

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

MICROSCOPIO AUTOMATIZADO: CONTEO DE BACILOS DE TUBERCULOSIS

Tesis para optar el Título de Ingeniero Electrónico, que presenta el bachiller:

Juan José Sato Yamada

ASESORES: Dr. Ing. Benjamín Castañeda Aphan – Ing. Kristians Díaz Rojas

Lima, agosto del 2013

RESUMEN

La prueba bacilosκόpica de la tuberculosis es la forma de diagnóstico microscópico más utilizado para combatir la enfermedad en los países pobres o subdesarrollados debido a su bajo costo y rapidez. Sin embargo, la realización de esta prueba es un proceso tedioso, extenuante y requiere de un especialista debidamente capacitado. Por ello, en el presente trabajo se presenta un algoritmo automatizado para la detección y conteo de bacilos de tuberculosis presentes en imágenes de muestras de esputo mediante la utilización de técnicas de procesamiento de imágenes digitales. Se analizaron diferentes espacios de color para hallar aquella capa o canal de color que posea un mayor contraste entre las intensidades de color de los píxeles de los bacilos y del fondo. Para esto se hizo un análisis de los histogramas mediante las gráficas de las características operativas del receptor. Para la segmentación de los bacilos, el presente trabajo desarrolló una técnica de umbralización adaptativa utilizando el método de Otsu para hallar el óptimo valor umbral. Luego, los objetos detectados son clasificados como bacilos o no-bacilos mediante un árbol de clasificación utilizando características de área y excentricidad. El algoritmo desarrollado presenta niveles de sensibilidad, especificidad y exactitud mayores a 90% y tiene un tiempo de ejecución de aproximadamente 9 segundos por campo (15 minutos para 100 campos). Cabe resaltar que, a diferencia de investigaciones previas, la presente tesis buscó desarrollar un algoritmo tanto de segmentación de los bacilos, como de su clasificación, e implementarlo en un microscopio automatizado para el diagnóstico automático de la enfermedad en tiempo real. Con esta finalidad, se implementó el algoritmo desarrollado con el programa Matlab® en un lenguaje de programación C++, obteniendo un programa capaz de interactuar con otros programas como el del control de la cámara digital. Se espera que este trabajo sirva de base para próximos estudios orientados a automatizar el proceso de diagnóstico de la enfermedad de una manera más óptima y veloz.

AGRADECIMIENTOS

Deseo agradecer y dedicar este trabajo a mis padres por haberme dado todo lo que tengo y hacer un gran esfuerzo para darme la oportunidad de estudiar en esta gran universidad. Todo lo que soy es gracias a ustedes y siempre espero que se sientan orgullosos de mí.

A mis compañeros de grupo del curso de Proyecto electrónico - Tesis, Igor Anyosa, Manuel Flores, Michael García y Miguel Veliz, por su amistad, apoyo y trabajo en equipo. A pesar de que no conseguimos alcanzar todos los objetivos propuestos, sé que todos sacrificamos muchas cosas para hacer nuestro trabajo y estoy seguro de que será de mucha utilidad para futuras investigaciones.

A mis asesores y profesores, el Dr. Ing. Benjamín Castañeda, el Ing. Kristians Díaz y el Ing. Ricardo Sánchez, por su apoyo, guía y asesoramiento. Sin su ayuda el proyecto nunca hubiese llegado tan lejos ni hubiésemos aprendido tanto.

A la Dra. Luz Huaroto y al Lic. Julio Tanta del Hospital Nacional Dos de Mayo, por brindarnos su tiempo y enseñarnos el proceso de diagnóstico de la tuberculosis.

Al Laboratorio de Imágenes Médicas de nuestra casa de estudios por permitirnos trabajar en sus ambientes y utilizar sus equipos.

Por último, quiero agradecer a los Proyectos DGI 2010-0081 (Diagnóstico Automatizado de Tuberculosis) y 2010-0105 (Mejoras en el tratamiento de Leishmaniasis Cutánea), por el financiamiento de los equipos y materiales utilizados para el desarrollo de nuestra investigación.

ÍNDICE

INTRODUCCIÓN.....	1
CAPÍTULO 1: CAPÍTULO 1: LA TUBERCULOSIS Y EL PROCESAMIENTO DE IMÁGENES DIGITALES	
1.1 TUBERCULOSIS.....	2
1.2 DIAGNÓSTICO DE LA TUBERCULOSIS.....	4
1.3 DEFINICIÓN DE LA PROBLEMÁTICA.....	7
1.4 PROCESAMIENTO DE IMÁGENES DIGITALES	8
1.5 ESTADO DEL ARTE	9
1.5.1 Presentación del asunto de estudio.....	9
1.5.2 Estado de la investigación	9
CAPÍTULO 2: CONOCIMIENTOS TEÓRICOS SOBRE PROCESAMIENTO DE IMÁGENES DIGITALES	
2.1 IMAGEN DIGITAL	21
2.2 ESPACIO DE COLOR.....	21
2.3 DETECTOR DE BORDE	24
2.3.1 Detector de borde de Roberts.....	25
2.3.2 Detector de borde de Prewitt	25
2.3.3 Detector de borde de Sobel	25
2.3.4 Detector de borde de Canny.....	26
2.4 OPERACIONES MORFOLÓGICAS	26
2.4.1 Dilatación.....	27
2.4.2 Erosión	27
2.4.3 Apertura	27
2.4.4 Cierre	27
2.5 DESCRIPTORES DE FORMA.....	28
2.5.1 Descriptores de Fourier	28
2.5.2 Momentos.....	29
2.5.3 Atributos geométricos.....	31

CAPÍTULO 3: DISEÑO E IMPLEMENTACIÓN	
3.1	OBJETIVOS 32
3.1.1	Objetivo general 32
3.1.2	Objetivos específicos..... 32
3.2	MATERIALES..... 32
3.3	ANÁLISIS DE ESPACIOS DE COLOR..... 35
3.4	DETECCIÓN DE BORDES Y OPERACIONES MORFOLÓGICAS..... 39
3.5	UMBRALIZACIÓN ADAPTATIVA 40
3.6	EXTRACCIÓN DE CARACTERÍSTICAS Y CLASIFICACIÓN..... 43
3.7	IMPLEMENTACIÓN DEL ALGORITMO EN LENGUAJE C++ 47
CAPÍTULO 4: PRUEBAS Y RESULTADOS	
4.1	ALGORITMO EN FUNCIONAMIENTO..... 49
4.2	PRUEBAS 49
4.3	RESULTADOS 53
CONCLUSIONES..... 55	
RECOMENDACIONES..... 56	
BIBLIOGRAFÍA..... 57	

INTRODUCCIÓN

La Tuberculosis es una enfermedad bacteriana infecto-contagiosa que afecta principalmente a los países subdesarrollados como el Perú. La prueba más utilizada para realizar el diagnóstico de la tuberculosis es la prueba de baciloscopia. Este método consiste en analizar una muestra de esputo del paciente y efectuar un conteo de los bacilos presentes en ella. Uno de los problemas más graves en los países subdesarrollados o zonas rurales es que presentan una ausencia o un número muy reducido de especialistas adecuadamente capacitados para realizar el diagnóstico de esta enfermedad. A esto se suma la reducción de la precisión del diagnóstico por el cansancio que provoca analizar una gran cantidad de muestras al día y la variabilidad de los resultados debido al criterio y experiencia del laboratorista.

Con la finalidad de solucionar este problema, el presente trabajo logró diseñar e implementar un algoritmo capaz de realizar un diagnóstico automático de la tuberculosis mediante la utilización de técnicas de procesamiento de imágenes digitales, tales como la selección del espacio de color más adecuado, detección de bordes, umbralización, operaciones morfológicas, selección de descriptores de forma e implementación de un sistema de clasificación.

La implementación de este algoritmo en un equipo médico permitiría automatizar el proceso de diagnóstico de la enfermedad y podría eliminar la necesidad de contar con un especialista en aquellas zonas rurales que no cuentan con la cantidad adecuada de personal para combatir la enfermedad.

En el Capítulo 1 se muestra la problemática del diagnóstico de la tuberculosis y se explica cómo el procesamiento de imágenes digitales puede ayudar a solucionar este problema. En el Capítulo 2 se definen algunos conceptos teóricos importantes sobre el procesamiento de imágenes digitales y que serán de mucha utilidad para comprender el siguiente capítulo. En el Capítulo 3 se explican las técnicas utilizadas para desarrollar e implementar el algoritmo de detección y conteo de bacilos de tuberculosis presentes en imágenes digitales de los campos de las muestras de esputo. Finalmente, en el Capítulo 4 se muestran las pruebas y resultados obtenidos con el algoritmo implementado tanto en Matlab[®], como en el lenguaje de programación C++.

CAPÍTULO 1

LA TUBERCULOSIS Y EL PROCESAMIENTO DE IMÁGENES DIGITALES

1.1 TUBERCULOSIS

La tuberculosis o TBC es una enfermedad bacteriana infecto-contagiosa causada por el bacilo *Mycobacterium tuberculosis*, conocido también como bacilo de Koch. Afecta principalmente a los pulmones (TBC pulmonar), pero puede afectar a otros órganos o tejidos (TBC extrapulmonar) como la piel, los huesos, los riñones y el cerebro [1].

El bacilo de tuberculosis se propaga a través del aire cuando personas infectadas expelen la bacteria al toser, estornudar, hablar o escupir. La infección se produce cuando se inhalan estas bacterias que pueden permanecer en el aire hasta por varias horas. Después de la infección las bacterias se multiplican lentamente, al cabo de uno a dos meses se producen las primeras lesiones, es en ese momento en el cual el sistema inmunológico se activa para combatir las bacterias. En el 90% de los casos la situación es estabilizada y la tuberculosis se puede mantener inactiva o latente hasta por varios años, en este estado la persona infectada no presenta síntomas ni tampoco puede contagiar la enfermedad. Por otro lado, si el sistema inmunológico no logra controlar la infección, se deriva a la tuberculosis activa [2]. Los síntomas de la tuberculosis pulmonar son: tos y expectoración por más de 15 días, debilidad, cansancio, fiebre, pérdida de peso, sudoraciones nocturnas, dolor en el pecho o de espalda, dificultad respiratoria y hemoptisis (expectoración de sangre) [3].

La tuberculosis es una enfermedad que afecta principalmente a países pobres y en desarrollo, ya que su fácil propagación combinado con una baja defensa inmunológica debido a la mala alimentación y sumado a malos hábitos de higiene, problemas de saneamiento y un mal sistema de salud, permiten que la enfermedad se propague con mucha mayor facilidad y que no sea tratada de manera oportuna y adecuada.

En 1993, la Organización Mundial de la Salud (OMS) declaró a la tuberculosis una emergencia de salud pública global. En esa época se presentaron aproximadamente de 7 a 8 millones de casos y de 1.3 a 1.6 millones de muertes por año debido a esta enfermedad. En el 2011, según el último reporte presentado por la OMS, se estimaron aproximadamente 9 millones de casos nuevos y 1.4 millones de muertes. La tuberculosis es considerada la segunda mayor causa de muerte provocada por una enfermedad contagiosa a nivel mundial y se estima que un tercio de la población mundial está infectada.

Durante las últimas décadas el Ministerio de Salud ha implementado una de las mejores estrategias para la prevención y control de la tuberculosis, reconocida y premiada a nivel internacional por la OMS, lo que ha permitido reducir de manera significativa la tasa de incidencia y mortalidad debido a la tuberculosis en nuestro país [3], (ver Figuras 1.1 y 1.2).

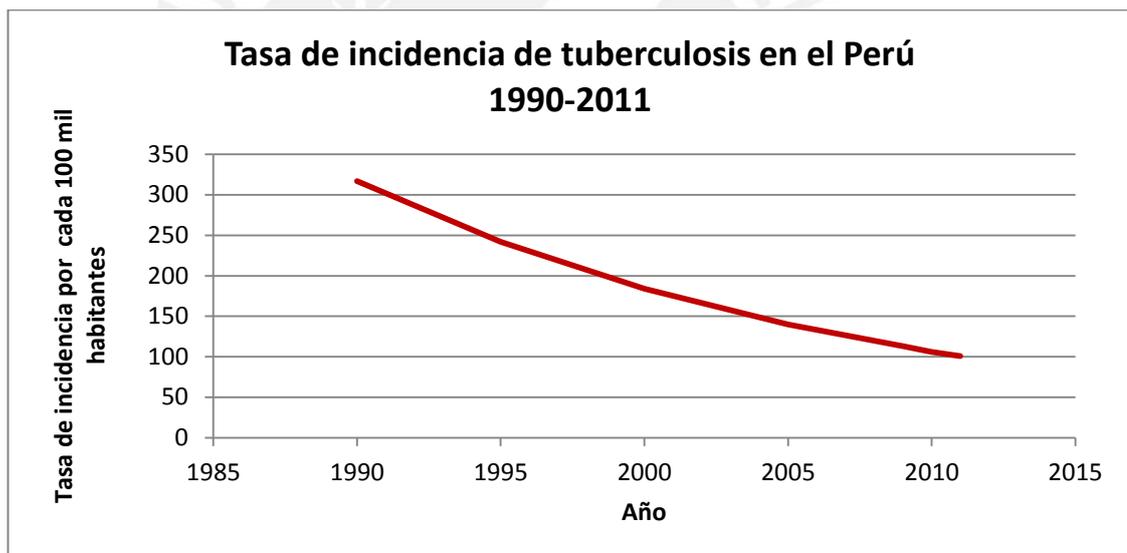


Figura 1.1: Tasa de incidencia de la tuberculosis en el Perú. Elaboración propia, datos extraídos de [1].

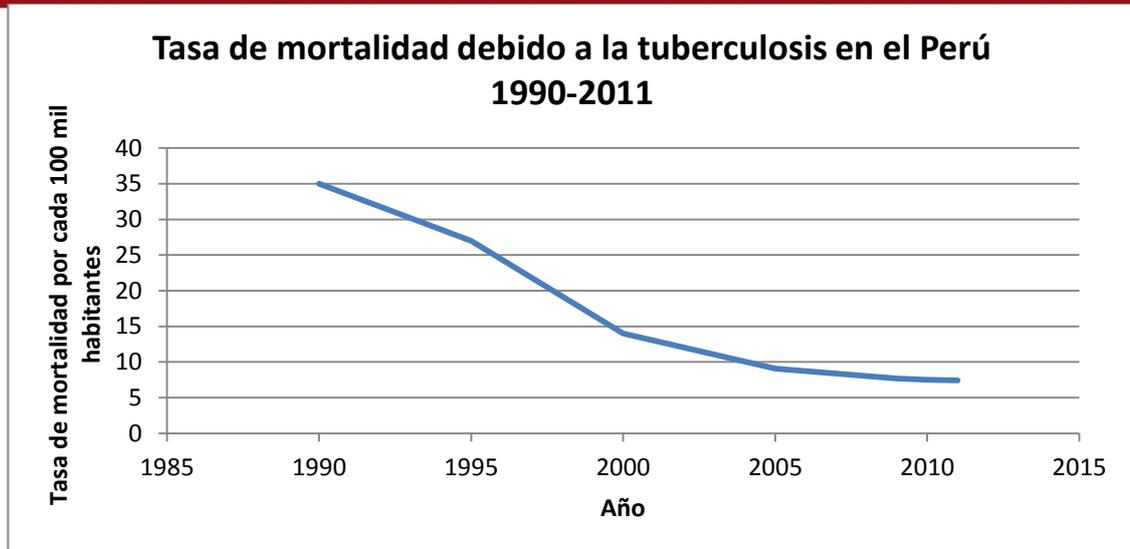


Figura 1.2: Tasa de mortalidad de la tuberculosis en el Perú. Elaboración propia, datos extraídos de [1].

Sin embargo, a pesar de los esfuerzos realizados por el Ministerio de Salud, el Perú sigue presentando una de las mayores tasas de incidencia y mortalidad de América. En el año 2011, se estimaron unos 30 mil casos nuevos y 2.2 mil muertes debido a esta enfermedad [1]. Por ello, la tuberculosis sigue siendo considerada un problema de salud pública en nuestro país. La concientización de la población y la capacitación y entrenamiento constante del personal de salud desempeñan un rol gravitante para garantizar el control de la enfermedad.

1.2 DIAGNÓSTICO DE LA TUBERCULOSIS

Existen diferentes procesos para el diagnóstico de la tuberculosis. Estos procesos varían dependiendo de su grado de sensibilidad y especificidad, costo y velocidad. El examen bacteriológico (baciloscopia) es el más utilizado para el diagnóstico de la tuberculosis debido a su bajo costo y rapidez [4].

La prueba de baciloscopia consiste en el conteo de bacilos presentes en una muestra de esputo a través de un microscopio, y dependiendo del número de bacilos se determina si el paciente presenta o no la enfermedad y su grado de infección y contagio. Para ello, las muestras de esputo pasan por un proceso de tinción mediante el cual se aprovecha la propiedad de la ácido-alcohol resistencia de las micobacterias para captar en su pared sustancias colorantes y retenerlas aun ante la acción de decolorantes como la mezcla de ácido y alcohol [4].

Existen dos tipos de microscopías para el diagnóstico baciloscópico de la tuberculosis: La microscopía de fluorescencia y la microscopía convencional. La microscopía de fluorescencia utiliza un microscopio de fluorescencia para analizar las muestras teñidas con la sustancia Auramina O para que el bacilo tome un color amarillo fluorescente (ver Figura 1.3). Por otro lado, La microscopía convencional utiliza un microscopio óptico convencional y la técnica de tinción de Ziehl-Neelsen, en la cual se aplica la fucsina fenicada para que se adhiera a las paredes de los bacilos, dándole un color fucsia, y además se le añade azul de metileno para dar una coloración celeste o azul al fondo, creando un mejor contraste entre el bacilo y el fondo [2], (ver Figura 1.4).

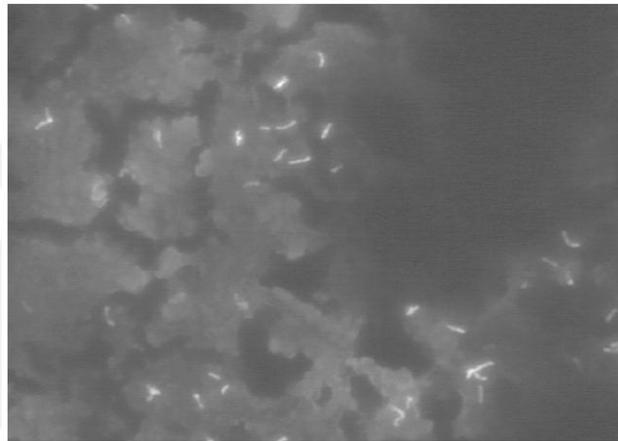


Figura 1.3: Tinción con Auramina O. Los bacilos toman una coloración fluorescente que se aprecia por la utilización de un microscopio de fluorescencia. Extraído de [5].

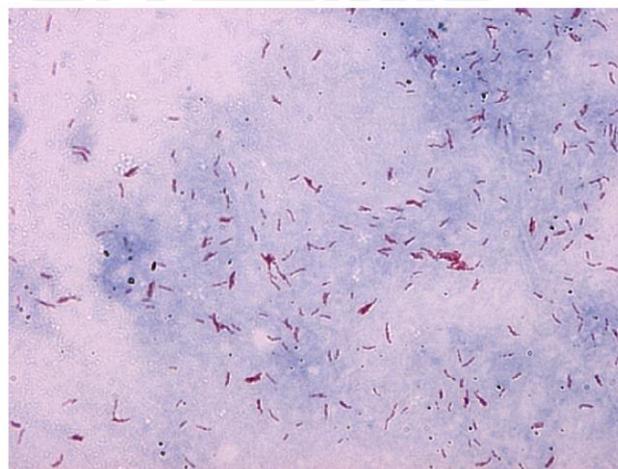


Figura 1.4: Tinción de Ziehl-Neelsen. Los bacilos adquieren una coloración fucsia debido a la fucsina fenicada y el fondo azulado debido al azul de metileno. Extraído de [5].

La microscopía de fluorescencia es en promedio 10% más sensible que la microscopía convencional [6], y utiliza un lente de aumento más bajo (típicamente 25x) que la microscopía convencional (típicamente 100x) [7]. Sin embargo, el costo relativamente alto de la unidad de microscopía y su mantenimiento comparado con el de la unidad de microscopía convencional [8], hacen que la tinción de Ziehl-Neelsen sea la técnica más apropiada para ser utilizada en los laboratorios de los países pobres o en desarrollo como el Perú, y es la recomendada por la OMS y la Unión Internacional Contra la Tuberculosis y Enfermedades Respiratorias [4].

Por otro lado, la confiabilidad de una prueba es expresada en términos de sensibilidad y especificidad. La sensibilidad es la probabilidad de clasificar un resultado como positivo cuando en realidad es positivo. La especificidad es la probabilidad de clasificar un resultado como negativo cuando en realidad es negativo [9]. La sensibilidad de la prueba de baciloscopía para la tuberculosis reportado en los estudios es muy variable: de 32% a 94% con la microscopía convencional y de 52% a 97% con la microscopía de fluorescencia. En promedio la especificidad de la prueba es similar para ambos tipos de microscopía y se encuentra en el rango de 94% a 100% [6]. Además, podemos definir la exactitud de una prueba como la proporción de resultados correctos sobre el total de resultados. Todas estas proporciones se hallan con el número de verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos [5].

Tabla 1.1: Resultado esperado vs Resultado de la prueba. Extraído de [5].

		Resultado esperado	
		Positivo	Negativo
Resultado de la prueba	Positivo	Verdadero Positivo (VP)	Falso Positivo (FP)
	Negativo	Falso Negativo (FN)	Verdadero Negativo (VN)

Para el caso del diagnóstico de la tuberculosis, un resultado es clasificado dependiendo de si la prueba realizada se ha clasificado correctamente o no. Tal como muestra la Tabla 1.1, el verdadero positivo (VP) es el número de muestras positivas clasificadas correctamente, el verdadero negativo (VN) es el número de muestras negativas clasificadas correctamente, el falso positivo (FP) es la cantidad de muestras negativas clasificadas como positivas, y por último, el falso negativo (FN) es la cantidad de muestras positivas clasificadas como negativas. De esta manera, utilizando las siguientes ecuaciones, podemos hallar los valores de sensibilidad, especificidad y exactitud [5]:

$$\text{Sensibilidad} = \frac{VP}{VP + FN} \quad (1.1)$$

$$\text{Especificidad} = \frac{VN}{VN + FP} \quad (1.2)$$

$$\text{Exactitud} = \frac{VP + VN}{VP + VN + FP + FN} \quad (1.3)$$

Cabe resaltar que el resultado de las pruebas puede ser medida a nivel de muestras o de objetos detectados. Además, para el caso del diagnóstico de la tuberculosis, el parámetro más importante a tomar en cuenta es la sensibilidad dado que es más importante diagnosticar acertadamente un caso positivo que uno negativo, ya que un paciente infectado necesita tratamiento oportuno, en cambio a un paciente sano las medicinas no le afectarían negativamente.

1.3 DEFINICIÓN DE LA PROBLEMÁTICA

En el Perú, la técnica más utilizada para el diagnóstico de la tuberculosis es la prueba de baciloscopia debido a su bajo costo y rapidez en comparación a otras técnicas de diagnóstico. Sin embargo cada prueba puede tomar entre 40 minutos a 3 horas dependiendo del grado de infección del paciente [10]. Además, la baciloscopia es considerada una herramienta de diagnóstico muy variable y de baja sensibilidad debido a que depende mucho de la experiencia y criterio del laboratorista para poder discriminar correcta y rápidamente los bacilos. A todo esto se le suma el cansancio o la fatiga visual provocada por la gran cantidad de muestras que debe analizar, conllevándolo a cometer errores en el proceso. Cabe mencionar que por norma existe una cantidad diaria límite de baciloscopías que puede realizar una persona (30 baciloscopías), pero lamentablemente, debido a la falta de personal, los encargados realizan más pruebas de las que indica la norma.

El tipo de microscopía más utilizado para la prueba de baciloscopia en nuestro país es la microscopía convencional junto con la técnica de tinción de Ziehl-Neelsen, debido a su bajo costo y facilidad de mantenimiento de los equipos comparado con la microscopía de fluorescencia, a pesar de que este último posee una mayor sensibilidad debido a que presenta un mejor contraste entre el fondo y el bacilo. Uno de los problemas en la preparación de la muestra de esputo, y que se

evidencia más en la tinción de Ziehl-Neelsen, es que si no se realiza un procedimiento adecuado de limpieza de los instrumentos, de extendido y de aplicación de las sustancias, y si además se utilizan materiales de baja calidad, conllevan a que se presenten manchas o restos indeseados en las muestras que pueden confundir al laboratorista.

En nuestro país, las zonas rurales o más alejadas son las más afectadas por la tuberculosis, ya que no suelen contar con el personal capacitado, ni con los materiales, ni recursos necesarios para poder realizar diagnósticos oportunos y acertados de la enfermedad [3].

1.4 PROCESAMIENTO DE IMÁGENES DIGITALES

Uno de los campos del procesamiento de imágenes digitales más aplicado y avanzado es la Visión Artificial o Visión por Computadora, la cual tiene como finalidad utilizar una computadora para emular la visión humana, incluyendo el poder de aprendizaje y la capacidad de realizar inferencias y acciones basadas en una entrada visual. Para ello, la visión por computadora hace uso de un sistema de reconocimiento de patrones, el cual permite clasificar objetos presentes en una imagen en diferentes clases establecidas convenientemente según el propósito [11]. De esta manera es posible discriminar diferentes formas de objetos y clasificarlos.

Las etapas de un sistema de reconocimiento de patrones son: adquisición, procesamiento, extracción de características y clasificación [12].

- Adquisición: Consiste en la obtención de imágenes digitales a partir de señales analógicas.
- Procesamiento: Consiste en la aplicación de filtros para mejorar el contraste de las imágenes y minimizar el ruido presente en ellas. En esta etapa se realiza la detección de los objetos y la segmentación de la imagen [11].
- Extracción de características: Es un procesamiento de alto nivel. Consiste en la representación matemática de las características de los objetos presentes en la imagen [13].
- Clasificación: Consiste en la aplicación de una función de decisión para determinar la clase de un objeto desconocido [13].

Una de las aplicaciones más importantes del reconocimiento de patrones y del procesamiento de imágenes en general, es el Diagnóstico Asistido por Computadora, que tiene como objetivo asistir a doctores y laboratoristas en la toma de decisiones diagnósticas. El diagnóstico asistido por computadora es aplicado en una gran variedad de imágenes médicas, como los rayos X, imágenes de tomografías computarizadas, imágenes de ultrasonido, electrocardiogramas y electroencefalogramas. La necesidad de un diagnóstico asistido por computadora se basa en el hecho de que las imágenes médicas en muchos casos no son fácilmente interpretables, y la interpretación puede depender en gran medida en la habilidad y experiencia del doctor o laboratorista [14]. De esta manera, si se contaran con imágenes digitales de las vistas microscópicas utilizadas para el diagnóstico de la tuberculosis, es posible implementar un algoritmo basado en procesamiento de imágenes digitales capaz de diagnosticar de manera automática la enfermedad.

1.5 ESTADO DEL ARTE

1.5.1 Presentación del asunto de estudio

La aplicación de los diferentes avances de la ingeniería en la medicina no solo ha logrado desarrollar tecnologías capaces de mejorar la calidad de vida de las personas, sino que permite que estas solucionen problemas que afectan gravemente a las personas de zonas pobres o poco desarrolladas. En ese sentido, la creación de un equipo médico con algoritmos basados en procesamiento de imágenes digitales para el diagnóstico asistido por computadora de la tuberculosis lograría automatizar el proceso de detección y conteo de los bacilos en una muestra, permitiendo eliminar los diversos problemas que afectan negativamente al diagnóstico de esta enfermedad, obteniendo como resultado un diagnóstico más objetivo, rápido y con un alto grado de sensibilidad y especificidad de la enfermedad. Además, este equipo eliminaría la necesidad de contar con la presencia de un especialista para realizar estos diagnósticos en aquellas zonas rurales o muy alejadas con muy bajos recursos y que no cuentan con el suficiente personal médico para combatir la enfermedad.

1.5.2 Estado de la investigación

Existen diversos trabajos sobre algoritmos para la detección de bacilos de tuberculosis en muestras de esputo mediante procesamiento de imágenes digitales. Estas investigaciones varían tanto en las técnicas utilizadas como en sus resultados. A continuación se mencionan alguna de ellas.

a) Veropoulos, 2001 [5]

Veropoulos fue el primero en proponer un algoritmo de aprendizaje automático o aprendizaje de máquinas para la identificación de los bacilos de tuberculosis. Para esto utilizó una red neuronal artificial y máquinas de vectores de soporte.

Una red neuronal artificial trata de simular el poder de aprendizaje y procesamiento automático del cerebro humano, y está conformada por neuronas (procesadores) y nodos interconectados que trabajan para producir un estímulo o respuesta de salida determinada dependiendo de los parámetros de entrada. La red neuronal artificial que utilizó Veropoulos tiene un algoritmo de aprendizaje de Retropropagación (Back-Propagation, BP), este algoritmo es un método de descenso de gradiente y utiliza una función de error que representa la diferencia entre la salida calculada por la red neuronal y la salida deseada.

Las redes neuronales convencionales presentan problemas de generalización, eficiencia de entrenamiento, eficiencia de pruebas y afinación del parámetro del algoritmo [15]. Para tratar de solucionar todos estos problemas Veropoulos utilizó una máquina de vector de soporte, el cual se basa en el principio de la teoría de aprendizaje estadístico de la Minimización del Riesgo Estructural. Este principio minimiza un límite de error de la prueba, permitiendo una mejor generalización que una red neuronal convencional. El algoritmo utilizado por la máquina de vector de soporte para la clasificación utiliza las funciones de base radial (Radial Basis Functions, RBF).

Además de los sistemas de clasificación ya mencionados, Veropoulos utilizó clasificadores de k-vecinos más cercanos (K-Nearest-Neighbour, KNN) y la regresión de núcleo (Kernel Regression, KR). KNN es un algoritmo estadístico que asigna un ejemplar desconocido a la clase que esté más cercana a ella. KR es un algoritmo relativamente nuevo que emula una máquina de vector de soporte.

Veropoulos analiza los dos tipos de tinciones, para cada caso utiliza procedimientos similares para la segmentación de los bacilos. Cabe mencionar que se utilizó un microscopio con un aumento de 100x y 400x.

El procedimiento utilizado por Veropoulos para las muestras teñidas con Auramina O fue el siguiente:

- Cantidad de imágenes analizadas: 100 imágenes de 720x512 píxeles con 256 niveles de gris. Espacio de color RGB (Rojo, Verde, Azul).
- Detección de bordes: Utiliza el Operador de Canny, el cual trabaja con los valores de la gradiente, con una desviación estándar $\sigma = 0.65$ y un tamaño de máscara de 3x3. Después de la detección de borde, cada región es etiquetada y se eliminan las regiones que estén fuera del rango de tamaño del bacilo.
- Conexión de píxeles de borde: Se realiza esta operación debido a que a pesar de utilizar parámetros cuidadosamente seleccionados para la detección de bordes, existen regiones con bordes abiertos. Los límites son necesarios para la derivación de los descriptores de forma. Se traza el límite interior de las regiones y la información sobre la posición de los píxeles de los límites es colectado para la última etapa del procesamiento.
- Descriptor de forma: Los descriptores de forma son un conjunto de parámetros que dan una representación numérica de cada región. Se utilizó 14 descriptores de Fourier.

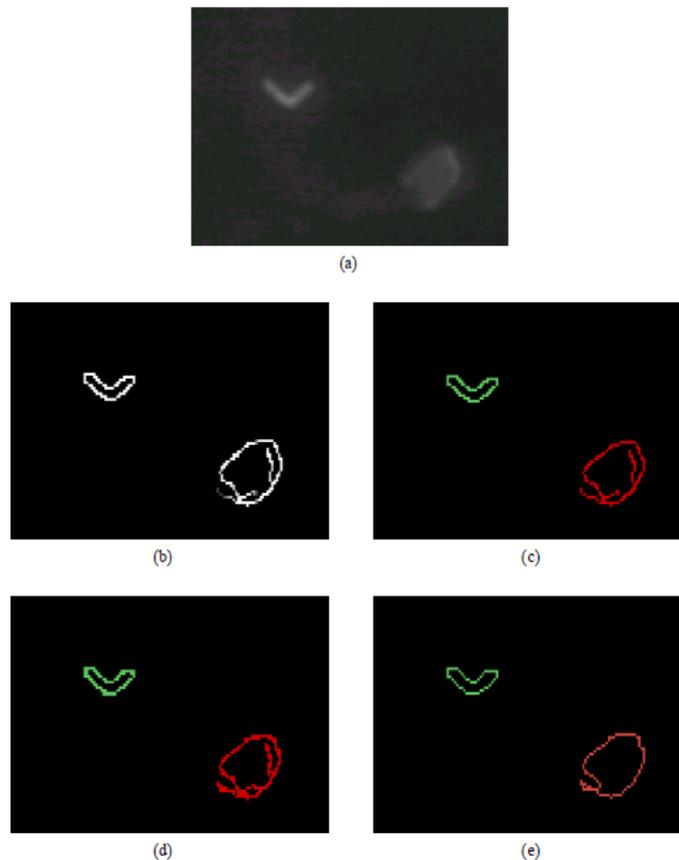


Figura 1.5: Proceso de segmentación de una muestra teñida con Auramina. (a) detalles de la imagen original: el objeto brillante de forma abastionada en el

lado izquierdo es un típico bacilo de tuberculosis; en algunos casos los bacilos son curvados como en este caso, (b) detección de borde de Canny, (c) etiquetado de las regiones, (d) conexión de píxeles de borde, (e) trazado del límite interno. Extraído de [5].

Como resultado de las pruebas mencionadas, Veropoulos una exactitud del 97.57% con el algoritmo de retropropagación, 88.06% con las funciones de base radial, 91.80% con el algoritmo de k-vecinos más cercanos y 95.24% con la regresión de núcleo.

b) Forero et al., 2006 [16]

Forero sostiene que el problema de la segmentación realizada por Veropoulos consistía en que no consideraban las imágenes cromáticas. Forero afirma que la forma del bacilo no es suficiente para realizar una discriminación porque otras partículas y especies de bacteria comparten la misma morfología. Por lo tanto, es necesario considerar también su color para mejorar la exactitud de la discriminación. Es por ello que Forero utiliza una umbralización de color adaptativa para mejorar la segmentación.

Los 2 últimos trabajos realizados por Forero utilizan técnicas muy similares, solo difieren en los descriptores de forma y en el sistema de clasificación utilizados. Como resultado, el último trabajo [16], presenta una mejora en los niveles de sensibilidad a comparación de su trabajo anterior [9].

A continuación describiremos los procedimientos utilizados por Forero en su último trabajo [16]:

- La tinción de los bacilos se realizó con Auramina O y se utilizó un fotomicroscopio estándar equipado con un sistema de iluminación de fluorescencia con un aumento de 250x.
- De los 43 pacientes se obtuvieron 781 imágenes de color RGB de 1600x1200 píxeles por cada paciente, de los cuales 308 imágenes presentaban bacilos de tuberculosis. Se extrajeron manualmente 1412 bacilos de 139 imágenes positivas y los 169 restantes se utilizaron para realizar las pruebas.
- Detección de bordes: Antes de aplicar un detector de bordes, se descubrió mediante un escaneo espectral que los bacilos presentaban un brillo intenso en las frecuencias de color verde; mientras que en esta misma frecuencia,

los restos y otras partículas no presentaban fluorescencia. En las frecuencias rojas aumentaba el brillo de los restos y partículas. Las frecuencias azules no brindaron ninguna información significativa. De esta manera, se eligió la capa de color verde para realizar la detección de bordes mediante el operador de Canny con una desviación estándar $\sigma = 1$, pues con este operador se obtuvieron mejores resultados que con otros operadores utilizados. Para finalizar esta etapa, se utilizaron operaciones morfológicas de cierre, rellenado de las regiones cerradas y apertura.

- Segmentación por color: Con el objetivo de poder extraer regiones con características de color de bacilo, se realizó una umbralización de color adaptativa. Ya que los bacilos presentan una mayor intensidad en la capa de color verde, el valor umbral se halló utilizando los valores promedio y máximo del canal verde. Cabe resaltar que otras alternativas de espacio de color, como por ejemplo HSI o CIE-Lab, no mejoraron los resultados.

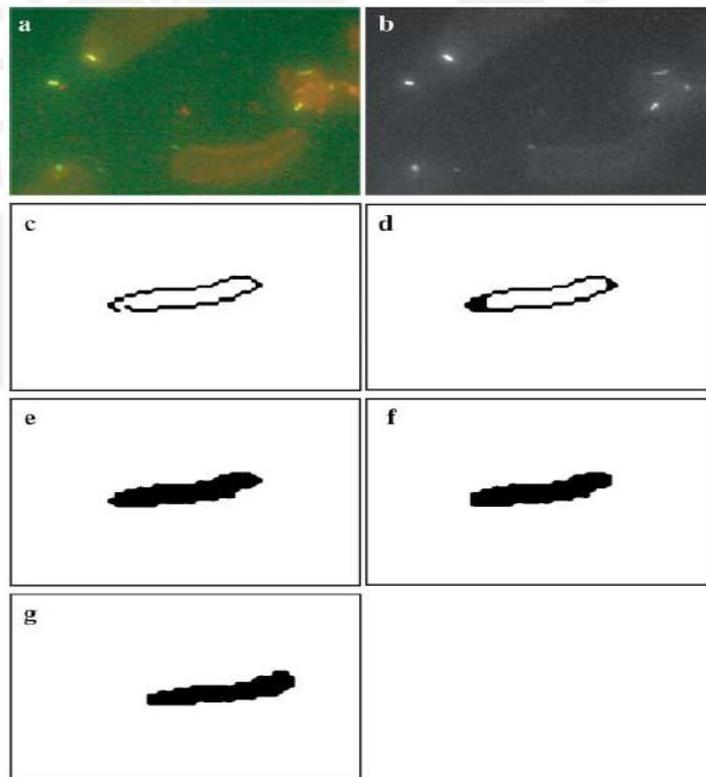


Figura 1.6: (a) Imagen original, (b) canal verde, (c) detector de bordes, (d) cierre, (e) rellenado de las regiones cerradas, (f) apertura, (g) segmentación por color. (c-g) detalles ampliados para una mejor apreciación de la importancia de las operaciones morfológicas. Extraído de [16].

- Filtrado: Los objetos restantes fueron filtrados de acuerdo a su área y excentricidad, eliminando aquellos objetos con un área muy pequeña o muy grande o con una excentricidad muy baja. De esta manera, se hallaron los siguientes umbrales: área mínima igual a 43 y área máxima igual a 207, y excentricidad mayor o igual a 0.8924.
- Extracción de características y análisis del bacilo: Los siguientes descriptores de forma fueron evaluados: área, compacidad, longitud del eje mayor y menor, excentricidad, perímetro, solidez, momentos de Hu y descriptores de Fourier. De estos descriptores, se descartaron aquellos que no fueran invariantes a escala, orientación y traslación, quedando los descriptores de compacidad, excentricidad, descriptores de Fourier y momentos de Hu (similar a los utilizados en su trabajo anterior [9]). Por último se descartaron los descriptores de Fourier, debido a que son muy susceptibles al ruido; se descartó la excentricidad por no ser un descriptor de forma independiente (se puede definir con los 2 primeros momentos de Hu); y se descartó la compacidad porque su utilización no mejoró los resultados de la clasificación. Así, para la extracción de características solo se utilizaron los momentos de Hu de orden uno, dos, tres y cuatro, específicamente los momentos uno, dos, tres y once, respectivamente.
- Análisis de agrupamiento: Debido a su simplicidad, se utilizó el algoritmo de agrupamiento de K-means para obtener el número estimado de agrupaciones que definan mejor a los bacilos de entrenamiento. Como resultado de este análisis se identificaron 4 agrupaciones.
- Clasificación: Para la etapa de clasificación se utilizaron los modelos de mixtura Gaussiana para caracterizar las agrupaciones. Un modelo de mixtura Gaussiana es una aproximación semiparamétrica a la estimación de la densidad basado en la utilización de un modelo que mejor se adapte al dato de entrada. Finalmente, la identificación del bacilo es llevada a cabo considerando un clasificador Bayesiano de error mínimo.

Como resultado, Forero logra obtener un alto nivel de sensibilidad y especificidad (mayor al 90% y 96%, respectivamente). Cabe resaltar que se mejoraron considerablemente los niveles de sensibilidad obtenidos en su trabajo anterior.

c) **Sadaphal et al., 2008 [17]**

El equipo de Sadaphal desarrolló una técnica de reconocimiento computacional automatizado de bacilo ácido-alcohol resistente en imágenes digitales de muestras

teñidas con Ziehl-Neelsen. La ácido-alcohol resistencia es una característica de los bacilos de tuberculosis, pero existen otros objetos o bacterias que poseen esta característica y podrían afectar los resultados, pues también mantienen la misma coloración después del proceso de tinción. Estos fueron los procedimientos:

- Segmentación por color: Sadaphal afirma que la segmentación y detección de bordes basados en la gradiente de color no funcionan adecuadamente para muestras teñidas con Ziehl-Neelsen, debido a que estas poseen mayores detalles de fondo que las imágenes con tinción de fluorescencia. Por esto, Sadaphal utilizó una segmentación bayesiana para predecir la probabilidad de un píxel que representa a un bacilo usando un conocimiento previo de los colores de una tinción con Ziehl-Neelsen. Las probabilidades del píxel fueron derivadas de los histogramas de las capas de color rojo, verde y azul, pues los valores de colores RGB de los píxeles de un bacilo ácido-alcohol resistente son significativamente diferentes que la de los objetos que no son bacilos. La umbralización utilizando las probabilidades de píxel generó una máscara binaria, el cual se mejoró utilizando la operación morfológica de la dilatación.
- Extracción de forma: Para discriminar los bacilos de tuberculosis de los otros objetos ácido-alcohol resistentes se utilizaron 2 descriptores de forma invariantes en orientación, traslación y escala: Relación entre ejes y la excentricidad. De esta manera, siguiendo un árbol de clasificación, los objetos con una relación entre ejes menor a 1.25 fueron clasificados como no-bacilos; luego, los objetos restantes con valores de excentricidad menores a 0.65 también fueron clasificados como no-bacilos. Finalmente se calculó la media μ y la desviación estándar σ del tamaño del bacilo de un conjunto de imágenes y, de los objetos restantes, aquellos con un área mayor a $\mu \pm 1.5\sigma$ fueron clasificados como posibles bacilos y los que se encontraban dentro del rango de $\mu \pm 1.5\sigma$ fueron clasificados como bacilos definidos.

Los resultados obtenidos por el equipo de Sadaphal se muestran en la Figura 1.7. Sin embargo, Sadaphal no nos brinda información sobre la sensibilidad o especificidad de su algoritmo.

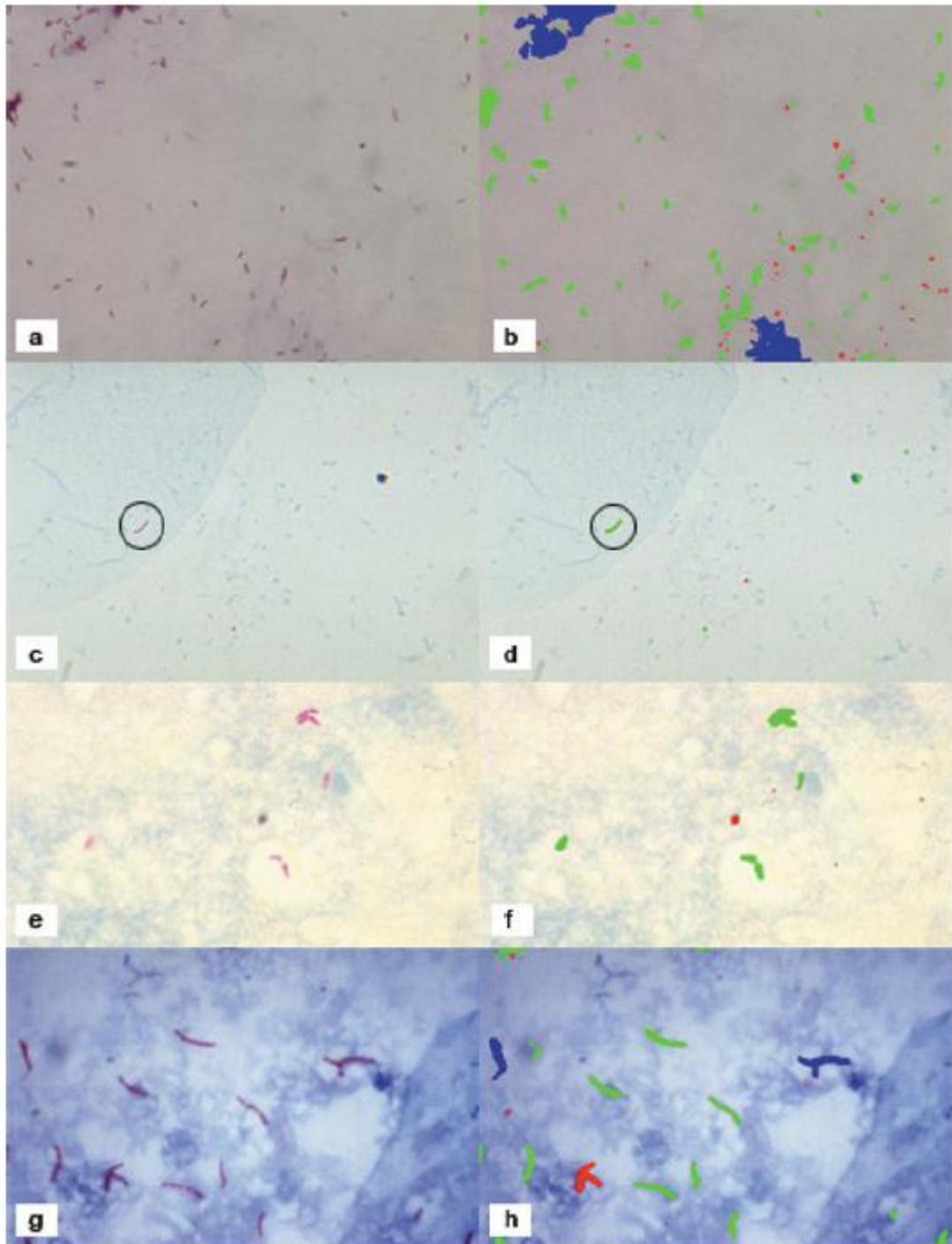


Figura 1.7: Los objetos clasificados como no-bacilo son etiquetados de color rojo, los objetos clasificados como posibles bacilos son etiquetados de color azul y los objetos clasificados como bacilos definidos son etiquetados de color verde. Extraído de [17].

d) Sotaquirá et al., 2009 [10]

Sotaquirá implementó un algoritmo inmune a la variación de condiciones de iluminación en las imágenes, ya que los colores que adquieren los bacilos y otros objetos, en una tinción con Ziehl-Neelsen, varían en intensidad dependiendo de la iluminación. Además, el algoritmo no solo permite realizar un diagnóstico de la enfermedad, sino que muestra el grado de infección a través del conteo de bacilos, ya sea si están asilados o en aglomeraciones.

- Se trabajó con un microscopio de luz Axioskop 2 con un aumento de 100x y un rango de nivel de luminosidad de 3 a 6 (con el fin de desarrollar un algoritmo inmune a la variación de condiciones de iluminación).
- Se adquirieron 1400 imágenes con diferentes niveles de resolución (1600x1200 píxeles y 640x480 píxeles), a partir de 14 pacientes con diferentes grados de infección (4 pacientes muy infectados, 4 pacientes medianamente infectados, 3 pacientes poco infectados y 3 pacientes no infectados).
- Cada capa de los espacio de color RGB, HSV, YCbCr, Lab y YIQ fueron evaluados. Los espacios de color RGB, HSV y YIQ fueron descartados debido a que presentaron un alto número de falsos positivos después de realizar la segmentación. Sin embargo, el espacio de color YCbCr, específicamente la capa Cr, transmitía la mayor cantidad de información relacionada a los bacilos y aglomeraciones de bacilos, y rechazaba la mayor cantidad de objetos que no son bacilos, pero contenía otros objetos como resultado de la variación de las condiciones de iluminación. Por otro lado, el espacio de color Lab, específicamente la capa a, presentaba una mayor inmunidad a la iluminación, pero no rechazaba los objetos que no eran bacilos presentes en la imagen.
- Debido a que la variación de la iluminación afecta la distribución del histograma, se utilizó una técnica de umbralización basada en la primera derivada del histograma. De esta manera, se calculó los valores umbrales para la capa Cr y a, y se efectúa la umbralización, obteniendo una imagen correspondiente a cada capa. Luego, se realiza una operación lógica AND entre las dos imágenes para aprovechar las características presentes en ambas capas.

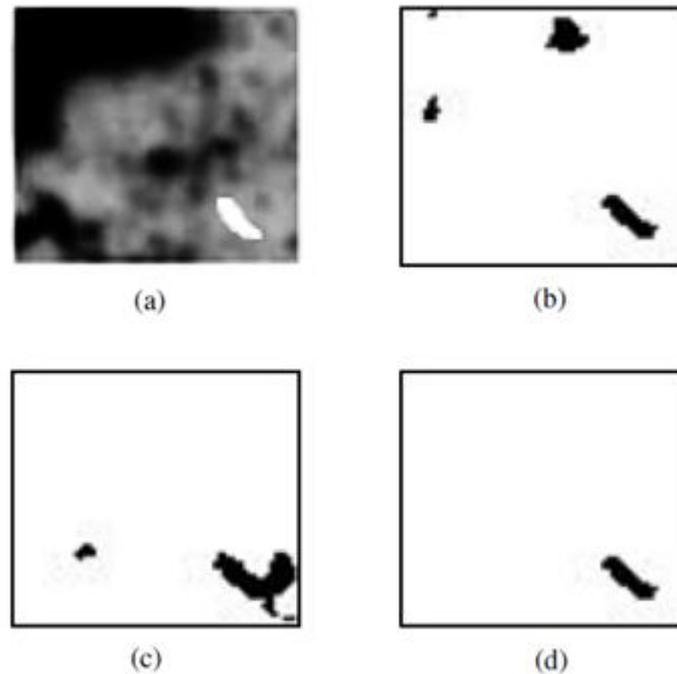


Figura 1.8: (a) imagen original, (b) imagen segmentada del espacio de color Lab, (c) imagen segmentada del espacio de color YCbCr, (d) resultado final (operación lógica AND entre la capa Cr y a). Extraído de [10].

- **Cuantificación de los bacilos:** Para cuantificar los bacilos que conforman una aglomeración, se calculó un tamaño de bacilo promedio a partir del conjunto de imágenes obtenidas. Como resultado, para cada resolución, 640x480 y 1600x1200, se halló un tamaño de bacilo promedio de 41.4 y 52.3 píxeles respectivamente. Usando el tamaño de bacilo promedio y conociendo el área en píxeles del objeto, se aproximó el número de bacilos que conformaban una aglomeración.
- **Diagnóstico:** Habiendo realizado el conteo de los bacilos, se utilizó la regla de decisión presentada en la Tabla 1.2 para diagnosticar el grado de infección del paciente.

Tabla 1.2: Regla de decisión para diferentes grados de infección de tuberculosis. Extraído de [10].

Grado de infección	Regla de decisión
3 (alto)	Más de 10 bacilos por campo en 20 campos observados
2 (medio)	1 a 10 bacilos por campo en 50 campos observados
1 (bajo)	Menos de 1 bacilo por campo en 100 campos observados
0 (no infectado)	Ningún bacilo en 100 campos observados

Como resultado, se obtuvo una sensibilidad del 90.9% y una especificidad del 100%. Además, la relación entre el número de diagnósticos correctos y el número de pacientes fue del 85.7%.

e) Otros estudios

Además de los estudios mencionados anteriormente, existen otros trabajos e investigaciones sobre algoritmos de detección de bacilos de tuberculosis. De nuestra propia casa de estudios tenemos los trabajos de Kanashiro, D. [18], y Aguilar, N. [19] quienes utilizaron técnicas similares a las desarrolladas por Veropoulos y Forero para segmentar imágenes de muestras de tuberculosis de pacientes del Hospital 2 de Mayo. Kanashiro desarrolló la etapa de selección de una capa de color a través de un análisis de los histogramas y las características operativas del receptor, pero este análisis puede no ser muy óptimo, pues no se consideró normalizar los histogramas para compensar la gran diferencia en la cantidad de píxeles del fondo y de los bacilos. La segmentación de las imágenes se realizó mediante una umbralización con un valor umbral fijo, lo cual no era muy eficiente dada la considerable variabilidad de las intensidades de color de las imágenes. Posteriormente, Aguilar se encargó de seleccionar los descriptores de forma e implementar un sistema de clasificación para los objetos segmentados con el algoritmo de Kanashiro. Sin embargo, Aguilar utiliza 8 descriptores de forma, entre los cuales se tienen momentos geométricos de alto orden, lo cual podría representar un alto costo computacional. Finalmente, de ambos trabajos, se obtiene como resultado una sensibilidad menor al 70%. El algoritmo posee una muy baja sensibilidad comparado con los otros estudios mencionados anteriormente y no podría ser considerado como un método de diagnóstico confiable, pero se debe tomar en cuenta que las técnicas utilizadas también pueden verse afectadas por la calidad de las imágenes con las que se trabajaron.

Otros trabajos como el de Costa et al. [20] desarrolla un algoritmo simple implementando una umbralización de la imagen que se obtiene al restar la capa verde de la capa roja, obteniendo una sensibilidad del 76.65%. También existen investigaciones que utilizan técnicas mucho más complejas como en el caso del trabajo de Khutlang et al. [21], quienes utilizaron clasificadores de píxeles (clasificador Bayesiano, clasificador de regresión lineal, clasificador lineal logístico y clasificador de la discriminante cuadrática) para segmentar y clasificar los objetos y, además, se utilizaron diferentes algoritmos de selección de descriptores de forma para seleccionar un subconjunto de descriptores, del total de descriptores

estudiados, que sean suficientes y permitan describir adecuadamente los objetos, obteniendo así sensibilidades mayores al 95%.

Por otro lado, existen algoritmos desarrollados para mejorar la calidad y contraste de las imágenes de las muestras de tuberculosis como los trabajos de Osman et al. [22] y Wahab et al. [23], las cuales usan técnicas de estiramiento del brillo y la oscuridad, y ecualización del histograma.

Cabe mencionar que la calidad y la intensidad de los colores de las muestras de tuberculosis teñidas con la tinción de Ziehl-Neelsen de los trabajos estudiados varían notablemente, esto debido a diferentes factores: diferentes procedimientos de preparación de la muestra, calidad de los suministros utilizados para la tinción, nivel de iluminación, la calidad del microscopio y la calidad de la cámara fotográfica. Por ello es muy importante tener en cuenta que algunas de las técnicas utilizadas en determinados trabajos podrían no funcionar adecuadamente para imágenes obtenidas a diferentes condiciones. Además, se debe tener siempre en consideración que implementar un algoritmo demasiado complejo implica un costo computacional muy alto, lo cual afectaría notablemente a la velocidad de ejecución del mismo.

CAPÍTULO 2

CONOCIMIENTOS TEÓRICOS SOBRE PROCESAMIENTO DE IMÁGENES DIGITALES

2.1 IMAGEN DIGITAL

Una imagen puede ser definida como una función de dos dimensiones, $f(x, y)$, donde x e y son las coordenadas espaciales, y la amplitud de f de cualquier par de coordenadas es llamada la intensidad o nivel de gris de la imagen en ese punto. Cuando x , y y el valor de amplitud de f son finitos y discretos, la imagen es llamada imagen digital y cada punto de esta imagen es conocido como píxel [11]. De esta manera, una imagen digital puede ser representada como un arreglo matricial de $N \times M$ píxeles.

2.2 ESPACIO DE COLOR

Un espacio de color se define como una base de N componentes vectoriales cuyas combinaciones generan todos los posibles colores que pueden ser representados en el espacio. De esta manera es posible dividir una imagen en N capas en escala de grises de las cuales la intensidad de gris de un mismo píxel de las diferentes capas representa la intensidad de cada componentes que combinadas dan como resultado el color de dicho píxel [11].

En el caso del procesamiento de imágenes digitales los espacios de color tienen 3 componentes vectoriales, cada una de ellas con diferentes características y propiedades. El espacio de color más común y utilizado por la mayoría de cámaras digitales es el espacio RGB (rojo, verde y azul). Sin embargo, existen otros espacios de color que también son frecuentemente utilizados y pueden ser obtenidas a partir del espacio de color RGB, tales como el espacio de color NTSC o YIQ (luminancia, matiz y saturación), el espacio de color YCbCr (luminancia, azul croma y rojo croma), el espacio de color HSV (matiz, saturación y valor), el espacio de color CMY (cian, magenta y amarillo) y el espacio de color HSI (matiz, saturación e intensidad). A continuación se muestran las ecuaciones utilizadas para obtener cada uno de los espacios de color mencionados [11]:

- Espacio de color NTSC o YIQ:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.1)$$

- Espacio de color YCbCr:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.2)$$

- Espacio de color HSV:

$$H = \begin{cases} \left(\frac{G - B}{\max - \min} \right) / 6, & \text{si } R = \max \\ \left(2 + \frac{B - R}{\max - \min} \right) / 6, & \text{si } G = \max \\ \left(4 + \frac{R - G}{\max - \min} \right) / 6, & \text{si } B = \max \end{cases} \quad (2.3)$$

$$S = \frac{\max - \min}{\max} \quad (2.4)$$

$$V = \max \quad (2.5)$$

Donde \max y \min , son el valor máximo y mínimo de las intensidades de las capas rojo, verde y azul, respectivamente.

- Espacio de color CMY:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.6)$$

- Espacio de color HSI:

$$H = \begin{cases} \theta, & B \leq G \\ 360 - \theta, & B > G \end{cases} \quad (2.7)$$

$$\theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R - G) + (R - B)]}{\sqrt{[(R - G)^2 + (R - B)(G - B)]}} \right\} \quad (2.8)$$

$$S = 1 - \frac{3}{(R + G + B)} [\min(R, G, B)] \quad (2.9)$$

$$I = \frac{1}{3}(R + G + B) \quad (2.10)$$

Por otro lado, algunos de las investigaciones vistas en la sección anterior, como los estudios de Forero y Sotaquirá, nos mencionan el análisis y/o utilización del espacio de color CIE $L^*a^*b^*$, un espacio de color menos conocido que las mencionadas anteriormente. En 1931, la Comisión Internacional de la Iluminación (CIE, por sus siglas en francés) estudió la percepción de color humana y estandarizó el espacio de color conocido como CIE XYZ. Este estándar define un espacio tridimensional donde tres valores (X, Y y Z), llamados valores de triple estímulo, definen un color. En 1976, en un esfuerzo por obtener un espacio de color perceptualmente uniforme que pueda ser correlacionado con la apariencia visual de los colores, la CIE creó el espacio de color $L^*a^*b^*$ [24] [25].

La transformación utilizada para la conversión del espacio de color RGB al espacio de color CIE XYZ está dada por:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.35758 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.11)$$

Finalmente, para la conversión del espacio de color CIE XYZ al CIE $L^*a^*b^*$ se utilizan las siguientes ecuaciones:

$$L^* = \begin{cases} 116 \left(\frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16, & \frac{Y}{Y_n} > 0.008856 \\ 903.3 \left(\frac{Y}{Y_n} \right) - 16, & \frac{Y}{Y_n} \leq 0.008856 \end{cases} \quad (2.12)$$

$$a^* = 500(f(X/X_n) - f(Y/Y_n)) \quad (2.13)$$

$$b^* = 200(f(Y/Y_n) - f(Z/Z_n)) \quad (2.14)$$

Donde:

$$f(t) = \begin{cases} t^{\frac{1}{3}}, & t > 0.008856 \\ 7.787t + 16/116, & t \leq 0.008856 \end{cases} \quad (2.15)$$

Los valores X_n , Y_n y Z_n , son los valores de triple estímulo del color blanco de referencia [24] [25].

2.3 DETECTOR DE BORDE

Un detector de borde permite la demarcación de los límites o bordes del objeto que se desea analizar, para ello se detectan las discontinuidades significativas en los valores de intensidad de la imagen. Estas discontinuidades son detectadas usando la primera y segunda derivada. La primera derivada utilizada en el procesamiento de imágenes digitales es la gradiente [11].

La gradiente de una función de dos dimensiones, $f(x,y)$, es definido como el vector:

$$\nabla F = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (2.16)$$

La magnitud de este vector es:

$$\nabla f = \text{mag}(\nabla F) = [G_x^2 + G_y^2]^{1/2} \quad (2.17)$$

El vector gradiente apunta a la dirección del máximo cambio de f en las coordenadas (x,y) . El ángulo en la cual dicho máximo cambio ocurre es:

$$\alpha(x,y) = \tan^{-1} \left(\frac{G_y}{G_x} \right) \quad (2.18)$$

La segunda derivada en el procesamiento de imágenes es generalmente calculada usando el Laplaciano. De esta manera, el Laplaciano de una función $f(x,y)$ es el siguiente:

$$\nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} \quad (2.19)$$

Sin embargo, el Laplaciano es raramente utilizado para la detección de borde porque, como una segunda derivada, es muy sensible al ruido, su magnitud produce doble borde y no permite detectar la dirección del borde.

Dado que la información de una imagen digital es discreta, es necesario aproximar las operaciones anteriores para la detección de bordes. Para ello, los detectores de borde utilizan máscaras para calcular el valor de la derivada de cada píxel analizando sus vecindades. La Figura 2.1 representa a un píxel (en el centro) y sus vecindades.

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

Figura 2.1: Representación de un píxel (Z_5) y sus ocho píxeles vecinos. Extraído de [11].

2.3.1 Detector de borde de Roberts

El detector de borde de Roberts es uno de los detectores de borde más simple y menos utilizado por ser muy limitado ya que no es simétrico y no permite detectar los bordes múltiples de 45° [11].

Utiliza las siguientes ecuaciones para aproximar las primeras derivadas G_x y G_y :

$$G_x = Z_9 - Z_5 \quad (2.20)$$

$$G_y = Z_8 - Z_6 \quad (2.21)$$

A partir de estos valores se calcula la magnitud de la gradiente g del píxel:

$$g = [G_x^2 + G_y^2]^{1/2} \quad (2.22)$$

Luego, se compara el valor de la magnitud de la gradiente con un valor umbral T . De esta manera, si $g \geq T$, entonces es un píxel del borde del objeto.

2.3.2 Detector de borde de Prewitt

El detector de borde de Prewitt utiliza las siguientes ecuaciones para aproximar digitalmente las primeras derivadas G_x y G_y :

$$G_x = (Z_7 + Z_8 + Z_9) - (Z_1 + Z_2 + Z_3) \quad (2.23)$$

$$G_y = (Z_3 + Z_6 + Z_9) - (Z_1 + Z_4 + Z_7) \quad (2.24)$$

2.3.3 Detector de borde de Sobel

El detector de borde de Sobel, a diferencia de los detectores anteriores, utiliza coeficientes con valor 2, lo que le permite suavizar el ruido presente en la imagen [11].

Las ecuaciones utilizadas para aproximar las primeras derivadas G_x y G_y son:

$$G_x = (Z_7 + 2Z_8 + Z_9) - (Z_1 + 2Z_2 + Z_3) \quad (2.25)$$

$$G_y = (Z_3 + 2Z_6 + Z_9) - (Z_1 + 2Z_4 + Z_7) \quad (2.26)$$

2.3.4 Detector de borde de Canny

El detector de Canny es considerado uno de los detectores de borde más poderosos porque utiliza también la información de la dirección de la gradiente, a comparación de los otros detectores que sólo utilizan la magnitud. Este método emplea los siguientes pasos para realizar la detección [11] [26]:

- 1) La imagen es suavizada usando un filtro Gaussiano con una determinada desviación estándar, σ , para reducir el ruido.
- 2) Se calcula la magnitud de la gradiente y la dirección del borde para cada píxel.
- 3) Se realiza el proceso conocido como la Supresión de no-máximos para obtener bordes más definidos y delgados. Para esto, se consideran cuatro direcciones identificadas por las orientaciones de 0° , 45° , 90° y 135° con respecto al eje horizontal y para cada píxel se encuentra la dirección que mejor se aproxime a la dirección de la gradiente. Si el valor de la magnitud de la gradiente es más pequeño que al menos uno de sus dos vecinos en la dirección del ángulo obtenido se asigna el valor 0 a dicho píxel, en caso contrario se asigna el valor de la magnitud de la gradiente.
- 4) Se utiliza el concepto de histéresis con dos valores umbrales (T_1 y T_2) determinados para definir el borde. Si el valor de la magnitud de la gradiente es mayor a T_2 es considerado un píxel de borde fuerte, si se encuentra entre los valores de T_1 y T_2 es considerado un píxel de borde débil, y si es menor a T_1 no es detectado como borde.
- 5) Finalmente, se analiza la vecindad de los píxeles de borde débil para realizar la conexión con los píxeles de borde fuerte.

2.4 OPERACIONES MORFOLÓGICAS

Una operación morfológica es una herramienta que permite extraer componentes de una imagen que son útiles en la representación y descripción de la forma de una región como los bordes. Las operaciones morfológicas fundamentales son la dilatación y la erosión. Estas operaciones morfológicas y sus diferentes combinaciones son los que usualmente se utilizan en el procesamiento de

imágenes digitales. Las combinaciones más comunes de estas dos operaciones son la Apertura (Opening) y el Cierre (Closing) [11]. Las operaciones de cierre y apertura son usadas para eliminar detalles de la imagen que son más pequeños que el elemento estructural [27].

2.4.1 Dilatación

La dilatación es una operación que hace crecer o engrosar los objetos en una imagen binaria. La forma específica y el alcance de este crecimiento son controlados por un elemento estructural. Si el origen o el centro del elemento estructural coincide con un píxel de valor 1 de la imagen, el elemento estructural solapa a la imagen original [11].

2.4.2 Erosión

La erosión es una operación morfológica que encoge o adelgaza los objetos en una imagen binaria. Como en la dilatación, la forma específica y el alcance de este encogimiento son controlados por un elemento estructural. Si todo el elemento estructural coincide con la imagen, se obtiene en la salida sólo la posición del origen del elemento estructural [11].

2.4.3 Apertura

La operación morfológica de apertura se realiza aplicando una operación de erosión seguida de una dilatación. Sea la imagen A y el elemento estructural B , la operación de apertura se denota por $A \circ B$ y puede ser interpretada como la unión de todas las traslaciones de B que encajen completamente en A . La Figura 2.2 ilustra el funcionamiento de la operación morfológica de apertura [11].

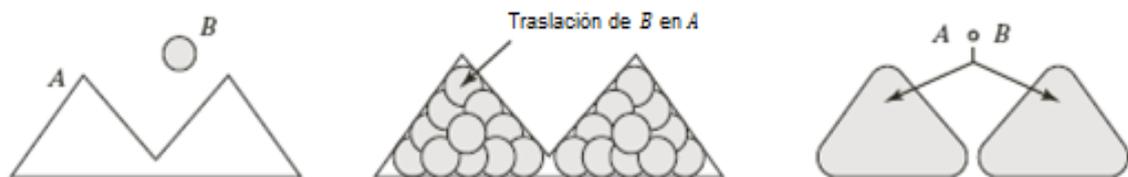


Figura 2.2: Ilustración de la operación de apertura. Extraído de [11].

2.4.4 Cierre

La operación morfológica de cierre está dada por una operación de dilatación seguida de una erosión. Sea la imagen A y el elemento estructural B , la operación de cierre se denota por $A \bullet B$ y puede ser interpretada como el complemento de la

unión de todas las traslaciones de B que no solapan a A . La Figura 2.3 muestra el funcionamiento de la operación morfológica de cierre [11].

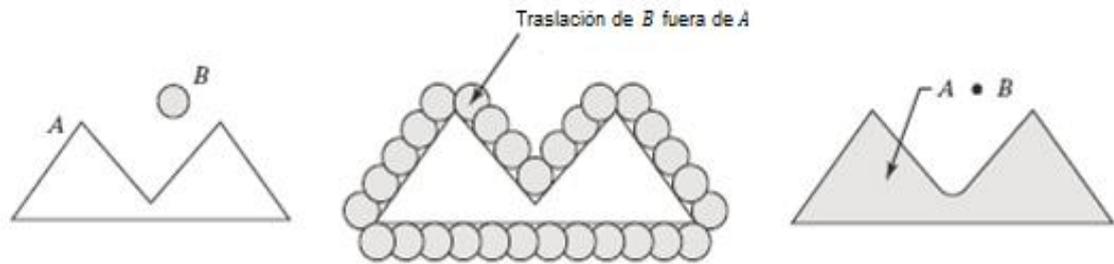


Figura 2.3: Ilustración de la operación de cierre. Extraído de [11].

2.5 DESCRIPTORES DE FORMA

Los descriptores de forma son una herramienta ampliamente utilizada en la etapa de extracción de características. Estos permiten obtener una representación numérica o matemática de las características de los objetos presentes en una imagen digital [13]. Existen un gran número de descriptores de forma diferentes utilizadas dependiendo de la aplicación. Unas de las características más importantes de los descriptores de forma más utilizados en las imágenes médicas es que son invariantes a escala, traslación y orientación [14].

2.5.1 Descriptores de Fourier

Un punto k del contorno digital del objeto analizado puede ser representado como una secuencia de pares de coordenadas $s(k) = [x(k), y(k)]$, para $k = 1, 2, \dots, N$ (ver Figura 2.4). La transformada de Fourier discreta de $s(k)$ está dada por:

$$a(u) = \frac{1}{N} \sum_{k=1}^N s(k) e^{-i2\pi uk/N} \quad (2.27)$$

Donde $a(u)$ son los coeficientes de Fourier (o descriptores de Fourier). Estos coeficientes están influenciados por la forma de la curva y el punto inicial de la curva. Para hacer que los descriptores de Fourier sean invariantes a traslación y rotación, se utiliza la siguiente transformación:

$$r(u) = \sqrt{|a_x(u)|^2 + |a_y(u)|^2} \quad (2.28)$$

Finalmente, para que sean invariantes a escala se utiliza la siguiente transformación:

$$w(u) = \frac{r(u)}{r(1)} \quad (2.29)$$

El número mínimo de coeficientes necesarios para reconstruir correctamente un objeto depende de la complejidad de la curva. Se ha comprobado que de 10 a 15 descriptores son suficientes para la descripción de las características de los bacilos [5] [11] [14].

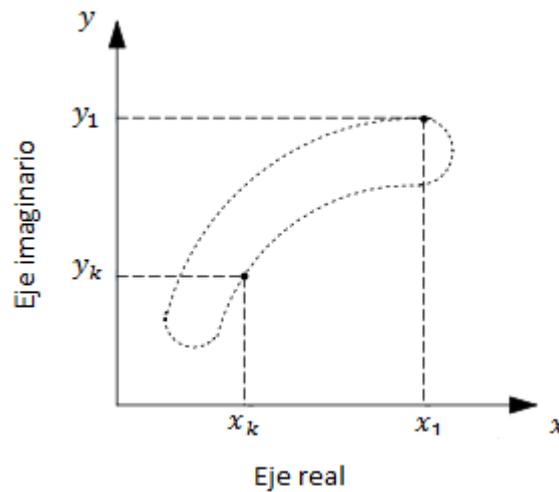


Figura 2.4: Un contorno digital representado como una secuencia de números complejos. Extraído de [5].

2.5.2 Momentos

Los momentos invariantes a escala, traslación y orientación son calculados por medio de sumatorias que consideran las coordenadas y el valor de cada uno de los píxeles de la imagen. Son muy usados en el reconocimiento de patrones debido al hecho de que los momentos proveen una representación equivalente a la imagen, es decir, esta puede ser reconstruida a partir de sus momentos [14].

a) Momentos de Hu

Los momentos de Hu son descriptores de forma que brindan información de las características geométricas de los objetos. Son invariantes a escala, traslación y orientación, y están basados en los momentos geométricos [28].

Los momentos geométricos de dos dimensiones de orden $(p + q)$ de una imagen digital $f(x, y)$ es definido como:

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \quad (2.30)$$

El correspondiente momento central es definido como:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (2.31)$$

$$\bar{x} = \frac{m_{10}}{m_{00}}$$

$$\bar{y} = \frac{m_{01}}{m_{00}}$$

Los momentos centrales de orden $(p + q)$ son definidos como:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\frac{p+q}{2}+1}} \quad (2.32)$$

Finalmente, los momentos de Hu, invariantes a escala, traslación y orientación, son definidos de la siguiente manera:

$$\phi_1 = \eta_{20} + \eta_{02} \quad (2.33)$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (2.34)$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (2.35)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (2.36)$$

$$\begin{aligned} \phi_5 = & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2] \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} - \eta_{03})^2] \end{aligned} \quad (2.37)$$

$$\begin{aligned} \phi_6 = & (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ & + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \end{aligned} \quad (2.38)$$

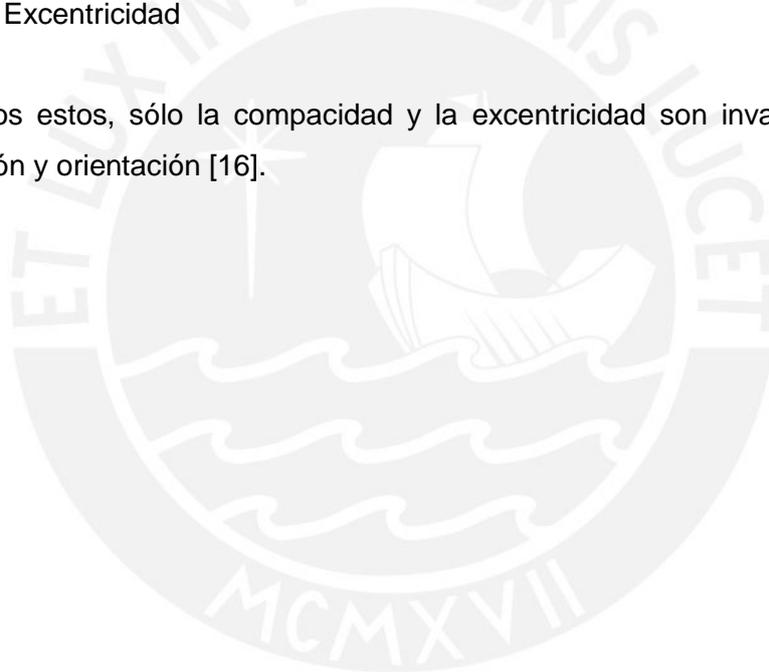
$$\begin{aligned} \phi_7 = & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2] \\ & - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (2.39)$$

2.5.3 Atributos geométricos

Los atributos geométricos, a pesar de que son más simples y de que algunos de estos no son invariantes a escala, traslación y orientación, en ciertas aplicaciones han demostrado brindar características suficientes para describir objetos y diferenciarlos correctamente a un costo computacional menor que utilizando los descriptores mencionados anteriormente [11] [16]. Algunos de los atributos geométricos más utilizados en el procesamiento de imágenes digitales son:

- Perímetro
- Área
- Centroide
- Compacidad
- Menor rectángulo que contiene al objeto (Minimum bounding box)
- Excentricidad

De todos estos, sólo la compacidad y la excentricidad son invariantes a escala, traslación y orientación [16].



CAPÍTULO 3

DISEÑO E IMPLEMENTACIÓN

3.1 OBJETIVOS

3.1.1 Objetivo general

Desarrollar un algoritmo capaz de realizar automáticamente la detección y conteo de bacilos de tuberculosis en muestras de esputo mediante el procesamiento de imágenes digitales.

3.1.2 Objetivos específicos

- Desarrollar un algoritmo que realice el conteo automático de bacilos de tuberculosis con un alto grado de exactitud (mayor a 85%).
- Reducir la duración del análisis de las muestras de esputo a tiempos menores a 15 minutos por muestra.
- Implementar el algoritmo en un lenguaje de programación C++ para obtener un programa capaz de funcionar en tiempo real y en comunicación con otros programas como el del control de la cámara digital.

3.2 MATERIALES

Las imágenes fueron adquiridas de muestras de esputo de pacientes del Hospital 2 de Mayo, dado que es el hospital con la mayor cantidad de pacientes de tuberculosis en Lima y además cuenta con un laboratorio de micobacterias donde se analizan las muestras. El técnico encargado brindó 3 muestras de esputo, teñidas con la técnica de tinción de Ziehl-Neelsen, de diferentes pacientes con un alto grado de infección. Además, en el laboratorio de imágenes médicas de nuestra casa de estudios se contaba con una base de datos de imágenes de muestras de esputo sin la aplicación del azul de metileno, puesto que esta sustancia, además de teñir el fondo de una coloración azul claro, suele generar manchas azules mucho más oscuras que podrían afectar el procesamiento (ver Figuras 3.1 y 3.2).

Para adquirir las imágenes digitales de las muestras se utilizó un microscopio convencional Olympus CX31 con un lente de aumento de 100x y una cámara digital Olympus EVOLT E-520 acoplado a éste. Se adquirieron 45 imágenes de las

muestras brindadas por el hospital (con aplicación de azul de metileno) y también se analizaron 60 imágenes de las muestras sin la aplicación de azul de metileno de la base de datos, todas a una resolución de 3648 x 2736 píxeles y almacenadas en formato de archivo Tiff mediante el software de transferencia y conversión de archivos de la misma corporación Olympus. El procesamiento se realizó mediante el software Matlab® y sus herramientas de procesamiento de imágenes.

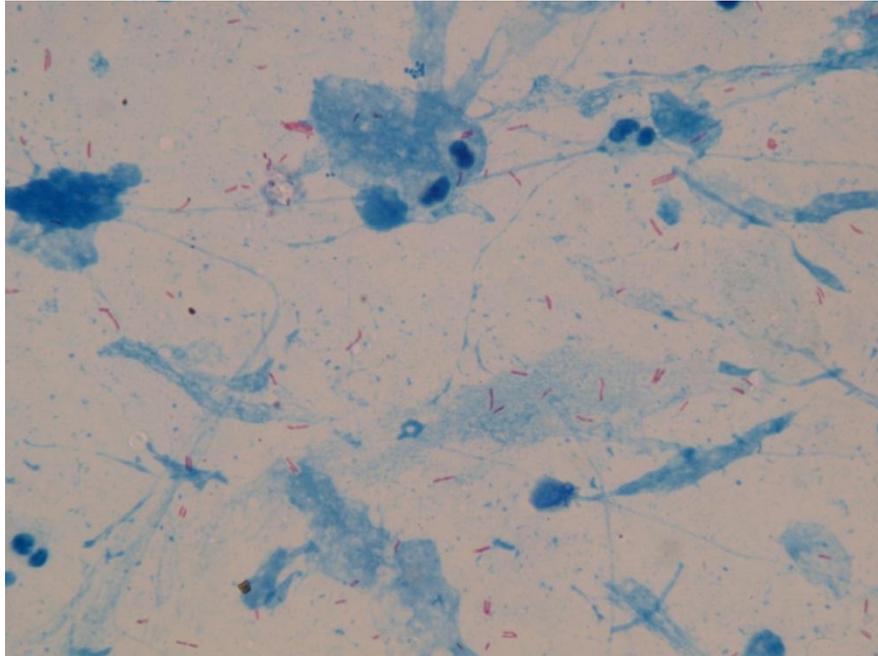


Figura 3.1: Imagen de un campo de una muestra de esputo teñida con la técnica de tinción de Ziehl-Neelsen. Se pueden apreciar las manchas azules producidas por la aplicación del azul de metileno.

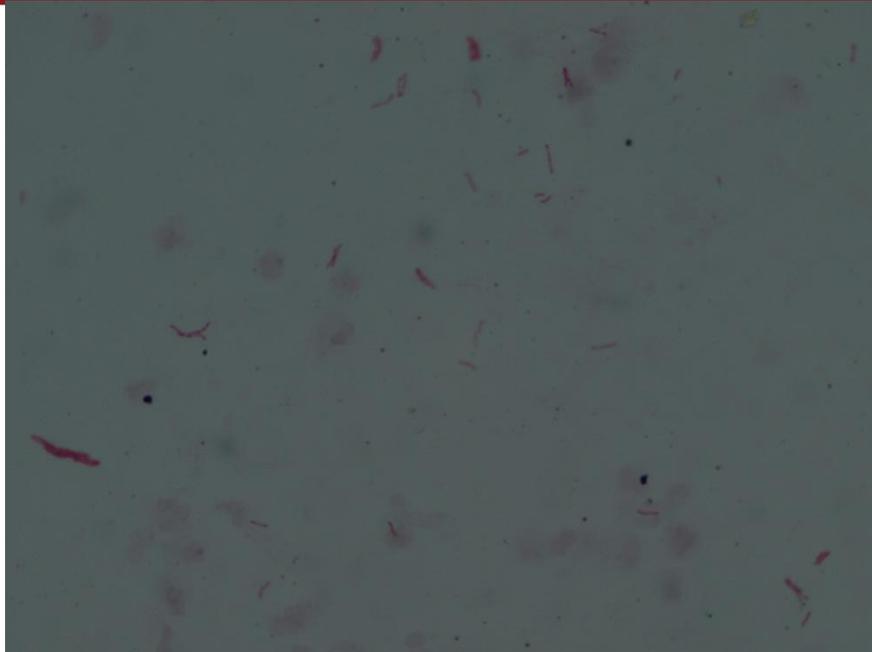


Figura 3.2: Imagen de un campo de una muestra de esputo sin la aplicación de azul de metileno. A pesar de que no se presentan manchas azules como en el proceso de tinción convencional, existen manchas de color fucsia que disminuyen el contraste entre el fondo y los bacilos.

Luego de analizar y procesar las imágenes de las muestras sin la aplicación de azul de metileno (análisis de espacios de color, aplicación de diferentes detectores de borde como el detector de borde de Roberts, Prewitt, Sobel y Canny, y aplicación de operaciones morfológicas), los mejores resultados se obtuvieron al trabajar con la capa de color verde del espacio de color RGB y utilizando el detector de borde de Canny y las operaciones morfológicas de cierre y apertura. Sin embargo, no existía un buen contraste entre los valores de intensidad de los píxeles del fondo y de los bacilos, debido a la gran cantidad de manchas fucsias presentes en la muestra e incluso en algunos campos estas manchas no permitían diferenciar los bacilos en lo absoluto. Como consecuencia, luego del procesamiento sólo se lograban detectar un 72% de los bacilos presentes en la imagen.

Por otro lado, las imágenes de las muestras de esputo con la aplicación de azul de metileno presentaron mejores resultados, porque a pesar de las manchas azules, el contraste entre el fondo y los bacilos es mucho mayor. De esta manera, se descartó la opción de prescindir del azul de metileno como mejora para el procesamiento de las imágenes. Por ello, en los puntos siguientes nos enfocaremos en las técnicas

utilizadas para segmentar las imágenes de las muestras de esputo con la aplicación de azul de metileno.

3.3 ANÁLISIS DE ESPACIOS DE COLOR

Con el fin de encontrar una capa o canal de un espacio de color que presente el mayor contraste entre los píxeles del fondo y de los bacilos, se analizaron las diferentes capas de los 7 espacios de color mencionados en el capítulo anterior (sección 2.2). Para ello primero se segmentaron manualmente cada bacilo de 15 imágenes con el software libre basado en Java para el procesamiento de imágenes *ImageJ* (<http://rsbweb.nih.gov/ij/>), obteniendo una imagen binaria para cada una de las imagen (ver Figura 3.3). Multiplicando la imagen binaria y la negada de la imagen binaria con una de las capas de la imagen original se logró obtener una imagen con píxeles sólo de los bacilos y otra con píxeles de sólo el fondo, para posteriormente computarizar el histograma de ambas imágenes. Se sumaron los histogramas de las 15 imágenes para ambos casos y se les dividió entre el número de píxeles de fondo y bacilos, respectivamente, con el fin de obtener valores normalizados más adecuados para su comparación. Luego de obtener el histograma total normalizado, se hizo un análisis de las gráficas de las Características Operativas del Receptor (ROC, por sus siglas en inglés) o Curvas de ROC.



Figura 3.3: Imagen binaria obtenida de la segmentación manual de la imagen de la Figura 3.1 con ayuda del software libre *ImageJ*. En el Anexo A se

muestran otros ejemplos de imágenes de muestras de esputo con su respectiva segmentación manual.

La curva de ROC es una técnica para la visualización, organización y selección de clasificadores basados en sus rendimiento, y son ampliamente utilizados en la teoría de detección de señales para representar la relación entre las tasas de verdaderos positivos y falsos positivos de los clasificadores. Esta curva es una representación gráfica de la sensibilidad frente al $(1 - \text{especificidad})$ o razón de falsos positivos para un sistema clasificador binario según se varía el umbral de discriminación. La propiedad más utilizada de esta gráfica es el área bajo la curva, pues es equivalente a la probabilidad de que un clasificador ordenará una instancia positiva elegida aleatoriamente más alto que una negativa. Es decir, para nuestro caso, el área bajo la curva representa el poder de discriminación entre el fondo y los bacilos de una determinada capa de color y puede tener un valor comprendido en el rango de 0.5 a 1, donde un área de valor 1 representa un poder de discriminación perfecto y un área de valor 0.5 corresponde a un clasificador sin capacidad de discriminación. [29]

La Figura 3.4 nos brinda un ejemplo de la distribución de clases de los histogramas del fondo y de los bacilos, donde la variable c representa el valor umbral de discriminación que se varía para generar la curva de ROC, usando las ecuaciones definidas en la sección 1.2 para calcular la sensibilidad y el $(1 - \text{especificidad})$.

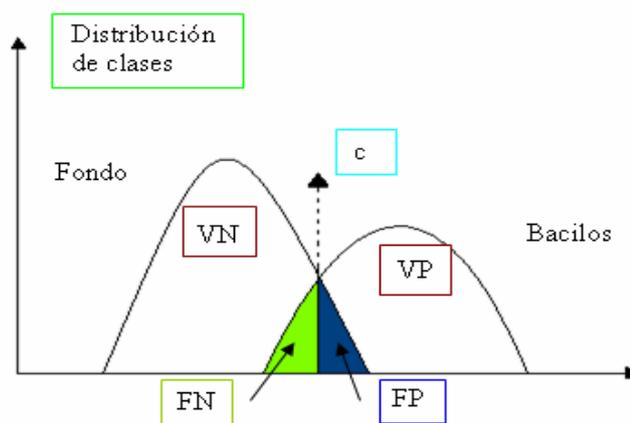


Figura 3.4: Ejemplo de la distribución de clases de los histogramas del fondo y de los bacilos, donde VP es el número de píxeles de bacilos clasificados correctamente, VN es el número de píxeles del fondo de la imagen clasificados correctamente, el FP es el número de píxeles del fondo

clasificados como bacilos y el FN es el número de píxeles de bacilos clasificados como fondo. Extraído de [18].

La Figura 3.5 muestra el análisis realizado para el espacio de color NTSC (en el Anexo B se muestran las figuras con el análisis de todos los espacios de color estudiados). En la Tabla 3.1 se muestran las áreas bajo la curva de las gráficas de ROC para cada espacio de color analizado. Se puede observar que la capa de color con mayor área bajo la curva de ROC, y por lo tanto con mayor poder de discriminación, es la capa Q del espacio de color NTSC. De esta manera podemos concluir que la mejor capa de color para procesar las imágenes de muestras de esputo con la tinción de Ziehl-Neelsen es la capa Q del espacio de color NTSC.



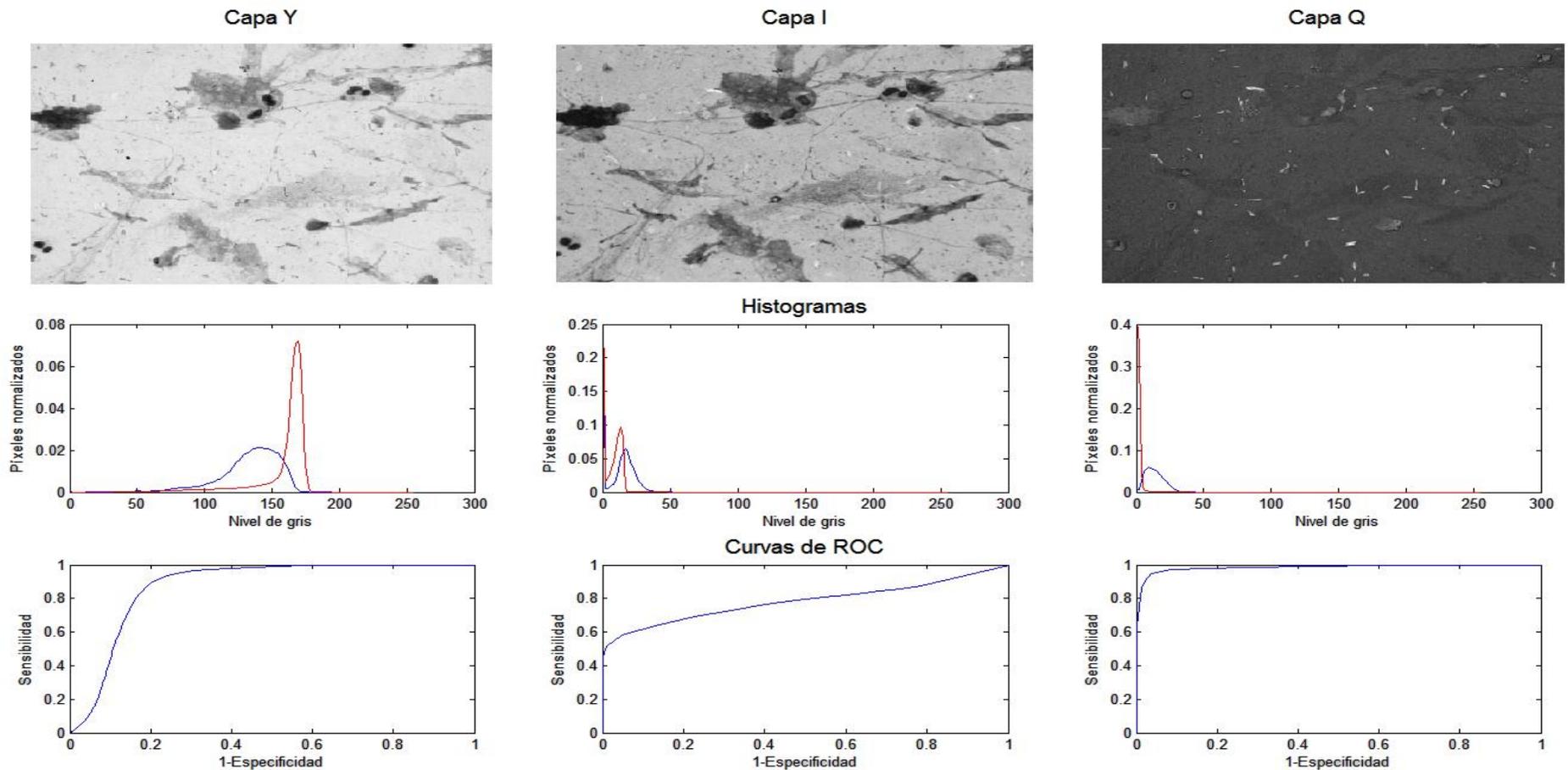


Figura 3.5: Análisis del espacio de color NTSC (YIQ). Cada una de las columnas corresponde a una capa de color (de izquierda a derecha tenemos la capa Y, I y Q). En la fila superior tenemos la representación de la imagen de la Figura 3.1 en las distintas capas de color. En la fila central tenemos los histogramas acumulados normalizados de los píxeles del fondo (curva roja) y de los bacilos (curva azul). Finalmente, en la fila inferior encontramos las curvas de ROC obtenidas a partir de los histogramas.

Tabla 3.1: Áreas bajo la curva de las gráficas de ROC para cada capa de los diferentes espacios de color analizados.

Espacio de color	Capa 1	Capa 2	Capa 3
RGB	0.70764	0.92397	0.95591
NTSC (YIQ)	0.87527	0.78298	0.98436
YCbCr	0.87522	0.73818	0.86898
HSV	0.86426	0.84178	0.75463
CMY	0.70762	0.92395	0.95589
HSI	0.86426	0.8154	0.86754
CIE L*a*b*	0.88703	0.94165	0.75497

3.4 DETECCIÓN DE BORDES Y OPERACIONES MORFOLÓGICAS

Luego de haber elegido la capa de color óptima para el procesamiento de las imágenes, se aplicaron las operaciones de detección de bordes con el fin de segmentar las imágenes. Se realizaron pruebas con los diferentes detectores de borde mencionados en la sección 2.3 (detector de borde de Roberts, Prewitt, Sobel y Canny), obteniendo un mejor resultado con el detector de borde de Canny. Sin embargo, a pesar de que en los estudios mencionados anteriormente se habían conseguidos muy buenos resultados con la aplicación de detectores de borde en muestras con Auramina [5] [16], en nuestro caso no se logró segmentar de manera óptima las imágenes dado que, tal como lo menciona Sadaphal et al. en su trabajo, la detección de bordes basados en la gradiente de color no funcionan adecuadamente para muestras teñidas con Ziehl-Neelsen, debido a que estas poseen mayores detalles de fondo que las imágenes con tinción de fluorescencia [17]. Además, la utilización de la operación morfológica de cierre para unir los bordes rotos tampoco generaba buenos resultados; pues si se utilizaba un elemento estructural grande, los objetos muy cercanos terminaban uniéndose, generando cúmulos de bacilos y unión de bacilos con detalles del fondo provocando anomalías en la forma de los bacilos; y si se utilizaba un elemento estructural pequeño, los bordes rotos no se unían correctamente provocando también anomalías en la forma de los bacilos. Por lo tanto, la utilización de detectores de bordes como método de segmentación fue descartada.

3.5 UMBRALIZACIÓN ADAPTATIVA

Debido a que no se consiguieron buenos resultados al utilizar la detección de bordes, se buscó segmentar las imágenes mediante una umbralización de color. Además, esta umbralización debería ser capaz de adaptarse a las variaciones de los valores de intensidad de las imágenes por lo que no era viable utilizar un valor umbral fijo. Así, se decidió implementar el método de Otsu para calcular automáticamente el valor umbral óptimo para una imagen. El método de Otsu es un algoritmo que asume que la imagen a ser umbralizada contiene dos clases de píxeles (píxeles de fondo y de bacilos, en nuestro caso), para así calcular el óptimo valor umbral que separe ambas clases [30].

En general, este valor umbral es calculado de la siguiente manera [30]:

Sean los píxeles de la imagen representados en L niveles de gris $[1, 2, \dots, L]$, la cantidad de píxeles con un nivel de intensidad i sea n_i , y N , el número total de píxeles de la imagen. Entonces el histograma es normalizado y representado como una distribución probabilística:

$$p_i = \frac{n_i}{N}, \quad \sum_{i=1}^L p_i = 1 \quad (3.1)$$

Ahora, asumiendo que las dos clases de píxeles sean C_0 y C_1 para un valor umbral k . C_0 denota los píxeles con niveles $[1, \dots, k]$, y C_1 denota los píxeles con niveles $[k + 1, \dots, L]$. Entonces, los momentos acumulativos de orden cero y uno del histograma hasta el nivel k están dados, respectivamente, por:

$$\omega(k) = \sum_{i=1}^k p_i \quad (3.2)$$

$$\mu(k) = \sum_{i=1}^k ip_i \quad (3.3)$$

Y el nivel promedio total de la imagen original es:

$$\mu_T = \mu(L) = \sum_{i=1}^L ip_i \quad (3.4)$$

Por último, el óptimo valor umbral que separe ambas clases estará dado por el umbral k que maximice σ_B^2 , donde:

$$\sigma_B^2(k) = \frac{[\mu_T \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \quad (3.5)$$

Es decir, el óptimo valor umbral k^* es:

$$\sigma_B^2(k^*) = \max_{1 \leq k < L} \sigma_B^2(k) \quad (3.6)$$

Sin embargo, si se aplicara el método de Otsu en toda la imagen, no se obtendría el óptimo valor umbral que separe las clases de píxeles que deseamos (píxeles del fondo y de los bacilos), pues en el histograma de la imagen sólo se pueden apreciar los píxeles del fondo, esto debido al hecho de que la cantidad de píxeles del fondo es mucho mayor que el de los bacilos. Por ello, fue necesario obtener una imagen en la cual la cantidad de píxeles de fondo y de bacilos sean aproximadamente proporcionales, para luego aplicar el método de Otsu a dicha imagen, todo esto de manera automática.

Para lograr dicho propósito, primero se aplicó un filtro gaussiano para suavizar la imagen de la Capa Q, luego se utilizó el detector de bordes de Sobel, pues este detector, a pesar de no detectar correctamente todos los bacilos presentes en la imagen, permite generar una imagen binaria con bordes gruesos. Así, multiplicando dicha imagen binaria con la capa Q de la imagen original, es posible obtener una imagen en la que la cantidad de píxeles del fondo y de los bacilos sea proporcional. La Figura 3.6 ilustra el procedimiento realizado.

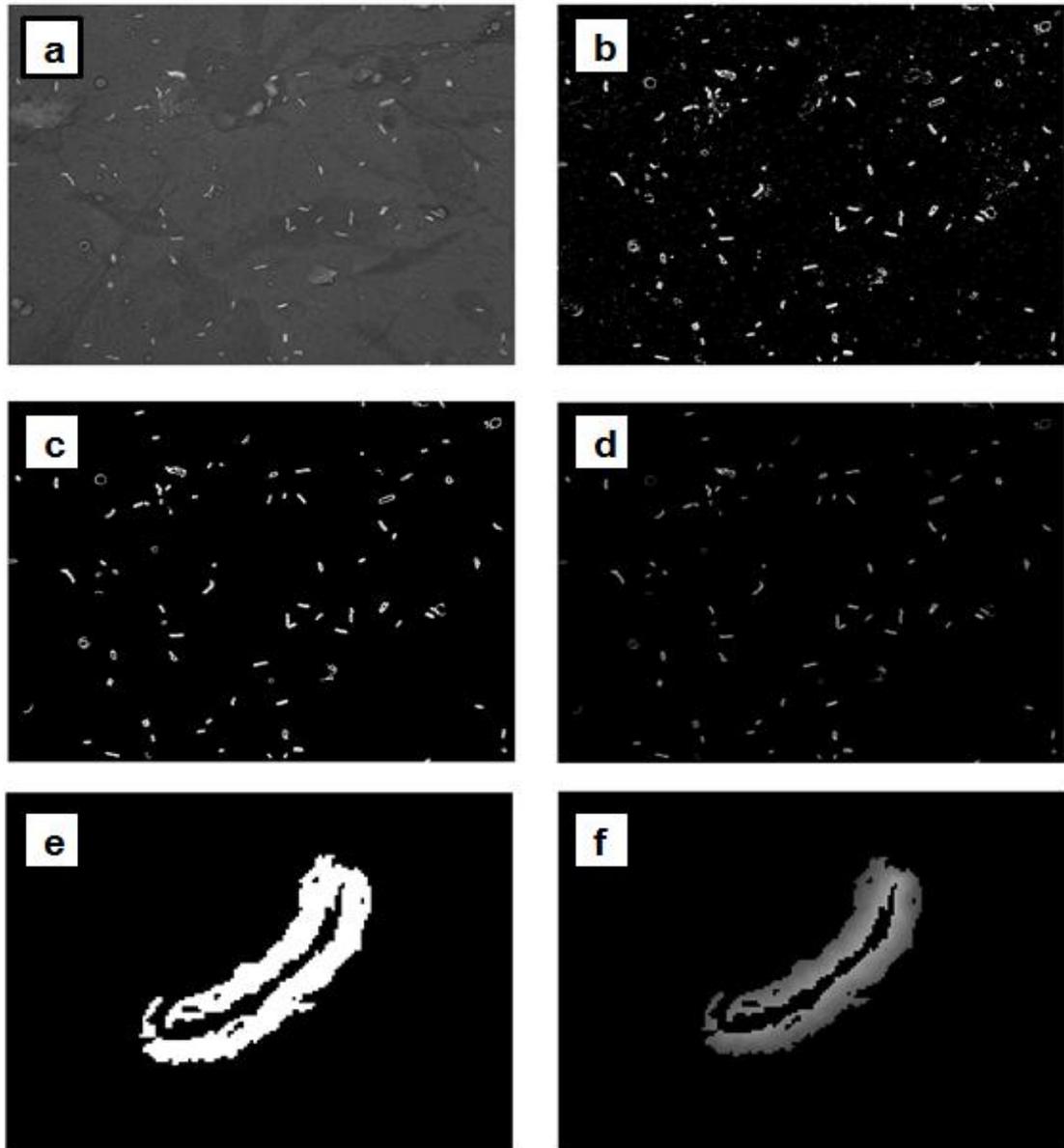


Figura 3.6: (a) Capa Q de la imagen original, (b) aplicación del filtro gaussiano y detector de bordes de Sobel, (c) filtrado de objetos pequeños, (d) producto de multiplicar la imagen (a) con la imagen (c), (e – f) detalle de un bacilo de las imágenes (c) y (d), respectivamente.

Como se puede apreciar en la Figura 3.6, el resultado de este procedimiento es una imagen con bordes gruesos de algunos bacilos en la que se tienen tanto píxeles del propio bacilo, como del fondo, de manera proporcional. Luego, se aplica el método de Otsu a la imagen resultante para hallar el óptimo valor umbral con el que se umbralizará la capa Q de la imagen original. La figura 3.7 muestra el resultado final de la umbralización adaptativa.



Figura 3.7: Resultado final de la umbralización adaptativa aplicada a la capa Q de la imagen de la Figura 3.1. Se puede apreciar que los bacilos poseen una forma abastonada y aproximadamente uniforme.

Los resultados obtenidos con la umbralización adaptativa no presentan tantas pérdidas de bacilos y anomalías en la forma de las mismas, como en el caso de la utilización de detectores de bordes y operaciones morfológicas para la segmentación.

3.6 EXTRACCIÓN DE CARACTERÍSTICAS Y CLASIFICACIÓN

Después de umbralizar la imagen, es necesario extraer las características de los objetos presentes en ella con el fin de clasificar correctamente los objetos que son bacilos y los que no lo son, teniendo siempre en cuenta de elegir la menor cantidad de descriptores suficientes que permitan una buena clasificación para reducir el costo computacional del algoritmo. Así mismo, la utilización de un sistema de clasificación sencillo permitiría reducir el tiempo de procesamiento del algoritmo y será menos complicado de implementar en otro lenguaje de programación, como el lenguaje C++, para poder ser implementado en un equipo médico.

Ahora, según los estudios de Forero et al. [16], los momentos geométricos y los descriptores de Fourier no son frecuentemente usados juntos, debido a que los descriptores de Fourier son muy susceptibles al ruido y los momentos geométricos por sí solos proveen una representación más compacta y robusta ante el ruido. Además, nos dice que para objetos con forma simétrica, los momentos de Hu de tercer orden y los de orden impar mayores a tres son cercanos a cero, por lo que el primer momento de Hu de tercer orden sería suficiente para reconocer objetos con forma simétrica. Finalmente, luego de analizar otros descriptores, Forero et al. concluyen que los primeros tres momentos de Hu (ϕ_1 , ϕ_2 y ϕ_3) que se muestran en las ecuaciones 2.30 , 2.31 y 2.32 de la sección 2.3, son suficientes para describir la forma de los bacilos y la utilización de un momento de Hu de mayor orden (como el momento Hu número 11) puede mejorar la separación entre clases.

Con el fin de implementar un sistema de clasificación que no sea muy complejo, con un costo computacional bajo y que sea veloz, se decidió utilizar un árbol de clasificación similar al utilizado en el trabajo de Sadaphal et al. [17]. Considerando, además de los momentos de Hu mencionados anteriormente, otros atributos geométricos como el área, la compacidad y la excentricidad, donde la compacidad provee una medida de cuánto la forma del objeto se asemeja a un círculo, y la excentricidad de un objeto se define como la relación entre el foco y la longitud del eje mayor de la menor elipse que contiene a dicho objeto [16]. Para ello se extrajeron manualmente 227 bacilos de las imágenes (ver Figura 3.8), para ser utilizados como bacilos de entrenamiento para calcular los valores máximos y mínimos de los diferentes descriptores de forma analizados.

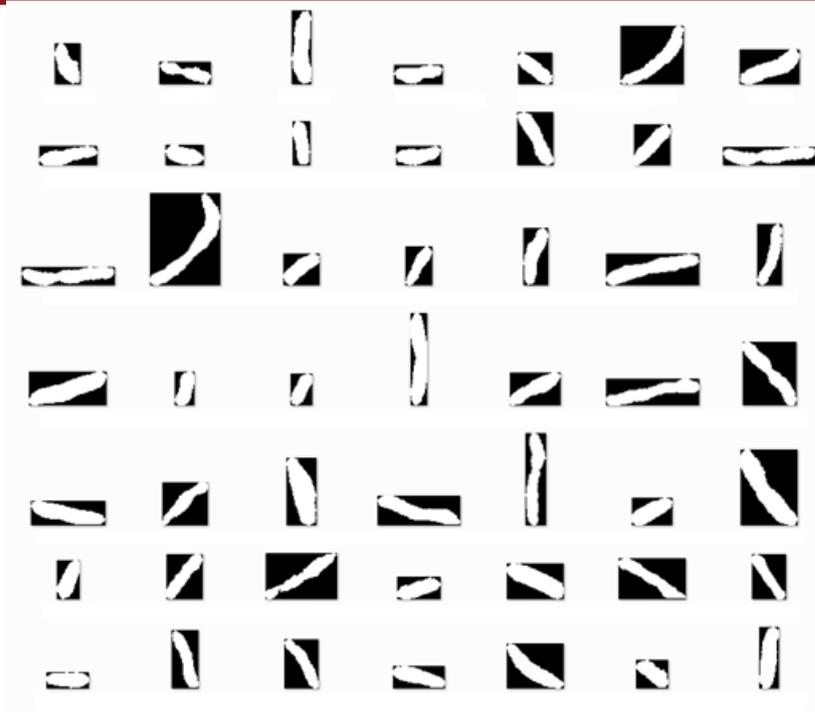


Figura 3.8: La imagen ilustra algunos de los bacilos de entrenamiento utilizados.

Luego de analizar los descriptores para los 227 bacilos de entrenamiento, sólo los atributos de área y excentricidad fueron elegidos para clasificar los objetos mediante un árbol de clasificación. Dado que se descartaron los primero dos momentos de Hu para evitar redundancias, pues la excentricidad e se puede calcular utilizando los primero dos momentos de Hu mediante la siguiente ecuación [16]:

$$e = \frac{2\sqrt{\phi_2}}{\phi_1 + \sqrt{\phi_2}} \quad (3.7)$$

Además, la compacidad y los momentos de Hu tres y once fueron descartados debido a que no brindaban información relevante para ser usados en un árbol de clasificación. Pero podrían ser de mayor utilidad si se utilizaran en otros sistemas de clasificación más complejos.

De esta manera, los valores máximos y mínimos de las áreas y excentricidades de los 227 bacilos de entrenamiento fueron calculados. Las áreas máximas y mínimas calculadas de bacilos típicos estaban dadas por 1708 y 313 píxeles, respectivamente; sin embargo, dado que el área no es un atributo geométrico invariante a escala, fue necesario considerar un rango más amplio, así, el área de

los bacilos fue definido en el rango de 200 y 2000 píxeles. Las excentricidades máximas y mínimas calculadas de bacilos típicos estaban definidas por 0.9925 y 0.8929, respectivamente, y no fue necesario hacer ningún tipo de ajuste, dado que la excentricidad es un descriptor invariante a escala, orientación y traslación. Finalmente, la Figura 3.9 nos muestra el sistema de clasificación utilizado para clasificar los objetos como bacilos u objetos que no son bacilos.

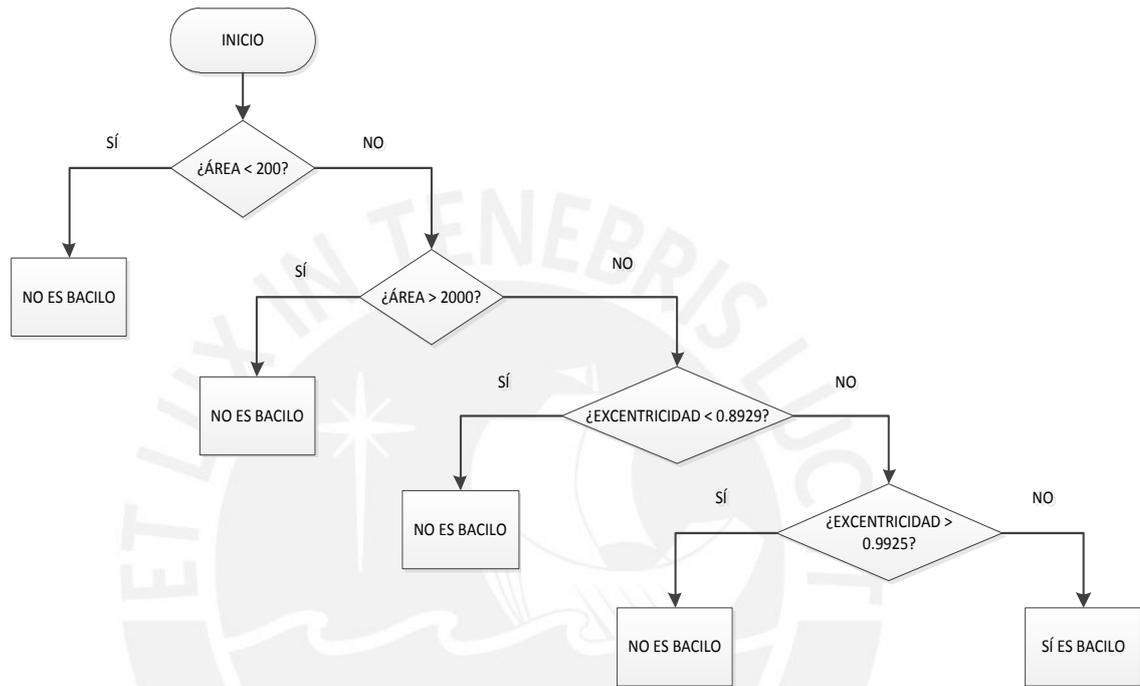


Figura 3.9: Árbol de clasificación utilizado para clasificar los objetos presentes en la imagen umbralizada. Los objetos que no se encuentren en el rango de las áreas y excentricidades son considerados como objetos que no son bacilos.

El Anexo C contiene el código para el análisis de los espacios de color y el algoritmo para la detección y conteo de bacilos implementados con el software Matlab®. Los resultados del algoritmo se verán en el siguiente capítulo.

3.7 IMPLEMENTACIÓN DEL ALGORITMO EN LENGUAJE C++

Luego de haber desarrollado un algoritmo capaz de detectar y contabilizar de manera satisfactoria el número de bacilos en imágenes de muestras de esputo, se procedió a implementar dicho algoritmo en un lenguaje de programación C++, con la finalidad de crear un programa o ejecutable que pueda ser utilizado para desarrollar un equipo médico capaz de diagnosticar automáticamente la enfermedad en tiempo real, interactuando con otros programas como los del control de motores para el movimiento de los ejes del microscopio, autoenfoco y control de la cámara digital. El entorno de desarrollo elegido para este propósito fue el Code::Blocks (<http://www.codeblocks.org/>), pues es un entorno de desarrollo integrado de código abierto y multiplataforma desarrollado en C++, que soporta múltiples compiladores, incluyendo MinGW, GCC, Digital Mars, Microsoft Visual C++, Borland C++, Watcom, LCC y el Intel C++ Compiler.

La programación inicial se realizó utilizando las librerías del software *Insight Sementation and Registration Toolkit* (ITK), dado que cuenta con varias de las funciones utilizadas para desarrollar nuestro algoritmo. ITK es un software orientado a objetos, de código abierto y multiplataforma creado en lenguaje C++, que provee de una amplia variedad de herramientas de procesamiento, segmentación y registro de imágenes digitales [31]. Luego de las pruebas de programación iniciales, la utilización del software ITK fue dejada de lado, esto debido al hecho de que varias de las funciones que ofrece ITK no brindaban exactamente los mismos resultados que las funciones utilizadas en Matlab® ya que no han sido desarrolladas con las mismas técnicas. Sin embargo, se siguió utilizando las librerías del software ITK para la lectura de las imágenes y la extracción de las capas de color rojo, verde y azul.

Así, se buscó implementar personalmente cada uno de los algoritmos y funciones utilizadas por nuestro algoritmo de detección y conteo de bacilos desarrollado en el software Matlab®. Se verificó siempre que los resultados de cada etapa sean los mismo que se obtuvieron en Matlab®. De esta manera, la conversión del espacio de color RGB al espacio NTSC (YIQ), aplicación de filtro gaussiano, detección de bordes, método de Otsu, umbralización, etiquetado y cálculo de áreas y excentricidades, fueron desarrollados en lenguaje de programación C++, obteniendo resultados aproximadamente iguales a los que se consiguieron con nuestro algoritmo desarrollado en Matlab®. La principal diferencia que se presentó fue el del valor umbral calculado por el método de Otsu con la función *graythresh* de

Matlab[®] y el calculado con el algoritmo implementado en lenguaje C++, ya que para aplicar el método de Otsu en lenguaje C++ fue necesario ajustar y redondear los valores de las intensidades de los píxeles de la capa Q para que se encuentren en un rango de valores enteros de 0 a 255, con el fin de poder generar un histograma.

Con la finalidad de mejorar el tiempo de ejecución, se aplicaron diferentes cambios al algoritmo implementado en lenguaje C++. Los cambios más resaltantes fueron la simplificación de las operaciones matemáticas para hallar los atributos geométricos de los bacilos, la optimización del análisis de los objetos detectados reduciendo el rango de análisis al mínimo rectángulo que contiene a cada objeto, y la eliminación de los objetos que se encuentren fuera del rango de áreas con la finalidad de reducir significativamente el número de objetos a ser etiquetados y analizados. Este último fue el cambio más significativo que permitió reducir el tiempo de ejecución del algoritmo, esta eliminación se realiza antes del etiquetado de los objetos y se logra mediante un barrido de la imagen binaria resultante de la umbralización. Así, cada vez que se encuentre un píxel de valor '1', se almacena en memoria la posición del píxel y se analizan sus vecindades, y de tener píxeles vecinos también con valor '1', se almacenan en memoria las posiciones de dichos píxeles y se analizan sus vecindades, y así sucesivamente hasta recorrer la totalidad del objeto. Finalmente, si el número de píxeles del objeto no se encuentra dentro del rango de valores de área para ser considerado como bacilo, se les da valor '0' a todos los píxeles de dicho objeto. (El código en C++ se encuentra en el Anexo D).

Por último, el ejecutable desarrollado en C++ fue integrado a un microscopio automatizado para la automatización del proceso de análisis de las muestras de esputo, interactuando con programas de control de motores de paso, de autoenfoco y de control de la cámara digital. Para este último, se utilizó el kit de desarrollo de software (SDK, por sus siglas en inglés) que brinda la misma corporación Olympus para el control de sus cámaras digitales. Cabe resaltar que el algoritmo implementado en C++ se ajustó para trabajar con imágenes en formato Jpeg dado que la cámara no podía entregar archivos en formato Tiff directamente mediante el controlador desarrollado con el SDK. Por otro lado, debido a la falta de precisión de los motores de paso, la cámara no lograba capturar imágenes adecuadas para el procesamiento, pues en la mayoría de casos estas imágenes estaban totalmente desenfocadas y era imposible analizarlas.

CAPÍTULO 4

PRUEBAS Y RESULTADOS

4.1 ALGORITMO EN FUNCIONAMIENTO

En resumen, el algoritmo se inicia con la extracción de la capa Q del espacio de color NTSC (YIQ) de la imagen original. Luego se procede a aplicar el filtro gaussiano y el detector de bordes de Sobel para obtener una imagen binaria de bacilos con bordes gruesos, para que al multiplicarlo por la Capa Q se pueda conseguir una imagen en la que la cantidad de píxeles del fondo y de los bacilos sean proporcionales, permitiendo así calcular el óptimo valor umbral utilizando el método de OTSU, y con este valor umbral calculado, se procede a realizar la umbralización de la Capa Q. Cabe resaltar que luego de umbralizar la imagen, se eliminan los objetos que se encuentren particionados por los bordes de la imagen, ya que, aunque el objeto parezca un bacilo, no son contabilizados por el laboratorista, pues puede inducir a errores. Finalmente, se etiquetan los objetos en la imagen para analizarlos uno por uno y clasificarlos como bacilos u objetos que no son bacilos según el árbol de clasificación implementado. La Figura 4.1 ilustra las etapas del algoritmo.

4.2 PRUEBAS

La Tabla 4.1 y 4.2 nos muestran las pruebas realizadas ejecutando el algoritmo en Matlab® y en lenguaje C++, respectivamente, para 25 imágenes distintas a las imágenes utilizadas para extraer los bacilos de entrenamiento. Como se mencionó en el capítulo anterior, las imágenes fueron adquiridas mediante un microscopio Olympus CX31 con un lente de aumento de 100x y una cámara digital Olympus EVOLT E-520 a una resolución de 3648 x 2736 píxeles. Además, las imágenes fueron almacenadas en formato Jpeg debido a que la cámara no puede entregar archivos en formato Tiff directamente mediante el controlador desarrollado con el SDK sin utilizar el software de transferencia y conversión de archivos de la misma corporación Olympus. Los programas se ejecutaron en una computadora personal con procesador Intel Core i7 de 2.20GHz y 6GB de RAM. Cabe mencionar que para realizar la adquisición de las imágenes se agregó una gota de aceite de inmersión a las muestras y se colocó el nivel 5.5 de intensidad de luz de la lámpara del microscopio. El recorrido de las muestras se realizó empezando desde el extremo

inferior derecho de la pátina y avanzando hacia la izquierda hasta llegar al otro extremo de la pátina, para luego subir y comenzar a avanzar hacia la derecha y así sucesivamente. Se capturaron aproximadamente 3 campos con una gran cantidad de bacilos por cada recorrido horizontal, haciendo un enfoque antes de capturar cada imagen. Obteniendo así 15 imágenes de cada una de las 3 muestras, de las cuales se tomaron 25 imágenes, distintas a las utilizadas para extraer los bacilos de entrenamiento, para las pruebas.

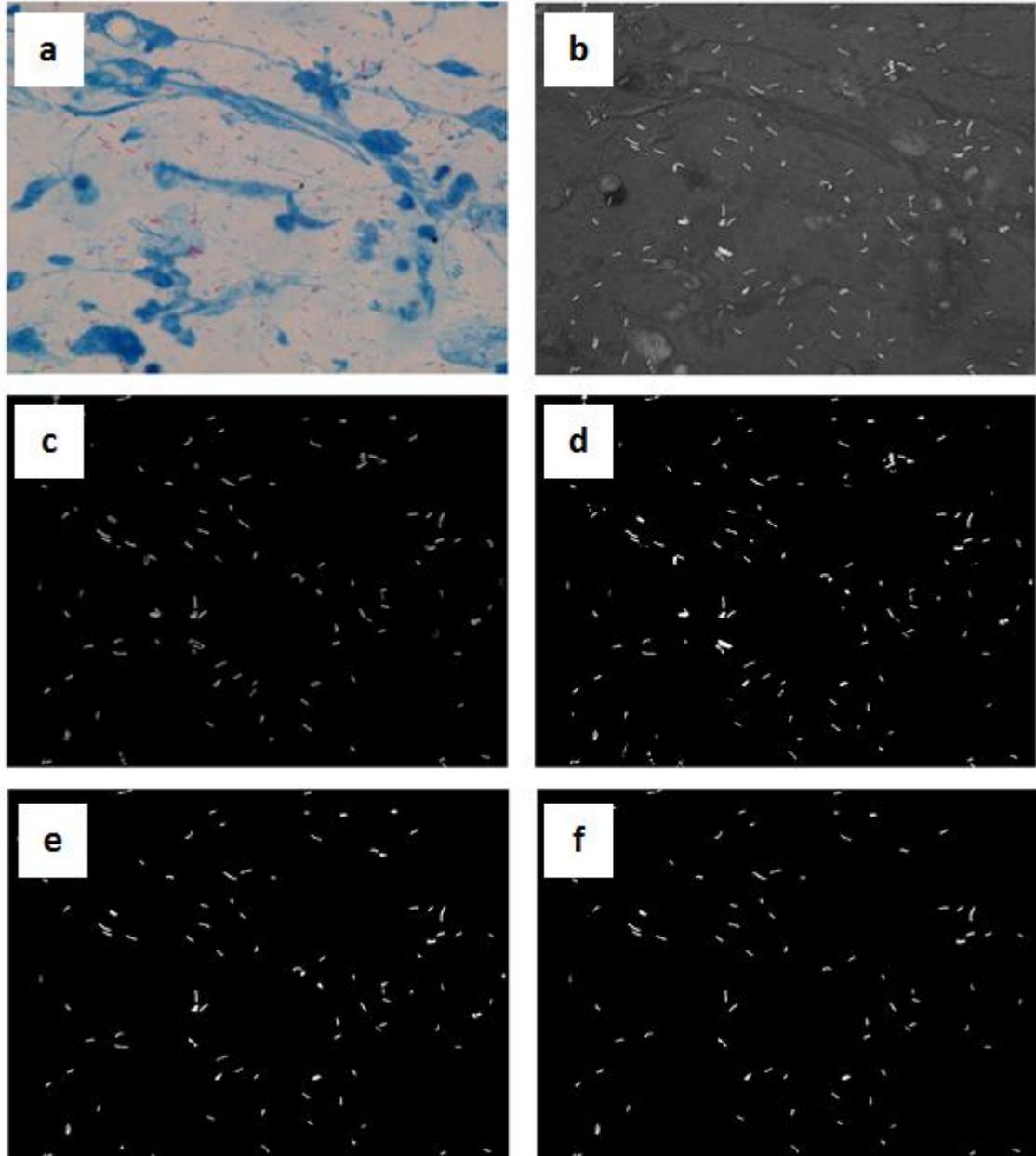


Figura 4.1: El algoritmo en funcionamiento. (a) imagen original, (b) capa Q de la imagen original, (c) producto de multiplicar la capa Q con la imagen binaria obtenida de la detección de bordes para luego aplicar el método de Otsu, (d) umbralización de la capa Q, (e) eliminación de objetos que no se encuentran

en el rango de áreas de los bacilos, (f) eliminación de objetos que no se encuentren en el rango de excentricidades de los bacilos.

Tabla 4.1: Objetos detectados y clasificados con el algoritmo en Matlab® y su tiempo de ejecución para 25 imágenes diferentes.

Imagen	Bacilos totales en la imagen	Objetos detectados al umbralizar	Bacilos perdidos al umbralizar	Bacilos clasificados correctamente	No-bacilos clasificados como bacilos	Tiempo (segundos)
1	68	1619	3	63	3	89.568
2	72	316	1	70	4	22.942
3	26	104	4	19	0	8.843
4	57	159	3	52	3	13.753
5	27	1127	0	25	9	69.632
6	31	435	0	30	7	28.584
7	44	1024	1	42	2	63.423
8	43	2030	2	38	10	102.971
9	39	788	4	34	2	49.388
10	75	388	3	67	1	26.894
11	62	267	3	55	3	20.217
12	36	263	1	32	1	20.004
13	44	809	2	39	5	51.099
14	58	1682	2	53	7	103.366
15	69	297	1	65	3	22.313
16	45	355	1	43	4	24.278
17	34	1120	1	30	3	68.579
18	53	991	5	45	1	61.073
19	59	1066	1	56	8	66.431
20	50	216	0	49	4	16.441
21	43	98	0	41	4	8.807
22	38	254	2	34	2	18.339
23	26	269	3	22	5	19.188
24	42	96	2	38	2	8.899
25	21	328	0	21	2	22.499
TOTAL	1162	16101	45	1063	95	1007.531

Tabla 4.2: Objetos detectados y clasificados con el algoritmo en C++ y su tiempo de ejecución para 25 imágenes diferentes.

Imagen	Bacilos totales en la imagen	Objetos detectados al umbralizar	Bacilos perdidos al umbralizar	Bacilos clasificados correctamente	No-bacilos clasificados como bacilos	Tiempo (segundos)
1	68	1603	3	63	2	11.123
2	72	316	1	70	3	11.138
3	26	95	4	19	0	6.256
4	57	152	5	50	1	9.142
5	27	1212	0	25	9	8.58
6	31	621	0	30	8	7.816
7	44	749	3	40	4	8.983
8	43	2019	2	38	10	9.516
9	39	787	4	34	2	8.174
10	75	354	3	65	1	10.468
11	62	144	4	52	1	9.516
12	36	187	2	30	1	7.207
13	44	803	2	39	3	9.142
14	58	1713	2	53	9	12.059
15	69	298	1	65	2	10.483
16	45	182	2	42	3	8.72
17	34	1123	1	30	3	8.424
18	53	1008	5	45	3	9.001
19	59	1068	1	56	8	10.639
20	50	213	0	49	4	9.282
21	43	98	0	41	1	8.096
22	38	255	2	34	2	7.831
23	26	271	3	22	5	7.301
24	42	109	2	35	1	8.018
25	21	335	0	20	2	7.16
TOTAL	1162	15715	52	1047	88	224.075

Además, se realizaron pruebas aumentando el rango de valores de áreas y de excentricidades para la cual un objeto podría ser clasificado como bacilo. Sin embargo, sólo se lograba aumentar en una cantidad reducida el número de bacilos clasificados correctamente (aproximadamente 3 bacilos más por imagen), mientras que el número de objetos clasificados incorrectamente como bacilos aumentaban considerablemente (aproximadamente 30 objetos más por imagen), viéndose reducido de manera significativa la exactitud del algoritmo.

4.3 RESULTADOS

A partir de los datos obtenidos de las pruebas y que se muestran en la Tabla 4.1 y 4.2, podemos hallar los valores de verdaderos positivos, verdaderos negativos, falsos positivos y falsos negativos, con la finalidad de calcular la sensibilidad, especificidad y exactitud del algoritmo.

Sea para el algoritmo implementado en Matlab®:

$$VP_1 = 1063 \quad (4.1)$$

$$FN_1 = 1162 - 1063 = 99 \quad (4.2)$$

$$VN_1 = 16101 - 1162 - 45 - 95 = 14799 \quad (4.3)$$

$$FP_1 = 95 \quad (4.4)$$

Finalmente, a partir de las ecuaciones definidas en la sección 1.2 (ecuaciones 1.1 – 1.3), podemos calcular la sensibilidad, especificidad y exactitud del algoritmo:

$$Sensibilidad_1 = \frac{VP_1}{VP_1 + FN_1} = \frac{1063}{1063 + 99} = 0.9148 \quad (4.5)$$

$$Especificidad_1 = \frac{VN_1}{VN_1 + FP_1} = \frac{14799}{14799 + 95} = 0.9936 \quad (4.6)$$

$$Exactitud_1 = \frac{VP_1 + VN_1}{VP_1 + VN_1 + FP_1 + FN_1} = \frac{1063 + 14799}{1063 + 14799 + 95 + 99} = 0.9879 \quad (4.7)$$

De manera análoga para el algoritmo implementado en lenguaje C++:

$$VP_2 = 1047 \quad (4.8)$$

$$FN_2 = 1162 - 1047 = 115 \quad (4.9)$$

$$VN_2 = 15715 - 1162 - 52 - 88 = 14413 \quad (4.10)$$

$$FP_2 = 88 \quad (4.11)$$

$$\text{Sensibilidad}_2 = \frac{VP_2}{VP_2 + FN_2} = \frac{1047}{1047 + 115} = 0.9010 \quad (4.12)$$

$$\text{Especificidad}_2 = \frac{VN_2}{VN_2 + FP_2} = \frac{14413}{14413 + 88} = 0.9939 \quad (4.13)$$

$$\text{Exactitud}_2 = \frac{VP_2 + VN_2}{VP_2 + VN_2 + FP_2 + FN_2} = \frac{1047 + 14413}{1047 + 14413 + 88 + 115} = 0.9870 \quad (4.14)$$

Podemos observar que la sensibilidad y exactitud son menores para el caso del programa implementado en lenguaje C++. Esta variabilidad se debe a que el valor umbral calculado por el método de Otsu no es exactamente el mismo que el que se calcula con la función del software Matlab[®], pero los niveles de sensibilidad, especificidad y exactitud siguen siendo mayores al 90%. Por otro lado, se ha logrado reducir considerablemente el tiempo de ejecución del algoritmo al implementarlo en lenguaje C++. El tiempo promedio para el procesamiento de las imágenes es de 40.301 segundos y 8.963 segundos para el algoritmo implementado en Matlab[®] y el algoritmo implementado en lenguaje C++, respectivamente.

CONCLUSIONES

- ❖ Se ha desarrollado e implementado un algoritmo de detección y conteo automático de bacilos de tuberculosis presentes en imágenes de muestras de esputo, obteniendo resultados con niveles de sensibilidad, especificidad y exactitud mayores al 90%.
- ❖ El algoritmo implementado en lenguaje C++ posee niveles de sensibilidad y exactitud menores al algoritmo implementado con el software Matlab® debido a que las técnicas utilizadas para calcular el valor umbral no son exactamente los mismos. Sin embargo, la diferencia es mínima y los niveles de sensibilidad, especificidad y exactitud siguen siendo superiores al 90%.
- ❖ Se logró reducir el tiempo de ejecución del algoritmo al implementarlo en lenguaje C++. El algoritmo en C++ posee un tiempo de ejecución promedio de 8.963 segundos por imagen; es decir que el tiempo que tomaría analizar una muestra, en el peor de los casos (se deben analizar hasta 100 campos si no se encuentran suficientes bacilos para realizar un diagnóstico), se demoraría aproximadamente 15 minutos. Este tiempo es menor al que le toma a un especialista en promedio diagnosticar la enfermedad (de 40 minutos a 3 horas [10]).
- ❖ El algoritmo implementado en lenguaje C++ fue utilizado para la automatización del proceso de análisis de las muestras de esputo, interactuando con programas de control de motores de paso, de autoenfoco y de control de la cámara digital. Sin embargo, debido a la poca precisión de los motores de paso, no se pudo lograr una automatización satisfactoria del proceso. A pesar de esto, sí se logró una interacción satisfactoria entre el algoritmo de captura de la imagen (control de la cámara digital) y el algoritmo de conteo y detección de bacilos de tuberculosis.

RECOMENDACIONES

- ❖ Debido a que la calidad de imagen es un aspecto de suma importancia para el procesamiento de imágenes de muestras de esputo teñidas con la técnica de tinción de Ziehl-Neelsen, es recomendable contar con una cámara de alta resolución y, de ser posible, una que esté especialmente diseñada para ser utilizada con un microscopio.
- ❖ La utilización de más descriptores de forma debe ser considerada con la finalidad de discriminar con mayor precisión los objetos presentes en las imágenes. Además, se debería tomar en cuenta la utilización de algoritmos de selección del menor conjunto de descriptores suficientes para la descripción de los objetos.
- ❖ La división de la forma de los bacilos en subclases mejoraría la discriminación, dado que la forma de los bacilos no es completamente uniforme. Este procedimiento podría realizarse mediante el algoritmo de agrupamiento de k-medias.
- ❖ Se debería considerar implementar un sistema de clasificación mucho más eficiente y complejo que considere un espacio de características con varios vectores. Teniendo siempre en consideración que un sistema muy complejo podría significar un costo computacional demasiado elevado para nuestro propósito.
- ❖ La implementación del algoritmo en el lenguaje de programación C++ debe ser mejorado utilizando técnicas de programación más avanzadas como punteros inteligentes con la finalidad de reducir el tiempo de ejecución.

BIBLIOGRAFÍA

- [1] World Health Organization
2012 *Global Tuberculosis Report - 2012*. Francia: WHO Press.
- [2] Médecins Sans Frontières
2010 *Tuberculosis – Practical guide for clinicians, nurses, laboratory technicians and medical auxiliaries*. 5th Edition. Paris, Francia.
- [3] Ministerio de Salud del Perú
2006 *Manuales de capacitación para el manejo de la tuberculosis*. Módulo 1: La tuberculosis: Introducción. Lima, Perú.
- [4] Organización Panamericana de la Salud
2008 *Manual para el Diagnóstico Bacteriológico de la Tuberculosis. Parte I: Baciloscopía*.
- [5] Veropoulos, K.
2001 *Machine Learning approaches to medical decision making*. Tesis de doctorado. Reino Unido: Universidad de Bristol, Facultad de Ingeniería, Departamento de Ingeniería Matemáticas.
- [6] Steingart, K. R., Henry, M., Ng, V., Hopewell, P. C., Ramsay, A., Cunningham, J., Urbanczik, R., Perkins, M., Aziz, M. A. & Pai, M.
2006 “Fluorescence versus conventional sputum smear microscopy for tuberculosis: a systematic review”. *Lancet Infectious Diseases*. Vol 6, pp. 570-581.
- [7] Bennedsen, J. & Larsen, S. O.
1996 “Examination for tubercle bacilli by fluorescence”, *Scandinavian Journal of Respiratory Disease*. Vol. 47, pp. 114-120.
- [8] Toman, K.
2004 “What are the advantages and disadvantages of fluorescence microscopy?”, *Toman’s Tuberculosis Case detection, treatment, and monitoring – questions and answers*, T. Frieden, pp. 7-10, World Health Organization, Hong Kong, China.

- [9] Forero, M., Sroubek, F. & Cristóbal, G.
2004 "Identification of tuberculosis bacteria based on shape and color".
Real-Time Imaging. Vol. 10, pp. 251-262.
- [10] Sotaquirá, M., Rueda, L. & Narvarez, R.
2009 "Detection and quantification of bacilli and clusters present in sputum
smear samples: a novel algorithm for pulmonary tuberculosis
diagnosis". *Proceedings of International Conference on Digital Image
Processing*. Bangkok, Tailandia, pp. 117-121.
- [11] Gonzalez, R., Woods, R., Eddins, S.
2004 *Digital Image Processing Using MATLAB®*. 1st Edition. USA: Pearson
Education, Inc.
- [12] Mercimek, M., Gulez, K. & Mumcu, T.
2005 "Real object recognition using moment invariants". *Sadhana*. India,
vol. 30, part 6, pp. 765-775.
- [13] Bow, S.
2002 *Pattern Recognition and Image Processing, 2nd Edition*. New York:
Marcel Dekker Inc.
- [14] Theodoridis, S. & Koutroumbas, K.
2009 *Pattern Recognition*. 4th Edition. Elsevier: Academic Press.
- [15] Cristianini, N. & Shawe-Taylor, J.
2000 *An introduction to Support Vector Machines and other kernel-based
learning methods*. Cambridge, Reino Unido: Cambridge University
Press.
- [16] Forero, M., Cristóbal, G. & Desco, M.
2006 "Automatic identification of *Mycobacterium tuberculosis* by Gaussian
mixture models". *Journal of Microscopy*. Vol. 223, part 2, pp. 120-132.

- [17] Sadaphal, P., Rao, J., Comstock, G. W. & Beg, M. F.
2008 "Image processing techniques for identifying Mycobacterium tuberculosis in Ziehl-Neelsen stains". *International Journal of Tuberculosis Lung Disease*. Paris, vol. 12, n. 5, pp. 579-582.
- [18] Kanashiro, D.
2007 *Detección de bacilos de tuberculosis en las muestras de esputo mediante procesamiento de imágenes digitales*. Tesis para optar el Título de Ingeniero Electrónico. Lima: Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería. [Documento sin publicar].
- [19] Aguilar, N.
2010 *Detección de bacilos de tuberculosis presentes en muestras de esputo por medio de descriptores de forma*. Tesis para optar el Título de Ingeniero Electrónico. Lima: Pontificia Universidad Católica del Perú, Facultad de Ciencias e Ingeniería.
- [20] Costa, M., Costa Filho, C., Sena, J., Salen, J. & Lima, M.
2008 "Automatic identification of mycobacterium tuberculosis with conventional light microscopy". *Proceedings of the 30th Annual International Conference of the IEE EMBS*, pp. 382-385.
- [21] Khutlang, R., Krishnan, S., Dendere, R., Whitelaw, A., Veropoulos, K., Learmonth, G. & Douglas, T. S.
2010 "Classification of Mycobacterium tuberculosis in Images of ZN-Stained Sputum Smears". *IEEE Transactions on Information Technology in Biomedicine*. Vol.14, No. 4, pp. 949-957.
- [22] Osman, M. K., Mashor, M. Y., Saad, Z. & Jaafar, H.
2009 "Contrast enhancement for Ziehl-Neelsen tissue slide images using linear stretching and histogram equalization technique". *2009 IEEE Symposium on Industrial Electronics and Applications (ISIEA 2009)*. Kuala Lumpur, Malaysia, vol. 1, pp. 431-435.

- [23] Wahab, A. S. W., Mashor, M. Y., Salleh, Z., Shukor, S., Rahim, N. A., Idris, F. M., Hasan, H. & Noor, S. S.
2008 "Colour image enhancement for Ziehl-Neelsen slide images". *2008 International Conference on Electronic Design*. Penang, Malaysia, pp. 1-5.
- [24] Ford, A. & Roberts, A.
1998 "Colour space conversions". Reporte técnico. Universidad de Westminster. Londres, Inglaterra.
- [25] Tkalcic, M. & Tasic J.
2003 "Colour spaces – perceptual, historical and applicational background". *EUROCON 2003*. Liubliana, Eslovenia, pp. 304-308.
- [26] Canny, J.
1986 "A Computational Approach to Edge Detection". *Ieee transactions on pattern analysis and machine intelligence*. Vol. PAMI-8, No. 6, pp. 679-698.
- [27] Sonka, M., Hlavac, V. & Boyle, R.
1999 *Image Processing, Analysis and Machine Vision*. Pacific Grove, CA: PWS Publishing Co.
- [28] Hu, M-K.
1962 "Visual pattern recognition by moment invariants". *IRE Transactions on Information Theory*. Vol. 8, pp. 179-187.
- [29] Fawcett, T.
2006 "An introduction to ROC analysis". *Pattern Recognition Letters*. Elsevier B.V., vol. 27, pp. 861-874.
- [30] Otsu, N.
1979 "A threshold selection method from gray-level histograms". *IEEE transactions on systems, man, and cybernetics*. Vol. 9, No. 1, pp. 62-66.

- [31] Ibáñez, L., Schroeder, W., Ng, L., Cates, J. & Insight Software Consortium
2005 *The ITK Software Guide, Second Edition, Updated for ITK version
2.4.*

