



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PROGRAMA DE MAESTRÍA Y
DOCTORADO EN INGENIERÍA

FACULTAD DE INGENIERÍA

RECONOCIMIENTO DE TEXTURAS EN IMÁGENES MÉDICAS

T E S I S

QUE PARA OPTAR POR EL GRADO DE
MAESTRA EN INGENIERÍA
ELÉCTRICA – INSTRUMENTACIÓN

P R E S E N T A

CAROLINA GONZÁLEZ GONZÁLEZ



DIRECTORA DE TESIS
DRA. TETYANA BAYDYK

México, D. F., Ciudad Universitaria, 2011

JURADO ASIGNADO

Presidente: Dr. Ernst Kussul Mikhailovich

Secretario: Dra. Lucía Medina Gómez

Vocal: Dra. Tetyana Baydyk Mykolaivna

1^{er}. Suplente: Dr. Fernando Arámbula Cosío

2^o. Suplente: Dr. Miguel Moctezuma Flores

México, Distrito Federal, Ciudad Universitaria.
Centro de Ciencias Aplicadas y Desarrollo Tecnológico (CCADET)

TUTORA DE TESIS

Dra. Tetyana Baydyk

FIRMA

Agradecemos el apoyo de los proyectos CONACYT 50231, PAPIIT IN110510-3, PAPIIT IN11961-0, el proyecto de ICyTDF 330/2009, y al Centro de Investigaciones Regionales Dr. Hideyo Noguchi de la Universidad Autónoma de Yucatán por la base de imágenes de tejido.

Dedicatorias

Primera y principalmente le quiero dar gracias a *Dios*, como mi Padre Celestial y Creador, porque en todo tiempo me he dado cuenta que Él cumple los deseos más profundos del corazón cuando le amas con ternura y hoy que culmino esta etapa de estudio, sé que Él ha estado en todo momento conmigo. Le doy gracias por cada aprendizaje, tanto en lo profesional como en lo personal, y porque hasta este momento me ha dado aliento de vida para poder lograr cada meta que existe en mi mente y corazón.

Agradezco a mi padre, *Benito Gerardo*, porque sé que esto no hubiera sido posible sin su ayuda. Por ser mi padre, por cada lágrima, por cada palabra de ánimo, por cada reprensión, por cada momento que estuvo en mi vida desde mi nacimiento hasta ahora como profesional. Miles de gracias a mi madre, *Josefina*, porque siempre fue mi fuente de inspiración. Gracias a su ejemplo de ímpetu y de carácter para seguir adelante pese a las adversidades, por ser mi madre, por su amor, por sus palabras motivacionales y por cada uno de sus cuidados. A ambos, gracias por su amor, por ser incondicionales conmigo, por su ejemplo que me ha llevado a ser una mujer con valores. Gracias por cada sacrificio que tuvieron que hacer para que esta meta se logre. Este logro es suyo también. A, *Leonardo* y *Rita*, mis hermanos, gracias.

A mi hermosa sobrina *Melanie Daniela*, porque siempre me alegraba mi vida en un día de estrés y agobio aunque a veces no me dejaba estudiar por jugar con ella, o por llevarla a los parquecitos, o porque jugaba en mi computadora o se ponía a ver películas. Gracias por enseñarme la inocencia, la paciencia, la humildad, por disfrutar la vida, por tus besos, por tus pláticas... Te amo princesa.

A mis abuelos: *Ismael* y *Rita*, *Antonio* y *Herminia* (q.e.d.); gracias por su amor a mi vida.

A mis amig@s de la Facultad de Ingeniería y del Posgrado en Ingeniería: *Enrique Melchor Rueda*, *Jhonatan Ponce Miranda*, *Ing. Alfonso Celestino Martínez*, *Ing. Amelia Pérez Islas*, *Ing. Iván Vladimir Villagrana Macías*, *Ing. Mayelly Reynoso Andrade*, *Ing. Orlando Zaldívar Zamorategui* y *Dr. Francisco Leonel Silva González*; a cada uno de ustedes agradezco las palabras de ánimo, por las veces que se sentaron a mi lado y me escucharon angustiada y me dieron valor para no desistir; por cada ocasión en que me brindaron un hombro donde poder llorar para después levantarme y continuar en la carrera de la vida; por los abrazos, por los besos y los consejos, por su apoyo; por los asesoramientos en ingeniería; por ser mis cómplices durante diferentes etapas de mi vida; por sus llamadas y correos electrónicos cuando no estábamos físicamente cerca, por preocuparse por mi familia (en especial por la salud de mi mamá), por todas las experiencias vividas, por su amistad y amor incondicional. Cada uno de ustedes tiene un lugar muy especial en mi corazón y sé que indudablemente puedo contar con ustedes siempre.

A mis amig@s de la ICM: *Fam. Bárcenas Guel*, *Fam. Hernández Mancilla (Charly y Areli)*, *Alfredo Baxin Cagal*, *Blanca Adelfa Hernández González*, *Eurídice Claudine García Lemus*, *Geovana Peláez Méndez*, *Jorge Humberto Rangel Sierra*, *Mayeli Estrada Villegas*, *Marlene Velázquez González*, *Mirella Vázquez Torres* y *Oscar Ignacio Plazola Ruiz (mi hijo)*; gracias a cada uno de ustedes por sus oraciones, ayunos, consejos, por su apoyo, por las palabras de aliento cuando sentía desfallecer, por su amistad incondicional, por cada experiencia vivida con ustedes... Gracias por todo.

A *Beatriz Ancira* le agradezco el apoyo brindado y sus comentarios siempre oportunos. A *Alejandro Pérez Meléndez* le agradezco el curso de inglés y el apoyo en mi vida.

A mis amigas del Colegio de Ciencias y Humanidades: *María de los Ángeles Hernández Hernández* y *María del Rocío López Santiago*; gracias por sus palabras siempre de ánimo para ser mejor cada vez, no sólo en lo profesional sino en mi vida personal; por su apoyo en todo momento, por estar conmigo en cada “caída” de mi vida en donde tuve que levantarme para continuar mi andar; por las risas, por las lágrimas, por los conflictos y diferencias de opinión que llegamos a tener porque por medio de ellas nos unimos y fortalecimos más la amistad, por cada experiencia vivida, por la compañía, por los abrazos, por las anécdotas, por los correos electrónicos y las llamadas que alegraban mi día... Porque a pesar de tantos años transcurridos, seguimos siendo amigas incondicionales. Las amo con todo mi corazón, friends forever...

A mi incondicional amiga, *Jessica Salguero Samperio*, gracias nena por tu amor y amistad en todo momento; gracias por hacerme ver las cosas de una manera diferente y ayudarme a salir adelante en los momentos más difíciles, emocionalmente hablando. Gracias por escucharme y por darme un hombro donde poder llorar. Gracias por cada experiencia vivida a tu lado, porque todas han sido inigualables e irrepetibles. Je t’aime beaucoup. Les agradezco su amistad, su apoyo y su amor a mi vida no sólo a ti sino a cada miembro de tu familia; realmente han sido para mí como una segunda familia.

A mis amigos de parranda: *Fermín González Muñoz*, *Raúl Rubio* y *Ricardo García*, gracias por el apoyo en los momentos más difíciles de mi vida, por un cálido abrazo cuando lloraba sin parar y aún por las palabras que a veces lastimaron mi corazón pero que me hicieron reflexionar y entender qué era lo mejor para mi vida; por los tacos, la comida y las chelas, por las salidas a divertirnos y las pláticas donde siempre me hacían ver que podía dar siempre más y lo mejor de mí en todo, por hacerme saber que soy una mujer de gran valor e inigualable. Por ser mis confidentes en el tiempo más difícil de mi vida. Son la bandita, los aprecio y quiero mucho.

A mi tutora, *Dra. Tetyana Baydyk Mykolaivna*: gracias por la confianza, el apoyo incondicional, la paciencia, la humildad, el asesoramiento, la comprensión, pero sobre todo por los conocimientos y enseñanzas que me brindo a lo largo de mi maestría; por la inversión de tiempo para la propuesta, desarrollo y culminación de este proyecto. La admiro y respeto en demasía. Fue un placer trabajar con usted.

A mis sinodales: *Dr. Ernst Kussul Mikhailovich*, *Dra. Lucía Medina Gómez* y *Dr. Miguel Moctezuma Flores*: gracias por el tiempo dedicado a la revisión y corrección, asesoramiento, opiniones y sugerencias a este trabajo y por la confianza brindada.

A cada uno de *mis profesores* a lo largo de mi trayectoria académica, por el empeño y dedicación empleados en la formación de nuevos profesionistas; porque a través de cada uno de ustedes forje mi carácter y profesión.

A cada uno de mis *compañeros de generación 2010–1* de la maestría, en especial a *Ing. Guillermo Ramírez Zúñiga* e *Ing. Edgar Alfredo Galindo González*, gracias por la convivencia, la amistad y las experiencias en esta etapa de nuestra vida.

A mi alma mater, ***Universidad Nacional Autónoma de México (U.N.A.M.)***, por ser una Institución de gran reconocimiento y prestigio; por la calidad y el apoyo siempre brindado durante mi formación profesional y, por el orgullo de pertenecer a ella.

Al *Consejo Nacional de Ciencia y Tecnología (CONACYT)*, gracias por el apoyo económico brindado a lo largo de mis estudios de maestría.

Reconocimiento de texturas en imágenes médicas

Índice General

Prólogo.....	10
Resumen.....	11
Abstract.....	12
1. Enfermedad de Chagas.....	13
1.1 Introducción.....	13
1.2 Descubrimiento de la enfermedad.....	14
1.3 Chinches, el vehículo o vector.....	14
1.4 Descripción de la enfermedad y diagnóstico.....	15
1.5 Clasificación de la enfermedad de Chagas.....	16
1.5.1 Características de la enfermedad en el período agudo.....	16
1.5.2 Características de la enfermedad en el período intermedio o de latencia.....	17
1.5.3 Características de la enfermedad en el período crónico.....	17
1.6 La enfermedad de Chagas en México.....	18
1.7 Reconocimiento de imágenes de tejidos.....	24
1.7.1 Imágenes médicas.....	24
1.7.2 Reconocimiento de imágenes médicas por medio de tejido.....	25
1.7.3 Técnicas para la detección de enfermedades por medio de imágenes.....	27

2.	Imágenes.....	31
2.1	Introducción.....	31
2.2	Definición de imagen.....	32
2.3	Características de las imágenes.....	32
2.4	Reconocimiento de imágenes.....	34
	2.4.1 Extracción de características.....	35
	2.4.2 Cálculo de similitud.....	35
	2.4.3 Resultados de la búsqueda.....	35
2.5	Segmentación y etiquetado.....	35
2.6	Métodos de segmentación.....	36
	2.6.1 Umbralización.....	37
	2.6.2 Región creciente.....	37
	2.6.3 Clasificadores.....	38
	2.6.4 Agrupamiento.....	39
	2.6.5 Campos aleatorios de Markov.....	39
	2.6.6 Redes neuronales artificiales.....	40
3.	Redes Neuronales Artificiales.....	41
3.1	Introducción.....	41
3.2	¿Qué es una neurona biológica?.....	41

3.3	¿Qué son las redes neuronales artificiales?.....	42
3.4	Objetivo de las RNA.....	43
3.5	Aplicaciones de redes neuronales artificiales en medicina.....	45
3.6	Clasificador neuronal de umbrales aleatorios RTC (Random Threshold Classifier).....	46
3.6.1	Histograma de brillo.....	48
3.6.2	Histograma de contraste.....	49
3.6.3	Histograma de orientación de microcontornos.....	49
4.	El sistema desarrollado y sus resultados.....	52
4.1	Introducción.....	52
4.2	Objetivo principal.....	52
4.3	Descripción de la base de imágenes.....	53
4.4	Descripción de los experimentos.....	54
4.5	Análisis de resultados.....	56
	Conclusiones.....	67
	Trabajos a futuro.....	68
	Anexo. Programa fuente.....	69
	Bibliografía y referencias.....	93

Índice de Figuras

Figura 1.1 Seroprevalencia de infección por <i>T. cruzi</i> en México hasta 2004.....	13
Figura 1.2 a) <i>Triatoma dimidiata</i> b) Ninfas de <i>Triatoma dimidiata</i> c) <i>Eratyrus cuspidatus</i> d) <i>Panstrongylus rufotuberculatus</i> ; todos ubicados en el estado de Veracruz.....	15
Figura 1.3. Vectores causantes de Chagas en México.....	19
Figura 1.4. Prevalencia de la enfermedad de Chagas en México según estudios seroepidemiológicos y de patología.....	20
Figura 1.5 Casos de <i>Trypanosoma cruzi</i> , por Estado, hasta 2004.....	21
Figura 1.6 Distribución de tres especies de <i>Triatóminos</i> en México.....	22
Figura 1.7 Principales vectores localizados en México.....	23
Figura 1.8 Algunos ejemplos de imágenes médicas según su espectro electromagnético.....	26
Figura 2.1 Modelo de sistema de reconocimiento de imágenes.....	34
Figura 3.1 Neurona biológica.....	41
Figura 3.2 Modelo de una Red Neuronal Artificial.....	42
Figura 3.3. Estructura del clasificador neuronal RTC.....	46
Figura 3.4 Representación geométrica del Clasificador RTC.....	47
Figura 3.5 Histograma de brillo.....	48
Figura 3.6 Histograma de contraste.....	49
Figura 3.7 Píxeles vecinos.....	49
Figura 3.8 Nuevos píxeles vecinos.....	50
Figura 3.9 Histograma de orientación de microcontornos.....	51
Figura 4.1 Imagen de tejido.....	53
Figura 4.2 Imagen marcada.....	54
Figura 4.3 Imagen del programa que realiza el sistema desarrollado.....	56
Figura 4.4 Imágenes aleatorias.....	57
Figura 4.5 Imágenes restantes.....	57
Figura 4.6 Ventana de escaneo.....	58
Figura 4.7 Gráfica de porcentaje de reconocimiento.....	66

Índice de Tablas

Tabla 1.1 Estudios de seroprevalencia realizados en bancos de sangre en México.....	20
Tabla 3.1. Cálculo de los ángulos.....	51
Tabla 4.1 Resultados de entrenamiento para 4 experimentos (pesos positivos).....	59
Tabla 4.1 Resultados de entrenamiento para 4 experimentos (pesos positivos).....	60
Tabla 4.2 Resultados promedio de entrenamiento promedio para 4 experimentos (pesos positivos).....	61
Tabla 4.3 Resultados promedio de reconocimiento para 4 experimentos (pesos positivos).....	62
Tabla 4.4 Resultados de entrenamiento para 4 experimentos (pesos positivos y negativos).....	63
Tabla 4.4 Resultados de entrenamiento para 4 experimentos (pesos positivos y negativos).....	64
Tabla 4.5 Resultados promedio de entrenamiento para 4 experimentos (pesos positivos y negativos).....	65
Tabla 4.6 Resultados promedio de reconocimiento para 4 experimentos (pesos positivos y negativos).....	65

Prólogo

La presente tesis tiene como objetivo principal el desarrollo e investigación de un sistema de reconocimiento de texturas en imágenes médicas basado en el clasificador neuronal RTC realizado en Borland® C++ 6.0 para el caso particular de la enfermedad de Chagas.

En el *Capítulo 1* se da una breve introducción a la enfermedad de Chagas: su descubrimiento, descripción de la enfermedad y diagnóstico, así como su distribución en México. Se dió una breve descripción de las imágenes médicas y reconocimiento de éstas.

En el *Capítulo 2* se indican conceptos básicos, prácticos y necesarios sobre imágenes y los diferentes métodos para trabajar con ellas, enfocando esta tesis principalmente a las redes neuronales artificiales.

El *Capítulo 3* está dedicado al clasificador neuronal de umbrales aleatorios (RTC, *Random Threshold Classifier*). Dentro de este capítulo están presentados la estructura del clasificador RTC, sus algoritmos de funcionamiento y su interpretación geométrica.

En el *Capítulo 4* es presentado el sistema de visión computacional desarrollado y los resultados de reconocimiento de texturas de imágenes médicas, analizándose el funcionamiento del sistema.

Finalmente se presentan las Conclusiones, los Trabajos a futuro y el texto del programa en el Anexo.

Resumen

La imagenología es una disciplina de la medicina que emplea diferentes modalidades de imágenes del cuerpo humano, obtenidas mediante un conjunto de equipos y métodos para llegar en forma rápida y segura a la detección de diversas enfermedades, siendo una herramienta imprescindible para la atención adecuada y calificada de los pacientes.

Es por ello que el ámbito médico necesita de diversos instrumentos y software especializado para la detección y diagnóstico de las diversas enfermedades existentes. Es entonces donde la ingeniería se involucra, pudiendo desarrollar un software especializado en el reconocimiento de texturas en imágenes, que ayude a la detección temprana de enfermedades.

En la presente tesis se analizó un caso clínico particular: la enfermedad de Chagas, e ingenierilmente se realizó el reconocimiento de texturas basándose en el entrenamiento de una red neuronal artificial (un clasificador neuronal), utilizando como software Borland® C++ 6.0.

Para el reconocimiento de texturas en imágenes médicas se utilizó y analizó una base de imágenes de tejido infectado por la enfermedad de Chagas, calculándose las siguientes características: histograma de brillo, histograma de contraste e histograma de orientación de microcontornos.

El principal objetivo fue el desarrollo de un sistema de visión computacional para el reconocimiento de texturas de tejido dañado con la enfermedad de Chagas.

Se investigó la posibilidad de reconocimiento de la enfermedad utilizando imágenes de muestras de tejido dañado. Los resultados obtenidos muestran la posibilidad de utilizar el clasificador neuronal RTC (*Random Threshold Classifier*) como sistema de visión computacional para reconocer texturas de imágenes médicas, diferenciando el tejido sano del tejido infectado con la enfermedad de Chagas.

Se concluyó que el tejido infectado puede ser reconocido por el sistema de visión computacional siendo el mejor porcentaje de reconocimiento de 97.54%.

Abstract

The imagenologic is a medicine discipline that uses different modalities of human body images that are obtained by set of equipments and methods to detect in rapid and sure form the diverse diseases, being an indispensable tool for the suitable and qualified attention of the patients.

For this reason the medicine needs different instruments and specialized software for the detection and diagnosis of the diverse diseases. The engineering can develop a specialized software for the recognition of image textures that can help in the early detection of diseases.

In the present thesis we analyze a clinical particular case: Chagas's disease, and from the engineering point of view we realize the recognition of textures based on the training of a neural artificial network (a neural classifier), using the software Borland[®] C ++ 6.0.

It was used and analyzed the base of images of tissue infected by Chagas's disease. The following characteristics for the texture recognition in medical images were calculated: brightness histogram, contrast histogram and histogram of micro contour orientations.

The principal aim was the development of a computer vision system for the image recognition of tissue damaged with Chagas's disease.

There was investigated the quality of texture recognition using images of tissue samples. The obtained results show the possibility of the RTC (Random Threshold Classifier) neural classifier application in computer vision system to classify textures of medical images, differentiating the healthy tissue from the tissue infected with Chagas's disease.

The best result in image recognition, the best recognition rate, was obtained of 97.54 %.

1.2 Descubrimiento de la enfermedad

El descubrimiento de la enfermedad de Chagas fue realizado por el médico brasileño Carlos Justino Ribeiro Das Chagas, el 23 de abril de 1909, del parásito denominado más tarde *Trypanosoma (Schizotrypanum) cruzi* [3]. La aportación del científico fue triple, ya que describió una nueva enfermedad, identificó al agente causal y reconoció la sintomatología en el huésped humano; Carlos Chagas, además de sus descubrimientos, llamó la atención de las autoridades sobre el asunto de las viviendas infestadas del vector (chinche) y de la importancia social de la enfermedad. La cumbre endémica de la enfermedad de Chagas se dio entre 1940 y 1970, cuando se llegó a una incidencia de más de 200 mil casos anuales y una mortalidad elevada. Los programas de control iniciaron hasta 1960 y las iniciativas gubernamentales hasta 1991.

1.3 Chinchas, el vehículo o vector

La enfermedad de Chagas es ocasionada por el parásito *Trypanosoma (Schizotrypanum) cruzi* (el cual mide aproximadamente 0.02 milímetros) y afecta tanto a seres humanos como animales domésticos, y el vehículo o vector para la introducción al cuerpo humano es una chinche [4]. Dicho insecto (Figura 1.2) suele vivir en jardines, selvas y especialmente en techos de paja y pisos de tierra; el nombre que recibe depende de la región de ubicación. Por ejemplo, en Argentina, Chile, Paraguay y Uruguay la llaman “vinchuca”; en Brasil, “chupao”; en México se les conoce también como “chinche palota” (Nuevo León), “chinche compostela” (Nayarit), “pick” (Yucatán), “chinchón” (estados del sur-este), o “besucona”; en Centroamérica y Ecuador, “chirima”. Normalmente, la chinche pica en la zona de los pómulos de las personas cuando están dormidas; inyecta un anestésico para después succionar sangre, y a una distancia de alrededor de un centímetro y medio defecan, descargando junto con las heces a los parásitos en el estado infectante conocido como *trypomastigote metacíclico*, que mide aproximadamente 20 micras de longitud; éstos penetran al torrente sanguíneo por frotamiento a través de la herida producida por la picadura, o bien por la mucosa conjuntival o bucal o alguna vía de continuidad (heridas pequeñas). El problema es que el individuo, al terminar el efecto del anestésico siente un escozor por la picadura, lo que ocasiona que se rasque y lleve las heces hacia el agujero donde picó la chinche y se autoinfecta. Otras fuentes de infección en humanos son: transfusión sanguínea, transmisión intrauterina, transmisión lacto-génica, rara vez por alimentos contaminados por heces de triatóminos y en el laboratorio por errores o accidentes durante el manejo del parásito.

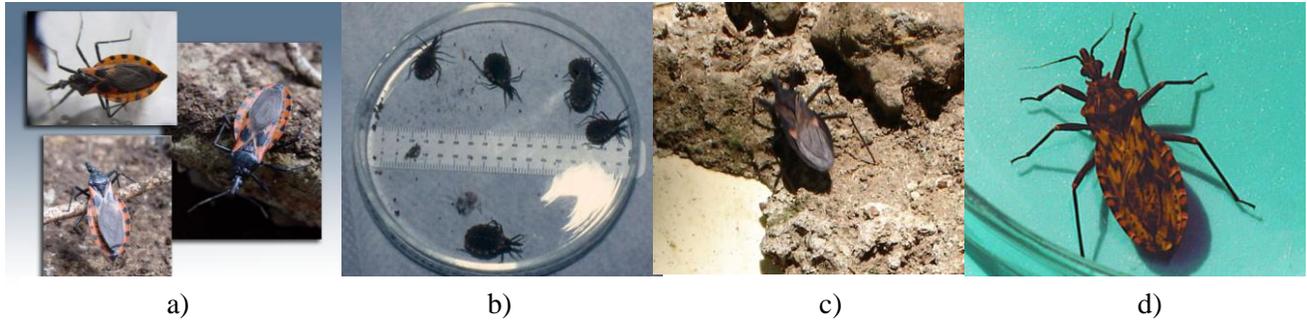


Figura 1.2 a) *Triatoma dimidiata* b) Ninfas de *Triatoma dimidiata* c) *Eratyrus cuspidatus*
 d) *Panstrongylus rufotuberculatus*; todos ubicados en el estado de Veracruz

1.4 Descripción de la enfermedad y diagnóstico

La enfermedad asociada con *Trypanosoma (Schizotrypanum) cruzi* (también conocida *Trypanosomiasis Americana*) es crónica, y causa inhabilidad severa. Se ha estimado que en el Continente Americano hay de 15 a 20 millones de personas infectadas; pruebas serológicas revelan que de 15 a 50 % de la población en áreas endémicas está infectada, 100 millones de personas están en riesgo y se presentan 50000 casos mortales por año [2]. Hay tres etapas principales en las personas: la etapa aguda, la etapa asintomática y la etapa crónica. En la primera y tercera etapas pueden verse afectados diversos órganos y la enfermedad puede ser mortal en cualquiera de ellas.

En ocasiones, se diagnostica por medio del ojo chagásico, que consiste en una inflamación aguda, ya que el sujeto se suele rascar también esa zona y causa la hinchazón. Aunque después de la inoculación del parásito son frecuentes el signo de Romaña (inflamación del párpado de un ojo u ojo chagásico) y el chagoma de inoculación (edema o roncha en la piel). El parásito invade al hospedador y destruye las células de tejido muscular cardíaco, fibra muscular lisa y células del sistema fagocitario mononuclear (macrófagos). El periodo inicial de infección está caracterizado por parasitemia intensa; los signos y síntomas son transitorios debido a la respuesta inmune del hospedador y por un cuadro febril. De acuerdo con la Organización Mundial de la Salud (OMS), ese padecimiento, prácticamente exclusivo de América, afecta a unos 18 millones de personas, aunque se sospecha que hay más casos que no se diagnostican correctamente. La enfermedad de Chagas causa un estimado de cincuenta mil muertes al año, lo cual significa que aproximadamente por cada hora se presentan cinco defunciones [5,6].

1.5 Clasificación de la enfermedad de Chagas

- Chagas agudo: Desde el momento de la picadura del insecto hasta alrededor de veinte a treinta días.
- Chagas indeterminado o asintomático: De duración variable, puede alcanzar varios años.
- Chagas con cardiopatía (Chagas crónico): La duración va a depender de la gravedad que alcance el proceso. Con pronósticos bien definidos correspondiendo a Grupo A y Grupo B.
 - Grupo A sin cardiomegalia: Arritmias, trastornos de conducción.
 - Grupo B con cardiomegalia: Insuficiencia cardíaca congestiva.

1.5.1 Características de la enfermedad en el período agudo

La enfermedad afecta mayoritariamente a los niños por estar más expuestos a ser infectados por la “chinche” o vector. El período de incubación (lapso de tiempo que media entre la introducción del *Trypanosoma* en el organismo y la aparición de los primeros síntomas) tiene una duración variable. El comienzo de los síntomas es súbito, ocurren entre la cuarta y octava semana y luego desaparecen siempre sin tratamiento. Cabe mencionar que muchas veces la persona no tiene síntomas que evidencien la enfermedad. En el periodo agudo, los síntomas pueden ser moderados o ausentes; dos o tres semanas después se presenta linfadenitis y edema generalizado, dolor muscular, fiebre, escalofríos, dolor de cabeza y de los músculos del cuerpo, adenopatías, malestar general e inapetencia y el complejo oftalmoganglionar, o signo de Romana, vómito y diarrea, bronquitis y miocarditis de intensidad variable: la fiebre es alta (intermitente, remitente o continua (37.5° a 39°)), puede haber hepato–esplenomegalia, mialgias y exantema eritematoso. Se puede observar también agrandamiento del hígado, y del bazo. El corazón suele presentar alteraciones variables de leves a graves. Los exámenes de laboratorio muestran anemia, linfocitosis, monocitos e incremento de las concentraciones de anticuerpos IgM. El periodo agudo de la enfermedad de Chagas termina en pocas semanas con la recuperación del paciente, o bien comienza con el estado crónico de la infección afectando el cerebro y conducir incluso a la muerte. El diagnóstico en el periodo agudo se establece por: alta parasitemia, incrementos de anticuerpos, anti *T. (S.) cruzi* tipo IgM por inmunofluorescencia, detección del parásito en sangre en un frotis de gota gruesa, búsqueda de amastigotes en ganglios y músculo estriado, xenodiagnóstico, entre otros.

1.5.2 Características de la enfermedad en el período intermedio o de latencia

Este periodo abarca de ocho a diez semanas, el enfermo entra en un segundo periodo que puede durar años. Durante ese tiempo no hay ningún síntoma; solamente se puede poner en evidencia la enfermedad por medio del análisis de sangre en la que se comprueba las alteraciones provocadas por la enfermedad. La mayor parte de las personas permanece en este período todo el resto de sus vidas.

Aunque el periodo indeterminado o de latencia se caracteriza por positividad serológica en individuos asintomáticos con electrocardiograma (ECG) y radiología normal para corazón, esófago y colón, pueden presentarse periodos variables de remisión, exacerbación marcada por la fiebre y la aparición de tripomastigotes en la sangre. La forma clínica en el periodo crónico se caracteriza por cardiopatía, formación de “megas”: megaesófago (esofagopatía) y megacolon (colopatía), se presentan alteraciones del sistema nervioso central y bronquio–ectasias (dilatación crónica de los bronquios). La glomerulonefritis por *T. (S.) cruzi*, es el resultado de depósitos de complejos inmunes en las membranas basales de los túbulos renales.

El diagnóstico en el periodo indeterminado o de latencia se establece solamente por serología positiva de cuando menos dos técnicas: hemaglutinación indirecta, ELISA, Western blot, inmunofluorescencia y prueba C reactiva (PCR).

1.5.3 Características de la enfermedad en el período crónico

De 10 a 20 años después de la infección, pueden desarrollarse los síntomas más severos. En personas con problemas inmunológicos (como el VIH (SIDA)), pueden ser aún mucho más complicados, además de que no todas las personas desarrollan síntomas en esta etapa de la enfermedad. Los principales síntomas son: problemas cardíacos (incluyendo dilatación del corazón), cambios en la frecuencia cardíaca o ritmo, fallas cardíacas, paros cardíacos, agrandamiento de distintas partes del tracto digestivo (con resultados tales como el estreñimiento (megacolon) o problemas para deglutir (megaesófago)). Los síntomas más comunes a lo largo de esta etapa tan larga son: palpitaciones, disnea, dolores referidos al área cardíaca, dolor en la zona hepática y manifestaciones típicas que se observan en el electrocardiograma. Las arritmias cardíacas más comunes en la miocardiopatía de Chagas son: bloqueo de rama derecha, hemibloqueo anterior izquierdo, arritmias ventriculares complejas, bradiarritmias secundarias a enfermedad del nódulo sinusal, bloqueos auriculo

ventriculares, prolapso de válvula mitral, miocardiopatía dilatada, entre otras. Para poder diagnosticar la enfermedad, así como la etapa en la que se encuentra se tienen una serie de estudios propuestos:

A nivel cardíaco:

- Estudio de Holter.
- Ecocardiografía.
- Prueba ergométrica graduada.
- Promediación de señales.
- Estudio del sistema nervioso autónomo.
- Ecodoppler arterial para detectar disfunción endotelial.
- Estudios de medicina nuclear (pool cardiovascular, perfusión miocárdica).
- Estudio electrofisiológico.
- Cateterismo cardíaco.
- Biopsia endomiocárdica.

A nivel digestivo:

- Estudios de manometría esofágica y rectal.
- Estudios contrastados baritados y con radiocoloides.
- Endoscopias.

La terapéutica de la enfermedad de Chagas es un problema complejo, ya que no existen fármacos específicos para su tratamiento en todas sus etapas. Actualmente hay dos medicamentos activos contra *T. (S.) cruzi*: Nifurtimox, del grupo de los nitrofuranos, y Benznidazol, del grupo de los nitroimidazoles. Ambos medicamentos suelen ser eficaces sólo en la fase aguda. Es necesario que los pacientes sean vigilados por un médico [3].

1.6 La enfermedad de Chagas en México

En México, la información sobre el predominio de Chagas (enfermedad de Chagas) es escasa e indica que se han presentado casos de Chagas en casi todos los Estados de la República Mexicana. Unos cuantos entomológicos han reportado treinta y nueve especies de triatómineos presentes en México, comprendidas en siete géneros. Exceptuando *Triatoma hegneri* en la Isla de Cozumel (Quintana Roo), todas las especies han sido detectadas en el área continental del país entre las latitudes 32°43'5'' y 14°32'45''. Muchas especies de triatóminos han sido encontradas naturalmente infectadas con *Trypanosoma cruzi*, entre ellas *Dipetalogaster maximus*, *Triatoma hegneri*, *Triatoma lecticularia*, *Triatoma nitida*, *Triatoma peninsularis*, *Triatoma protracta*, *Triatoma recurva*, *Triatoma rubida* y

Triatoma sinaloensis, y (con relevancia epidemiológica por la transmisión del parásito a los humanos) *Rhodnius prolixus*, *Triatoma barberi*, *Triatoma dimidiata*, *Triatoma gerstaeckeri*, *Triatoma longipennis*, *Triatoma Mazzotti*, *Triatoma mexicana*, *Triatoma pallidipennis*, *Triatoma phyllosoma* y *Triatoma picturata* (Figura 1.3).

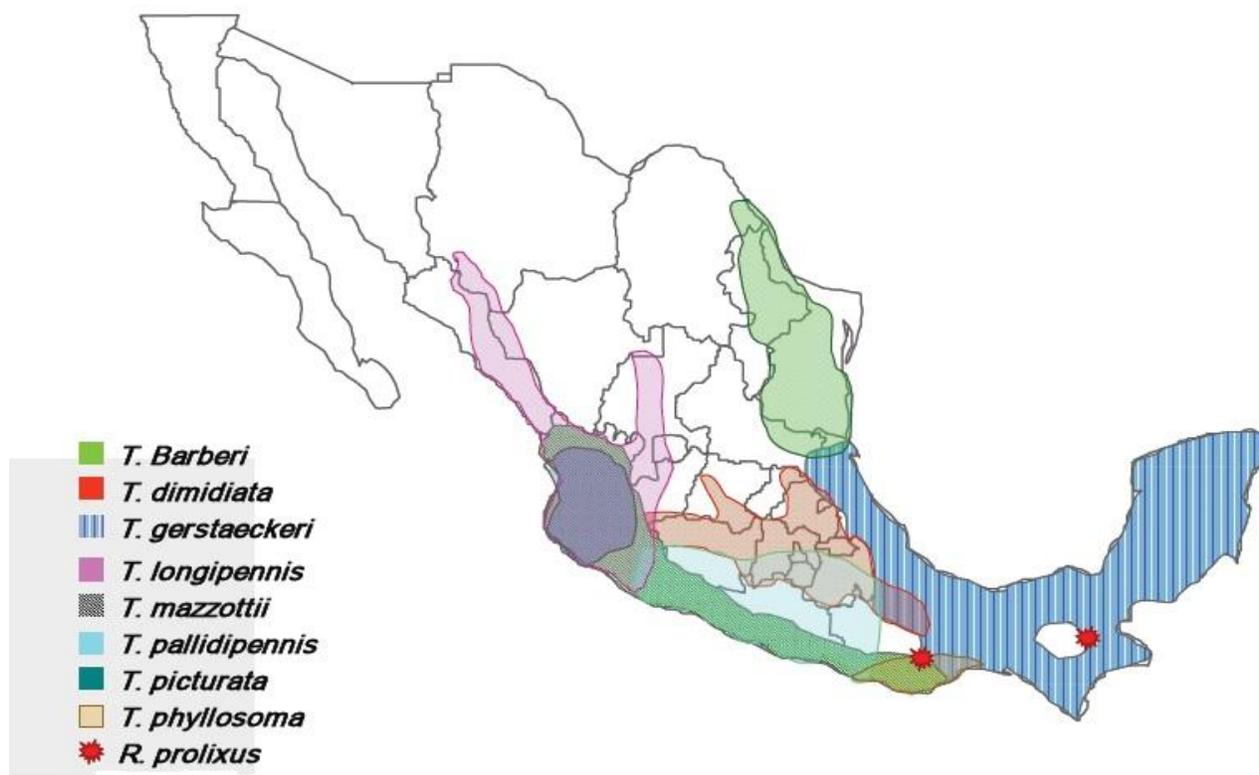


Figura 1.3. Vectores causantes de Chagas en México

Desde el descubrimiento de *Triatoma dimidiata* por Hoffman en 1928 en Veracruz [7] y los trabajos de Mazzotti en 1936 y 1940 en Oaxaca, en donde describe por primera vez al vector infectado, con dos casos de infección aguda en el humano y dos vertebrados naturalmente infectados, a la fecha, el estudio de la enfermedad de Chagas en México aún no ha tenido una justa valoración. Dadas las características geográficas, climáticas y socioculturales de las poblaciones rurales y suburbanas, así como las biológicas del gran número de transmisores existentes, de la confirmación de los múltiples hallazgos de triatóminos naturalmente infectados con *Trypanosoma cruzi* en todo el país y, desde

luego el reporte de casos humanos que no han dejado de publicarse en la literatura correspondiente, con estos precedentes desde hace tiempo debería haberse considerado el impacto sanitario de la misma [8,9]. Por otro lado, se han realizado encuestas seroepidemiológicas en diferentes estados de la República Mexicana por investigadores de varios grupos de trabajo, con el empleo de antígenos de diversas naturalezas, así como, variadas técnicas y procedimientos en el serodiagnóstico, lo que ocasiona dificultades en la comparación y evaluación no sólo de técnicas, sino de resultados; aún así, estos hallazgos pueden proporcionar una idea de la situación de esta enfermedad en México (Figura 1.4) [10, 11, 12]. Con estos resultados, se aprecia regionalmente en el país, una distribución heterogénea de la infección, donde Dumonteil refiere un promedio de seroprevalencias entre el 5 y 20% en el área rural y calcula una frecuencia de 20% en las áreas rurales endémicas donde se han reportado casos crónicos y serologías con títulos elevados; la transmisión en estas áreas es de manera natural o vectorial (Chagas Rural), siendo este mecanismo el más frecuente en México. La transmisión transfusional con sangre o sus componentes (Chagas Urbano), es el segundo mecanismo en importancia; donde se calcula que el riesgo de adquirir la infección por esta vía oscila entre el 20 y 25%, situación debida a la cada vez mayor migración poblacional del área rural a la urbana [5, 13].

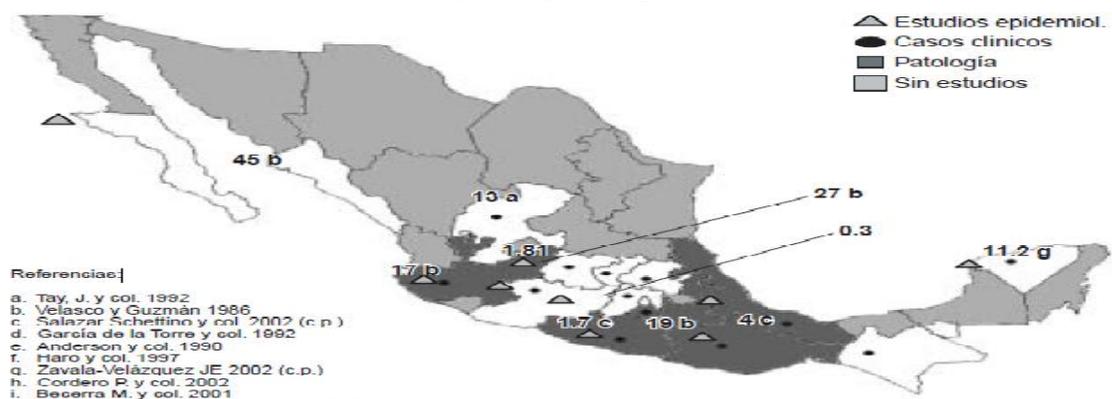


Figura 1.4. Prevalencia de la enfermedad de Chagas en México según estudios seroepidemiológicos y de patología.

Tabla 1.1

Estudios de seroprevalencia realizados en bancos de sangre en México					
Lugar	Autor	Año	Técnica*	Donadores	%
Oaxaca	Goldsmith	1978	FC, HAI, AD	298	4.4
D.F.	Monteón	1987	IFI, CIE	265	1.1
D.F.	Ramos	1993	ELISA, dot-ELISA, W-b	1 076	0.28
Jalisco	Trujillo	1993	HAI	3 419	1.28
Yucatán	Rodríguez	1995	IFI	215	5.6
Cuernavaca	Rancel	1998	ELISA	318	17.0
D.F.	Guzmán	1998	HAI, IFI	64 969	1.5
D.F.	Monteón	1999	IFI, ELISA, W-b	3 300	0.3

FC-Fijación del Complemento, HAI-Hemaglutinación Indirecta, AD-Aglutinación Directa, IFI-Inmunofluorescencia Indirecta, CIE-Contrainmunolectroforesis, ELISA-Inmunoensayo Enzimático, W-b-Western-blot.

En zonas urbanas, los escasos estudios realizados en bancos de sangre, muestran una seroprevalencias entre el 0.28 y 17%, según se muestra en la Tabla 1.1 [14–21]. El primer caso de transmisión por transfusión sanguínea en México, fue reportado en 1989 por Salazar–Schettino y colaboradores. En relación a la tasa de incidencia de la enfermedad, Schofield en el año 2000 estima que en México hay 10854 casos nuevos por año, con una seroprevalencia de 540000 individuos infectados (Figura 1.5) [22].

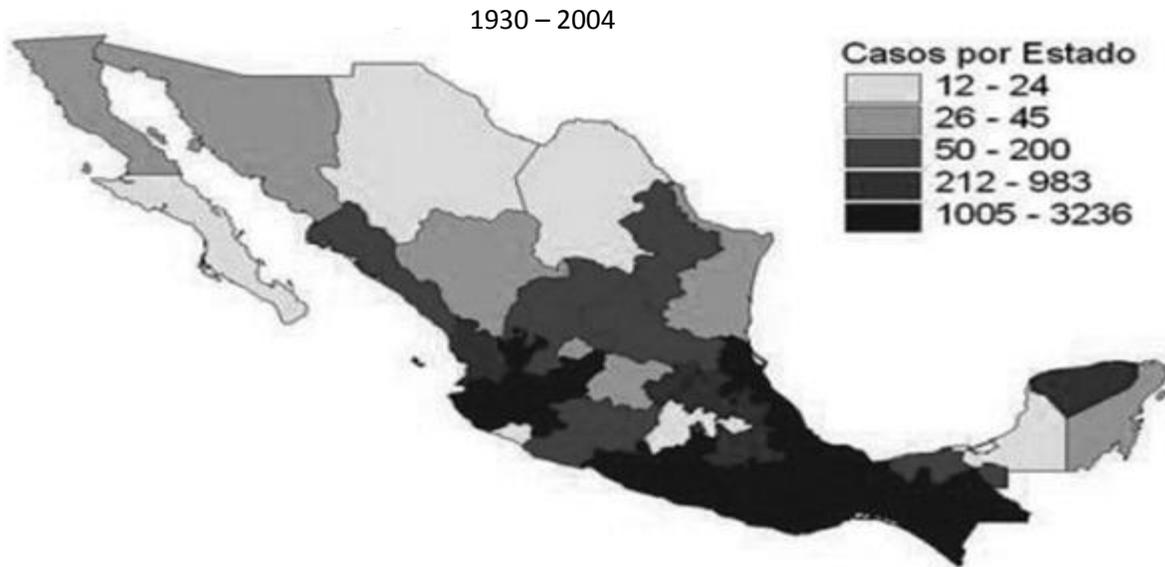


Figura 1.5 Casos de *Trypanosoma cruzi*, por Estado, hasta 2004

Referente al transmisor, Hoffmann en 1928 correlaciona por primera vez en México, al triatómino como transmisor de *T. cruzi*; sin embargo, Lent y Wygodzinky en 1979, señalando la importancia de los transmisores, consideran a México como el país Hispanoamericano con mayor número de especies de triatóminos. Este reporte es sumamente importante ya que marca el hecho de la gran variedad de transmisores existentes, aunada posteriormente a la descripción de otros tres transmisores: *Triatoma brailovskyi* (1984), *Triatoma gomeznunezi* (1994) y *Triatoma bassolsae* (1999). A la fecha, México no sólo es el país con mayor número de especies transmisoras (39 especies hasta el momento identificadas), sino que además varias presentan hábitos intradomiciliados, con diferentes comportamientos biológicos, lo que dificulta aún más las estrategias para el control. Esta situación nos diferencia notablemente de los países del Cono Sur, donde sólo presentan un transmisor de

importancia epidemiológica, *T. infestans* y de los países Andinos y de Centro América, donde existen dos, *T. dimidiata* y *Rhodnius prolixus* [5].

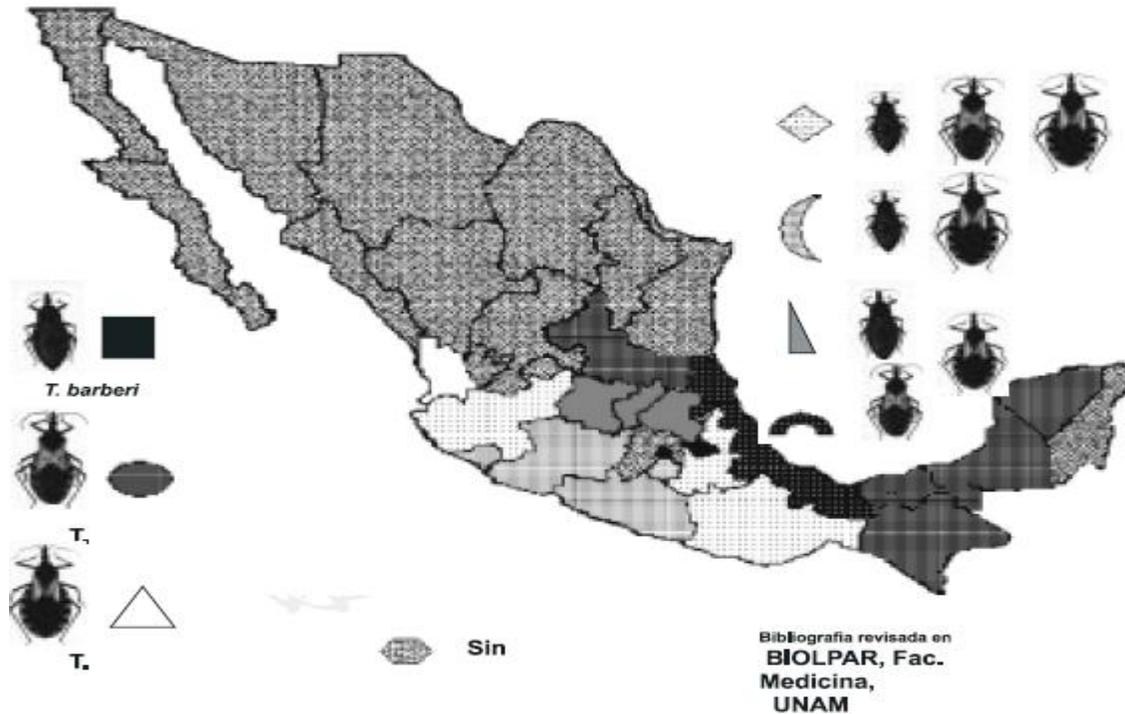


Figura 1.6 Distribución de tres especies de *Triatóminos* en México

En la República Mexicana, han sido reportados siete géneros: *Belminus*, *Dipetalogaster*, *Eratyrys*, *Panstrongylus*, *Paratriatoma*, *Rhodnius* y *Triatoma* distribuidas en todos los estados, la mayoría de los cuales, han sido identificados naturalmente infectadas con *Trypanosoma cruzi*. La distribución de las 3 especies intradomiciliadas que se consideran más importantes por su distribución y eficacia biológica como transmisores en el sur del país, se presentan en la Figura 1.6.

Hasta 1997, se habían identificado 18 mamíferos reservorios de *T. cruzi* principalmente en los estados de Oaxaca, Morelos, Jalisco, Michoacán, Veracruz, Colima, Nuevo León y Distrito Federal: *C. familiares* (perro), *D. novemcinctus mexicanus* (armadillo), *D. marsupiales* (tlacuache), *S. vulgaris* (ardilla), *Ph. Opposum* (marta), *F. domesticus* (gato); diversos ratones y ratas de los géneros *Rattus*, *Mus*, *Neotoma*, *Sigmodon*, *Peromyscus*, *Ototylomys*, *Tyloma*, *Liomys*; *C. perspicillata* (murciélago) *B. taurus* (vaca), *E. asinus* (asno) y *S. scrofa* (cerdo).

El Centro Nacional de la Transfusión Sanguínea y la Facultad de Medicina de la UNAM editaron en el año 2000, un Manual de Laboratorio para el Diagnóstico de la Infección por *Trypanosoma cruzi*, publicado por la OPS/OMS México, donde contempla, especialmente la problemática normativa que existe en el país para la realización del tamizaje y su obligatoriedad, la cual aún cuando está bien definida, aún no es de observancia en todo el país como sería deseable para el control del Chagas Urbano.

Acerca del control de Chagas Rural, en el año 2001 se publica la *Iniciativa México*, por la Subsecretaría de Prevención y Control de Enfermedades de la Secretaría de Salud y la Facultad de Medicina de la UNAM, en donde se contempla el estudio en individuos menores de 18 años y la presencia de transmisores intradomiciliados, para evaluar la transmisión activa con los casos seropositivos. Estos estudios, se plantearán como parámetros para establecer las medidas de Control y Vigilancia Epidemiológica de la enfermedad de Chagas. En el estado de Veracruz, esto ya fue realizado conjuntamente por la Facultad de Medicina de la UNAM y la Secretaría de Salud del estado y actualmente está por iniciarse de manera similar en el estado de Guanajuato [5].

En México la Enfermedad de Chagas es un problema de salud pública que afecta fuertemente a 11 Estados de la República Mexicana. En la Figura 1.7 se muestran los vectores descubiertos hasta el momento.

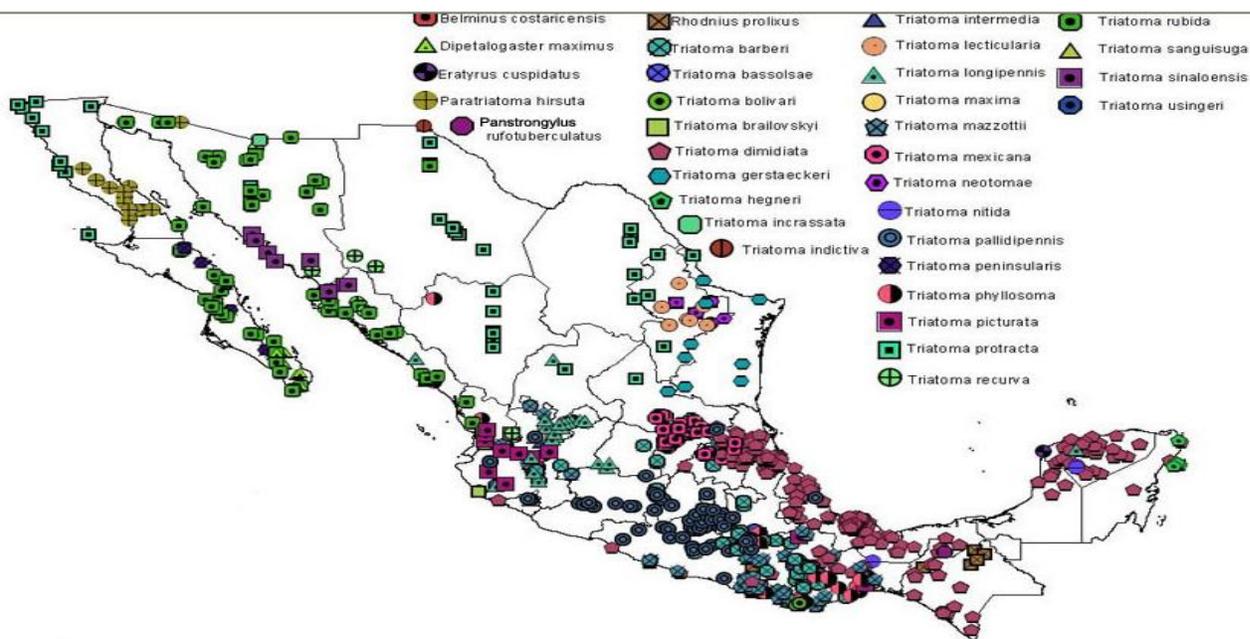


Figura 1.7 Principales vectores localizados en México.

1.7 Reconocimiento de imágenes médicas

1.7.1 Imágenes médicas

Se puede afirmar dejando poco lugar a dudas que las exploraciones “no–invasivas” del interior del cuerpo humano nacen en 1895 con las primeras imágenes obtenidas por Wilhelm Röntgen con los rayos–X, denominando así a la radiación electromagnética que descubriera ese mismo año [23]. Desde entonces el desarrollo de la especialidad médica conocida hoy como “diagnóstico por imágenes”, ha sido incesante, y ha cobrado un impulso vertiginoso a partir de la producción masiva de imágenes digitales en la década de 1970.

Si bien las primeras “digitalizaciones” de fotografías comenzaron a inicios la década de 1920 para lograr su transmisión telegráfica, no fue sino hasta principios de los años 60 que comenzó a aplicarse el procesamiento digital de imágenes por computadora con el desarrollo de hardware y lenguajes de programación. Una de las motivaciones principales era reducir el ruido y corregir distorsiones presentes en las imágenes obtenidas y enviadas por sondas espaciales [23, 24].

Sin embargo, pocas veces se tiene en cuenta el rol jugado por el diagnóstico por imágenes, que puede llevar a la detección temprana de enfermedades y/o sus causas, resolver la acción terapéutica de ser necesaria o en su defecto descartarla, disminuyendo así los costos y evitando riesgos asociados a métodos invasivos tales como intervenciones quirúrgicas.

Es así que la disciplina conocida como procesamiento o tratamiento de imágenes es considerada parte de otra más general llamada procesamiento de señales, compartiendo incumbencias con campos conocidos como visión computacional y análisis de imágenes, a su vez y con frecuencia enmarcadas en la Inteligencia Artificial.

En todo este contexto, las herramientas y las bases conceptuales del tratamiento digital de imágenes médicas se nutren en gran medida de aquellas previamente desarrolladas para otras aplicaciones quizá más decisivas, tales como las agropecuarias, militares o de vigilancia, refiriendo en particular a las que aplican y hacen uso del reconocimiento de patrones. No obstante, las peculiaridades e importancia del tratamiento de señales biomédicas ameritan su consideración como una disciplina separada.

1.7.2 Reconocimiento de imágenes médicas por medio de tejido

Antes de abordar el análisis de imágenes para uso médico, es preciso tener en cuenta los aspectos físico-técnicos involucrados en su generación. Una buena parte de los estudios realizados in vivo, hace uso de la interacción de la radiación con la materia, detectando fotones en forma externa al cuerpo examinado. Por ejemplo, las técnicas más difundidas aprovechan la atenuación diferencial de Rayos X y Gamma (~20–511 kilo electro Volts [keV]) en tejidos con distinto número atómico y densidad; en estos casos la absorción fotoeléctrica provee el contraste radiográfico [25]. En otras modalidades menos difundidas se detectan también los fotones dispersados en dichos tejidos mediante el efecto Compton [25].

En el cuerpo humano y de otras especies, a grosso modo se distinguen tres tipos de tejidos biológicos por su densidad y número atómico Z : óseo (alto Z), tejido blando (similar al agua) y pulmonar (similar al aire), sin olvidar que pueden encontrarse implantes metálicos en la región del cuerpo a explorar. Aquí cabe distinguir entre modalidades que producen imágenes por transmisión, de aquellas que detectan la emisión de fotones provenientes del cuerpo, como es el caso de aquellos estudios en los que al paciente se le administra un radiofármaco durante un estudio de medicina nuclear.

Prácticamente se aprovecha todo el espectro electromagnético para la producción de imágenes médicas, tal como se esquematiza en la Figura 1.8 [25]. Para nuestra investigación nos interesa manejar las imágenes dentro del área visible (que es la región del espectro electromagnético que el ojo humano es capaz de percibir), las cuales podemos tomar con una cámara digital o con un microscopio. Entre las muchas técnicas de análisis de imágenes médicas se han utilizado clasificadores basados en redes neuronales, tanto de aprendizaje supervisado como no supervisado. Generalmente estas redes se entrenan de forma tal que puedan reconocer los bordes de distintos tejidos [25]. En cuanto a los modelos y al etiquetado, pueden ser determinísticos, probabilísticos o difusos, en los cuales más de una clase de tejido puede resultar asignada a un píxel.

Una imagen médica puede representar información funcional o anatómica, procesos bioquímicos o metabólicos, en forma estática o en función del tiempo. La interpretación de estos datos para producir un diagnóstico, puede facilitarse y ampliarse mediante distintos procesamientos digitales.

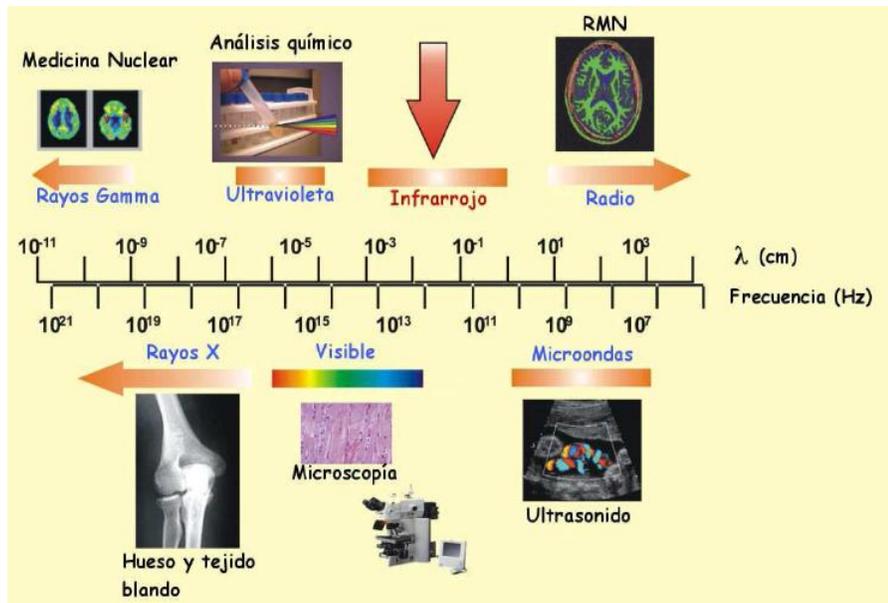


Figura 1.8 Algunos ejemplos de imágenes médicas según su espectro electromagnético [25]

Si bien el imagenólogo especialista puede realizar algunos procesos en forma manual, la complejidad de ciertas imágenes, entre ellas las cerebrales, motiva el desarrollo de algoritmos computacionales que automaticen y optimicen varias tareas.

Muchas de estas tareas concernientes al análisis de imágenes pueden ser realizadas en forma automática o semi-automática, mediante la aplicación de métodos computacionales basados en modelos físico-estadísticos. Pero además es deseable que estos procesos sean optimizados en cuanto a simplicidad, rapidez, exactitud y robustez. En la rutina de diagnóstico por imágenes, es común que se cometan errores o se trabaje de manera sub-óptima al no validar el alcance y las limitaciones de ciertos algoritmos, lo cual puede tener consecuencias directas en la interpretación clínica.

Entre dichos algoritmos, tanto las tareas de segmentación como de registro de imágenes médicas mediante cálculo numérico han sido material de tratamiento extensivo en la literatura de las últimas dos décadas [26]. Su complejidad intrínseca y la cantidad de situaciones que pueden suscitarse hacen que continúen siendo temas abiertos y que no exista un único método óptimo para todos los casos.

Por otra parte, la ejecución de estas tareas en forma automática o semi-automática facilita la difusión y adopción de estas herramientas en los servicios de diagnóstico, donde no suele disponerse de mucho tiempo para que los especialistas realicen procesamientos largos y tediosos [27].

Una red neuronal artificial (ANN, por sus siglas en inglés) es una estructura de neuronas simuladas con un patrón de conexiones inspirado en los patrones de algunas regiones cerebrales. Las ANN ofrecen características especiales que hacen su atractivo su empleo en un gran número de situaciones: procesamiento no lineal, adaptivo y en paralelo. Cada neurona de las ANN recibe estímulos de otras neuronas o en forma externa. La suma ponderada de esas “entradas” pasa a través de una función de base y el argumento resultante se aplica a una función de activación sináptica que finalmente representa las “salidas” de las neuronas. La forma o topología de las conexiones entre estas neuronas define la arquitectura de la red. Sus configuraciones pueden ser mono-capa, multi-capa, con alimentación directa (feed-forward), retro-alimentadas (feedback) y de conectividad lateral [28].

Estas capas pueden ser de entrada, ocultas y de salida. Una red será tanto más poderosa cuantas más capas tenga, pero esto puede causar una lenta convergencia en el aprendizaje, el cual puede ser supervisado o no supervisado [28].

Una ANN puede ser entrenada para reconocer distintas texturas o clasificar distintos componentes de una imagen. Una especie particular de redes no supervisadas, los SOM (*Self-Organizing Maps*) han sido aplicados exitosamente en segmentación de RM cerebral [29].

Entre sus varias aplicaciones en imágenes médicas, cabe destacar el uso extensivo de redes neuronales en la detección y clasificación de microcalcificaciones en mamografías, de especial relevancia en la detección precoz del cáncer de mama.

1.7.3 Técnicas para la detección de enfermedades por medio de imágenes

La imagenología es una disciplina de la medicina que emplea diferentes modalidades de imágenes del cuerpo humano, obtenidas mediante un conjunto de equipos y métodos como ultrasonido, tomografía

axial computarizada, resonancia magnética nuclear, radiología convencional y digital, para llegar en forma rápida y segura a la detección de diversas enfermedades, volviéndose una de las herramientas imprescindibles para la atención adecuada y calificada de los pacientes. Comprende la realización de todo tipo de exámenes diagnósticos y terapéuticos, en los cuales se utilizan equipos que reproducen imágenes del organismo.

Las primeras imágenes fueron obtenidas a finales del siglo XIX gracias al descubrimiento de los rayos X [30]. Esto dio un aporte insospechado al avance y el desarrollo de las ciencias de la salud originando lo que actualmente se conoce como la especialidad de la imagenología. Sin duda, la mayor ventaja que tiene la imagenología es la posibilidad de avanzar en el tratamiento de alguna enfermedad humana sin intervenir quirúrgicamente al paciente; esto se le conoce como métodos no invasivos. Prácticamente, la imagenología nació en el mismo momento que fueron descubiertos los rayos X.

Desde la primera radiografía lograda por Röntgen, se han hecho esfuerzos para obtener imágenes que muestren partes del interior del cuerpo humano con una mejor calidad y resolución. Para lograr una buena imagen en los términos antes mencionados, se deben tomar en cuenta cuatro tipos de resolución: espacial, contraste, temporal y estadística. La resolución espacial está relacionada con la geometría del proceso de formación de la imagen y, dependiendo de la técnica, de la pantalla intensificadora. El contraste depende del tejido y la forma en que atenúa los rayos X; asimismo, de la energía de estos rayos y de su dispersión por el tejido. La resolución temporal depende de cuán borrosa es la imagen causada por el movimiento voluntario e involuntario del paciente. Finalmente, la resolución estadística se relaciona con el número de rayos absorbidos por la pantalla; entre más rayos se absorban, menos ruido estadístico habrá.

Algunas de las técnicas para obtener imágenes del cuerpo humano son: tomografía axial computarizada (TAC), tomografía por emisión de positrones (PET), tomografía computarizada de emisión de fotones únicos (SPECT), ultrasonido y resonancia magnética nuclear (RMN) [31]. A continuación describiremos estas técnicas.

La tomografía axial computarizada (TAC) es una exploración de rayos X que produce imágenes detalladas de cortes axiales del cuerpo. En lugar de obtener una imagen como la radiografía

convencional, la TAC obtiene múltiples imágenes al rotar alrededor del cuerpo. Una computadora combina todas estas imágenes en una imagen final, que representa un corte del cuerpo como si fuera una rodaja. Esta máquina crea múltiples imágenes en rodajas (cortes) de la parte del cuerpo que está siendo estudiada.

La tomografía por emisión de positrones, TEP (*Positrón Emission Tomography*, PET por sus siglas en inglés), es una técnica no invasiva de diagnóstico por imagen, capaz de medir la actividad metabólica de los diferentes tejidos del cuerpo humano, especialmente del sistema nervioso central. Al igual que el resto de técnicas de diagnóstico en medicina nuclear, la TEP se basa en detectar y analizar la distribución que adopta en el interior del cuerpo un radioisótopo administrado a través de una inyección. Lo que hace es medir la producción de fotones gamma (resultado de la interacción de un positrón con algún electrón del tejido).

La tomografía computarizada de emisión de fotones únicos (SPECT, *Single-Photon Emission Computed Tomography*) es una técnica para obtener imágenes bidimensionales de un cuerpo tridimensional a través de un dispositivo llamado cámara gamma. Las imágenes son, en realidad, proyecciones de ángulos múltiples del cuerpo en cuestión. Posteriormente se utiliza una computadora para aplicar un algoritmo de reconstrucción tomográfica de estas proyecciones múltiples obteniendo así la imagen tridimensional.

El ultrasonido (también conocido como ecografía) emplea los ecos de una emisión de ultrasonidos dirigida sobre un cuerpo u objeto como fuente de datos para formar una imagen de los órganos o masas internas. Un pequeño instrumento llamado transductor emite ondas de ultrasonidos. Estas ondas sonoras de alta frecuencia se transmiten hacia el área del cuerpo bajo estudio y se recibe su eco. El transductor recoge el eco de las ondas sonoras y una computadora lo convierte en imagen. En la ecografía no se emplea radiación.

La resonancia magnética nuclear (RMN) se basa en las propiedades magnéticas que poseen los núcleos atómicos (generalmente el hidrógeno) para crear imágenes. Esta técnica alinea los campos magnéticos de los átomos en la dirección de un campo magnético externo. Al regresar dichos átomos

a su estado original, emiten una señal (podría decirse que resuenan) que registra una computadora para producir la imagen.

El principal problema que se tiene para la detección de diversas enfermedades es la diversidad de técnicas de imagenología que se tienen y el reconocimiento de la enfermedad por medio de cambios en el tejido.

El problema central del reconocimiento de imágenes consiste en el reconocimiento de sus propiedades, y éstas pueden ser utilizadas en el reconocimiento de imágenes médicas por medio de cambios en el tejido. Las texturas generalmente proporcionan información determinada sobre determinadas secciones de la imagen. Las características de textura que se pueden utilizar para el reconocimiento de imágenes médicas por medio de tejido son las que se representan en los histogramas.

En el siguiente capítulo se analizarán los diferentes métodos de tratamiento de imágenes.

CAPÍTULO 2

2. Imágenes

2.1 Introducción

Los ambientes de trabajo médico permanecen rodeados de una gran cantidad de información visual como las radiografías, tomografías computarizadas, imágenes por resonancia magnética, mamografías, ecografías, entre otras. Las posibilidades de producción de imágenes crecen más rápido que las metodologías para administrar y procesar la información visual, utilizando técnicas apropiadas para el almacenamiento y representación [32]. Las bases de datos de imágenes han sido estudiadas desde hace varios años. Las primeras aproximaciones para indexar grandes volúmenes de imágenes se realizaron utilizando palabras clave, pero la construcción del índice se convertía en una tarea costosa y subjetiva. El Photobook [33] y el QBIC [34] han sido algunos de los primeros trabajos para bases de datos de imágenes en donde los autores se preocuparon por las propiedades visuales de la imagen y sus características de forma, área y textura, implementando un sistema de recuperación de imágenes que utiliza operadores visuales. En el área de las imágenes médicas se han desarrollado diversos trabajos para permitir al personal médico el acceso al contenido de la imagen de acuerdo a sus propiedades y su significado.

La exploración de grandes cantidades de imágenes es una tarea en donde los sistemas de información pueden contribuir a organizar y clasificar los registros automáticamente. Una base de datos de imágenes debe permitir al usuario recuperar una imagen del archivo a partir de sus propiedades visuales, como la forma o composición estructural. Los usuarios necesitan encontrar imágenes con ciertas características, sin tener que explorar demasiados registros, obteniendo aquellas que tengan un alto grado de importancia de acuerdo a los criterios definidos. Varios trabajos se han realizado para representar el significado de la imagen a partir de sus características visuales [32, 35, 36, 37, 38], obteniendo resultados apropiados principalmente para especialidades artísticas y gráficas. Por otro lado los trabajos de visión artificial orientados hacia otras aplicaciones de tiempo real y de control, tienen requerimientos más específicos para su desarrollo, como la fuerte identificación de objetos, el seguimiento de los mismos en secuencias de imágenes y el reconocimiento de patrones. Muchos de estos problemas aún continúan sin resolver, principalmente por la dependencia que existe del completo entendimiento de las imágenes, en donde los investigadores han identificado dos vacíos fundamentales: sensorial y semántico. La creciente necesidad de implementar sistemas que permitan

acceder a las imágenes a partir de su contenido visual, ha llevado a impulsar uno de los campos de investigación más activos de hoy en día: Recuperación de Imágenes por Contenido (*CBIR* por sus siglas en inglés). La investigación en esta área comparte conceptos y resultados provenientes de trabajos de visión computacional, reconocimiento de rostros, biometría, exploración de extensos álbumes fotográficos, recuperación y clasificación de secuencias de video, entre otras. Las aplicaciones particulares en el área médica van desde la medicina basada en evidencia, el razonamiento por casos médicos, el diagnóstico asistido por computadora y el entrenamiento y capacitación por computadora al personal médico [39].

2.2 Definición de imagen

Una imagen es una colección de medidas o valores en el espacio bidimensional (2D) o tridimensional (3D). En imágenes médicas, estas medidas o intensidades de imagen pueden ser la absorción de radiación de imágenes de Rayos X, presión acústica en ultrasonido, o amplitud de señal de radiofrecuencia (*RF: radio-frecuency*) en MRI. Si se hace una medida simple para cada posición de la imagen, entonces la imagen es una imagen escalar. Si se hace más de una medida (por ejemplo: dual-eco MRI), la imagen es llamada imagen vector o imagen multicanal. Las imágenes pueden ser adquiridas en el dominio continuo como las películas de Rayos X, o en el discreto como en MRI. En imágenes discretas 2D, la posición de cada medida se conoce como píxel y en imágenes 3D, se le llama voxel. Por simplicidad, se usa el termino píxel con frecuencia para referirse a ambos casos [40].

2.3 Características de las imágenes

Las características visuales de una imagen pueden clasificarse de acuerdo a su ámbito como globales o locales y su nivel de abstracción como lógicas o físicas [41].

- 1) Físicas: Son aquellas que pueden expresarse cuantitativamente, y se extraen mediante la aplicación de técnicas en la imagen original. También son llamadas características de bajo nivel [42].
 - a) Color: Se utiliza para describir las distribuciones cromáticas de la imagen, construyendo el histograma de frecuencias. También son aplicados a imágenes en escalas de grises. Se han propuesto diversos espacios de colores, para poder describirlos de la misma forma como los percibe el ojo humano.

- RGB (*Red–Green–Blue*) Contiene la codificación de los colores, de acuerdo a su intensidad en 3 componentes. En una misma escena el mismo color puede cambiar en este espacio debido a problemas de iluminación o contrastes.
 - YIQ–YUV: Basados en la iluminación y cromaticidad, son utilizados por los comités de estándares en televisión tanto en Europa como en América. Es una transformación lineal del espacio RGB.
 - HSV (*Hue–Saturation–Intensity*). Este espacio de color es el más cercano a la percepción humana, pero no es un modelo perfecto.
- b) Textura: También se encuentran en la categoría de características locales. Aunque existen varias técnicas para extraer la textura, ésta puede definirse en general como una propiedad de homogeneidad en las regiones de la imagen. Las técnicas para análisis de textura incluyen energía, entropía, homogeneidad, contraste, correlación, agrupamiento, direccionalidad, repetitividad, granularidad y otras más. La textura y otras características, fueron estudiadas para la recuperación de imágenes médicas en [43].
- 2) Lógicas: Las características lógicas son también llamadas características de alto nivel. Por lo general contienen información de los objetos en la imagen y sus relaciones espaciales.
- a) Curvatura: Puede ser medida tanto en contornos identificados como en una región local de la imagen, utilizando la razón de cambio en la dirección de la línea tangente al contorno o curva.
- b) Forma: Para la identificación de formas en una imagen existen diferentes aproximaciones y técnicas. Los descriptores elípticos de Fourier son utilizados para describir contornos cerrados en los objetos [44]. También existe segmentación por textura, clustering y otra serie de técnicas que componen un amplio campo de investigación.
- c) Puntos interesantes: Dado que la identificación fuerte de las formas y objetos en una imagen es todavía un problema abierto, se han realizado trabajos para representar la imagen a partir de sus puntos interesantes [45], reduciendo la complejidad de la imagen y enfocándose en las regiones con mayor interés visual.
- d) Posición de las regiones: Basado en la identificación previa de formas, la posición de las mismas es una característica interesante para algunas aplicaciones, la cual puede ser medida como posición absoluta (por cuadrantes) o posición relativa (con respecto a otros objetos). Al medir la posición debe tenerse en cuenta la robustez frente a rotaciones de la

figura por errores de captura. La posición también incluye relaciones de contenedora e intersección.

- 3) Locales: Las características están basadas en las características físicas o de bajo nivel. Estas características pueden medirse con respecto a:
 - a) Cada píxel
 - b) Una ventana de tamaño fijo
 - c) Una región previamente identificada
- 4) Globales: Pueden ser características simples o compuestas. Las compuestas son una combinación de características físicas, lógicas y locales. Este tipo de características dan información sobre la totalidad de la imagen tal como su tamaño, composición de color, orientación y número de objetos entre otras. Para las aplicaciones de imágenes biomédicas poco a poco se han identificado características útiles y otras que no lo son tanto.

La segmentación automática de imágenes es utilizada ampliamente para extraer información de las imágenes médicas. La segmentación también se puede utilizar para recuperar las imágenes médicas o para calcular la forma aproximada de las regiones en imágenes médicas [46].

2.4 Reconocimiento de imágenes

Un modelo de sistema para reconocimiento de imágenes se muestra en la Figura 2.1 [47].



Figura 2.1 Modelo de sistema de reconocimiento de imágenes [47]

En el modelo se observa que se tiene la captura de la imagen digital, posteriormente se guarda esa imagen, se realiza la extracción de características y se almacena en una nueva imagen. Después se hace el cálculo de la similitud entre imágenes y se obtiene un resultado de la búsqueda; con ello se

tiene un soporte para realizar un diagnóstico al comparar las imágenes de la base de imágenes con la imagen capturada.

A continuación se explican los módulos del sistema de la Figura 2.1:

2.4.1 Extracción de características: La extracción de características es el primer paso en todo el proceso de recuperación de imágenes, y desde luego, toda información que pueda ser útil debe ser evaluada de acuerdo al dominio de la aplicación, consolidando criterios para la selección de las características que representarán el contenido de una imagen. Este módulo se encarga de procesar las imágenes digitales para cuantificar en ellas las propiedades que permitirán distinguirla de las demás. En este sistema se incluyeron características de color, textura y bordes representados por medio de sus histogramas. Estas características son extraídas y almacenadas en el sistema para su utilización durante el proceso de recuperación. Estas características son de bajo nivel con información global acerca de la imagen.

2.4.2 Cálculo de similitud: Cada imagen tiene asociados histogramas con información de su contenido. Para comparar 2 imágenes se utiliza la información obtenida mediante los histogramas.

2.4.3 Resultados de la búsqueda: La consulta inicia con una imagen de ejemplo sobre la cual se calculan las características para compararlas con las características de las imágenes en la base de imágenes.

2.5 Segmentación y etiquetado

Clásicamente, la segmentación de imágenes se define como la partición de una imagen en regiones constituyentes no traslapadas, las cuales son homogéneas con respecto a alguna característica como una intensidad o una textura [50]. Si el dominio de la imagen esta dado por I , entonces el problema de segmentación consiste en determinar el conjunto $S_k \subset I$ cuya unión es la imagen I completa. Por lo tanto, el conjunto que conforma la segmentación debe satisfacer

$$I = \bigcup_{k=1}^K S_k, \quad (2.1)$$

Donde $S_k \cap S_j = \emptyset$ para $k \neq j$, y cada S_k está conectado. Idealmente, un método de segmentación encuentra aquellos conjuntos que corresponden a distintas estructuras o regiones anatómicas de interés en la imagen.

Al eliminar la restricción de que las regiones estén conectadas, determinar los conjuntos S_k es llamado clasificación de píxel y a los conjuntos se les llama clases. La clasificación de píxeles frecuentemente es un objetivo deseable en el tratamiento de imágenes médicas, particularmente cuando se necesita clasificar regiones desconectadas que pertenecen al mismo tejido. La determinación del número de clases K en la clasificación de píxeles puede ser un problema complejo, por lo que generalmente se asume conocida, basado en conocimientos previos de la anatomía que se está tomando en consideración.

El etiquetado es el proceso de asignar una designación significativa a cada región clase y puede ser llevada a cabo separadamente de la segmentación. Este proceso mapea el índice numérico del conjunto S_k , a una designación anatómica. En imágenes médicas, frecuentemente las etiquetas son obvias y pueden ser determinadas mediante inspección del técnico o fisiólogo. El etiquetado automático por computadora es deseable cuando las etiquetas no son obvias o el sistema realiza procesamiento automático de imágenes. Una situación típica en la que se utiliza el etiquetado ocurre en la mamografía digital cuando se segmenta la imagen en distintas regiones y las regiones se etiquetan como sanas o tumorosas.

2.6 Métodos de segmentación

En esta sección se describen resumidamente varios métodos de segmentación que han aparecido en la literatura reciente de segmentación en imágenes médicas [51]. Se definirá cada método y se discutirán sus ventajas y desventajas. Aunque cada técnica fue creada separadamente, frecuentemente se utilizan múltiples técnicas en conjunto con otras para resolver diferentes problemas de segmentación.

Los métodos de segmentación de imágenes médicas se pueden clasificar en ocho categorías: métodos de umbralización, métodos de región creciente, clasificadores, métodos de agrupamiento (clustering methods), modelos de campos aleatorios de Markov, Redes Neuronales Artificiales (RNA), modelos deformables y métodos guiados por plantillas (atlasguided methods). Al final de esta sección se

describen otros métodos notables que no pertenecen a ninguna de estas categorías. De los métodos mencionados anteriormente, los de umbralización, clasificación, agrupamiento, y campos aleatorios de Markov, pueden considerarse métodos de clasificación de píxeles.

La mayoría de los métodos de segmentación que se describirán pueden ser vistos como problemas de optimización donde la segmentación deseada es la que minimiza alguna función de energía o de costo, definida para una aplicación en particular. La ventaja de ver la segmentación como un problema de optimización es que define de manera precisa los aspectos deseables de la segmentación. Es muy claro que, para diferentes aplicaciones, se necesitan diferentes funciones de energía o costo.

2.6.1 Umbralización

La umbralización (thresholding) es un método que busca segmentar imágenes escalares creando una partición binaria de las intensidades de las imágenes [50]. Una umbralización trata de determinar un valor de intensidad, llamado umbral (threshold), que separa las clases deseadas. La segmentación se logra agrupando todos los píxeles con mayor intensidad al umbral en una clase, y todos los otros píxeles en otra clase. La determinación de más de un valor umbral es un proceso llamado multiumbralización (multithresholding). La umbralización es una técnica efectiva para obtener la segmentación de imágenes donde estructuras diferentes tienen intensidades contrastantes u otras características diferenciables. La partición usualmente es generada interactivamente, pero también existen métodos automáticos. Para imágenes escalares, los métodos interactivos pueden estar basados en la apreciación visual del usuario ya que la operación de umbralización es implementable en tiempo real. Generalmente, la umbralización es el paso inicial de una secuencia de operaciones de procesamiento de imágenes. Su principal limitación es que en su forma más simple sólo se generan dos clases y por lo tanto no se puede aplicar a imágenes multicanal. En adición, la umbralización usualmente no toma en cuenta las características espaciales de la imagen. Esto causa que sea sensible al ruido e inhomogeneidades de intensidad, las cuales pueden ocurrir en imágenes de resonancia magnética. Estos factores corrompen el histograma de la imagen, haciendo la separación más difícil [46].

2.6.2 Región creciente

La región creciente (region growing) es una técnica para extraer regiones de la imagen que están conectadas según cierto criterio predefinido [50]. Este criterio puede estar basado en información de

intensidades y/o bordes de la imagen. En su forma más simple, este método requiere un punto semilla (seed point) seleccionado manualmente por el usuario, y extrae todos los píxeles conectados a la semilla, que tengan el mismo valor de intensidad. Al igual que la umbralización, por lo general no se utiliza la región creciente solamente en una imagen, sino que se utiliza como parte de un conjunto de operaciones de procesamiento de imágenes, particularmente en la delineación de pequeñas y simples estructuras como tumores y lesiones. Su desventaja principal es que requiere interacción manual para obtener el punto semilla. Los algoritmos de división y mezcla (split and merge) están relacionados con la región creciente pero no requieren una semilla. La región creciente también puede ser sensible al ruido, causando que las regiones extraídas tengan agujeros e inclusive que se desconecten.

2.6.3 Clasificadores

Los métodos clasificadores son técnicas de reconocimiento de patrones que buscan particionar un espacio característico derivado de la imagen usando datos con etiquetas conocidas. Un espacio característico es un rango espacial de cualquier función de la imagen, siendo las intensidades de la imagen el más común de los espacios característicos. Los clasificadores son conocidos como métodos supervisados debido a que requieren datos de entrenamiento que son segmentados manualmente, para luego ser utilizados en la segmentación automática de nuevos datos. Hay una gran cantidad de maneras en las que los datos de entrenamiento pueden ser aplicados en los métodos de clasificación.

Un clasificador simple es el clasificador del vecino más cercano, donde cada píxel o voxel es clasificado en la misma clase que el dato de entrenamiento con la intensidad más cercana. Los k vecinos más cercanos ($kNN - k$ Nearest Neighbor) es una generalización de este método. Otros clasificadores utilizados son la ventana de Parzen y el clasificador de Bayes. Los clasificadores estándares requieren que la estructura a ser segmentada posea características cuantificables distintas. Debido a que los datos de entrenamiento pueden ser etiquetados, los clasificadores pueden transferir estas etiquetas a los nuevos datos siempre que el espacio característico distinga cada etiqueta lo suficiente. No son iterativos, por lo que son relativamente eficientes computacionalmente y pueden ser aplicados a imágenes multicanal. Una desventaja de los clasificadores es que no obedecen a ningún modelo espacial. Otra desventaja es la necesidad de la interacción manual para obtener los datos de entrenamiento. Los conjuntos de entrenamiento pueden ser adquiridos para cada imagen que requiera segmentación, pero esto puede tomar mucho tiempo y esfuerzo. En adición, el uso del mismo

conjunto de entrenamiento para un gran número de imágenes puede llevar a resultados viciados, ya que no toman en cuenta las variaciones anatómicas y fisiológicas entre diferentes sujetos [50].

2.6.4 Agrupamiento

Los algoritmos de agrupamiento (clustering) llevan a cabo esencialmente la misma función que los métodos clasificadores, pero sin utilizar datos de entrenamiento. Por lo tanto, son métodos no supervisados. Para compensar la falta de los datos de entrenamiento, los métodos de agrupamiento iteran entre segmentar la imagen y caracterizar las propiedades de cada clase. En este sentido, los métodos de agrupamiento se entrenan a sí mismos usando los datos disponibles. Un algoritmo de agrupamiento común es el algoritmo de las K -medias o algoritmo ISODATA. Este algoritmo agrupa datos calculando iterativamente la media de la intensidad para cada clase y segmentando la imagen mediante la clasificación de cada píxel en la clase con la media más cercana.

Aunque los algoritmos de agrupamiento no requieren que los datos se entrenen, si requieren una segmentación inicial (o de manera equivalente, requiere parámetros iniciales). Como los métodos de clasificación, los algoritmos de agrupamiento no incorporan directamente un modelo espacial. De cualquier forma, esta falta de modelado espacial puede proveer ventajas significativas para realizar los cálculos velozmente. Es posible incorporar robustez al ruido usando campos aleatorios de Markov [50], como se describe en la sección siguiente.

2.6.5 Campos aleatorios de Markov

Los modelos de campos aleatorios de Markov (*MRF*– *Markov Random Fields*) no son un método de segmentación en sí mismos, pero son un modelo estadístico que puede ser usado dentro de los métodos de segmentación [50]. Los MRF modelan las interacciones espaciales entre vecinos o píxeles cercanos. Estas correlaciones locales proveen un mecanismo para modelar una variedad de propiedades de la imagen. En el tratamiento de imágenes médicas, se utilizan frecuentemente para tomar en cuenta el hecho que la mayoría de los píxeles pertenecen a la misma clase a la que pertenecen sus píxeles vecinos. En términos físicos, esto implica que bajo la asunción del MRF, estructura anatómica que consista de un solo píxel tiene una probabilidad muy baja de ocurrir. Los MRF son incorporados frecuentemente en los algoritmos de segmentación por agrupamiento, como las K -medias bajo un modelo Bayesiano previo. La segmentación es obtenida maximizando la

probabilidad a posteriori de la segmentación a partir de los datos de la imagen, obtenidos mediante métodos iterativos como los Modos Condicionales Iterados, presentados por Besag [49], o el Templado Simulado, presentado por Geman [52]. Una dificultad asociada con los modelos MRF es la selección apropiada de los parámetros que controlan la fuerza de las interacciones espaciales. Una selección muy alta puede resultar en segmentación excesivamente suave y una pérdida de los detalles estructurales. En adición, los métodos MRF usualmente requieren algoritmos computacionalmente intensivos. A pesar de estas desventajas, los MRF son ampliamente utilizados no solo para modelar clases de segmentación, sino también para modelar propiedades de texturas e inhomogeneidades de las intensidades.

2.6.6 Redes Neuronales Artificiales (RNA, ANN)

Las Redes Neuronales Artificiales (ANN—Artificial Neural Network) son redes masivamente paralelas de procesamiento de elementos o nodos que simulan el aprendizaje biológico. Cada nodo en una RNA es capaz de llevar a cabo cálculos elementales. El aprendizaje se alcanza a través de la adaptación de pesos asignados a las conexiones entre nodos. Haykin [53] realiza un tratamiento más profundo de las redes neuronales. Las RNA representan un paradigma para el aprendizaje de las máquinas y pueden ser usadas en una variedad de formas de segmentación de imágenes. El uso que más se le da en procesamiento de imágenes médicas es el de un clasificador, donde los pesos son determinados usando datos de entrenamiento y luego se utiliza la RNA para segmentar nuevos datos. Las RNA también pueden ser usadas de una manera no supervisada como método de agrupamiento o como modelo deformable. Debido al gran número de interconexiones utilizadas en una red neuronal, se puede incorporar fácilmente información espacial en los procedimientos de clasificación. Aunque las RNA son inherentemente paralelas, pero frecuentemente se implementan en computadoras seriales, y esto reduce su potencial computacional.

CAPÍTULO 3

3. Redes Neuronales Artificiales

3.1 Introducción

En el procesamiento de imágenes médicas es indispensable, como una herramienta, el reconocimiento de texturas como parte de un problema que a simple vista no podría ser diagnosticado. Este proceso da como resultado una gran aportación multidisciplinaria, entre ellas médicas, que por consiguiente se puede llegar al diagnóstico de diversas patologías de una forma más precisa, concreta y específica.

Para llevar a cabo dicho proceso se tienen diferentes técnicas entre las cuales se encuentran las redes neuronales artificiales, la visión computacional, análisis de regresión lineal (con la finalidad de obtener modelos matemáticos o modelos de predicción que describan el fenómeno o problema), algoritmos de retropropagación, inteligencia artificial, entre otros.

Todas las técnicas anteriormente mencionadas han aumentado el auge de su aplicación en el ámbito médico en últimos tiempos.

3.2 ¿Qué es una neurona biológica?

Una neurona es la célula fundamental del sistema nervioso, especializada en procesar información. El cerebro humano se compone de 10^{11} neuronas interconectadas entre sí por 10^{13} conexiones formando redes que desarrollan funciones específicas complejas [54].

Los elementos principales que componen la neurona biológica son: las neuronas y las conexiones entre ellas. Una neurona se presenta en la Figura 3.1. Las dendritas son las entradas a la neurona, el axón es la salida de la neurona y la sinapsis son las conexiones con otras neuronas.

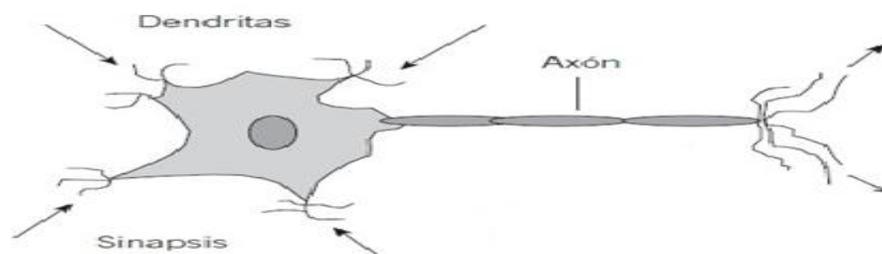


Figura 3.1 Neurona biológica

Esquemáticamente una neurona recoge las señales procedentes de otras neuronas, por medio de las dendritas. Estas señales se combinan, y en función de la estimulación total recibida, la neurona toma cierto nivel de activación, que se traduce en la generación de breves impulsos nerviosos con cierta tasa de disparo o determinada frecuencia.

Si estos impulsos son superiores al “umbral de activación”, la neurona propaga su señal a través del axón. Las extremidades de estas ramificaciones llegan hasta las dendritas de otras neuronas y establecen una conexión llamada sinapsis, que transforma al impulso eléctrico en un mensaje neuroquímico mediante la liberación de una sustancia llamada neurotransmisor [55].

El efecto sobre la neurona receptora puede ser de dos tipos: inhibitorio o excitador. El aprendizaje se produce mediante la variación de la efectividad de la sinapsis.

3.3 ¿Qué son las redes neuronales artificiales?

Una Red Neuronal Artificial (RNA) denominada en inglés como *Artificial Neural Network (ANN)* es un algoritmo de cálculo que se basa en una analogía con el cerebro humano. Son, también, un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funciona el sistema nervioso de los animales. Se trata de un sistema de interconexión de neuronas en una red que colabora para producir un estímulo de salida, tal como se muestra en la Figura 3.2 (Red neuronal con n neuronas de entrada, m neuronas en la capa oculta y una neurona de salida).

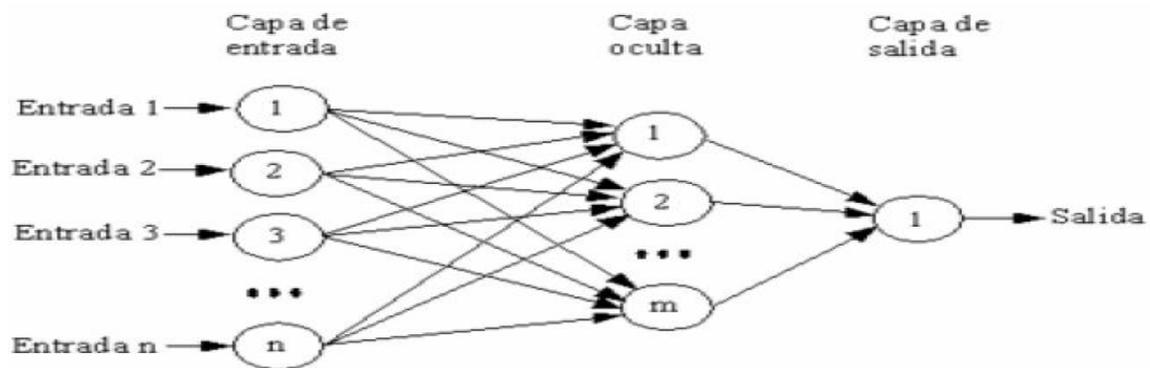


Figura 3.2 Modelo de una Red Neuronal Artificial

La capa de entrada es la que recibe directamente la información proveniente de las fuentes externas de la red.

Las capas ocultas son internas en la red y no tienen contacto directo con el entorno exterior. El número de niveles externos varía entre cero y tres, dependiendo de la tarea a resolver. Las neuronas de las capas ocultas pueden estar interconectadas de distintas maneras, lo que determina, junto con su número, las distintas topologías de redes neuronales.

La capa de salida es la que transfiere información de la red hacia el exterior [56].

3.4 Objetivo de las RNA

El objetivo general de la RNA consiste en emular la capacidad de aprendizaje del sistema nervioso, de manera que ésta pueda aprender a identificar un patrón de asociación entre los valores de un conjunto de variables predictoras (entradas) y los estados que se consideran dependientes de dichos valores (salidas). Desde un punto de vista técnico, la RNA consiste en un grupo de unidades de proceso (neuronas) que se asemejan a las neuronas al estar interconectadas por medio de un entramado de relaciones (pesos) análogos al concepto de conexiones sinápticas en el sistema nervioso. A partir de las neuronas de entrada, la señal progresa a través de la red hasta proporcionar una respuesta en forma de nivel de activación de las neuronas de salida. Los valores de salida proporcionan una predicción del resultado en función de las variables de entrada. Desde el punto de vista de implementación práctica, las neuronas son elementos computacionales simples que emulan la respuesta de una neurona a un determinado estímulo. Estos elementos, como las neuronas en el sistema nervioso, funcionan como interruptores: cuando la suma de señales de entrada es suficientemente alta (en el caso físico de una neurona se interpretaría como que se acumula suficiente neurotransmisor), la neurona manda una señal a las neuronas con las que mantiene contacto (se genera un potencial de acción). Esta situación se modela matemáticamente como una suma de pesos de todas las señales de llegada a la neurona que se compara con un umbral característico. Si el umbral se supera, entonces la neurona se dispara, mandando una señal a otras neuronas, que a su vez procesarán esa información juntamente con la que reciben de neuronas adyacentes. Evidentemente, la respuesta de cada neurona dependerá del valor de las interacciones con las neuronas precedentes dentro de la estructura de la red. Al igual que el sistema nervioso, el trabajo que desarrolla una RNA deriva no de la complejidad de cada unidad de proceso sino de la densidad y complejidad de sus interconexiones.

La primera implementación práctica de estas ideas se describe en los trabajos de McCulloch y Pitts en 1946 [57]. A partir de este punto, algunos de los hitos principales en la investigación de este tipo de técnicas fueron: el diseño por Widrow y Hoff (1961) [58] de la red conocida como Adalina (capaz de resolver problemas de regresión lineal), el desarrollo de la red con estructura de perceptrón simple en 1959 (con equivalencia al análisis discriminante y regresión logística) y las redes multicapa por Rosenblatt en 1986 (que permiten la resolución de situaciones no lineales). Por otra parte, los trabajos teóricos de Bishop y la aportación sobre redes autoorganizadas de Kohonen [59] dotaron de fundamentos formales a este tipo de técnica. A partir de los trabajos pioneros, el interés sobre esta metodología se ha difundido a casi todos los ámbitos de la ciencia. Los distintos aspectos técnicos y las implicaciones de su utilización han sido investigados desde muchos puntos de vista, interesando, entre otros, a matemáticos, físicos, neurólogos, ingenieros, programadores y filósofos. En medicina las áreas de aplicación se han multiplicado rápidamente en la última década. Desde un punto de vista práctico, existen muchos tipos de RNA, cada una de ellas con diferentes aplicaciones; algunos ejemplos de esto pueden ser: reconocimiento de voz, reconocimiento de texturas, aplicaciones textiles, aplicaciones interdisciplinarias. Para clasificarlas, podemos considerar dos criterios básicos: el modo de aprendizaje y el flujo de información. En una red, el modo de aprendizaje puede ser supervisado, es decir, la red recibe los patrones de entrada y la respuesta observada que debe aprender; o no supervisado si la red reconoce automáticamente en los datos el patrón que debe aprender.

Por otra parte, el flujo de información que manejan puede ser unidireccional, cuando la información sigue una dirección única desde las neuronas de entrada a los de salida; o realimentado, donde el flujo de información no es único al incorporar circuitos de realimentación entre capas de la red. En una primera aproximación, se indicará que las redes unidireccionales con aprendizaje supervisado pueden utilizarse para muchos problemas de interés médico. El desarrollo de una RNA necesita planificarse adecuadamente para conseguir una red convenientemente entrenada que alcance una precisión óptima.

El entrenamiento consiste en la presentación repetida de un conjunto suficientemente amplio de datos de entrenamiento formado por las entradas y los valores correspondientes de las variables a predecir, hasta conseguir que los pesos internos conduzcan a resultados óptimos en la capa de salida, acercándose lo más posible a los resultados esperados. En un contexto médico, el entrenamiento consistiría en presentar a la red, de forma iterativa, los valores de las distintas variables clínicas (en forma de valores de la capa de entrada) de cada paciente y conseguir que la red sea capaz de predecir

el estado final observado en cada paciente (indicados por el estado de las capas de salida de la red) de la manera más precisa posible).

3.5 Aplicaciones de redes neuronales artificiales en medicina

- Predicción de resultados [60]
 - Cirugías (cardíacas, hígado, cirrosis, pulmonar, entre otras).
 - Trasplantes (cornea, hígado, corazón, entre otros).
 - Neoplasias (colón, páncreas, entre otros).
 - Monitorización en cirugías.

- Diagnósticos
 - Apendicitis.
 - Tuberculosis.
 - Análisis sanguíneo.
 - Alteraciones psicológicas y psiquiátricas.
 - Detección de deformaciones de la cornea.
 - Detección de taquicardias ventriculares y supraventriculares.

- Aplicaciones diversas (análisis de ondas y análisis de imágenes)
 - Procesamiento de imágenes médicas para diagnóstico temprano preventivo.
 - Resonancias magnéticas.
 - Reconocimiento de imágenes en ultrasonidos.
 - Electrocardiograma.
 - Reconocimiento de lesiones en imágenes de mamografía.
 - Detección, monitoreo, procesamiento y análisis de registros en los encefalogramas.

- Farmacología
 - Optimización de dosis y predicción de reacciones adversas a los medicamentos.

3.6 Clasificador neuronal de umbrales aleatorios RTC (*Random Threshold Classifier*)

El clasificador neuronal de umbrales aleatorios RTC (*Random Threshold Classifier*) es uno de los que se pueden utilizar para el reconocimiento de texturas en imágenes [61].

A continuación se describirá la estructura de este clasificador (Figura 3.3). Consta de unas entradas, a las que llamaremos X_i ($i=1 \dots n$), estas entradas estarán constituidas por características principales de la imagen. Estas entradas son las mismas para todos los grupos neuronales (1, S). Estarán conectadas a las neuronas con umbrales de valores l_{ij} y h_{ij} (*low* (*bajo*) y *high* (*alto*) respectivamente); el índice i representa la característica y el índice j representa el grupo neuronal al que se encuentra conectado.

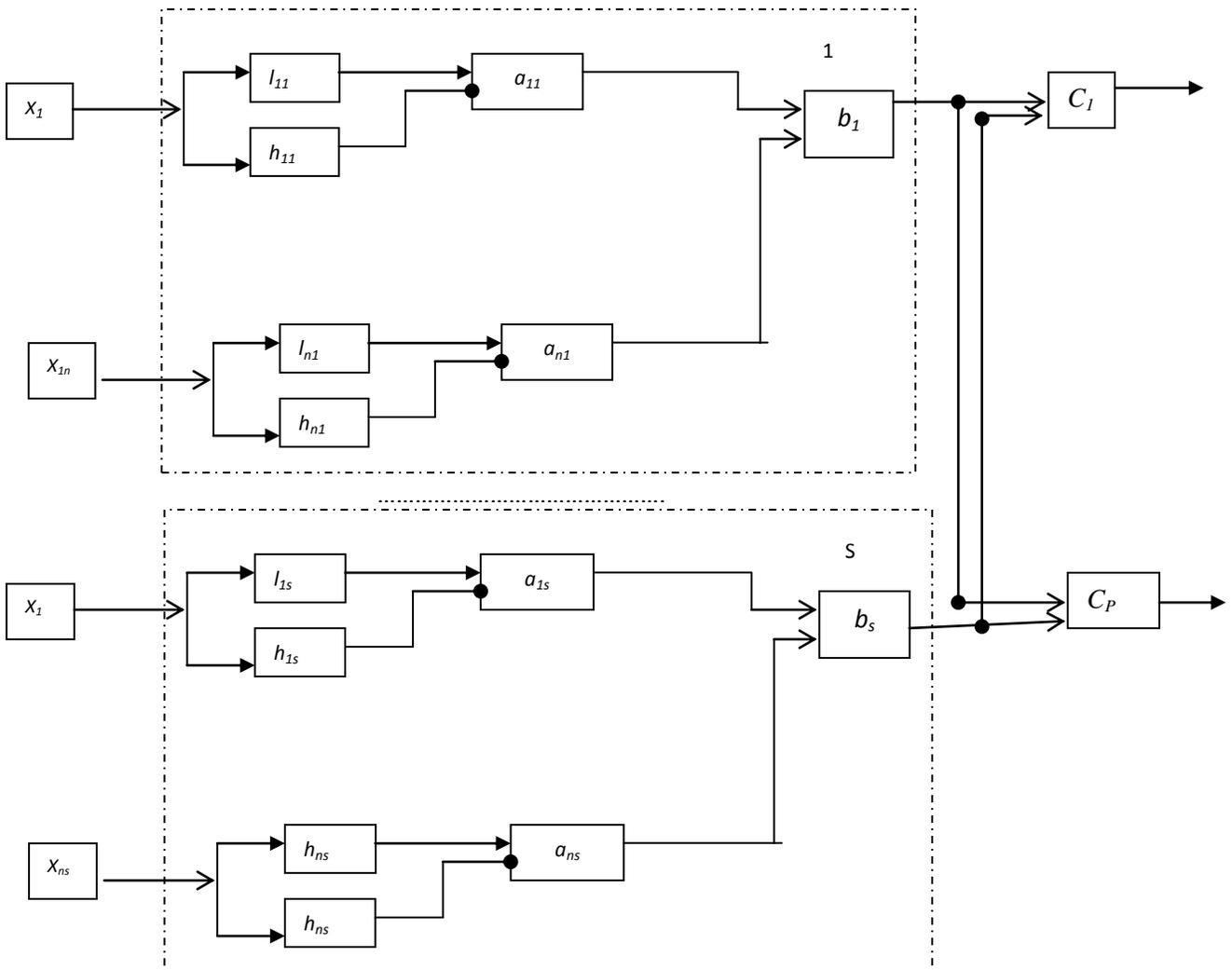


Figura 3.3. Estructura del clasificador neuronal RTC

Los umbrales l_{ij} y h_{ij} se eligen de manera aleatoria cumpliendo siempre la regla que l_{ij} sea menor a h_{ij} .

$$l_{ij} < h_{ij}, \tag{3.1}$$

Los umbrales l_{ij} y h_{ij} están conectados a la neurona a_{ij} . La neurona a_{ij} está excitada cuando X_i pertenece a la diapazone (l_{ij}, h_{ij}) .

Si X_i es menor que l_{ij} , la neurona con este umbral no da respuesta. Si X_i es mayor que h_{ij} , la neurona con este umbral funciona con conexión de inhibición, es decir, prohíbe a la neurona a_{ij} funcionar.

La neurona a_{ij} estará conectada a la neurona b_j , la cual únicamente está funcionando cuando las neuronas que están conectadas con ella tienen excitaciones. Con solo una de las neuronas a_{ij} que no se encuentre excitada, no se va a tener respuesta la neurona b_j .

Todas las neuronas a_{ij} deben estar excitadas para que pueda existir respuesta en la neurona b_j . Entonces la neurona b_j funciona como una compuerta lógica “AND” (“Y”).

De manera geométrica podemos observar un valor en X_1 y X_2 (Figura 3.4) con sus respectivos valores l_1 y l_2 (generados de manera aleatoria) y su grafica nos indica que el área de cruce entre las 2 zonas es la parte donde funcionan ambos y la neurona, por ende también funcionará.

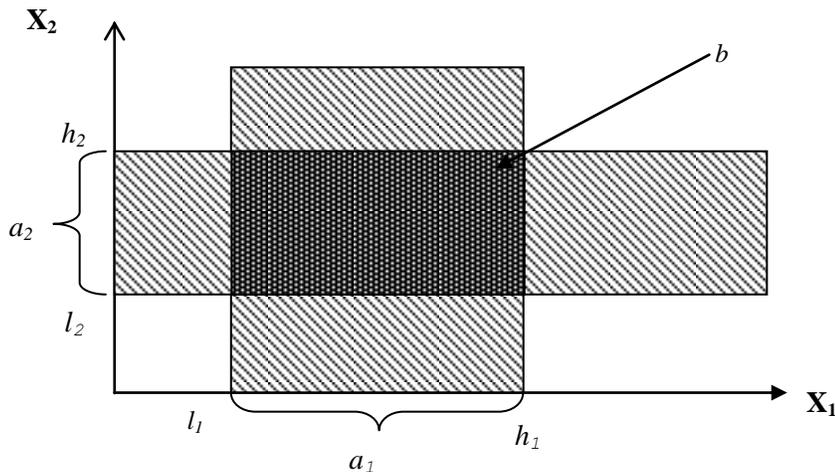


Figura 3.4 Representación geométrica del Clasificador RTC

El número de rectángulos que tengamos va a ser igual al número de vectores binarios b que tengamos. Los rectángulos van a ser de diferente tamaño debido a que son generados de manera aleatoria.

La estructura del clasificador neuronal RTC contiene, entre las capas B y C, las conexiones con pesos w_{ij} . El principio de conectividad es de “todas” las neuronas de la capa B con “todas” las neuronas de la capa C. El proceso de entrenamiento se realiza por medio de un cambio a dichos pesos. Los pesos pueden tener tanto valores positivos como valores negativos. Para dicho proceso utilizamos la regla de entrenamiento de Hebb, donde el índice c representa la clase correcta y el índice i representa la clase incorrecta.

$$\begin{aligned} w_{jc}(t+1) &= w_{jc}(t) + a_j \\ w_{ji}(t+1) &= w_{ji}(t) - a_j \end{aligned} \tag{3.2}$$

3.6.1 Histograma de brillo

Un histograma es una representación gráfica y estadística que ofrece información cuantitativa sobre una imagen. En el eje X vamos a representar los intervalos de nivel de gris (que va de 0 a 255, para una escala de gris de 8 bits) y en el eje Y vamos a tener el número de píxeles que corresponde a cada intervalo. El histograma de brillo nos proporciona una distribución gráfica de los niveles de gris de los píxeles en una imagen. Con este histograma vamos a realizar un barrido de toda la imagen.

Vamos a tener intervalos a analizar (para este caso se van a generar 16 intervalos). Se va a generar una variable X_{ij} donde i y j son los índices de barrido mediante dos ciclos “for”. Los resultados del barrido se guardan en una matriz $B[R]$ donde R es el intervalo. Se analizan 2 píxeles contiguos, si cae dentro de un intervalo se grafica y se pasa a analizar el siguiente par de píxeles. Se verifica dentro de que intervalo cae y se aumenta en uno y se grafica en el histograma (Figura 3.5).

$B[R]++ \leftarrow$ aumenta en uno al intervalo donde está el píxel analizado.

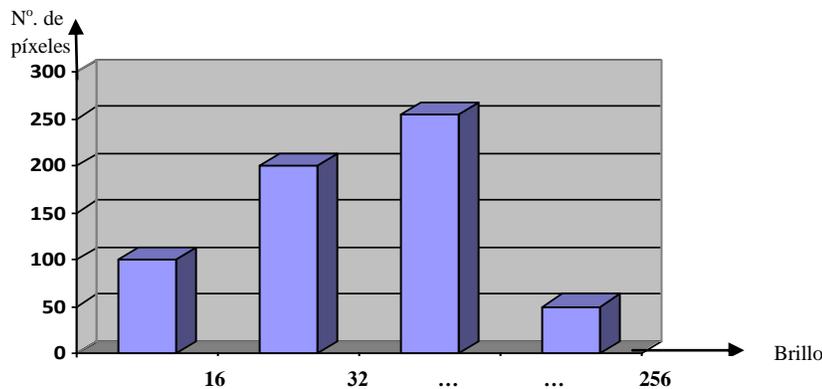


Figura 3.5 Histograma de brillo

3.6.2 Histograma de contraste

El contraste es la diferencia de brillo entre dos píxeles vecinos. Para obtener el histograma de contraste vamos a tener intervalos a analizar (para este caso se van a generar 16 intervalos). Se va a generar una variable X_{ij} donde i y j son los índices de barrido mediante dos ciclos “for”. Los resultados del barrido se van a guardar en una matriz B [R] donde R es el intervalo. Se analizan 2 píxeles que estén contiguos, y se realiza la diferencia entre ambos. Si cae dentro de un intervalo se grafica y se analiza el siguiente par de píxeles.

Se verifica dentro de que intervalo cae y se aumenta en uno y se grafica en el histograma (Figura 3.6).

B [R] ++ ← aumenta en uno al intervalo donde está el píxel analizado.

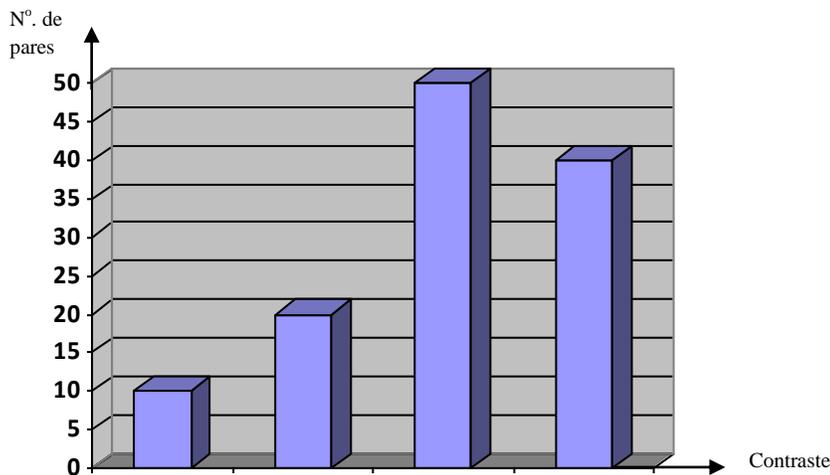


Figura 3.6 Histograma de contraste

3.6.3 Histograma de orientación de microcontornos

Este histograma puede utilizarse como característica para el reconocimiento de texturas.

Como primer paso necesitamos analizar píxeles vecinos (Figura 3.7):

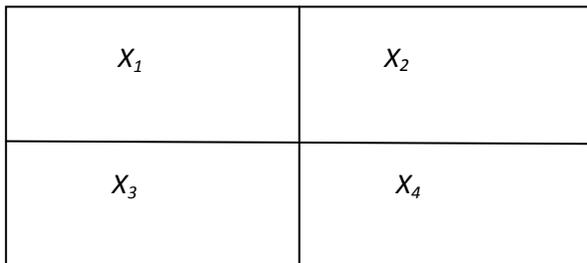


Figura 3.7 Píxeles vecinos

Este algoritmo está basado en el algoritmo de Schwartz [61] y contiene los siguientes pasos:

1. Sumar los elementos de las diagonales (sumas de brillos)

$$y_1 = X_1 + X_4, \tag{3.3}$$

$$y_2 = X_2 + X_3.$$

2. Definir una constante experimental c_1 (dependiendo de la imagen que tengamos). La constante c_1 depende de las imágenes con las que trabaja el sistema. Cabe aclarar que la constante se elige sólo una vez para cada base de imágenes.

3. Si el valor absoluto de la diferencia de $|y_1 - y_2|$ es mayor a la constante c_1 , el contorno no existe y se analizan cuatro nuevos píxeles vecinos. ($|y_1 - y_2| > c_1$) (Figura 3.8)

X_2	X_5
X_4	X_6

Figura 3.8 Nuevos píxeles vecinos

En caso de que el valor absoluto de la diferencia de $|y_1 - y_2|$ sea menor a la constante c_1 el contorno existe y se puede definir su orientación ($|y_1 - y_2| < c_1$). Cuando la diferencia cae dentro de esta condición, se pasa al siguiente paso. De lo contrario, el contorno no existe y se tienen que analizar los siguientes cuatro píxeles vecinos.

4. Posteriormente calculamos las deltas, que son el valor absoluto las diferencias de los valores de los píxeles vecinos en diagonal, es decir:

$$\delta_1 = X_1 - X_4,$$

$$\delta_2 = X_3 - X_2.$$

$$\tag{3.4}$$

5. Se elige el valor máximo y mínimo aplicando las siguientes fórmulas:

$$\Delta_1 = \max(|\delta_1|, |\delta_2|),$$

$$\Delta_2 = \min(|\delta_1|, |\delta_2|).$$

$$\tag{3.5}$$

6. Inmediatamente procedemos a calcular el valor del ángulo y en radianes (el valor de y debe ser siempre menor de “1”, ($y < 1$))

$$y = \frac{\Delta_2}{\Delta_1}. \quad (3.6)$$

7. Procedemos a analizar la Tabla 3.1 en relación con los valores anteriormente calculados

Tabla 3.1. Cálculo de los ángulos

y	$0 - \frac{\pi}{4}$	$\frac{\pi}{4} - \frac{\pi}{2}$	$\frac{\pi}{2} - \frac{3\pi}{4}$	$\frac{3\pi}{4} - \pi$	$\pi - \frac{5\pi}{4}$	$\frac{5\pi}{4} - \frac{3\pi}{2}$	$\frac{3\pi}{2} - \frac{7\pi}{4}$	$\frac{7\pi}{4} - 2\pi$
y	(-)	(+)	(-)	(+)	(-)	(+)	(-)	(+)
	↘	↗	↘	↗	↘	↗	↘	↗
δ_1	+	+	+	-	-	-	-	+
δ_2	-	+	+	+	+	-	-	-
$ \delta_1 - \delta_2 $	+	+	-	-	+	+	-	-

Las flechas hacia arriba indican que se debe sumar el valor de y , y la flecha hacia abajo indica que se debe restar el valor de y ; en ambos casos es al intervalo.

Teniendo el valor del ángulo del contorno, podemos calcular todos los contornos en la imagen y calcular el histograma de orientación de microcontornos para toda la imagen. Posteriormente se grafica el histograma (Figura 3.9).

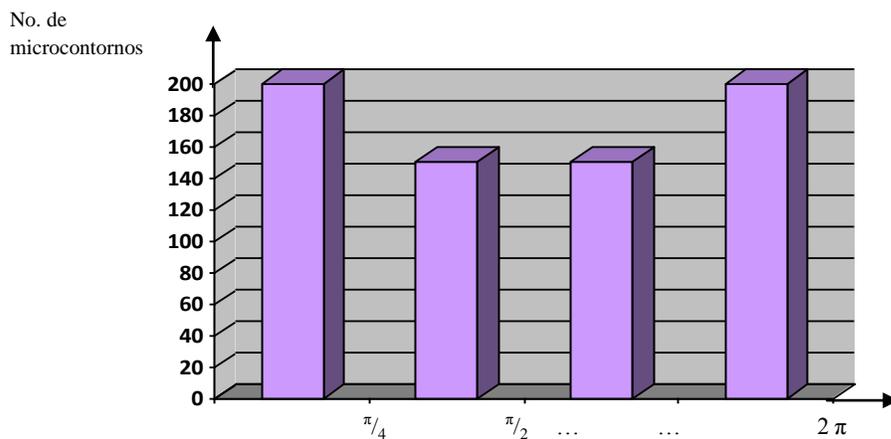


Figura 3.9 Histograma de orientación de microcontornos

CAPÍTULO 4

4. El sistema desarrollado y los resultados

4.1 Introducción

El ámbito médico necesita de diversas herramientas y software especializado para la detección, diagnóstico y tratamiento de diferentes enfermedades y aunado con el ámbito ingenieril se puede desarrollar software que ayude a la detección temprana de enfermedades y en este caso en particular, el reconocimiento de texturas en imágenes médicas.

Parte de la investigación dentro de la Universidad Nacional Autónoma de México pretende ser multidisciplinaria, es por ello que durante el desarrollo del presente trabajo de tesis se analizará un caso clínico particular: la enfermedad de Chagas [62], tanto desde un punto de vista médico al informar la definición y estatus de la enfermedad, así como ingenierilmente a partir del reconocimiento de imágenes basado en el entrenamiento de una red neuronal artificial (un clasificador neuronal).

Como parte del trabajo de tesis, se analizará una base de imágenes de tejido infectado por la enfermedad de Chagas (agradecemos al Centro de Investigaciones Regionales Dr. Hideyo Noguchi, de la Universidad Autónoma de Yucatán, por la base de imágenes de tejido). Para detectar la enfermedad está desarrollada una red neuronal artificial – el clasificador RTC para realizar un reconocimiento de texturas imágenes médicas, utilizando como software Borland® C++ 6.0. Para describir las imágenes calculamos las siguientes características: histogramas de brillo, de contraste y de orientación de microcontornos. La meta es construir un sistema computacional para reconocer la imagen con tejido dañado por la enfermedad correspondiente.

4.2 Objetivo principal

El objetivo principal de este trabajo es el desarrollo e investigación de un sistema de reconocimiento de texturas en imágenes médicas basado en el clasificador neuronal RTC realizado en Borland® C++ 6.0 para el caso particular de la enfermedad de Chagas.

4.3 Descripción de la base de imágenes

El problema de reconocimiento de texturas en imágenes médicas no es trivial; esto es debido a las diferentes texturas, colores, formas, tamaños y otros atributos de las muestras. A nuestra disposición tenemos un conjunto de imágenes con tejido dañado por la enfermedad de Chagas (agradecemos al Centro de Investigaciones Regionales Dr. Hideyo Noguchi, de la Universidad Autónoma de Yucatán, por la base de imágenes de tejido). Este conjunto contiene 10 imágenes, las cuales están presentadas en formato BMP (Mapa de bits) con una resolución de 533 x 400 píxeles y un tamaño de 625 KB. Las imágenes contenidas en la base son de diferentes texturas de tejido dañado con la enfermedad de Chagas. Se muestra un ejemplo de la imagen en la Figura 4.1.

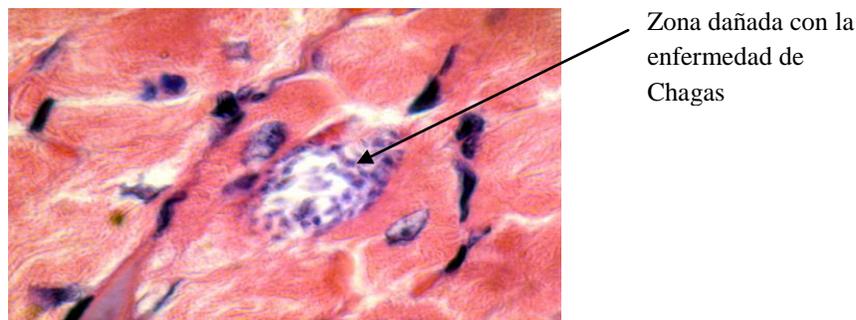


Figura 4.1 Imagen de tejido

Para poder trabajar con el clasificador neuronal RTC, se tiene la base de imágenes dividida en dos grupos: uno para el entrenamiento y otro para la validación del sistema.

Para nuestra base de imágenes se tienen 10 imágenes en total, de las cuales 5 son utilizadas para entrenamiento y otras 5 son usadas para prueba del sistema desarrollado. El entrenamiento utilizado en este caso es el entrenamiento supervisado o con maestro.

Para este propósito se prepararon las imágenes, marcando las zonas infectadas con color blanco. Se utilizaron imágenes marcadas (zona blanca en la imagen) de las cuales se presenta un ejemplo en la Figura 4.2. En este caso la zona blanca o marcada corresponde a la zona infectada por la enfermedad de Chagas. Todas las 10 imágenes son marcadas, es decir que cada una de ellas tiene la zona blanca.

Para aumentar el número de muestras para nuestro clasificador neuronal, se escaneo cada imagen con una ventana (también llamada área local) de dimensiones $h \times v$. En los experimentos realizados $h = v = 40$. Para cada ventana se calcularon 48 características o parámetros (16 valores de histograma de brillo, 16 valores de histograma de contraste y 16 valores de histograma de orientación de microcontornos).

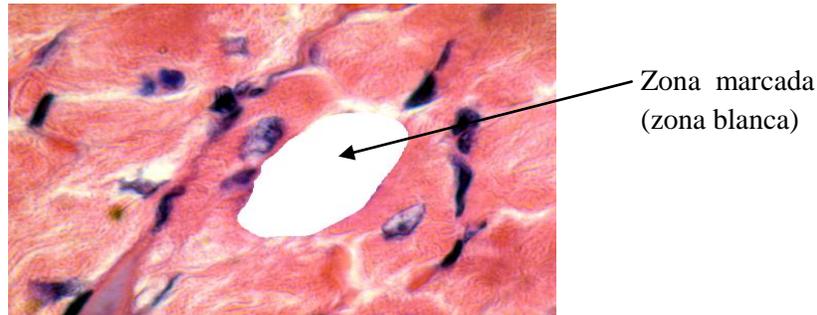


Figura 4.2 Imagen marcada

4.4 Descripción de los experimentos

La primera fase de los experimentos es la de entrenamiento del sistema. Dicha fase será supervisada para poder realizar la comparación con la salida obtenida mediante el clasificador y la salida correcta. Si ambas salidas son iguales se entiende que no es necesario realizar cambio alguno en el sistema, en caso contrario se tendrían que modificar apropiadamente los valores de los pesos de conexiones entre la capa *B* y la capa *C* del sistema.

En el proceso de entrenamiento del sistema, éste “sabe” si existe la presencia de tejido dañado con la enfermedad de Chagas. Esto es porque el sistema “reconoce” si existe presencia de tejido enfermo (marcada en blanco) o tejido sano (zona sin marcado blanco) (Figura 4.2).

El proceso de entrenamiento es posible realizar utilizando uno de los dos siguientes criterios:

1. Hasta obtener cero errores de reconocimiento en la fase de entrenamiento, o
2. Mediante un número fijo de ciclos de entrenamiento, es decir, decidiendo a priori cuántas veces será introducido todo el conjunto de imágenes de entrenamiento y una vez superado dicho número se detiene el proceso de entrenamiento.

Para el término de proceso de entrenamiento del sistema presentado en esta tesis se tomó como base el segundo criterio.

El proceso de validación del clasificador neuronal se hace con las imágenes que no participaron en el proceso de entrenamiento; esto es con el propósito de obtener el porcentaje de reconocimiento de texturas.

Es necesario decir que para cada serie de los experimentos con el sistema, se eligieron 5 imágenes aleatoriamente de 10 imágenes que conforman la base de imágenes.

Dentro del sistema se tienen las siguientes constantes necesarias para el análisis de la imagen:

- **H**; Alto de imagen, con valor igual a 400 que es el tamaño de alto de la imagen.
- **W**; Ancho de imagen, con valor igual a 533 que es el tamaño de ancho de la imagen.
- **NumOfClasses**; Número de clases con valor igual a 2: Tiene Chagas (1), No tiene Chagas (0).
- **GF**; Grupo de características, inicialmente con valor igual a 3 (histograma de brillo, histograma de contraste e histograma de orientación de microcontornos).
- **SG**; Tamaño del grupo de características, inicialmente con valor igual a 16.

Dentro del sistema se tienen las siguientes variables a considerar para el análisis de la imagen:

- **h**; Alto de la ventana (área local) con la que se escaneará la imagen, inicialmente con valor 40.
- **w**; Ancho de la ventana (área local) con la que se escaneará la imagen, inicialmente con valor 40.
- **NumOfNeurons**; Número de neuronas, inicialmente con valor igual a 32000.
- **SizeOfSubspace**; Tamaño de Subespacio – Número de entradas aplicadas para una neurona (7 de $GF*SG=48$, es un máximo de 7 de las 48 características a analizar en la imagen).
- **NumOfRecIm**; Número de imágenes para reconocimiento (máx.=10), inicialmente con valor igual a 5.
- **NI**; Número total de imágenes en la base de imágenes, contamos con 10 imágenes.
- **NumOfTrainCycles**; Número de ciclos de entrenamiento, inicialmente con valor igual a 10.

El sistema funciona de la manera que se describe a continuación.

4.5 Análisis de resultados

El programa que realiza el funcionamiento del clasificador neuronal RTC consiste de dos partes: forma y texto del programa. Con la forma se construyó un ambiente de trabajo a través de un menú (Figura 4.3).

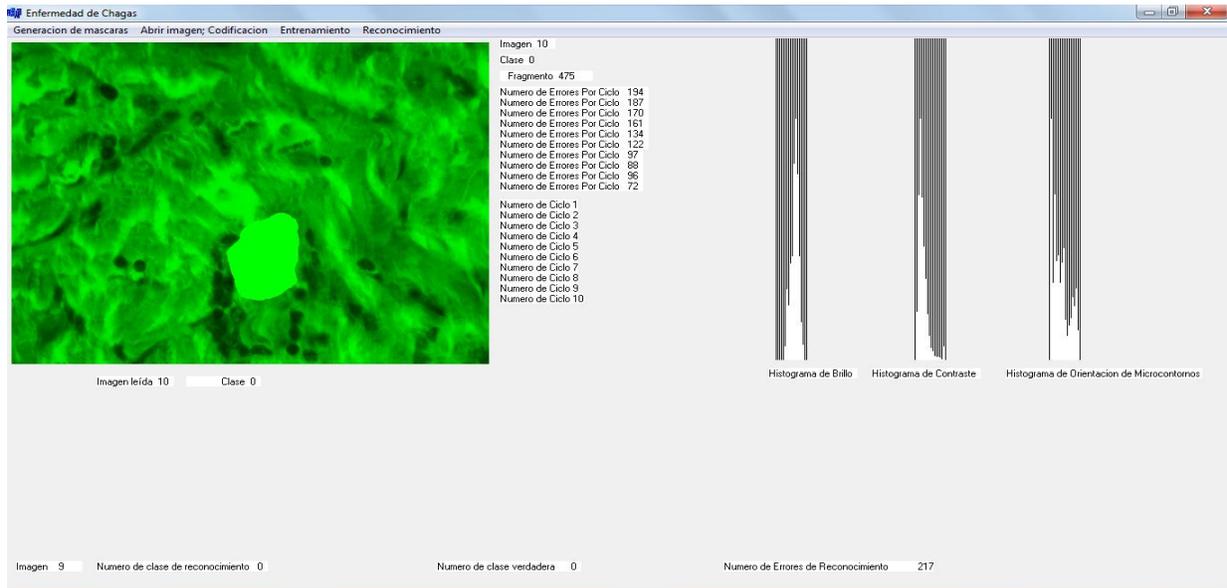


Figura 4.3 Imagen del programa que realiza el sistema desarrollado

Dentro del menú se tiene la opción “Generación de máscaras”. La máscara significa la primera parte de la estructura del clasificador neuronal RTC, es decir las neuronas con umbrales aleatorios h y l . Posteriormente se eligen aleatoriamente 5 imágenes para utilizarlas en el proceso de entrenamiento de la red.

El siguiente punto del menú es “Abrir imagen–Codificación”. Se abren las imágenes (10 en total en la base de imágenes). Por medio de un escaneo de una ventana de 40 x 40, se aumenta el número de muestras del tejido. Se maneja clase cero (0) para el caso donde no existe la enfermedad de Chagas y clase uno (1) donde el tejido está infectado con la enfermedad de Chagas y se calculan los histogramas de brillo, de contraste y de orientación de microcontornos para la ventana. Subsiguientemente se codifica esta información.

Posteriormente se inicia el proceso de entrenamiento con la opción “Entrenamiento” dentro del menú para 5 imágenes, dentro del cual se manejan 10 ciclos y se obtiene el número de errores por ciclo.

Finalmente se tiene la parte de reconocimiento (dentro de la opción “Reconocimiento”, dentro del menú del sistema desarrollado) donde se manejan las restantes imágenes y se obtienen los errores del mismo.

En la Figura 4.4 se muestra el resultado de elegir aleatoriamente 5 imágenes al compilar el programa que realiza el sistema desarrollado. Posteriormente en la Figura 4.5 se muestran las 5 imágenes restantes.

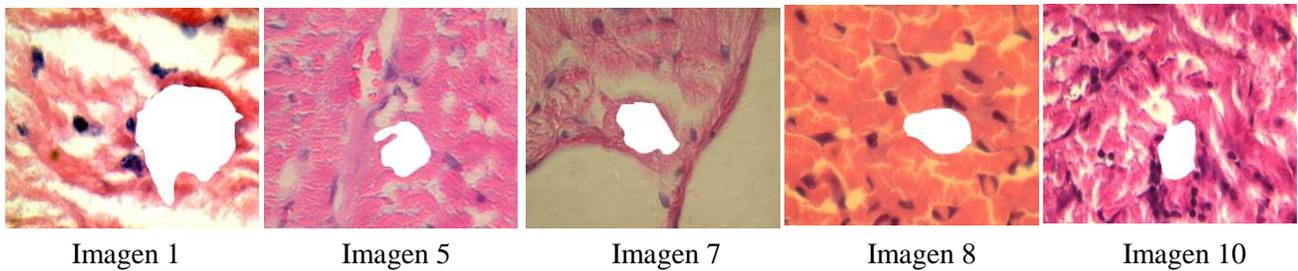


Figura 4.4 Imágenes aleatorias

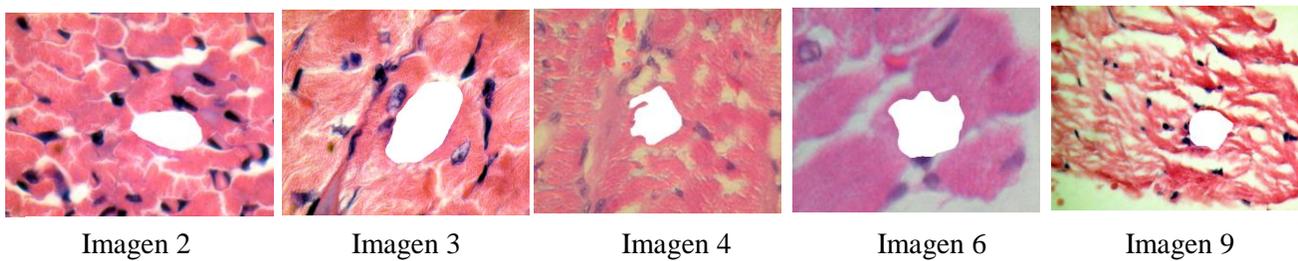


Figura 4.5 Imágenes restantes

Se obtuvo el número total de ventanas (NTV) para cinco imágenes de acuerdo a la siguiente fórmula:

$$NTV = \frac{\left(\left(\frac{W}{\frac{w}{2}} - 1 \right) \times \left(\frac{H}{\frac{h}{2}} - 1 \right) \right) \times (NIB)}{2} = \frac{\left(\frac{533}{20} - 1 \right) \left(\frac{400}{20} - 1 \right) (10)}{2} = \frac{(25.65)(19)(10)}{2} = 2470 \quad (4.1)$$

donde W y H corresponden al ancho y alto de la imagen, respectivamente; w y h corresponden al ancho y alto de la ventana (o área local) con la que se escanea la imagen con paso $\frac{w}{2}$ y $\frac{h}{2}$ (Figura 4.6), y finalmente NIB es el número de imágenes de la base (para nuestro caso se utilizaron 10 imágenes).

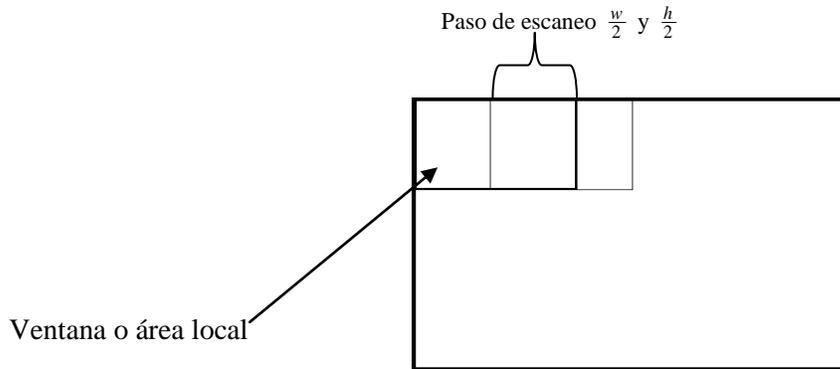


Figura 4.6 Ventana de escaneo

Para la investigación realizada en la presente tesis, se tiene que la imagen tiene dimensiones 533 x 400 píxeles, y la ventana (área local) tiene dimensiones de 40 x 40 píxeles. Por lo tanto, el $NTV=2470$ para nuestro caso.

El porcentaje de error X se calcula mediante la siguiente fórmula:

$$X = \frac{100 \times N_{error}}{NTV} \% \quad (4.2)$$

donde N_{error} , es el número de ventanas reconocidas con error para 5 imágenes. Cada ventana (o área local) pertenece a una clase: tejido sano o tejido enfermo. El sistema desarrollado trata de reconocer a qué clase pertenece la textura de esta ventana. Cuando la textura reconocida no coincide con la clase correcta, entonces tiene un error de reconocimiento. El porcentaje de error nos indicará el número de ventanas en las cuales el sistema erró al reconocer entre el fondo de la imagen y la parte de tejido dañada con dicha enfermedad.

El experimento consiste en seleccionar 5 imágenes. Para la primera versión del clasificador RTC la matriz de pesos entre la capa B y la capa C puede tener valores $w_{ij} \geq 0$, pesos positivos. Posteriormente se realiza el entrenamiento de nuestro clasificador neuronal RTC. Se eligen 10 ciclos

de entrenamiento. Esto significa que se va a presentar 10 veces una secuencia de 5 imágenes. Para cada ciclo se calcula el número de errores.

Se realizará dicho experimento 4 veces. Esto significa que vamos a cambiar 4 veces la secuencia de 5 imágenes. Los resultados de los experimentos se muestran en la Tabla 4.1.

Tabla 4.1 Resultados de entrenamiento para 4 experimentos (pesos positivos)

Número de Experimento	Número de Ciclos	Número de errores (N_{error})	% de Error
1	1	147	5.95
	2	144	5.83
	3	144	5.83
	4	136	5.51
	5	128	5.18
	6	110	4.45
	7	111	4.49
	8	102	4.85
	9	89	3.60
	10	87	3.52
2	1	151	6.11
	2	142	5.75
	3	130	5.26
	4	140	5.67
	5	114	4.61
	6	111	4.49
	7	94	3.81
	8	88	3.56
	9	85	3.44
	10	80	3.24

Tabla 4.1 Resultados de entrenamiento para 4 experimentos (pesos positivos)

Número de Experimento	Número de Ciclos	Número de errores (N_{error})	% de Error
3	1	137	5.54
	2	141	5.70
	3	139	5.68
	4	123	4.98
	5	125	5.06
	6	118	4.78
	7	115	4.66
	8	98	3.97
	9	96	3.89
	10	81	3.28
4	1	137	5.55
	2	153	6.19
	3	139	5.63
	4	127	5.14
	5	100	4.05
	6	98	3.97
	7	97	3.93
	8	87	3.52
	9	87	3.52
	10	80	3.24

A continuación se presentará la Tabla 4.2 con los resultados promedio para 4 experimentos (pesos positivos).

Tabla 4.2 Resultados promedio de entrenamiento para 4 experimentos (pesos positivos)

Número de ciclos	Número de errores (N_{error})	% de Error	% de Reconocimiento
1	143	5.79	94.21
2	145.25	5.88	94.12
3	138	5.59	94.41
4	131.5	5.32	94.68
5	141.75	5.74	94.26
6	109.25	4.42	95.58
7	104.25	4.22	95.78
8	93.75	3.79	96.21
9	89.25	3.61	96.39
10	82	3.32	96.68

Para probar el sistema se muestran otras 5 imágenes que no participaron en el proceso de entrenamiento. Los resultados se encuentran mostrados en la Tabla 4.3

Tabla 4.3 Resultados promedio de reconocimiento para 4 experimentos (pesos positivos)

Número de experimento	Número de errores por ciclo en reconocimiento (N_{error})	% de Error	% de Reconocimiento
1	252	10.20	89.80
2	235	9.52	90.48
3	234	9.47	90.53
4	231	9.36	90.64

Después de esta serie de experimentos, realizamos una nueva serie de experimentos para investigar nuestro sistema basado en el clasificador neuronal permitiendo ahora a la matriz de pesos entre la capa *B* y la capa *C* tener valores de pesos tanto positivos como negativos.

$$w_{ij} < 0 \text{ OR } w_{ij} \geq 0 \quad (4.2)$$

Realizamos nuevas pruebas con el sistema, seleccionando 5 imágenes para entrenamiento.

Realizamos el entrenamiento de nuestro clasificador neuronal RTC. Se eligieron 10 ciclos de entrenamiento. Para cada ciclo calculamos el número de errores. Realizamos dicho experimento 4 veces. Los resultados de los experimentos se muestran en la Tabla 4.4.

Tabla 4.4 Resultados de entrenamiento para 4 experimentos (pesos positivos y negativos)

Número de Experimento	Número de Ciclos	Número de errores (N_{error})	% de Error
1	1	200	8.09
	2	176	7.13
	3	138	5.59
	4	113	4.57
	5	111	4.49
	6	84	3.40
	7	77	3.12
	8	75	3.04
	9	66	2.67
	10	65	2.63
2	1	186	7.53
	2	171	6.92
	3	144	5.83
	4	140	5.67
	5	112	4.53
	6	100	4.05
	7	88	3.56
	8	85	3.44
	9	63	2.55
	10	58	2.35

Tabla 4.4 Resultados de entrenamiento para 4 experimentos (pesos positivos y negativos)

Número de Experimento	Número de Ciclos	Número de errores (N_{error})	% de Error
3	1	206	8.34
	2	170	6.88
	3	149	6.03
	4	120	4.86
	5	105	4.25
	6	99	4.01
	7	91	3.68
	8	93	3.76
	9	90	3.64
	10	86	3.48
4	1	47	1.91
	2	44	1.78
	3	43	1.74
	4	43	1.74
	5	39	1.58
	6	41	1.66
	7	35	1.42
	8	35	1.42
	9	31	1.26
	10	34	1.38

A continuación se presentará la Tabla 4.5 con los resultados promedio para 4 experimentos con pesos positivos y negativos.

Tabla 4.5 Resultados promedio de entrenamiento para 4 experimentos (pesos positivos y negativos)

Número de ciclos	Número de errores (N_{error})	% de Error	% de Reconocimiento
1	159.75	6.47	93.53
2	140.25	5.68	94.32
3	118.5	4.79	95.21
4	104	4.21	95.79
5	91.75	3.72	96.28
6	81	3.28	96.72
7	72.75	2.94	97.06
8	72	2.92	97.08
9	62.5	2.53	97.47
10	60.75	2.46	97.54

Para probar el sistema se muestran otras 5 imágenes que no participaron en el proceso de entrenamiento. Los resultados se encuentran mostrados en la Tabla 4.6.

Tabla 4.6 Resultados promedio de reconocimiento para 4 experimentos (pesos positivos y negativos)

Número de experimento	Número de errores por ciclo en reconocimiento (N_{error})	% de Error	% de Reconocimiento
1	257	10.40	89.60
2	198	8.02	91.98
3	285	11.54	88.46
4	288	11.66	88.34

De la Figura 4.7 se puede observar que el mejor porcentaje de reconocimiento se tiene cuando a la matriz de pesos entre la capa B y la capa C se le permite tener valores de pesos tanto positivos como negativos. Se puede observar que conforme avanzan los ciclos, el clasificador neuronal va “aprendiendo” a reconocer la imagen de manera más eficaz.

En tanto que cuando la matriz de pesos toma valores positivos, se observa una inestabilidad entre ciclos y un porcentaje de reconocimiento menor.

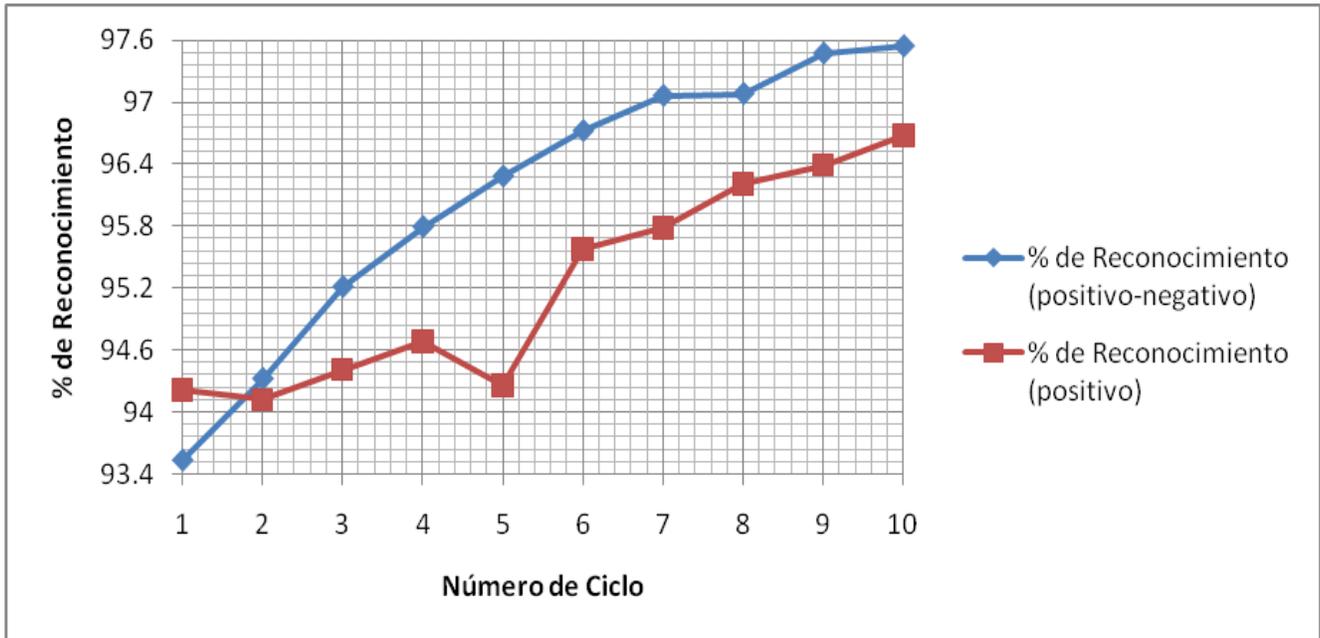


Figura 4.7 Gráfica de porcentaje de reconocimiento

La matriz con pesos positivos y negativos entrena más rápido y da mejores resultados.

El tiempo de respuesta del clasificador neuronal es otro de los parámetros importantes, ya que nos permite conocer el tiempo necesario que tarda el sistema para reconocer una imagen y realizar una clasificación de la misma.

En el desarrollo de la presente tesis se midió el tiempo de respuesta para el sistema con las siguientes características: ventana de 20 x 20, 10 ciclos, 5 imágenes para entrenamiento y 5 imágenes para reconocimiento. El tiempo de cada operación se desglosa de la siguiente manera: 7.8 segundos en abrir y codificar una imagen, 5 segundos en el entrenamiento para reconocer la imagen y 7 segundos en el reconocimiento de la imagen, para cuando la matriz de pesos entre la capa B y C toma valores tanto positivos como negativos.

Conclusiones

El reconocimiento de texturas es una tarea importante en el reconocimiento de imágenes médicas. Para este propósito se propuso utilizar el clasificador neuronal de umbrales aleatorios (RTC). Como entradas para el clasificador RTC calculamos los histogramas de brillo, de contraste y de orientación de microcontornos.

Durante el desarrollo del sistema se obtuvieron los resultados del cálculo de dichos histogramas, con lo que se pudo clasificar la base de imágenes de tejido sano y tejido dañado con la enfermedad de Chagas.

Se programó el algoritmo de trabajo del clasificador neuronal RTC usando Borland[®] C++ 6.0.

El entrenamiento y validación del sistema se realizó mediante una base de 10 imágenes: de las cuales 5 se utilizaban para el entrenamiento y 5 se utilizaban para el reconocimiento.

Los resultados obtenidos para la tarea de reconocimiento de texturas en imágenes médicas para el caso particular Chagas muestran la posibilidad de aplicar el clasificador neuronal RTC para el reconocimiento de texturas en imágenes médicas.

El mejor porcentaje de reconocimiento es 97.54%, que es cuando permitimos a la matriz de pesos entre la capa *B* y la capa *C* tener valores de pesos tanto positivos como negativos. En un futuro existe la posibilidad de mejorar los resultados obtenidos.

Para este propósito planeamos realizar los siguientes trabajos.

Trabajos a futuro

Como trabajos a futuro se proponen:

- Aumentar la base de imágenes, con la finalidad de mejorar los resultados obtenidos con el clasificador neuronal RTC.
- Investigar cómo influye el número de ciclos de entrenamiento, para ver la posibilidad de una mejora del sistema.
- Investigar cómo influye el tamaño de la “ventana (área local)” de escaneo, para observar si el entrenamiento y reconocimiento disminuyen en sus errores.
- Realizar modificaciones en el clasificador neuronal RTC para mejorar la tarea de reconocimiento.
- Adaptar el sistema para su uso práctico en el reconocimiento de la enfermedad de Chagas en sus diferentes etapas.
- Buscar la posibilidad de poder implementar el sistema dentro de zonas hospitalarias para que el trabajo del imagenólogo sea menos complicado en la detección de dicha enfermedad.

Anexo. Texto del programa

```
//-----Random Threshold Classifier for Chagas
//Subspace RTC

#include <stdlib.h>//random
#include <mem.h>//memset
#include <stdio.h>//sprintf
#include <math.h>
#include <vcl.h>
#include <algorithm>
#include <iostream>
#include <functional>

#pragma hdrstop

#include "TextureProjectC.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;
//-----
//-----
//-----
//!!!Constants

const c=15.99; // c < 16
const ContrastScale=5; //Contrast Histogram
const ContourExistConst=10; //Contour Orientation Histogram
const t=0.41; // tg(22,5)
const ImHeight=400; // image height
const ImWidth=533; // image width
const NormRate=300; //normalization rate for histogram elements
const NumOfClasses=2; //number of output classes Chaga(1)/NoChaga(0)
const GF=3; // GroupOfFeatures
const SG=16; //SizeOfFeatureGroup
//!!!Parameters

const h=40; //window height
const w=40; //window width

const NumOfNeurons=32000; //number of neurons
const SizeOfSubspace=3; //size of subspace - number of input array components applied to one neuron (7 from GF*SG=48)

float GAP=1; // tOn=x0-GAP*r0;tOff=x0+GAP*r0;
float TDS=0.1; //Etrue=Etrue*(1-TDS)

const NumOfRecIm=5; // number of images for recognition (max=10)
const NI=10; //Total Number of Images in Data Base
```

```

int NumOfTrainCycles=10; //number of training cycles

//!!!ImageProcessing

Graphics::TBitmap *ImageBitmap = new Graphics::TBitmap(); //original image

Graphics::TBitmap *ImageBitmap2 = new Graphics::TBitmap(); //marked image

/*Graphics::TBitmap *HistBitmap = new Graphics::TBitmap();
*/

int Image[ImHeight*ImWidth];
int ImageAdv[ImHeight*ImWidth];
int im[h*w][NumOfClasses];

int Image1[ImHeight][ImWidth];
int ImageAdv1[ImHeight][ImWidth];
int ImageAdv2[ImHeight][ImWidth];
int im1[h][w]; //original
int im2[h][w]; //marked: Chagas-white, background- original image

//!!!Histograms

int BrightnessHistogramArray[SG];
int ContrastHistogramArray[SG];
int GradientOrientationHistogramArray[SG];

/*void BrightnessHistogram(int, int);
void ContrastHistogram(int, int);
void GradientOrientationHistogram(int, int);
*/

void BrightnessHistogramAdv(int);
void ContrastHistogramAdv(int);
void GradientOrientationHistogramAdv(int);
void BrightHistOut(int,int);
void ContrHistOut(int,int);
void OrientHistOut(int,int);
void __fastcall Stop(void);

//!!!Random Subspace Classifier
FILE *st1;
FILE *st2;
FILE *st3;
FILE *st4;

int InputArray[GF*SG];
int NeuronArray[NumOfNeurons];
int MaskMatrix[NumOfNeurons][SizeOfSubspace];

//void CreatingOfMaskMatrix(void);
//void CreatingOfInputArray(void);

```

```

//void FillingOfNeuronArray(void);

int NumOfActN;
int NumberOfActiveNeuronsArray[NI*(2*ImHeight/h)*(2*ImWidth/w)+(2*ImHeight/h)*(2*ImWidth/w)];
int NonZeroNeuronsArray[NumOfNeurons];
char ClassDefinitionArray[ImHeight/(h/2)][ImWidth/(w/2)];
long ClassExcitArray[NumOfClasses];
long WeightMatrix[NumOfNeurons][NumOfClasses];

//void CreatingOfWeightMatrix(void);
void WeightMatrixUpdate(void);

int TrainImArray[NI-NumOfRecIm]; //numbers of images for training
int RecImArray[NumOfRecIm]; //numbers of images for recognition

int LimitArray[NumOfNeurons][SizeOfSubspace][2];

void CreatingOfImArrays(void); //creating of training and recognition arrays of image numbers
//void CreatingOfLimitArray(void); //creating of limits array

//void OneTrainingCycle(void);
void Training(void);
void RecognitionForTraining(void);
void Recognition(void);

int TrueClassNumber; //number of true class
int RecClassNumber; //number of recognized class

//!!!Diagnostics

int NumOfErrPerCycle; //number of errors per cycle
int NumOfErr; //number of error during recognition
int NumOfNonZeroNeurons; //number of non zero neurons in NeuronArray

//!!!Stop

//-----
__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
    randomize();
}
//-----

void __fastcall TForm1::OpenImage1Click(TObject *Sender)
{
    int i,i1,i2,in,j,j1,j2,br,br1,ih,iw,jh,jw,s1,s2,k,l;
    char NameOfImFile[50],NameOfImFile2[70],z;
    int ImMean, ImMeanCl[NI];
    int s[1];

```

```

int x0,r0,tOn,tOff;
int k1=0;

int p, q;

NumbOfActN=0;
    k=2*ImHeight/h;
    l=2*ImWidth/w;

if ((st1 = fopen("CodeVectors.dat", "wb")) == NULL)
    {
        fprintf(stderr, "Cannot open file CodeVectors\n");
    }
fclose(st1); /* close file */

if ((st2 = fopen("NumberOfActiveNeurons.dat", "wb")) == NULL)
    {
        fprintf(stderr, "Cannot open file NumberOfActiveNeurons\n");
    }
fclose(st2); /* close file */

if ((st4 = fopen("ClassDefinition.dat", "wb")) == NULL)
    {
        fprintf(stderr, "Cannot open file ClassDefinition\n");
    }
fclose(st4); /* close file */
//-----

    for(i=0;i<NI;i++)
        ImMeanCl[i]=0;

for(in=0;in<NI;in++)
    {
        ImMean=0;
        if (in<10)

sprintf(NameOfImFile,"C:\\Tesis\\TextureProject\\Carolina_project\\ImagesBMP\\Dibujo%d.bmp",in+1);
        else

sprintf(NameOfImFile,"C:\\Tesis\\TextureProject\\Carolina_project\\ImagesBMP\\Dibujo%d.bmp",in+1);
        //NI=10, "else" no trabaja, para futuro aumento de base
        ImageBitmap->LoadFromFile(NameOfImFile);
        Form1->Canvas->TextOutA(300,230,"Imagen abierta: "+AnsiString(in+1)+" ");
        //Form1->Canvas->TextOutA(180,230,"Clase "+AnsiString(z+1)+" ");
        try
        {
// Canvas->Draw(5,5,ImageBitmap);

```

```

    }
catch(...)
    {
        MessageBeep(0);
    }

    for(i1=0;i1<ImHeight;i1++)
        for(j=0;j<ImWidth;j++)
            {
                br=ImageBitmap->Canvas->Pixels[j][i1];
                br1=(br&0xFF)+((br>>8)&0xFF)+((br>>16)&0xFF);
                br1=br1/3;
                ImageAdv1[i1][j]=br1;
                ImMeanCl[in]=ImMeanCl[in]+br1;
            }
    ImMeanCl[in]=ImMeanCl[in]/(ImHeight*ImWidth);

}
ImMean=0;

//-----coding

for(in=0;in<NI;in++)
{
    if (in<10)

sprintf(NameOfImFile,"C:\\Tesis\\TextureProject\\Carolina_project\\ImagesBMP\\Dibujo%d.bmp",in+1);
    else

sprintf(NameOfImFile,"C:\\Tesis\\TextureProject\\Carolina_project\\ImagesBMP\\Dibujo%d.bmp",in+1);

    ImageBitmap->LoadFromFile(NameOfImFile);

    if (in<10)

sprintf(NameOfImFile2,"C:\\Tesis\\TextureProject\\Carolina_project\\ImagesMark\\Dibujo%d_mark.bmp",in+
1);
    else

sprintf(NameOfImFile2,"C:\\Tesis\\TextureProject\\Carolina_project\\ImagesMarkP\\Dibujo%d_mark.bmp",in
+1);

    ImageBitmap2->LoadFromFile(NameOfImFile2);

    try
    {
//        Canvas->Draw(5,5,ImageBitmap);
//        Canvas->Draw(5,400,ImageBitmap2);
    }
    catch(...)

```

```

    {
    MessageBeep(0);
    }

for(i1=0;i1<ImHeight;i1++)
    for(j=0;j<ImWidth;j++)
        {
        br=ImageBitmap->Canvas->Pixels[j][i1];
        br1=(br&0xFF)+((br>>8)&0xFF)+((br>>16)&0xFF);
        br1=br1/3;
        ImageAdv1[i1][j]=br1;
        //ImageAdv1[i1][j][z]=ImageAdv1[i1][j][z]*125/ImMean;

        br=br1=0;
        br=ImageBitmap2->Canvas->Pixels[j][i1];
        br1=(br&0xFF)+((br>>8)&0xFF)+((br>>16)&0xFF);
        br1=br1/3;
        ImageAdv2[i1][j]=br1;
        //ImageAdv1[i1][j][z]=ImageAdv1[i1][j][z]*125/ImMean;
        }

z=0; //0 no hay chagas; 1 hay chagas
p=-1;
q=-1;

for(i1=0;i1<ImHeight-h/2;i1=i1+h/2)
    {
    p++; q=-1;
    for(j1=0;j1<ImWidth-w/2;j1=j1+w/2)
        {
        q++;
        memset(im1,0,sizeof(int)*w*h);
        for(ih=0;ih<h;ih++)
            for(jw=0;jw<w;jw++)
                {
                im1[ih][jw]=ImageAdv1[i1+ih][j1+jw];
                br=im1[ih][jw];
                br1=(br<<8); //green
                ImageBitmap->Canvas->Pixels[j1+jw][i1+ih]=br1;

                im2[ih][jw]=ImageAdv2[i1+ih][j1+jw];
                br=im2[ih][jw];
                br1=(br<<8); //verde
                ImageBitmap2->Canvas->Pixels[j1+jw][i1+ih]=br1;
                }

        //Class definition (z) Maximum of texture in window h*w

        int ne=0; int ee=0; // (<255),no chaga;
            // (=255),est chaga

```

```

float npor=0;

for(ih=0;ih<h;ih++)
    for(jw=0;jw<w;jw++)
    {
        if(im2[ih][jw]==255)
            ee++;
        else
            ne++;
    }
    npor=0.1*ne; //from 0.1 (more samples) till 0.5 (less samples)

    if(ee>npor)
        z=1; //est chaga
    else z=0; //no chaga

    Canvas->Draw(5,5,ImageBitmap2);

    Form1->Canvas->TextOutA(100,420,"Imagen leída "+AnsiString(in+1)+" ");
    Form1->Canvas->TextOutA(200,420," Clase "+AnsiString(int (z))+" ");
    //ImageBitmap->LoadFromFile(NameOfImFile);
    // Stop;

ClassDefinitionArray[p][q]=z;

    BrightnessHistogramAdv(z);
    Form1->Canvas->TextOutA(850,410,"Histograma de Brillo ");
    BrightHistOut(850,400);

    ContrastHistogramAdv(z);
    Form1->Canvas->TextOutA(965,410,"Histograma de Contraste ");
    ContrHistOut(965,400);

    GradientOrientationHistogramAdv(z);
    Form1->Canvas->TextOutA(1115,410,"Histograma de Orientación de Microcontornos ");
    OrientHistOut(1115,400);

//Input vector formation for one window (3 histograms)
for(i=0;i<GF*SG;i++)
{
    if (i<SG)
        InputArray[i]=BrightnessHistogramArray[i];
    else if ((i>=SG)&&(i<(GF-1)*SG))
        InputArray[i]=ContrastHistogramArray[i-SG];
    else
        InputArray[i]=GradientOrientationHistogramArray[i-(GF-1)*SG];
}

//Active neurons for every window

```

```

memset(NeuronArray,0,sizeof(int)*NumOfNeurons);
memset(NonZeroNeuronsArray,0,sizeof(int)*NumOfNeurons);
k1=0;

for(i=0;i<NumOfNeurons;i++)
{
NeuronArray[i]=1;
NonZeroNeuronsArray[k1]=i+1;
k1++;

for(j=0;j<SizeOfSubspace;j++)
{
j=j;
if
((InputArray[MaskMatrix[i][j]]<=LimitArray[i][j][0])||(InputArray[MaskMatrix[i][j]]>=LimitArray[i][j][1]))
{
NeuronArray[i]=0;
k1--;
NonZeroNeuronsArray[k1]=0;
break;
}
}
}
NumOfNonZeroNeurons=k1;

NumbOfActN=NumbOfActN+k1;

if ((st1 = fopen("CodeVectors.dat", "ab")) == NULL)
{
fprintf(stderr, "Cannot open file CodeVectors\n");
}
//fseek(st1,sizeof(int)*(in*k*1+p*k+q),SEEK_SET);
fwrite(NonZeroNeuronsArray,sizeof(int),NumOfNeurons,st1);

fclose(st1); /* close file */

if ((st2 = fopen("NumberOfActiveNeurons.dat", "ab")) == NULL)
{
fprintf(stderr, "Cannot open file NumberOfActiveNeurons\n");
}
s[0]=NumbOfActN;
fwrite(s,sizeof(int),1,st2);
fclose(st2); /* close file */

for (i2=0;i2<k;i2++)
for (j2=0;j2<l;j2++)
{

```

```

//
        z=ClassDefinitionArray[i2][j2];

        if(z==1)
            {
                z=z;
            }
    }
//sleep(200);
}

}

if ((st2 = fopen("NumberOfActiveNeurons.dat", "rb")) == NULL)
    {
        fprintf(stderr, "Cannot open file NumberOfActiveNeurons\n");
    }
fseek(st2, in*k*1, SEEK_SET);
fread(NumberOfActiveNeuronsArray,sizeof(int)*k*1,1,st2);
fclose(st2);

if ((st4 = fopen("ClassDefinition.dat", "ab")) == NULL)
    {
        fprintf(stderr, "Cannot open file ClassDefinition\n");
    }
fseek(st4,(in)*k*1,SEEK_SET);
fwrite(ClassDefinitionArray,sizeof(char),k*1,st4);

fclose(st4); /* close file */
}
}
//-----

void OrientHistOut(int i0,int j0)
{
int i,j,k,l;
int GOH[SG];

memset(GOH,0,sizeof(int)*SG);
k=0;
Form1->Canvas->Brush->Color = (Graphics::TColor) (clWhite);
Form1->Canvas->FillRect(Rect(i0+50, 0, i0+50+SG*2, 400));

for(j=0;j<SG;j++)
    {
        l=GradientOrientationHistogramArray[j];
        if (l>k)k=l;
    }

for(j=0;j<SG;j++)

```

```

    {
        GOH[j]=(GradientOrientationHistogramArray[j]*NormRate)/k;
    }
Form1->Canvas->Pen->Color = (Graphics::TColor) (clNone);
for(j=0;j<SG;j++)
    {
        //k=GOH[j];
        Form1->Canvas->MoveTo(i0+j*2+50,400-j0);
        Form1->Canvas->LineTo(i0+j*2+50,400-GOH[j]);
        GradientOrientationHistogramArray[j]=GOH[j];
    }

Form1->Canvas->MoveTo(i0+(-1*2)+50,400-j0);
Form1->Canvas->LineTo(i0+(-1*2)+50,400);
Form1->Canvas->MoveTo(i0+(16*2)+50,400-j0);
Form1->Canvas->LineTo(i0+(16*2)+50,400);
    //Sleep(500);
}

```

//-----

```

void BrightHistOut(int i0,int j0)
{
int i,j,k,l;
int BH[SG];

memset(BH,0,sizeof(int)*SG);
Form1->Canvas->Brush->Color = (Graphics::TColor) (clWhite);
Form1->Canvas->FillRect(Rect(i0+10, 0, i0+10+SG*2, 400));
k=0;
for(j=0;j<SG;j++)
    {
        l=BrightnessHistogramArray[j];
        if (l>k)k=l;
    }

for(j=0;j<SG;j++)
    {
        BH[j]=(BrightnessHistogramArray[j]*NormRate)/k;
    }
Form1->Canvas->Pen->Color = (Graphics::TColor) (clNone);
for(j=0;j<SG;j++)
    {
        //k=BH[j];
        Form1->Canvas->MoveTo(i0+j*2+10,400-j0);
        Form1->Canvas->LineTo(i0+j*2+10,400-BH[j]);
        BrightnessHistogramArray[j]=BH[j];
    }

Form1->Canvas->MoveTo(i0+(-1*2)+10,400-j0);

```

```

Form1->Canvas->LineTo(i0+(-1*2)+10,400);
Form1->Canvas->MoveTo(i0+(16*2)+10,400-j0);
Form1->Canvas->LineTo(i0+(16*2)+10,400);
    //Sleep(500);
}

//-----
void ContrHistOut(int i0,int j0)
{
    int i,j,k,l;
    int CH[SG];

    memset(CH,0,sizeof(int)*SG);
    Form1->Canvas->Brush->Color = (Graphics::TColor) (clWhite);
    Form1->Canvas->FillRect(Rect(i0+50, 0, i0+50+SG*2, 400));
    k=0;
    for(j=0;j<SG;j++)
        {
            l=ContrastHistogramArray[j];
            if (l>k)k=l;
        }

    for(j=0;j<SG;j++)
        {
            CH[j]=(ContrastHistogramArray[j]*NormRate)/k;
        }

    for(j=0;j<SG;j++)
        {
            // k=CH[j];
            Form1->Canvas->MoveTo(i0+j*2+50,400-j0);
            Form1->Canvas->Pen->Color = (Graphics::TColor) (clNone);
            Form1->Canvas->LineTo(i0+j*2+50,400-CH[j]);
            ContrastHistogramArray[j]= CH[j];
        }

    Form1->Canvas->MoveTo(i0+(-1*2)+50,400-j0);
    Form1->Canvas->LineTo(i0+(-1*2)+50,400);
    Form1->Canvas->MoveTo(i0+(16*2)+50,400-j0);
    Form1->Canvas->LineTo(i0+(16*2)+50,400);
    //Sleep(500);
}

//-----
void __fastcall TForm1::Training1Click(TObject *Sender)
{
    Training();
}

```

```

//-----
void Training(void)
{
int i,i1,i2,j2,ic,ip,ip1,j,NT;
int s[1],sant[1];
int ind,ind1,ind2,ind3;

int n,m,br,br1,i0,j0,p,q,k,l;
char NameOfImFile[50],z;
int MaxExcit=0;
int SwapElem,ImMean;

    k=2*ImHeight/h;
    l=2*ImWidth/w;

memset(NonZeroNeuronsArray,0,sizeof(int)*NumOfNeurons);
memset(NumberOfActiveNeuronsArray,0,sizeof(int)*NI*k*1);

/*if ((st3 = fopen("Matrix.dat", "rb")) == NULL)
    {
        fprintf(stderr, "Cannot open file Matrix\n");
    }
    fread(WeightMatrix,sizeof(int)*NumOfNeurons*NumOfClasses,1,st3);

    fclose(st3); // close file
*/

NT=NI-NumOfRecIm;

if ((st2 = fopen("NumberOfActiveNeurons.dat", "rb")) == NULL)
    {
        fprintf(stderr, "Cannot open file NumberOfActiveNeurons\n");
    }
    fseek(st2,NI*k*1,SEEK_SET);
    fread(NumberOfActiveNeuronsArray,sizeof(int)*NI*k*1,1,st2);
fclose(st2);

for(ic=0;ic<NumOfTrainCycles;ic++)
{
    NumOfErrPerCycle=0;

    Form1->Canvas->TextOutA(550,200+13*ic,"Numero de Ciclo "+AnsiString(ic+1));

    // for(z=0;z<NumOfClasses;z++)
    //

for(i1=0;i1<NT;i1++)
{

```

```

if ((st4 = fopen("ImageForTraining.dat", "rb")) == NULL)
{
    fprintf(stderr, "Cannot open file ImageForTraining\n");
}
fread(TrainImArray, sizeof(int), NT, st4);

fclose(st4);

ip=TrainImArray[i1];

Form1->Canvas->TextOutA(550,0,"Imagen "+AnsiString(ip)+" ");

if ((st4 = fopen("ClassDefinition.dat", "rb")) == NULL)
{
    fprintf(stderr, "Cannot open file ClassDefinition\n");
}

fseek(st4, ip*k*1, SEEK_SET);
fread(ClassDefinitionArray, sizeof(char), k*1, st4);

fclose(st4); /* close file */

//

for (i2=0; i2<k; i2++)
    for (j2=0; j2<l; j2++)
    {
        z=ClassDefinitionArray[i2][j2];

Form1->Canvas->TextOutA(550,20,"Clase "+AnsiString(int(z)));
Form1->Canvas->TextOutA(550,40," Fragmento "+AnsiString(i2*j2)+" ");

        memset(NonZeroNeuronsArray, 0, sizeof(int)*NumOfNeurons);

        ind=NumberOfActiveNeuronsArray[ip*k*1+j2*k+i2];
        ind1=NumberOfActiveNeuronsArray[ip*k*1+j2*k+i2+1];

        NumbOfActN=ind1-ind;

if ((st1 = fopen("CodeVectors.dat", "rb")) == NULL)
{
    fprintf(stderr, "Cannot open file CodeVectors\n");
}

fseek(st1, sizeof(int)*(NumOfNeurons*(ip*k*1+i2+j2*k)), SEEK_SET);
fread(NonZeroNeuronsArray, sizeof(int), NumOfNeurons, st1);

fclose(st1); /* close file */

TrueClassNumber=z;

```

```

if (TrueClassNumber==1)
    z=z;

memset(ClassExcitArray,0,sizeof(int)*NumOfClasses);

RecognitionForTraining();

/*for(m=0;m<NumOfNeurons;m++)
{
if (NonZeroNeuronsArray[m]==0)
    break;
WeightMatrix[NonZeroNeuronsArray[m]][TrueClassNumber]=1;

ClassExcitArray[TrueClassNumber]=ClassExcitArray[TrueClassNumber]+WeightMatrix[NonZeroNeuronsArray[m]][TrueClassNumber];
}
ClassExcitArray[TrueClassNumber]=ClassExcitArray[TrueClassNumber]*(1-TDS);

MaxExcit=0;
RecClassNumber=0;

for(n=0;n<NumOfClasses;n++)
{
if (ClassExcitArray[n]>MaxExcit)
{
MaxExcit=ClassExcitArray[n];
RecClassNumber=n;
}
}
*/
if (RecClassNumber!=TrueClassNumber)
{
NumOfErrPerCycle++;
WeightMatrixUpdate();
}

} //

} //

Form1->Canvas->TextOutA(550,60+13*ic,"Numero de Errores Por Ciclo
"+AnsiString(NumOfErrPerCycle)+" ");

} //
if ((st3 = fopen("Matrix.dat", "wb")) == NULL)
{
fprintf(stderr, "Cannot open file Matrix\n");
}
fwrite(WeightMatrix,sizeof(int)*NumOfNeurons*NumOfClasses,1,st3);

fclose(st3); /* close file */

```

```

}
//}
//}

//-----
void RecognitionForTraining(void)
{
    int i,j,k,l,m,z,max;

    max=0;
    for(i=0;i<NumOfClasses;i++)
        ClassExcitArray[i]=0;

    for(i=0;i<NumOfClasses;i++)
    {
        for(m=0;m<NumOfNeurons;m++)
        {
            if (NonZeroNeuronsArray[m]==0)
                break;

            //WeightMatrix[NonZeroNeuronsArray[m]][TrueClassNumber]=1;

            ClassExcitArray[i]=ClassExcitArray[i]+WeightMatrix[NonZeroNeuronsArray[m]][i];
        }
        ClassExcitArray[i]=ClassExcitArray[i]*(1-TDS);
    }

    max=0;
    RecClassNumber=0;
    for(i=0;i<NumOfClasses;i++)
    {
        if(ClassExcitArray[i]>max)
        {
            max=ClassExcitArray[i];
            RecClassNumber=i;
        }
    }
}

//-----
void __fastcall TForm1::Recognition1Click(TObject *Sender)
{
    Recognition();
}

```

```

//-----
void Recognition(void)
{
int i,i1,i2,ip,j,j2,br,br1,i0,j0,n,m,k,l,z;
char NameOfImFile[50];
int MaxExcit=0;
int ind,ind1,ind2;

NumOfErr=0;

    k=2*ImHeight/h;
    l=2*ImWidth/w;

memset(NonZeroNeuronsArray,0,sizeof(int)*NumOfNeurons);
memset(NumberOfActiveNeuronsArray,0,sizeof(int)*NI*k*1);

if ((st3 = fopen("Matrix.dat", "rb")) == NULL)
    {
        fprintf(stderr, "Cannot open file Matrix\n");
    }
    fread(WeightMatrix,sizeof(int)*NumOfNeurons*NumOfClasses,1,st3);

    fclose(st3); /* close file */

    memset(NonZeroNeuronsArray,0,sizeof(int)*NumOfNeurons);

for(i1=0;i1<NumOfRecIm;i1++)
    {
        if ((st4 = fopen("ImageForRecognition.dat", "rb")) == NULL)
            {
                fprintf(stderr, "Cannot open file ImageForRecognition\n");
            }
            fread(RecImArray,sizeof(int),NumOfRecIm,st4);

            fclose(st4);

            ip=RecImArray[i1];
            //k=2*ImHeight/h;
            //l=2*ImWidth/w;

            Form1->Canvas->TextOutA(10,650,"Imagen "+AnsiString(ip)+" ");
            //ip=0;
            if ((st2 = fopen("NumberOfActiveNeurons.dat", "rb")) == NULL)
                {
                    fprintf(stderr, "Cannot open file NumberOfActiveNeurons\n");
                }
                fseek(st2, ip*k*1, SEEK_SET);
                fread(NumberOfActiveNeuronsArray,sizeof(int)*k*1,1,st2);
                fclose(st2);
    }
}

```

```

if ((st4 = fopen("ClassDefinition.dat", "rb")) == NULL)
{
    fprintf(stderr, "Cannot open file ClassDefinition\n");
}

fseek(st4, ip*k*1, SEEK_SET);
fread(ClassDefinitionArray, sizeof(char), k*1, st4);

fclose(st4); /* close file */

for (i2=0; i2<k; i2++)
for (j2=0; j2<l; j2++)
{
    z=ClassDefinitionArray[i2][j2];
    TrueClassNumber=z;

    if(z==1)
    z=z;

    ind=NumberOfActiveNeuronsArray[ip*k*1+j2*k+i2];
    ind1=NumberOfActiveNeuronsArray[ip*k*1+j2*k+i2+1];

    NumbOfActN=ind1-ind;

    if ((st1 = fopen("CodeVectors.dat", "rb")) == NULL)
    {
        fprintf(stderr, "Cannot open file CodeVectors\n");
    }

    fseek(st1, sizeof(int)*((ip*k*1+j2*k+i2)*NumOfNeurons), SEEK_SET);
    fread(NonZeroNeuronsArray, sizeof(int), NumOfNeurons, st1);

    fclose(st1); /* close file */

    memset(ClassExcitArray, 0, sizeof(int)*NumOfClasses);

    for(n=0; n<NumOfClasses; n++)
    for(m=0; m<NumOfNeurons; m++)
    {
        if (NonZeroNeuronsArray[m]==0)
            break;
        ClassExcitArray[n]=ClassExcitArray[n]+WeightMatrix[NonZeroNeuronsArray[m]][n];
    }

    MaxExcit=0;
    RecClassNumber=0;

    for(n=0; n<NumOfClasses; n++)
    {

```

```

        if (ClassExcitArray[n]>MaxExcit)
            {
                MaxExcit=ClassExcitArray[n];
                RecClassNumber=n;
            }
    }

// memset(ClassExcitArray,0,sizeof(int)*NumOfClasses);

z=ClassDefinitionArray[i2][j2];
    TrueClassNumber=z;

    if (RecClassNumber!=TrueClassNumber)
        {
            NumOfErr++;
        }
    if (RecClassNumber==1)
        z=RecClassNumber;

        Form1->Canvas->TextOutA(100,650,"Numero de clase de reconocimiento
"+AnsiString(RecClassNumber)+" ");
        Form1->Canvas->TextOutA(480,650,"Numero de clase verdadera "+AnsiString(TrueClassNumber)+"
");
        Form1->Canvas->TextOutA(800,650,"Numero de Errores de Reconocimiento
"+AnsiString(NumOfErr));
    } //cycle i2,j2
} //cycle i1

// }
}

//-----

void BrightnessHistogramAdv(int z)
{
int i,j,ih,jw;
int MaxElem;
int ImageUpd[h][w][NumOfClasses];

memset(BrightnessHistogramArray,0,sizeof(int)*SG);
/*for(j=0;j<SG;j++)
    BrightnessHistogramArray[j]=0;
*/
for(i=0;i<NumOfClasses;i++)
    {
        memset(ImageUpd[i],0,sizeof(int)*w*h);
    }

    MaxElem=0;
    for(ih=0;ih<h;ih++)

```

```

    for(jw=0;jw<w;jw++)
    {
        if(im1[ih][jw]>MaxElem) MaxElem=im1[ih][jw];
    }

for(ih=0;ih<h;ih++)
for(jw=0;jw<w;jw++)
    {
        ImageUpd[ih][jw][z]=(int)((im1[ih][jw]*c)/(MaxElem));
        BrightnessHistogramArray[ImageUpd[ih][jw][z]]++;
    }
}
//-----

void ContrastHistogramAdv(int z)
{
    int i,j,ih,jw;
    int d,d1,d2;
    int MaxElem;

    memset(ContrastHistogramArray,0,sizeof(int)*SG);
    /*for(j=0;j<SG;j++)
        ContrastHistogramArray[j]=0;
    */

    MaxElem=0;
    for(ih=0;ih<h;ih++)
    for(jw=0;jw<w;jw++)
        {
            if(im1[ih][jw]>MaxElem) MaxElem=im1[ih][jw];
        }

    for(ih=0;ih<(h-1);ih++)
    for(jw=0;jw<(w-1);jw++)
        {
            d1=(int)(((im1[ih][jw]-im1[ih+1][jw+1])*c*ContrastScale)/MaxElem);
            d2=(int)(((im1[ih+1][jw]-im1[ih][jw+1])*c*ContrastScale)/MaxElem);

            if(d1<0) d1=-d1;
            if(d2<0) d2=-d2;

            if(d1>d2)
                d=d1;
            else
                d=d2;
            if (d>15) d=15;
            ContrastHistogramArray[d]++;
        }
}
//-----

```

```

void GradientOrientationHistogramAdv(int z)
{
int ih,jw,k;
int d1,d2,y1,y2,y,absd1,absd2,dmin,dmax,ind;
float ang;
int MaxElem;

memset(GradientOrientationHistogramArray,0,sizeof(int)*SG);
dmin=0;
dmax=0;
ind=0;
MaxElem=0;
k=3; //d1>x srednee*2/^(k)

for(ih=0;ih<h;ih++)
for(jw=0;jw<w;jw++)
{
if(im1[ih][jw]>MaxElem) MaxElem=im1[ih][jw];
}

for(ih=0;ih<h-1;ih++)
for(jw=0;jw<w-1;jw++)
{
y1=(im1[ih][jw]+im1[ih+1][jw+1]);
y2=(im1[ih][jw+1]+im1[ih+1][jw]);

if ((y1-y2)<ContourExistConst)
{
y=(y1+y2)/4;
d1=(im1[ih][jw]-im1[ih+1][jw+1]);
d2=(im1[ih][jw+1]-im1[ih+1][jw]);
if(d1>0)
absd1=d1;
else
absd1=-d1;
if(d2>0)
absd2=d2;
else
absd2=-d2;

if ((absd1-absd2)>0)
{ dmax=absd1;dmin=absd2;}
else
{ dmax=absd2;dmin=absd1;}

if (dmax>y*2^(-k))
ang=float (dmin)/float(dmax);
else ang=0.0;
if(d1>0&&d2<0&&(absd1-absd2)>0)
ang=M_PI_4-ang;
if(d1>0&&d2>0&&(absd1-absd2)>0)

```

```

ang=M_PI_4+ang;
if(d1>0&&d2>0&&(absd1-absd2)<0)
ang=(M_PI_4*3)-ang;
if(d1<0&&d2>0&&(absd1-absd2)<0)
ang=(M_PI_4*3)+ang;
if(d1<0&&d2>0&&(absd1-absd2)>0)
ang=M_PI_4*5-ang;
if(d1<0&&d2<0&&(absd1-absd2)>0)
ang=M_PI_4*5+ang;
if(d1<0&&d2<0&&(absd1-absd2)<0)
ang=(M_PI_4*7)-ang;
if(d1>0&&d2<0&&(absd1-absd2)<0)
ang=(M_PI_4*7)+ang;

ind=int ((ang*c)/(2*M_PI));
GradientOrientationHistogramArray[ind]++;
}
}
}

//-----

void CreatingOfImArrays(void)
{
int i,j,n;
int k=0;
int RanNum;

memset(RecImArray,0,sizeof(int)*NumOfRecIm);

for(i=0;i<NumOfRecIm;i++)
{
RanNum=(random(NI)+1);

RecImArray[i]=RanNum;

for(j=0;j<i;j++)
{
if (RecImArray[j]==RanNum)
{
i--;
break;
}
}
}

for(j=0;j<(NumOfRecIm);j++) //
for(i=0;i<(NumOfRecIm-1);i++)
{
if (RecImArray[i]>RecImArray[i+1])

```

```

        {
        n=RecImArray[i];
        RecImArray[i]=RecImArray[i+1];
        RecImArray[i+1]=n;
        }
    }

for(j=1;j<=NI;j++)
{
    if (k==NI-NumOfRecIm) break;
    TrainImArray[k]=j;
    k++;
    for(i=0;i<NumOfRecIm;i++)
    {
        if (j==RecImArray[i])
        {
            k--;
            TrainImArray[k]=0;
            break;
        }
    }
}

for (i=0;i<NI-NumOfRecIm;i++)
Form1->Canvas->TextOutA(5,290+20*i,"Imagen " + AnsiString(TrainImArray[i]));

if ((st4 = fopen("ImageForTraining.dat", "wb")) == NULL)
{
    fprintf(stderr, "Cannot open file ImageForTraining\n");
}
fclose(st4);

if ((st4 = fopen("ImageForTraining.dat", "ab")) == NULL)
{
    fprintf(stderr, "Cannot open file ImageForTraining\n");
}
fwrite(TrainImArray,sizeof(int),(NI-NumOfRecIm),st4);

fclose(st4);

if ((st4 = fopen("ImageForRecognition.dat", "wb")) == NULL)
{
    fprintf(stderr, "Cannot open file ImageForRecognition\n");
}
fclose(st4);

if ((st4 = fopen("ImageForRecognition.dat", "ab")) == NULL)
{
    fprintf(stderr, "Cannot open file ImageForRecognition\n");
}
}

```

```

        fwrite(RecImArray,sizeof(int),NumOfRecIm,st4);

        fclose(st4);
    }

//-----

void WeightMatrixUpdate(void)
{
int i;
    for(i=0;i<NumOfNeurons;i++)
        {
            if (NonZeroNeuronsArray[i]==0)
                break;

WeightMatrix[NonZeroNeuronsArray[i]][RecClassNumber]=WeightMatrix[NonZeroNeuronsArray[i]][RecClassNumber]-1;

WeightMatrix[NonZeroNeuronsArray[i]][TrueClassNumber]=WeightMatrix[NonZeroNeuronsArray[i]][TrueClassNumber]+1;
        }
/*    for(i=0;i<NumOfNeurons;i++)
        {
            if (WeightMatrix[NonZeroNeuronsArray[i]][RecClassNumber]<0)
                WeightMatrix[NonZeroNeuronsArray[i]][RecClassNumber]=0;
            else if (WeightMatrix[NonZeroNeuronsArray[i]][TrueClassNumber]<0)
                WeightMatrix[NonZeroNeuronsArray[i]][TrueClassNumber]=0;
        }
*/
}

//-----

//-----

void __fastcall TForm1::MaskGeneration1Click(TObject *Sender)
{
int i,j;
int x0,r0,tOn,tOff;
//mask generation, weight matrix to "0",train and test image set formation
for(i=0;i<NumOfNeurons;i++)
for(j=0;j<NumOfClasses;j++)
    {
        WeightMatrix[i][j]=0;
    }

for(i=0;i<NumOfNeurons;i++)
for(j=0;j<SizeOfSubspace;j++)
    {

```

```

    MaskMatrix[i][j]=random(GF*SG);
    }

for(i=0;i<NumOfNeurons;i++)
{
    memset(LimitArray[i],0,sizeof(int)*SizeOfSubspace);
}

for(i=0;i<NumOfNeurons;i++)
for(j=0;j<SizeOfSubspace;j++)
{
    x0=random(NormRate+1);
    r0=random(NormRate+1);

    tOn=x0-GAP*r0;
    tOff=x0+GAP*r0;

    if(tOn<0)tOn=-tOn;
    if (tOff>NormRate)tOff=tOff-NormRate;
    if(tOn<tOff)
    {
        LimitArray[i][j][0]=tOn;
        LimitArray[i][j][1]=tOff;
    }
    else
    {
        LimitArray[i][j][0]=tOff;
        LimitArray[i][j][1]=tOn;
    }
}

Form1->Canvas->TextOutA(5,210,"Las Mascaras han sido leidas");

CreatingOfImArrays();

Form1->Canvas->TextOutA(5,230,"Sets de: Entrenamiento/Test, han sido leidos");
Form1->Canvas->TextOutA(5,250,"Set de entrenamiento");
}
//-----

```

Bibliografía y referencias

- [1] Dumonteil E., Update on Chagas's disease in Mexico, *Salud Pública de México*, vol. 41, No.4, 1999, pp. 322–327.
- [2] CHAGMEX[®] (Enfermedad de Chagas en México), Instituto de Biología UNAM, 2004, <http://unibio.unam.mx/chagmex/>
- [3] <http://www2.biomédicas.unam.mx/FrameGaceta2.htm>
- [4] Zavala Patricia. El mal de Chagas, enfermedad de la pobreza, *Gaceta UNAM*, 18 de noviembre de 2010, Número 4 (293), ISSN 0188–5138
- [5] Fundación ONCE, 2009, España,
<http://salud.discapnet.es/Castellano/Salud/Enfermedades/EnfermedadesEndemicas/Paginas/Chagas.aspx#a2>
- [6] Academia Nacional de Medicina de México A.C. (Paz María Salazar–Schettino), Simposio de enfermedad de Chagas, *Gaceta médica de México*, septiembre–octubre 2003, vol.139, suplemento 3, pp. 5–7, www.medigraphic.com/pdfs/gaceta/gm-2003/gms033.h.pdf
- [7] Hoffmann CC. Nota acerca de un probable transmisor de la tripanosomiasis humana en el estado de Veracruz. *Rev Mex Biol* 1928; 8:12–8.
- [8] Salazar–Schettino PM, Tay J, Ontiveros A, Jiménez J, Haro I de, Bucio MI, Ruíz Hernández AL. Enfermedad de Chagas en México. Presentación de casos clínicos, *Rev Fac Med (UNAM)* 1983; 25:77–82.
- [9] Salazar–Schettino PM, Haro I de, Sánchez Vega JT, Canepa–Chambochumi D y cols. Reporte de tres casos de enfermedad de Chagas en fase crónica. *Rev Patol Clin (Méx)* 1986; 33: 55–63
- [10] Salazar–Schettino PM y cols. Taller Internacional de Epidemiología, Diagnóstico y Control de la enfermedad de Chagas. 7–9 nov 2002 Oaxaca, Oax (En prensa) OPS/OMS. México.

- [11] Becerra IMA, Manrique MA, Padrón AG, Ortiz GJR. Megaesófago en la enfermedad de Chagas. *Rev Hosp Juárez Méx* 2001; 68(1):31–36
- [12] Cordero PLJ, Zárate CR, Ramos CMA, Cordero CJ. Miocardiopatía dilatada chagásica en el estado de Chiapas, México. *Rev Mex Cardiol* 2002 ;(4):153–57
- [13] Dumonteil E., Update on Chagas' disease in México. *Salud Pública, Méx*, 1999; 41:322–327.
- [14] Goldsmith RS, Zárate R, Kagan I, Cedeño–Ferreira J, Galindo–Vasconcelos M, Antonio E. El potencial de la transmisión en la enfermedad de Chagas por transfusión sanguínea: Hallazgos serológicos entre donadores en el estado de Oaxaca. *Sal Pub Méx* 1978; 20:439–444.
- [15] Monteón MV, Linares TC, Amador GFR, Ruessegger GL, Reyes PA. Anticuerpos séricos a *Trypanosoma cruzi* en donadores de sangre en la ciudad de México. *Bioquimia* 1987; 9:6–9.
- [16] Ramos–Echevarría A, Monteón–Padilla V, Reyes–López P. Detección de anticuerpos contra *Trypanosoma cruzi* en donadores de sangre. *Sal Pub Méx* 1993; 35:56–64.
- [17] Trujillo CF, Lozano KF, Soto MM, Hernández GR. The prevalence of *Trypanosoma cruzi* infection in blood donors in the state of Jalisco, Mexico. *Rev Soc Bras Med Trop* 1993; 26:89–92.
- [18] Rodríguez–Félix ME, Zavala–Velázquez J, Barrera–Pérez MA, Guzmán–Marín E, Ramírez–Sierra MJ, Álvarez Miguel R., Riesgo de transmisión de la enfermedad de Chagas por donantes de sangre. *Rev Biomed* 1995; 6:70–75.
- [19] Rangel H, Gatica R, Ramos C. Detection of Antibodies against *Trypanosoma cruzi* in Donors form a Blood Bank in Cuernavaca, Morelos, Mexico. *Archives of Medical Research* 1998; 29:79–82.
- [20] Guzmán C, García L, Floriani J, Guerrero S, Torres M, Ramírez C, Velasco O. Riesgo de transmisión de *Trypanosoma cruzi* por transfusión de sangre en México. *Rev Panam Salud Publica/Pan Am J Public Health* 1998; 4:94–98.
- [21] Monteón–Padilla V, Hernández–Becerril N, Guzmán–Bracho C, Rosales–Encina JI, Reyes–López PA. American Tripanosomiasis (Chagas 'disease) and Blood Banking in Mexico City: Seroprevalence and Its Potential Transfusional Transmission Risk. *Archives of Medical Research* 1999; 30:393–398.

- [22] Schofield C.J. Challenges of Chagas Disease Vector Control in Central America. WHO/WHOPEPES 2000; pp.35.
- [23] González R.C., Woods R.E. Digital Image Processing. 2nd edition, Prentice Hall, New Jersey, 2002.
- [24] Rosenfeld A. Picture Processing by Computer. ACM Computing Surveys, 1(3): 147–76, 1969.
- [25] Isoardi Roberto A., Tesis doctoral: Optimización de análisis y registración de imágenes topográficas, abril 2010, Argentina, pp.2 – 6.
- [26] Bankman I.N. Segmentation. En: Handbook of Medical Imaging: Processing and Analysis (editor: Bankman I.N.). Academic Press – 2nd edition, 2008.
- [27] Isoardi Roberto A., Tesis doctoral: Optimización de análisis y registración de imágenes tomográficas, Universidad Nacional de Cuyo, Argentina, pp.15-16, abril, 2010.
- [28] Hertz J.A., Krogh A.S., Palmer R.G. Introduction to the theory of neural computation. Westview Press, Redview City, CA, USA, 1991.
- [29] Dawant B.M., Zijdenbos A.P. Image Segmentation. En: Handbook of Medical Imaging Vol.2, SPIE Press, 2000.
- [30] Hendee, W. X rays in medicine. Physics Today. Noviembre, pp. 51–56 . 1995.
- [31] Gibson, A., Cook, E. y Newing, A. Teaching Medical Physics. Physics Education. 41(4): 301–306, 2006.
- [31] Caicedo Rueda Juan Carlos. Tesis: Recuperación de Imágenes Médicas por Contenido: arquitectura, técnicas y aproximaciones. Universidad Nacional de Colombia, pp.3, 5.
- [32] S. S. A. Pentland, W. Picard, “Photobook: Content–based manipulation of image databases, MIT, 1995.
- [33] C. W. Niblack and R. Barber, “Qbic project: querying images by content, using color, texture, and shape,” Medical Imaging: Storage and Retrieval for Image and Video Databases, vol. 1908, 1993.

- [34] M. T. Remco C. Veltkamp, Content-based image retrieval systems: A survey,” This is a revised and extended version of Technical Report UUC-2000-34, October 2000, 2002.
- [35] J. G. A. Mojsilovic and B. Rogowitz, Isee: Perceptual features for image library navigation, SPIE Human Vision and Electronic Imaging, 2002.
- [36] Y. Song, Sceneryanalyzer: A system supporting semantics-based image retrieval, Intelligent Multimedia Documents, 2002.
- [37] G. T. Andrea F. Abate, Michele Nappi and M. Tucci, Ime: an image management environment with content-based access, Image and Vision Computing, vol. 17, pp. 967-980, 1999.
- [38] H. Müller, “A review of content based image retrieval systems in medical applications clinical benefits and future directions,” International Journal of Medical Informatics, vol. 73, pp. 1-23, 2004.
- [39] Coto Ernesto. Métodos de Segmentación de Imágenes Médicas, Universidad Central de Venezuela. Facultad de Ciencias. Escuela de Computación. Laboratorio de Computación Gráfica (LCG), pp. 9-13, marzo 2003.
- [40] Indexing hierarchical structures using graph spectra, IEEE Trans Pattern Anal Mach Intell, vol. 27, pp. 1125-1140, 2005.
- [41] B. Johansson, A survey on: Contents based search in image databases, 2000.
- [42] P. Howarth and D. Heesch, Medical image retrieval using texture, locality and colour, 2004.
- [43] M. Carlin, Measuring the performance of shape similarity retrieval methods, Computer Vision and Image Understanding, vol. 84, 2001.
- [44] G. Heidemann, Unsupervised image categorization, Image and Vision Computing, vol. 23, 2005.
- [45] A. J. M., Content-based image retrieval using approximate shape of objects, 17th IEEE Symposium on Computer-Based Medical Systems (CBMS'04), 2004.
- [46] Recuperación de Imágenes Médicas por Contenido: arquitectura, técnicas y aproximaciones. Grupo de Investigación Bioingenium, Universidad Nacional de Colombia.

- [47] Sahoo, P.K.; Soltani, S.; Wong, A.K.C. A survey of thresholding techniques. *Computer Vision, Graphics and Image Processing*: vol. 41, pp. 233–260. 1988.
- [48] Besag, J. On the statistical analysis of dirty pictures. *Journal Royal Statistical Society B*, vol. 48, num. 3, pp. 259–302. 1986.
- [49] Geman, S.; Geman, D. Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*: vol. 6, pp. 721–741. 1984.
- [50] Coto Ernesto. Métodos de Segmentación de Imágenes Médicas, Universidad Central de Venezuela. Facultad de Ciencias. Escuela de Computación. Laboratorio de Computación Gráfica (LCG), pp. 6–13, marzo 2003.
- [51] Xu, Chenyang ; Pham, Dzung L ; Prince, Jerry L. Current Methods in Medical Image Segmentation. *Annual Review of Biomedical Engineering, Annual Reviews*, vol. 2, pp. 315-337, Annual Reviews, Palo Alto, CA, 2000.
- [52] Zhang, Y.J. A survey on evaluation methods for image segmentation. *Pattern Recognition*: vol. 29, num. 8, pp. 1335–1346. 1996.
- [53] Rueckert, D.; Burger, P. Contour fitting using an adaptive spline model. *British Machine Vision Conference 1995*, vol. 1, pp. 207–216. Birmingham, UK. Septiembre, 1995.
- [54] Maren Alianna, Craig Harston, Robert Pap. *Handbook of neural computing applications*, USA, Academic Press, 1990.
- [55] Rodríguez Flores Marco Antonio. Tesis: Desarrollo de un sistema basado en un clasificador neuronal para reconocimiento de orugas, México, 2010, pp.91
- [56] Hilera, J. R., Víctor J. Martínez. *Redes Neuronales artificiales. Fundamentos, modelos y aplicaciones*, Wilmington, Delaware, Addison–Wesley, 1995.
- [57] Steinbuch, K., Die Lematrix, *Kibernetik 1*, 1961, pp. 36–45
- [58] Steinbuch, K., U. Piske, Learning matrices and their application, *IEEE Transaction on Electronic Computers*, EC–12, 1963, pp.846–862

- [59] Grossberg, S. The theory of embedding fields with applications to psychology and neuropsychology, Rockefeller Institute of Medical Research, New York, 1964.
- [60] Widrow, B., M. E. Hoff. Adaptive Signal Processing, Prentice–Hall, Englewood Cliffs N. J., 1985.
- [61] T. Baydyk, E. Kussul, Redes neuronales, visión computacional, y micromecánica, editorial ITACA–UNAM, pp. 158.
- [62] González González Carolina, T. Baydyk, E. Kussul, Paper Recognition of Textures in Medical Images, Centro de Ciencias Aplicadas y Desarrollo Tecnológico, Universidad Nacional Autónoma de México, México, D.F., 2010, pp.8.