2017. p. 1-45.
- [13] Álvarez-Hernández DA, Franyuti-Kelly GA, Díaz-López-Silva R, González-Chávez AM, González-Hermosillo-Cornejo D, Vázquez-López R. Chagas disease: Current perspectives on a forgotten disease. *Revista Médica del Hospital General de México*. 2018 jul;81(3):154-64.
- [14] Choudhury SD. Nano-Medicines a Hope for Chagas Disease! *Frontiers in Molecular Biosciences*. 2021 jun;8.
- [15] Barcia JJ. The Giemsa Stain: Its History and Applications. *International Journal of Surgical Pathology*. 2007 jul;15(3):292-6.
- [16] JH B. *Microorganisms. Theory and Practice of Histological Techniques*. 2008.
- [17] Bezrukov A. Romanowsky staining, the Romanowsky effect and thoughts on the question of scientific priority. *Biotechnic & Histochemistry*. 2017 jan;92(1):29-35.
- [18] Giri D. Wright's Stain : Preparation, Principle, Procedure and Results. *Laboratory Tests*. 2019.
- [19] Molina I, Salvador F, Sánchez-Montalvá A. Actualización en enfermedad de Chagas. *Enfermedades Infecciosas y Microbiología Clínica*. 2016 feb;34(2):132-8.
- [20] Tyagi V. *Understanding Digital Image Processing*. CRC Press; 2018.
- [21] Gonzalez RC WR. *Digital Image Processing*. 3rd ed. Pearson, editor; 2009.
- [22] Sachs J. *Digital Image Basics*. Color DL, editor; 2003. Available from: <https://www.dl-c.com/Temp/downloads/Whitepapers/Basics.pdf>.
- [23] Stutz D LB Hermans A. Superpixels: An evaluation of the state-of-the-art. *Comput Vis Image Underst*. 2018 Jan.

- [24] Achanta R SKLAFPSS Shaji A. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *Journal of Latex class files*. 2011 Dec;6(1).
- [25] Wang M, Liu X, Gao Y, Ma X, Soomro NQ. Superpixel segmentation: A benchmark. *Signal Processing: Image Communication*. 2017 aug;56:28-39.
- [26] Chen Z, Guo B, Li C, Liu H. Review on Superpixel Generation Algorithms Based on Clustering. In: 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE). IEEE; 2020. .
- [27] Wang J, Ju L, Wang X. An Edge-Weighted Centroidal Voronoi Tessellation Model for Image Segmentation. *IEEE Transactions on Image Processing*. 2009 aug;18(8):1844-58.
- [28] González FA. Modelos de aprendizaje computacional en reumatología. *Revista Colombiana de Reumatología*. 2015 jun;22(2):77-8.
- [29] Vega MÁ, Mora LMQ, Badilla MVC. Inteligencia artificial y aprendizaje automático en medicina. *Revista Medica Sinergia*. 2020 aug;5(8):e557.
- [30] May M. Eight ways machine learning is assisting medicine. *Nature Medicine*. 2021 jan;27(1):2-3.
- [31] de Jesús Montero José Carlos. Algoritmos de aprendizaje automático de vanguardia para el diagnóstico de enfermedades. Instituto Tecnológico Superior de Misantla. 2019.
- [32] de Santana Maíra A. Breast cancer diagnosis based on mammary thermography and extreme learning Machines. Scielo. 2018.
- [33] Allan OP. Revisión de métodos de aprendizaje automático para detectar al parásito de la enfermedad de Chagas. *Revista Investigación y Ciencias*. 2020.
- [34] Alan HG. Implementación de algoritmo de superpíxeles para la segmentación de imágenes a color. *Boletín UPIITA-IPN*. 2017.
- [35] OMS. La enfermedad de Chagas (tripanosomiasis americana). Organización Mundial de la Salud. 2021.
- [36] Salazar-Schettino PM CBMAAMdCSDZGE Bucio-Torres MI. Enfermedad de Chagas en México. *Revista de la Facultad de Medicina de la UNAM*. 2016.
- [37] OMS. Enfermedad de Chagas. Available from: <https://www.paho.org/es/temas/enfermedad-chagas>.

- [38] Jesús EZJ. Aplicación de algoritmos Random Forest y XGBoost en una base de solicitudes de tarjetas de crédito. Scielo. 2020.
- [39] de León Marta GR. Análisis de sensibilidad mediante Random Forest. 2018.
- [40] César CMM. Automatic detection of the parasite Trypanosoma cruzi in blood smears using a machine learning approach applied to mobile phone images. Pubmed. 2022.
- [41] Jesús RSE. Análisis de Textura y Aprendizaje Computacional a partir de Microfotografía Chagásica. Memorias CNIB. 2021.