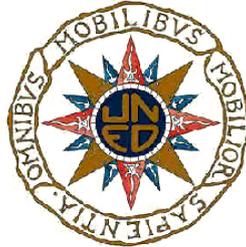


UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

Escuela Técnica Superior de Ingeniería Informática



**SEGMENTACIÓN AUTOMÁTICA DE LA
RED DE VASOS SANGUÍNEOS EN IMÁGENES DE RETINA
MEDIANTE REDES NEURONALES ARTIFICIALES EVOLUTIVAS
Y OPERADORES BASADOS EN *LOCAL BINARY PATTERNS* (LBP)**

TRABAJO FIN DE MÁSTER EN INTELIGENCIA ARTIFICIAL AVANZADA:
SISTEMAS INTELIGENTES DE DIAGNÓSTICO, PLANIFICACIÓN Y CONTROL

Antonio Rodríguez Jiménez

Director: Enrique J. Carmona Suárez

Septiembre 2013

Resumen

En este trabajo de Fin de Máster se presenta un nuevo método para la segmentación de los vasos sanguíneos de la retina en fotografías de fondo de ojo, basado en la utilización de Redes Neuronales Artificiales (RNA) y en el uso del operador LBP (*Local Binary Pattern*), un operador que, tradicionalmente, ha sido muy utilizado en el análisis de imágenes para la detección de texturas. En realidad, para ser más precisos, se ha evaluado no sólo el operador LBP clásico sino, además, un conjunto de operadores que son variaciones del original. Algunas de estas variaciones corresponden a operadores ya existentes en la literatura y otras han sido aquí propuestas. Aplicando este conjunto de operadores a las fotografías de fondo de ojo se obtiene, para cada pixel de las mismas, un conjunto de valores que conforman un vector de características. A partir de vectores de este tipo, se construirá un conjunto de datos para entrenar una RNA. Sin embargo, no se utilizará un método clásico de entrenamiento de RNAs sino que el aprendizaje de la RNA óptima se hará a partir de un algoritmo evolutivo basado en *Grammatical Evolution*. Mediante este paradigma evolutivo, no sólo se aprenderá la RNA más adecuada para clasificar cada pixel de la imagen como perteneciente o no a un vaso sanguíneo, sino que se seleccionará además, de forma automática, la topología de la red y las características de entrada más adecuadas para ello, simplificando así la tarea de determinar cuáles son las variantes de LBP más discriminantes que han de ser utilizadas como entrada a la red. La evaluación del método se ha llevado a cabo con dos de las bases de datos de imágenes digitales de retina más utilizadas al abordar la solución de este tipo de problema: DRIVE (*Digital Retinal Images for Vessel Extraction*), y STARE (*Structured Analysis of the Retina*).

El método obtiene unos resultados competitivos respecto a otros métodos existentes en la literatura, tanto desde el punto de vista de precisión y sensibilidad como desde el punto de vista del coste computacional.

Palabras Clave: Segmentación de vasos *sanguíneos*, Imágenes de Retina, Redes Neuronales Artificiales, *Local Binary Patterns*, *Grammatical Evolution*.

Abstract

In this Master's Thesis a new method is presented for automated segmentation of retinal blood vessels in fundus photographs based on the use of Artificial Neural Networks (ANN) and in the use of LBP operator (Local Binary Patterns), an operator that has traditionally been used in image analysis for the detection of textures. Actually, to be precise, it has been evaluated not only the classic LBP operator but also a set of operators which are variations of the original. Some of these variations correspond to existing operators in the literature and others have been proposed here. Applying this set of operators to the fundus photographs is obtained, for each pixel of the same, a set of values that form a feature vector. From such vectors, will be build a data set to train an ANN. However, don't will be used a classical method for training ANN, but learning the optimal ANN will be made from an evolutionary algorithm based on Grammatical Evolution. Through this evolutionary paradigm, will be not only learned the most appropriate ANN to classify each image pixel as belonging or not to a blood vessel, but also selected, automatically, the network topology and the input characteristics more suitable for this purpose, thus simplifying the task of determining the most discriminant LBP variants to be used as input to the network. The evaluation of the method was carried out with two of most used digital retinal images databases to approach the solution of this type of problem: DRIVE (Digital Retinal Images for Vessel Extraction), and STARE (Structured Analysis of the Retina).

The method obtains competitive performance over other methods available in the literature, both in terms of accuracy and sensitivity, and from the standpoint of computational cost.

Keywords: Segmentation of blood vessels, Retinal Images, Artificial Neural Networks, Local Binary Patterns, Grammatical Evolution.

Índice General

Capítulo 1	1
Introducción	1
Capítulo 2	5
Revisión de la Bibliografía.....	5
2.1. Métodos no supervisados.....	5
2.1.1. Métodos basados en filtros coincidentes.....	5
2.1.2. Métodos basados en técnicas morfológicas.....	8
2.1.3. Método de seguimiento de los vasos sanguíneos.....	11
2.1.4. Método del análisis del espacio escalar y regiones crecientes.....	11
2.2. Métodos supervisados.....	13
Capítulo 3.....	19
Descripción del Método de Segmentación.....	19
3.1. Técnicas utilizadas y su aplicación.....	20
3.1.1 <i>Local Binary Patterns</i>	20
3.1.2 Variantes de LBP.....	25
3.1.3. Aprendizaje de redes neuronales con Evolución Gramatical.....	31
3.2. Descripción del Método.....	37
3.2.1. Fase I. Obtención de las redes neuronales: RNA_V y RNA_VF... 38	
3.2.2. Fase II. Fusión de la salida de las RNAs (RNA_V y RNA_VF) 46	
Capítulo 4.....	53
Evaluación.....	53
4.1. Evaluación preliminar.....	53
4.1.1. Estudio preliminar de la aplicabilidad de los operadores LBP a la segmentación de vasos en imágenes de retina.....	53

4.1.2. Estudio preliminar de la validez del algoritmo evolutivo de generación de RNAs mediante Evolución Gramatical.....	57
4.2. Evaluación experimental.....	58
4.2.1. Método de evaluación.	60
Capítulo 5.....	64
Comparación y Discusión de Resultados	64
Capítulo 6.....	76
Conclusiones y Trabajo Futuro	76
Bibliografía.....	78

Índice de Figuras

Figura 1.1: (a) Imagen de fondo de ojo y (b) máscara binaria correspondiente a su red de vasos sanguíneos. (Imagen número 6 de la base de datos de imágenes de retina DRIVE)	4
Figura 2.1. Verificación basada en la toma de muestras multiumbral: Las regiones de la imagen sobreviven la prueba de verificación como un todo o son eliminadas. (a) Imagen de entrada I. (b) Imagen binaria B_T , $20 \leq T < 200$. (c) Imagen binaria B_T , $200 \leq T < 255$ (Jiang y Mojon. 2003)	6
Figura 3.1. Píxel central y sus P vecinos espaciados igualmente a su alrededor en un radio R. [Guo et al-2010]	21
Figura 3.2. Cálculo del LBP.....	23
Figura 3.3. Los 36 únicos patrones binarios invariantes a la rotación LBP8, Rri que se pueden producir en un conjunto de 8 píxeles vecinos distribuidos simétricamente en un círculo de radio R. Los puntos negros y blancos corresponden a los valores de bit 0 y 1 en la salida de 8 bits del operador. La primera fila contiene los nueve patrones “uniformes” y el número interior corresponde a su código único de LBP8, Rriu2 [Ojala et al-2002].	24
Figura 3.4. Diferentes primitivas de texturas detectadas por LBP. [M.Pietikäinen et al-2011]	24
Figura 3.5. Ejemplo del cálculo de los operadores LBP y LTP con $\delta =5$. (Tan & Triggs. 2010)	26
Figura 3.6. División de un código LTP en dos códigos LBP, positivo y negativo.	27
Figura 3.7. Resultado de aplicar el operador LTP al canal verde de una imagen de fondo de ojo. (a) Imagen original; (b) LTP signo positivo; (c) LTP signo negativo. $P=8$, $R=2$, $\delta=2$. (Imagen 10 de la base de datos de imágenes de retina DRIVE).....	27
Figura 3.8. (a) bloque ejemplo; (b) Diferencias locales; (c) Las componentes de signo y (d) magnitud. [Guo et al-2010]	28
Figura 3.9. Resultado de aplicar CLBP al canal verde de una imagen de fondo de ojo. (a) Imagen original; (b) CLBP_S; (c) CLBP_M; (d) CLBP_C. $P=8$, $R=2$. (Imagen 10 de la base de datos de imágenes de retina DRIVE).....	30
Figura 3.10. Los dos nuevos componentes de CLBPLBP obtenidos al aplicar el CLBPLTP al canal verde de una imagen de fondo de ojo. (a) CLBPLTP_PM ; (b) CLBPLTP_NM; $P=8$, $R=2$. (Imagen 10 de la base de datos de imágenes de retina DRIVE).....	31
Figura 3.11. Gramática expresada en BNF para construir una red neuronal de dos capas. [Tsoulos et al-2008].....	35
Figura 3.12. Ejemplo del procedimiento de decodificación seguido por la Evolución Gramatical para el BNF de una red Neuronal	37
Figura 3.13. Fase I. Arquitectura para la construcción y aprendizaje de las dos redes neuronales (RNA_V y RNA_VF)	39
Figura 3.14. Fase II. Arquitectura para la fusión de las dos redes, RNA_V y RNA_VF, y obtener así la segmentación final (imagen binaria) de la red de vasos sanguíneos.	40
Figura 3.15. Proceso obtención de máscara de vasos finos para la imagen 22 de la base de datos de imágenes de retina DRIVE.	43
Figura 3.16. Resultados de aplicar las dos redes neuronales de segmentación de vasos, RNA_V y RNA_VF, al canal G de la imagen 1 de la base de datos de imágenes de retina DRIVE.	47
Figura 3.17. Máscaras obtenidas al aplicar los valores umbral, a los resultados obtenidos de aplicar las dos redes neuronales de segmentación de vasos, al canal G de la imagen 1 de la base de datos de imágenes de retina DRIVE.	48
Figura 3.18. Clústeres producidos por el algoritmo K medias para la variable compuesta por los canales G y R, y la salida de la red umbralizada, normalizados en el intervalo [0 255] para la imagen 7 de la base de datos de imágenes de retina DRIVE.	50
Figura 3.19. Resultado de aplicar el operador OR entre la salida del algoritmo k medias y la salida generada al aplicar la red de vasos finos a la imagen número 19 de la base de datos de imágenes de retina DRIVE.....	51
Figura 3.20. Segmentación resultante de aplicar el método propuesto a la imagen 19 de la base de datos de imágenes de retina DRIVE.....	52
Figura 4.1. Corona elíptica alrededor del contorno papilar de 4 píxeles de grosor (blanca), y corona alejada 40 píxeles de la anterior (rosa). (Imagen tomada de la base de datos DRIONS).....	54

Figura 4.2. Histogramas de la red de vasos, el contorno papilar y coronas elípticas situadas alrededor del contorno papilar a una distancia de 40 y 80 píxeles, calculados para el operador LBPP = 8, $R = 6riu^2$	56
Figura 4.3. Curvas ROC para las redes RNA_V ($A_z=0.9024$) y RNA_VF. ($A_z=0.8969$)	63
Figura 4.4. Curva ROC RNA_V OR RNA_VF. ($A_z=0.9208$)	63
Figura 5.1. Curvas ROC obtenida para (a) nuestro método ($A_z=0.9319$) y (b) curvas ROC para los métodos de otros autores, [Staal et al-2004].	65
Figura 5.2. Variación de la Precisión, Sensibilidad y Especificidad en función del área de filtrado de blobs, para el conjunto de imágenes de test de la base de datos de imágenes de retina DRIVE.	68
Figura 5.3.i. Influencia del área de filtrado de eliminación de blobs en los valores de Sensibilidad, Especificidad y Precisión. Imagen número 20 de la base de datos de imágenes de retina DRIVE. ..	69
Figura 5.3.ii. Influencia del área de filtrado de eliminación de blobs en los valores de Sensibilidad, Especificidad y Precisión. Imagen número 20 de la base de datos de imágenes de retina DRIVE. ..	70
Figura 5.4.i. Segmentación de la imagen 20 de la base de datos de imágenes de retina DRIVE. a) Imagen original. b) Gold Standard.....	70
Figura 5.4.ii. Segmentación de la imagen 20 de la base de datos de imágenes de retina DRIVE. c) Chaudhuri. d) Jiang. e) Niemeijer. f) Pérez. g) Staal. h) Zana.....	71
Figura 5.5. Valores de Precisión, Sensibilidad y Especificidad obtenidos con el método RNA+LBP, para cada una de las imágenes del conjunto de test de la base de datos de imágenes DRIVE.....	74

Índice de Tablas

Tabla 2.1. Resultados comparativos entre diferentes métodos de segmentación. [Staal et al-2004].....	18
Tabla 5.1. Comparativa de resultados obtenidos por nuestro método (RNAE+LBP) con diferentes métodos de segmentación, en el conjunto de test de la base de datos de imágenes de retina DRIVE. (Niemeijer et al. 2004, Staal et al. 2004 y Fraz et al 2012).....	65
Tabla 5.2. Comparativa de resultados obtenidos por nuestro método (RNAE+LBP) y diferentes métodos de segmentación, en el conjunto de test de la base de datos STARE. [Niemeijer et al-2004, Staal et al-2004 y Fraz et al- 2012]	66
Tabla 5.3. Comparativa de resultados obtenidos por diferentes métodos de segmentación, para la imagen número 20 del conjunto de test de la base de datos de imágenes de retina DRIVE.	72
Tabla 5.4. Comparativa de resultados obtenidos por diferentes métodos de segmentación, para la imagen número 15 del conjunto de test de la base de datos de imágenes de retina DRIVE.	73
Tabla 5.5. Comparativa de tiempos medios de segmentación, empleados por los distintos métodos, para el conjunto de imágenes de test de la base de imágenes DRIVE.	75

Capítulo 1

Introducción

Los médicos oftalmólogos realizan el estudio de la red de vasos sanguíneos de la retina o vasculatura retiniana, para el diagnóstico y estimación de múltiples enfermedades, ya que algunas de ellas, como la retinopatía diabética del prematuro, la hipertensión, la arteriosclerosis o la degeneración macular asociada a la edad, afectan a la morfología de los vasos en sí misma, y en su presencia éstos cambian de diámetro, de ángulos de ramificación y de tortuosidad.

La red de vasos sanguíneos de la retina es la única red de vasos del cuerpo humano, que se puede estudiar de forma no invasiva, y para ello, existen básicamente dos técnicas. Por un lado tenemos la exploración del fondo de ojo, realizada con un oftalmoscopio, o con un retinógrafo si se quiere asimismo obtener una fotografía digital, que consiste en visualizar, a través de la pupila y de los medios transparentes del globo ocular, la retina y el disco óptico, y por otro lado, la angiografía fluoresceínica, o angiofluoresceinografía, que implica la inoculación intravenosa de fluoresceína, un colorante, que viaja por el torrente sanguíneo y se encarga de aumentar el contraste de los vasos sanguíneos contra el fondo, emitiendo una fosforescencia que se puede captar mediante unas cámaras especiales. La primera de las dos técnicas es la menos invasiva, y las imágenes que se obtienen a partir de ella son las que se han utilizado en este trabajo.

Se entiende por segmentación de una imagen la extracción de los objetos de interés insertados en la misma, y para realizarla, se agrupan los píxeles con sus vecinos en función de similitud en criterios tales como el color, la luminancia, los bordes, las texturas, etc.

Efectuar la segmentación de la red de vasos sanguíneos de la retina manualmente, a partir de una imagen de fondo de ojo, requiere mucho entrenamiento y habilidad, además de ser un proceso demasiado lento, por lo que, disponer de algún medio capaz de realizarla de manera automática, puede suponer una gran ayuda para el diagnóstico y seguimiento de las enfermedades retínicas, siempre que se lleve a cabo en un corto lapso de tiempo y de un modo eficiente. El hecho de realizar la segmentación de forma automática a partir de imágenes digitales, permite también a los oftalmólogos llevar a cabo, de una forma más ágil, el seguimiento del tratamiento aplicado al paciente que presenta alguna enfermedad ocular, por medio del estudio comparado de los cambios morfológicos producidos en sus vasos retinianos a lo largo del tiempo.

Cuando nos enfrentamos al problema de realizar la segmentación de la vasculatura retiniana, de manera automática, nos encontramos principalmente con dos dificultades, la primera, es que la red de vasos sanguíneos presenta un bajo contraste con respecto al fondo y a otras estructuras de la retina, como el disco óptico, el borde de la retina o incluso a patologías como hemorragias u otras lesiones que puedan estar presentes y, la segunda, que los vasos presentan un amplio rango de anchuras complicando aun más la tarea.

También la vasculatura retiniana se utiliza para la localización de otros elementos importantes de la retina como el disco óptico, la fovea y las lesiones rojas, como por ejemplo microaneurismas y hemorragias.

El problema a resolver. Objetivos, metodología y recursos utilizados.

Dada una imagen de fondo de ojo, el objetivo de este trabajo es clasificar cada uno de los píxeles de la imagen como vaso sanguíneo o como no vaso, obteniendo así una máscara binaria, como la mostrada en la Figura 1.1., con la red vascular completa de la retina. Para ello se ha desarrollado

un nuevo método de segmentación de la red de vasos sanguíneos de la retina, basado en Redes Neuronales Artificiales Evolutivas, y en un conjunto de operadores basados en el operador LBP, *Local Binary Patterns*, [Ojala et al.-2000]. Al aplicar los operadores basados en LBP obtenemos, para cada píxel de la imagen, un vector de características, que utilizamos como entrada a un generador de redes neuronales artificiales (RNA) realizado con un algoritmo evolutivo basado en Evolución Gramatical (*Grammatical Evolution*), [O'Neill&Ryan-2001], que se encargará de encontrar, de forma automática, la topología de la RNA más adecuada, para clasificar cada píxel como vaso o no vaso, de seleccionar las características que aplicadas a cada píxel son las más adecuadas para constituir la entrada a la red neuronal seleccionada y de ajustar los pesos entre las neuronas que la componen en la fase de entrenamiento.

Para entrenar la RNA es necesario disponer de imágenes de fondo de ojo, en las que ya estén etiquetados los píxeles como vasos o no vasos. Se ha elegido para este propósito parte de la base de datos de imágenes de retina del Instituto de Ciencias de la Imagen de la Universidad de Utrecht conocida como base de datos de imágenes de retina DRIVE (Digital Retinal Images for Vessel Extraction).

Para la evaluación y comparación de los resultados del método presentado en este trabajo se han utilizado dos bases de datos de imágenes de retina: DRIVE, ya mencionada anteriormente, y la base de datos de imágenes de retina del proyecto STARE (Structure Analysis of the Retina) de la Universidad de California, recopilada por Hoover. [Hoover et al-2000].

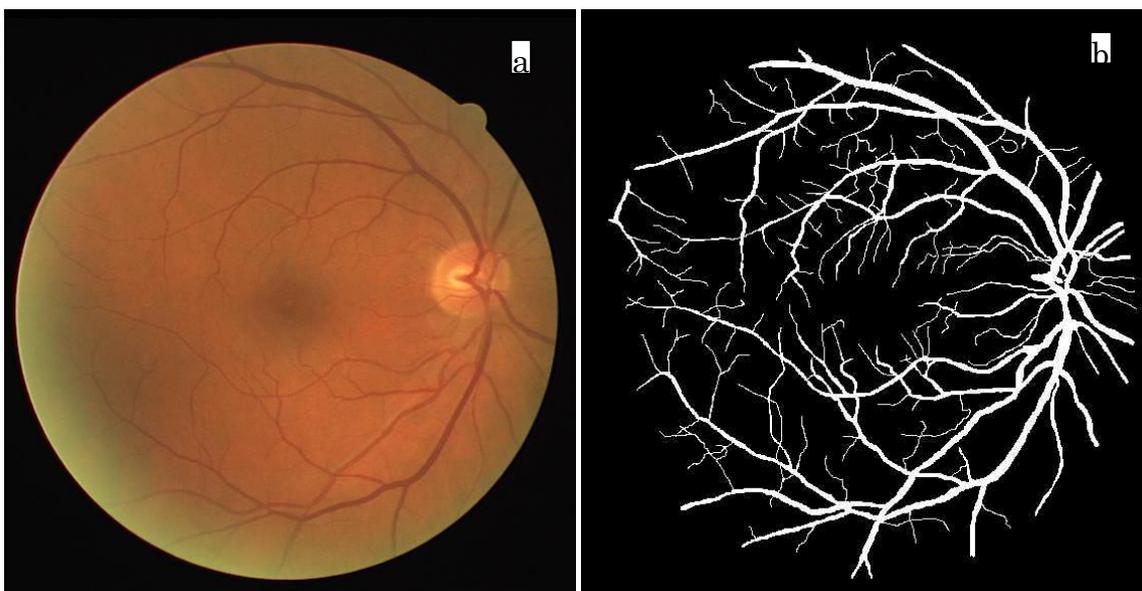


Figura 1.1: (a) Imagen de fondo de ojo y (b) máscara binaria correspondiente a su red de vasos sanguíneos. (Imagen número 6 de la base de datos de imágenes de retina DRIVE)

Capítulo 2.

Revisión de la Bibliografía.

A lo largo de las últimas décadas, se han ido desarrollando métodos para la segmentación de los vasos sanguíneos en imágenes de retina, basados en diversos enfoques, que podemos agrupar para su revisión en dos bloques, como se propone en los trabajos de [Staal et al.-2004] y [Salem et al.-2007]. Por un lado tenemos los métodos supervisados, los cuales necesitan imágenes etiquetadas manualmente para poder ser entrenados, y por otro el resto de métodos que están basados en diferentes conjuntos de técnicas. Vamos a ver una breve descripción de cada uno de ellos.

2.1. Métodos no supervisados

Resumimos en este apartado algunos de los métodos no supervisados más utilizados como son los basados en filtros coincidentes, los basados en técnicas morfológicas, los de seguimiento de vasos sanguíneos y por último los basados en el análisis del espacio escalar de regiones crecientes.

2.1.1. Métodos basados en filtros coincidentes.

Existen métodos basados en filtros coincidentes de una y de dos dimensiones. Un ejemplo de método utilizando filtros de una dimensión es el propuesto por Jiang y Mojon, [Jiang&Mojon-2003] que consiste en que se van realizando ensayos sucesivos, aplicando en cada uno de ellos diferentes umbrales a la imagen y extrayendo todos los objetos binarios resultantes. En la figura 2.1 se muestra un ejemplo de ello. Aplicando después una

clasificación a estos objetos, se pueden seleccionar aquellos que tengan características semejantes a los vasos sanguíneos. Y con la combinación de todos los objetos seleccionados se forma el árbol de vasos sanguíneos.

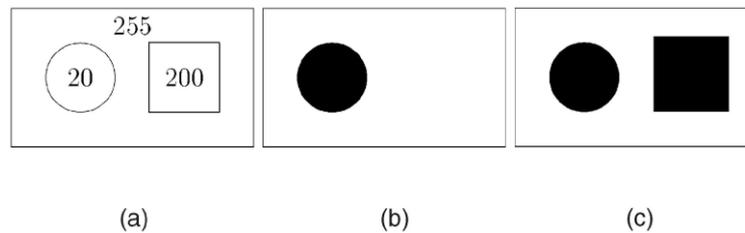


Figura 2.1. Verificación basada en la toma de muestras multiumbral: Las regiones de la imagen sobreviven la prueba de verificación como un todo o son eliminadas. (a) Imagen de entrada I . (b) Imagen binaria B_T , $20 \leq T < 200$. (c) Imagen binaria B_T , $200 \leq T < 255$ (Jiang y Mojon. 2003)

El algoritmo del método es el siguiente:

Entrada: Imagen de Intensidad I
 Conjunto de umbrales de prueba T_s
 Salida: Imagen binaria B del objeto y píxeles de fondo
 Para cada umbral $T \in T_s$ hacer {
 obtener la imagen binaria B_T aplicando a I el umbral T ;
 aplicar el procedimiento de verificación a B_T , obteniendo la imagen binaria
 B_T^* formada por las regiones parciales o enteras supervivientes al
 aplicar el umbral.
 }
 $B = \bigcup_{T \in T_s} B_T^*$

Otros ejemplos de utilización de filtros coincidentes de una dimensión son los expuestos en [Zhou et al-1994], [Gao et al-1997] y [Gao et al-2000], y [Tolias&Panas-1998].

Por otro lado, un filtro de dos dimensiones es básicamente una máscara representada por una matriz bidimensional, de un determinado tamaño, donde los valores de cada elemento se escogen para detectar una determinada propiedad de la imagen. El centro de la máscara se va

moviendo a lo largo de todos los píxeles de la imagen, y en cada posición se multiplica el valor de los píxeles bajo la máscara por los valores de la misma, a continuación se suman y el resultado es el nuevo valor del píxel que ocupa el lugar central de la máscara, obteniendo de esta forma una nueva imagen. Por ejemplo para detectar puntos aislados, que difieren en intensidad respecto a sus vecinos, se puede utilizar una máscara de 3x3 con todos los valores iguales a -1, excepto el del centro al que se le asigna el valor 8. Si se aplica esta máscara a una zona de una imagen en la que todos los píxeles tienen la misma intensidad, i , el resultado será 0, ya que el resultado de aplicar la máscara será $x = -8i + 8i = 0$. Sin embargo si el píxel central tiene una intensidad diferente a la de sus vecinos, el valor obtenido no será cero.

Como ejemplo de utilización de filtros de dos dimensiones para la segmentación de vasos sanguíneos tenemos, en primer lugar, el método expuesto en [Chaudhuri et al-1989], en el que utilizan filtros de dimensiones 15x16. Estos observaron tres importantes propiedades de los vasos sanguíneos en las imágenes de retina y, en base a ellas, diseñaron 12 plantillas diferentes de filtros coincidentes para detectar los posibles segmentos de vasos a lo largo de todas las direcciones. La segmentación de los vasos se lleva a cabo convolucionando la imagen con versiones giradas de los filtros coincidentes y reteniendo únicamente la máxima respuesta. Para una resolución angular de 15 grados se necesitan 12 convoluciones. Las propiedades en las que se basó Chaudhuri son:

- a) Los vasos sanguíneos tienen la propiedad de que sus dos bordes siempre discurren en paralelo uno del otro, y presentan curvaturas pequeñas. Las direcciones del gradiente para estos dos elementos están separadas 180°, y las denomina “anti-paralelos”. Los pares anti-paralelos pueden ser aproximados con segmentos lineales de ancho finito.

- b) Debido a que los vasos sanguíneos presenta una reflectancia baja comparada con la de otras partes de la retina, aparecen más oscuros en relación con el fondo. El perfil de intensidad de los vasos se puede aproximar por una Gaussiana, a pesar de que varía ligeramente de un vaso a otro.
- c) Aunque la anchura de los vasos decrece a medida que viajan radialmente hacia el exterior del disco óptico, el cambio es gradual. Según sus observaciones el ancho de los vasos está en el rango de 2 a 10 píxeles.

Este esquema mantiene la simplicidad computacional de los operadores de extracción de bordes de tipo realce/umbral, y a la vez, incorpora las ventajas de utilizar detectores de bordes basados en modelos. El problema que presentaba es que debido al gran tamaño del núcleo de convolución, el tiempo de cálculo en un computador normal, cuando se desarrolló el método, era muy elevado. [Chaudhuri et al-1989].

En el año 2000, [Hoover et al-2000] desarrollan su algoritmo de segmentación de vasos, basándose en el método expuesto en [Chaudhuri et al-1989], al que añaden el “sondeo de umbral”, que consiste en evaluar en cada imagen obtenida como respuesta a un filtro coincidente, un conjunto de criterios que sirvan para determinar el umbral de sondeo, con el que, finalmente, se decidirá si el área analizada es o no un vaso. Con su método aseguraron reducir el número de falsos positivos obtenido por el método en el que se basaron, en un factor de 15 veces.

2.1.2. Métodos basados en técnicas morfológicas.

La morfología matemática es útil para realizar análisis de las formas geométricas de una imagen, utilizando aspectos teóricos como la unión, la

intersección o la inclusión. Los operadores morfológicos también requieren un elemento estructurante, que será el que determine el efecto que producirá la operación morfológica sobre la imagen. Los elementos estructurantes más utilizados tienen formas simples como círculo, cuadrado o línea.

Las operaciones morfológicas elementales con respecto a un elemento estructurante son la *erosión*, en la que los elementos menores que el elemento estructurante son eliminados de la imagen, la *dilatación*, cuyo resultado es el conjunto de elementos tal que al menos algún elemento del conjunto estructurante está contenido en la imagen. La *apertura*, es una composición de un operador de erosión seguido de otro de dilatación, que sirve para romper uniones y eliminar protuberancias. El cierre, también es una composición de erosión y dilatación, pero se aplica primero la dilatación, y sirve para unir fisuras y eliminar pequeños huecos. Por último, la *transformación de sombrero alto (top-hat)*, sirve para resaltar detalles en zonas de poco contraste. Se puede conseguir un suavizado de la imagen, por ejemplo combinando una apertura y un cierre con el mismo objeto estructural.

Todos los autores combinan las técnicas morfológicas con otras, como la aplicación de la transformada de Hough, [Zana&Klein-1999], la detección de la línea central de los vasos, [Fraz et al-2008 y Fraz et al-2012], [Medoça&Campilho-2006], o la evaluación de la curvatura [Zana&Klein-2001].

El método propuesto por Zana y Klein, [Zana&Klein-1999], es un método general de segmentación que se basa en la utilización de un algoritmo que consiste en varias operaciones morfológicas, que se puede dividir en los siguientes pasos:

1. Reconocimiento de las partes lineales calculando el supremo de aperturas.

2. Reducción del ruido utilizando reconstrucciones geodésicas del supremo de aperturas en la imagen original.
3. Eliminación de los diferentes tipos de patrones indeseables aplicando el Laplaciano en el resultado de los pasos anteriores, seguido de un filtro basado en una secuencia de aperturas y cierres con elementos estructurales lineales de longitud creciente (supremo de apertura de tamaño 15 con reconstrucción, ínfimo de cierre de tamaño 15 con reconstrucción, y supremo de aperturas de tamaño 29).

El resultado obtenido con las operaciones anteriores se filtra aplicando un umbral para producir la segmentación de los vasos sanguíneos.

De las imágenes en formato RGB, explican como el canal azul es muy débil y en el canal rojo aparece mucho ruido, por lo que utilizan al igual que los médicos el canal verde para su estudio, al que aplican en primer lugar una inversión de los niveles de grises, ya que los vasos aparecen oscuros.

La detección se puede resumir en este caso en los siguientes pasos:

- Si la imagen es una imagen verde, realizar una inversión.
- Eliminar el ruido utilizando una apertura con reconstrucción.
- Realizar la suma de top-hats con elementos estructurantes lineales de direcciones variadas.
- Suavizar la imagen con un filtro de Gaussiana.
- Calcular el Laplaciano.
- Realizar un filtrado adicional con elementos estructurantes lineales y reconstrucción.
- Aplicar un umbral a los valores positivos.

Mendoça y Campillo [Medoça&Campilho-2006], utilizan, igualmente, un conjunto de operadores morfológicos en los que varían el tamaño de los elementos estructurantes, con el objetivo de realzar vasos sanguíneos de diferentes grosores. Los mapas binarios de los vasos se obtienen en cuatro

escalas diferentes, utilizando un método de reconstrucción morfológica binaria con operador de doble umbral. Los vasos se rellenan con un proceso iterativo de crecimiento de regiones, delimitado por los píxeles centrales de los vasos, que son utilizados como puntos de semilla inicial, limitando la fase de agregación con una de las máscaras binarias.

En su método, [Fraz et al-2012], explotan la aplicación de operadores *top-hat* multidireccionales con un elemento estructurante de rotación lineal, que se adapta al perfil del vaso. Este operador dilata los vasos en una determinada dirección y entonces se extraen utilizando planos de corte de bit. La imagen final de la vasculatura se obtiene utilizando los píxeles centrales de los vasos como semilla de un algoritmo de crecimiento de regiones con semilla. Los píxeles candidatos de la línea central de los vasos se obtienen aplicando filtros Gaussianos de diferencia de desplazamiento (DoOG) en la imagen normalizada.

2.1.3. Método de seguimiento de los vasos sanguíneos.

Esta técnica consiste en que el usuario fija un punto de inicio y otro de fin sobre un único vaso, y el algoritmo localiza el centro del vaso en cada una de las secciones transversales del mismo a lo largo de su eje longitudinal. Para la localización de los puntos de vasos se suele utilizar también un filtro coincidente. Estas técnicas se ven obstaculizadas por el cruce y la bifurcación de los vasos, pero sin embargo pueden aportar medidas muy precisas del ancho y la tortuosidad de los mismos. Ejemplos de este método son los trabajos expuesto en [Tamura et al-1988] y [Kochner et al-1998].

2.1.4. Método del análisis del espacio escalar y regiones crecientes.

En [Martínez-Pérez et al-1999] y [Martínez-Pérez et al-2007], recurren a una combinación del análisis espacio escalar y regiones crecientes para realizar la segmentación de la vasculatura. Las dos características que utilizan son la magnitud del gradiente de intensidad y la fuerza de la cresta. La primera derivada direccional describe la variación de la intensidad de la imagen en la vecindad de un punto. En un marco multiescalar, la magnitud del gradiente representa la pendiente de la intensidad de un valor determinado del parámetro de escala s .

$$\nabla I_s = \sqrt{(\partial_x I_s)^2 + (\partial_y I_s)^2}.$$

La fuerza de la cresta se determina calculando el mayor valor absoluto del vector propio (*eigenvector*) $|\lambda_1|$ de la matriz de las segundas derivadas parciales de la intensidad de la imagen, el Hessiano. Para tener en cuenta la diferencia en el ancho de vasos a lo largo de la retina estas dos características se normalizan por la escala s sobre el espacio escalar al tiempo que se conserva solo el máximo local. El máximo local de la magnitud del gradiente γ y el máximo local de la fuerza de la cresta κ son entonces:

$$\gamma = \max_s \left[\frac{|\nabla I(s)|}{s} \right], \quad \kappa = \max_s \left[\frac{|\lambda_1(s)|}{s} \right]$$

Los histogramas de ambas características se utilizan en el último paso de crecimiento de regiones de múltiples pasadas, que parte de una o varias regiones iniciales llamadas semillas, determinadas de forma manual o automática mediante un heurístico. Las regiones se van expandiendo a los píxeles vecinos mientras se cumple un determinado criterio de homogeneidad, el cual progresivamente segmenta los vasos sanguíneos utilizando la información característica junto con la información espacial sobre los ocho píxeles vecinos, obteniendo de esta manera una imagen segmentada binaria. Esto es llevado a cabo alternando las regiones de vasos y fondo de la imagen aumentando y reduciendo los umbrales de las características después de cada iteración. El proceso se repite hasta que no se añada ningún píxel más a ninguna de las dos regiones.

2.2. Métodos supervisados

Los métodos supervisados se basan en extraer un cierto número de características para cada píxel de la imagen, y luego, mediante una red neuronal, clasificarlos como pertenecientes o no a vasos sanguíneos. Para entrenar la red, y ajustar los *pesos* en las conexiones que unen sus neuronas, es necesario disponer de un conjunto de imágenes etiquetadas manualmente. En [Synthanayothin et al-1999], ya utilizaron una red neuronal en su método, para clasificar los píxeles de las imágenes de fondo de ojo, compuesta por tres capas, 200 nodos de entrada, 20 nodos ocultos y dos salidas.

En el método expuesto en [Niemeijer et al-2004], el vector característico para cada píxel de la imagen lo extraen del canal verde del espacio de color RGB. En los experimentos iniciales probaron dos tipos de características, la salida producida al aplicar filtros y los valores de los píxeles en relación a sus vecinos. Los resultados para la salida al aplicar filtros produjo mejores resultados.

También compararon el resultado de la clasificación con tres clasificadores diferentes, un clasificador kNN, un clasificador lineal y un clasificador cuadrático, seleccionando finalmente el clasificador kNN porque fue con el que obtuvieron mejores resultados. Los filtros consistieron en filtros de Gaussiana y sus derivadas hasta orden 2 y escalas $s=1, 2, 4, 8$ y 16 píxeles. El número total de características para cada píxel fue de 31, y cada una de ellas se normalizó con media igual a cero y varianza unidad antes de la clasificación. Con el uso del clasificador kNN, se obtiene un mapa de probabilidad que indica, para cada píxel de la imagen, la probabilidad de que un píxel sea vaso o no vaso. Aplicando un umbral al mapa de probabilidad se puede obtener un mapa binario de la segmentación de la vasculatura retiniana.

Método PBM.

En [Staal et al-2004], afirman que la representación en base a los píxeles no es la más adecuada para los vasos sanguíneos y por ello fundamentan su enfoque, al que denominan PBM, en la propiedad que tienen los vasos de ser estructuras alargadas. Su algoritmo se basa en la detección de crestas, apoyado en lo expuesto en [Eberly-1996] y [Kalitzin et al- 2001], aduciendo que las crestas son indicadores naturales de vasos en las imágenes de fondo de ojo. Utilizan para la extracción de crestas el canal verde de la imagen del espacio de color RGB, como en la mayoría de los casos anteriores, porque afirman también que es el que ofrece mayor contraste entre los vasos y el fondo de la imagen. Las crestas se definen como puntos donde la imagen tiene un valor extremo en la dirección de curvatura de mayor superficie, y su detección se lleva a cabo de forma similar a la explicada en el apartado 2.1.4., donde se describe el método de [Martínez-Pérez et al-2007].

En las imágenes de fondo de ojo debido a que los vasos son estructuras oscuras, se utilizan las crestas de los mínimos locales, y en las angiografías fluorescentes, donde los vasos son estructuras brillantes se deben utilizar las crestas de los máximos locales.

El siguiente paso, una vez determinadas las crestas de la imagen, es agrupar los píxeles de las mismas en función de su pertenencia al mismo vaso sanguíneo, en lo que denominan *conjunto convexo afín*. También aquí se utiliza para ello un algoritmo de crecimiento de región, que va comparando los píxeles de cresta, ya agrupados, con sus vecinos en un radio ε_c donde el subíndice c denota conectividad. En el caso de que no exista ningún píxel para agrupar, se elige uno al azar como semilla de los píxeles que restan por agrupar. La comparación entre los píxeles agrupados y los candidatos dentro de la vecindad se basa en dos condiciones:

- 1) Las direcciones del vector propio de los píxeles de cresta deben de ser similares y

- 2) Si se cumple la primera condición los píxeles deben estar en la misma cresta, y no en crestas paralelas.

Por último, detectadas las crestas y agrupados sus píxeles en conjuntos convexos afines, se divide la imagen en parcelas basándose en los conjuntos convexos, y cada píxel se asigna al conjunto convexo al que está más próximo. Estas parcelas se referencian como regiones de conjuntos convexos (CSRs).

Determinados los conjuntos convexos y las CSRs, el siguiente paso es elegir las características más apropiadas para utilizar con el clasificador elegido. A priori se definen muchas características, ya que de antemano no se sabe cuáles serán las más apropiadas, y se seleccionan mediante el método de *selección secuencial hacia delante* de Whitney. [Whitney-1971].

El algoritmo comienza con un conjunto vacío de características y, en cada paso, la característica que mejor satisfaga la función criterio elegida se incorpora al conjunto de características actual. Después de que se han agrupado todas las características, el conjunto que ofrezca un mejor rendimiento será el utilizado para la clasificación. La función criterio que han elegido en este caso es el área bajo la curva ROC, A_z , [Park et al-2004].

Partiendo de los conjuntos convexos, y en base a que los que forman parte de un vaso se pueden distinguir de los no vasos por su apariencia local, se extraen características basadas en los perfiles perpendiculares a los mismos. Para extraer la información, se muestrea el perfil para cada punto k en el conjunto convexo del canal verde de la imagen en la dirección del vector $\hat{v}(k)$ con $c(k)$ como punto medio. Los perfiles obtenidos se promedian y el perfil medio $\psi(n)$, con $n=-N, \dots, N$ se toma como perfil del conjunto convexo. La operación de promediado realiza cierto suavizado, así que no se considera necesaria la difuminación del perfil. Las características extraídas del perfil son 15, y algunas de ellas son las siguientes:

- 1) La altura del perfil: $h = \psi(0)$
 - 2) El ancho del perfil definido como la distancia entre los bordes extremos derecho e izquierdo del perfil: $w = n_{re} - n_{le}$ ($n_{re} = \arg \max_{n>0} \psi'(n)$, siendo ψ' la primera derivada del perfil. n_{le} se define de manera similar
 - 3) La intensidad del borde, definida como: $s_e = \psi'(n_{le}) + \psi'(n_{re})$
 - 4) La intensidad dividida por el ancho $\frac{s_e}{w}$
 - 5) La altura del borde: $h_e = \frac{1}{2}(\psi(n_{le}) + \psi(n_{re}))$
 - 6) La distancia entre el primer y el último punto del conjunto convexo: $d = \|c(1) - c(K)\|$
 - 7) La longitud del conjunto convexo: $l = \sum_{k=2}^K \|c(k) - c(k-1)\|$
 - 8) La curvatura del conjunto convexo, aproximada por: $\kappa = \sum_{k=2}^K \hat{v}(k) - \hat{v}(k-1)$
-

A partir de las CSRs se extraen otras 12 características. Ajustando una línea recta a lo largo del conjunto convexo, se puede establecer un sistema de coordenadas para cada CSR. Como origen de las coordenadas se elige el centro de masas del conjunto convexo. El primer eje está a lo largo de la dirección de la línea ajustada, el segundo eje es perpendicular al primero y su dirección se elige de tal forma que se establece un sistema de coordenadas a derecha.

Las características calculadas para los píxeles de la CSR se pueden subdividir en características que no tienen en cuenta información del conjunto convexo y características que explotan el uso del sistema de coordenadas. Algunas de estas 12 características son:

1. El valor del canal rojo de la imagen en la posición del píxel. $r(x)$
2. El valor del canal verde de la imagen en la posición del píxel. $g(x)$
3. La relación entre los valores de los canales rojo y verde

4. La diferencia entre los valores del canal rojo del píxel y el punto más cercano en el conjunto convexo: $r(x) - r(c_{closest})$.
5. La coordenada del píxel con respecto al primer eje x'_1 .
-

En [Staal et al, 2004] también probaron varios clasificadores, eligiendo finalmente un kNN y las pruebas del método se llevaron a cabo con las bases de datos de imágenes de retina, DRIVE y STARE.

De la base de datos de imágenes de retina DRIVE se tomaron 40 imágenes, de las cuales siete tenían alguna patología. Las imágenes fueron segmentadas manualmente por tres observadores, que fueron entrenados por un oftalmólogo, teniendo que marcar los píxeles, de los que tuvieran por lo menos el 70% de certeza, como vaso. Las 40 imágenes se dividieron en un conjunto de entrenamiento y un conjunto de test, cada uno formado por 20 imágenes. El conjunto de test fue segmentado dos veces, resultando en un conjunto A y un conjunto B. Los observadores del conjunto A marcaron 577.649 píxeles como vasos y 3.960494 como fondo (12,7% vasos), y en el conjunto B se marcaron 566.532 puntos como vasos y 3.981.611 como fondo (12,3% vasos). A la segmentación del conjunto A la denominan *gold-standard* y se utiliza, por comparación con ella, para medir la precisión de los diferentes métodos de segmentación. A la segmentación del conjunto B se le denomina “*segundo observador*”.

La base de datos STARE fue recopilada por Hoover, [Hoover et al-2000], y consiste en 19 imágenes. En este caso la segmentación se llevó a cabo por dos observadores, el primero segmentó 615.726 píxeles como vasos y 5.293.034 como fondo (10,4% vasos), y el segundo 879.695 píxeles como vasos y 5.029.065 como fondo (14,9% vasos). El primer observador segmentó muchos menos vasos finos, y de ahí la diferencia entre ambos.

Staal et al., aportan resultados comparativos con los obtenidos en [Jiang&Mojon-2003] y [Hoover et al-2000], y el segundo observador, en

ambas bases de datos de imágenes de retina, en cuanto a precisión y al área bajo la curva ROC, A_z . La precisión se define como la relación entre la suma de verdaderos positivos, puntos marcados como vasos que realmente son vasos, y los verdaderos negativos, puntos marcados como fondo que realmente son fondo, y el total de píxeles de la imagen.

Criterio	Método	Base de datos	
		DRIVE	Hoover
A_z	Hoover		0.7590
	Jiang	0.9327	0.9298
	PBM	0.9520	0.9614
Precisión	2º Observador	0.9473	0.9351
	Hoover		0.9275
	Jiang	0.8911	0.9009
	PBM	0.9441	0.9516

Tabla 2.1. Resultados comparativos entre diferentes métodos de segmentación. [Staal et al-2004]

Capítulo 3.

Descripción del Método de Segmentación.

Al enfrentarnos al problema de la segmentación de los vasos sanguíneos en imágenes de retina partimos de las siguientes hipótesis:

- La segmentación de la red de vasos sanguíneos en imágenes de retina es un proceso complicado que en general tiene un alto coste computacional.
- Los denominados *Local Binary Patterns* (LBP), patrones binarios locales y sus variantes, son lo suficientemente discriminantes para permitir determinar si un píxel, en una imagen de retina, pertenece o no a un vaso sanguíneo, con un coste computacional bajo.
- Decidir cuáles de esos patrones son los más adecuados, cómo se han de combinar en una red neuronal y entrenar esa red neuronal, es una tarea difícil que un algoritmo generador de redes neuronales desarrollado mediante Evolución Gramatical será capaz de resolver de la forma más óptima posible.

En base a estas hipótesis, se propone diseñar un método de segmentación basado en la utilización de una red neuronal, obtenida mediante un algoritmo evolutivo basado en Evolución Gramatical, entrenada para la detección de vasos sanguíneos en imágenes de fondo de ojo, capaz de determinar con la mayor precisión posible a la vez que con una sensibilidad elevada y un coste computacional bajo, si cada uno de los píxeles de una imagen de retina forma parte, o no, de la red de vasos sanguíneos. Las características de las imágenes de retina, que se utilizarán

para entrenar la red neuronal, se obtienen aplicando a las imágenes operadores basados en LBP, cuya eficacia en la determinación de texturas y su bajo coste computacional han sido ya ampliamente comprobados.

En este capítulo se exponen, en primer lugar, las técnicas que se han empleado y el modo en que han sido aplicadas y, finalmente, la descripción del método que engloba dichas técnicas.

3.1. Técnicas utilizadas y su aplicación.

Las técnicas en las que se apoya el método son el operador de patrones binarios locales, *local binary patterns* (LBP), y sus variantes LTP y CLBP, y la Evolución Gramatical y su aplicación en la construcción y entrenamiento de redes neuronales. A continuación se resumen brevemente todas ellas, y el modo en que han sido aplicadas en este trabajo.

3.1.1 *Local Binary Patterns*.

En [Ojala et al-2000] y [Ojala et al-2002] se desarrolla un operador invariante a cambios de iluminación uniforme y a la rotación, para detectar los parámetros característicos de una textura al que denominaron *Local Binary Patterns* (LBP).

El análisis de texturas bidimensionales tiene muchas aplicaciones potenciales, pero existen pocos casos de éxito debido, principalmente, a que la mayoría de los enfoques para la clasificación de texturas, se basan en comparar la escala espacial, la orientación y las propiedades en la escala de grises, de la textura a analizar, con muestras de texturas conocidas. Pero las texturas no siempre las tenemos con la misma resolución, con la misma rotación espacial o en las mismas condiciones de iluminación. Es por todo ello por lo que estos enfoques no son adecuados. El LBP es un operador de texturas muy simple, pero a la vez muy eficaz y que presenta una gran robustez a los cambios de iluminación, por lo cual, debido a su potencia

discriminativa y a su simplicidad computacional se ha vuelto muy popular en diversos tipos de aplicaciones, e incluso en el análisis de imágenes en tiempo real, como demuestra el gran número de artículos ya publicados, y disponibles en la página de la universidad de Oulu¹.

Para obtener el LPB, se comienza definiendo la textura T en un grupo local de píxeles vecinos de una imagen monocroma, como la distribución conjunta de los niveles de gris de P ($P > 1$) píxeles de la imagen:

$$T = t(g_c, g_0, g_1, \dots, g_{P-1}). \quad (1)$$

En la cual el valor g_c corresponde al valor de gris del píxel central del grupo local y g_p ($p = 0, \dots, P - 1$) corresponde a los valores de grises de P píxeles igualmente espaciados en un círculo de radio R ($R > 0$) que forman un conjunto vecinal circular simétrico. Ver figura 3.1.

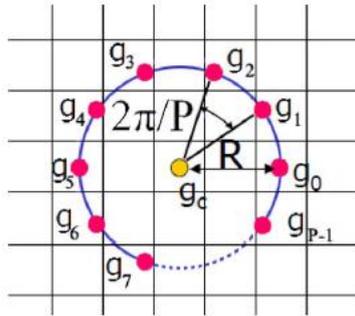


Figura 3.1. Píxel central y sus P vecinos espaciados igualmente a su alrededor en un radio R . [Guo et al-2010]

Ahora, como primer paso se resta el valor en la escala de grises del píxel central (g_c) de los valores de grises del conjunto circular simétrico de píxeles, obteniendo:

$$T = t(g_c, g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c). \quad (2)$$

Después se asume que las diferencias $g_i - g_c$ son independientes de g_c lo cual nos permite factorizar la expresión anterior (2) como:

¹ http://www.cse.oulu.fi/CMV/LBP_Bibliography

$$T \approx t(g_c)t(g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c). \quad (3)$$

Esta distribución factorizada es sólo una aproximación a la distribución conjunta, que no garantiza una independencia exacta, pero se acepta una pequeña pérdida de información ya que nos permite lograr la invarianza con respecto a cambios uniformes de iluminación. La distribución $t(g_c)$ en (3) describe la luminancia total de la imagen, la cual no está relacionada a la textura de la imagen local, y consecuentemente, no nos aporta ninguna información útil al análisis de la textura.

Por lo tanto, la mayoría de la información en la distribución conjunta original del nivel de grises (1) sobre las características de la textura se transmiten por la distribución del conjunto de diferencias:

$$T \approx t(g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c). \quad (4)$$

Este si es un operador muy discriminante de texturas.

Dado que las diferencias $g_i - g_c$ no se ven afectadas por los cambios en la luminancia media y la distribución de las diferencias es invariante a los desplazamientos de la escala de grises, se consigue la invarianza con respecto a la variación de la escala de grises considerando únicamente los signos de las diferencias, en lugar de sus valores exactos.

$$T \approx t(s(g_0 - g_c), s(g_1 - g_c), \dots, s(g_{P-1} - g_c)). \quad (5)$$

Donde

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (6)$$

Si asignamos un factor binomial 2^p por cada signo $s(g_p - g_c)$ podemos transformar (5) en un único número $LBP_{P,R}$ que caracteriza la estructura espacial de la textura local de la imagen. Ver figura 3.2.

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p. \quad (7)$$

El operador $LBP_{P,R}$ es por definición invariante ante cualquier transformación monótonica en la escala de grises.

Ejemplo	$s(g_p - g_c)$	2^p																												
<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td style="border: 1px solid black; padding: 2px 5px;">78</td><td style="border: 1px solid black; padding: 2px 5px;">99</td><td style="border: 1px solid black; padding: 2px 5px;">50</td></tr> <tr><td style="border: 1px solid black; padding: 2px 5px;">54</td><td style="border: 1px solid black; padding: 2px 5px;">54</td><td style="border: 1px solid black; padding: 2px 5px;">49</td></tr> <tr><td style="border: 1px solid black; padding: 2px 5px;">57</td><td style="border: 1px solid black; padding: 2px 5px;">12</td><td style="border: 1px solid black; padding: 2px 5px;">13</td></tr> </table>	78	99	50	54	54	49	57	12	13	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td style="border: 1px solid black; padding: 2px 5px;">1</td><td style="border: 1px solid black; padding: 2px 5px;">1</td><td style="border: 1px solid black; padding: 2px 5px;">0</td></tr> <tr><td style="border: 1px solid black; padding: 2px 5px;">1</td><td style="border: 1px solid black; padding: 2px 5px;">C</td><td style="border: 1px solid black; padding: 2px 5px;">0</td></tr> <tr><td style="border: 1px solid black; padding: 2px 5px;">1</td><td style="border: 1px solid black; padding: 2px 5px;">0</td><td style="border: 1px solid black; padding: 2px 5px;">0</td></tr> </table>	1	1	0	1	C	0	1	0	0	<table style="border-collapse: collapse; width: 100%; text-align: center;"> <tr><td style="border: 1px solid black; padding: 2px 5px;">32</td><td style="border: 1px solid black; padding: 2px 5px;">64</td><td style="border: 1px solid black; padding: 2px 5px;">128</td></tr> <tr><td style="border: 1px solid black; padding: 2px 5px;">16</td><td style="border: 1px solid black; padding: 2px 5px;"></td><td style="border: 1px solid black; padding: 2px 5px;">1</td></tr> <tr><td style="border: 1px solid black; padding: 2px 5px;">8</td><td style="border: 1px solid black; padding: 2px 5px;">4</td><td style="border: 1px solid black; padding: 2px 5px;">2</td></tr> </table>	32	64	128	16		1	8	4	2	Patrón = 01111000 $LBP = 8 + 16 + 32 + 64 = 120$
78	99	50																												
54	54	49																												
57	12	13																												
1	1	0																												
1	C	0																												
1	0	0																												
32	64	128																												
16		1																												
8	4	2																												

Figura 3.2. Cálculo del LBP.

El operador $LBP_{P,R}$ produce 2^P valores de salida diferentes, que se corresponden a los 2^P patrones binarios distintos que se pueden formar con los P píxeles vecinos del conjunto. Cuando se gira la imagen, los valores del nivel de gris g_p correspondiente se moverán a lo largo del perímetro del círculo alrededor de g_0 . Si designamos g_0 como el valor del nivel de gris del elemento $(0, R)$ a la derecha de g_c , rotando un patrón binario determinado obtendremos un valor diferente de $LBP_{P,R}$. Para eliminar este efecto de la rotación y asignar un único identificador a cada invariante de rotación de LBP se define:

$$LBP_{P,R}^{ri} = \min\{ROR(LBP_{P,R}, i) \mid i = 0, 1, \dots, P - 1\} \quad (8)$$

donde el superíndice ri significa “invariante a la rotación” y $ROR(x, i)$ realiza un desplazamiento circular a la derecha bit a bit del número binario x , i veces ($|i| < P$). En términos de píxeles de la imagen, (8) corresponde simplemente a rotar el conjunto de píxeles vecinos en el sentido de las agujas del reloj tantas veces hasta que el número máximo de los bits más significativos, comenzando por g_{p-1} sea 0.

$LBP_{P,R}^{ri}$ cuantifica la ocurrencia estadística de los patrones invariantes de rotación de los individuos correspondientes a ciertas micro características en la imagen, por lo tanto los patrones pueden ser considerados como detectores de características. A continuación, en la figura 3.3 se muestran los 36 patrones invariantes a la rotación obtenidos para $LBP_{8,R}^{ri}$.

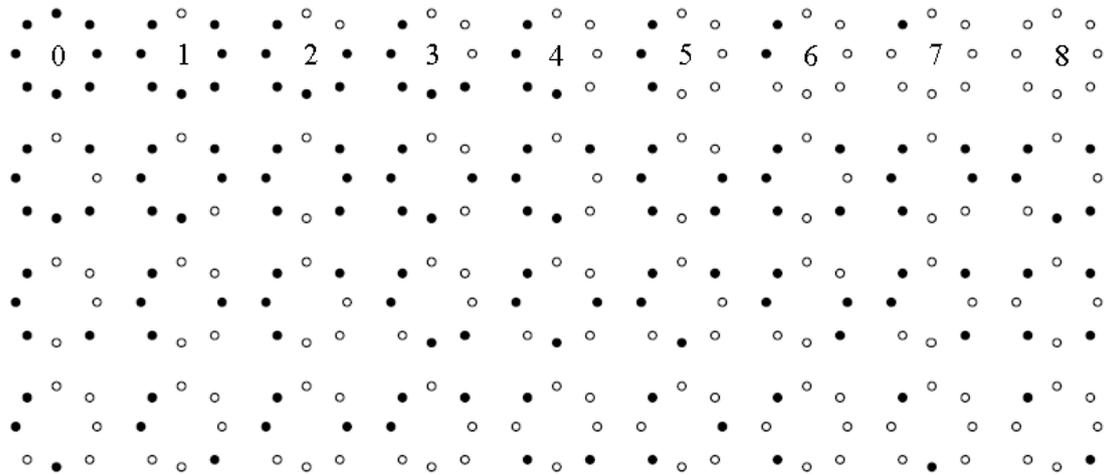


Figura 3.3. Los 36 únicos patrones binarios invariantes a la rotación $LBP_{8,R}^{ri}$ que se pueden producir en un conjunto de 8 píxeles vecinos distribuidos simétricamente en un círculo de radio R. Los puntos negros y blancos corresponden a los valores de bit 0 y 1 en la salida de 8 bits del operador. La primera fila contiene los nueve patrones “uniformes” y el número interior corresponde a su código único de $LBP_{8,R}^{riu^2}$ [Ojala et al-2002].

A modo de ejemplo, en la figura 3.3, se muestran algunas de las primitivas de texturas detectadas con $LBP_{8,R}^{ri}$. El patrón #0 detecta puntos brillantes, el #8 puntos oscuros y áreas planas, el #6 extremos de líneas, el #4 bordes y el #5 esquinas.

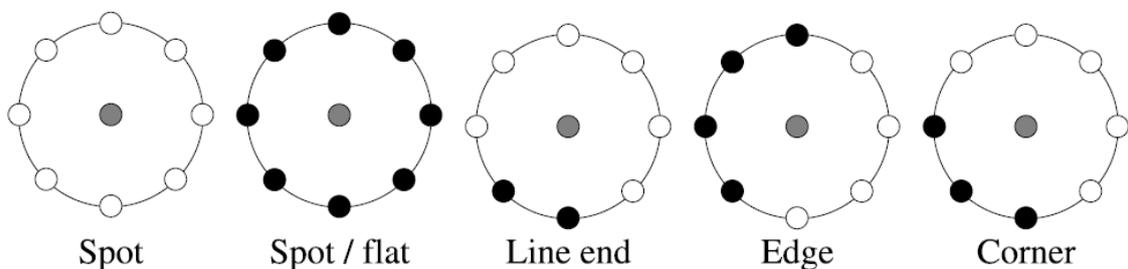


Figura 3.4. Diferentes primitivas de texturas detectadas por LBP. [M.Pietikäinen et al-2011]

En [Ojala et al-2000], los autores observan que ciertos patrones binarios locales aportan, la gran mayoría, en algunos casos hasta el 90% de todos los patrones de 3x3 que había en las texturas analizadas. A estos patrones, les denominaron “uniformes” y para definirlos formalmente introdujeron una medida de uniformidad $U(\text{“patrón”})$, la cual corresponde al número de transiciones espaciales (cambios bit a bit 0/1) en el “patrón”. Designando a los patrones que tienen un valor U menor o igual a 2, como uniformes,

propusieron el siguiente operador para la descripción del invariante de textura y rotación de la escala de grises en lugar del $LBP_{P,R}^{ri}$.

$$LBP_{P,R}^{riu2} = \begin{cases} \frac{\sum_{p=0}^{P-1} s(g_p - g_c)}{P + 1} & \text{si } U(LBP_{P,R}) \leq 2 \\ & \text{otherwise} \end{cases} \quad (9)$$

dónde:

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)| \quad (10)$$

El superíndice *riu2* refleja el uso de patrones “uniformes menor o igual a 2” invariantes a la rotación. Exactamente se pueden dar $P+1$ patrones binarios “uniformes” en un conjunto de P píxeles vecinos distribuidos simétricamente en un círculo de radio R . En la figura 3.1.3., las etiquetas de los $8+1$ patrones posibles para un radio $R=8$, están indicadas en el interior de cada uno de ellos. (0, 1,...8).

Muchos de los trabajos que emplean LBP para el análisis de texturas utilizan $LBP_{P,R}^{riu2}$, o sea los 9 patrones uniformes en el caso de 8 píxeles como hemos visto en la figura 3.3. En nuestro caso, no necesitamos clasificar un grupo de píxeles como en la identificación de texturas, sino a cada uno de los píxeles individualmente, por lo que utilizaremos todos los patrones binarios invariantes a la rotación que se pueden obtener para el conjunto de puntos, P , con el que estemos trabajando. Esto quiere decir que emplearemos el operador con *ri* en lugar de *riu2* para el cálculo del LBP y sus variantes.

3.1.2 Variantes de LBP.

El éxito obtenido por los métodos LBP en diversos problemas y aplicaciones de visión por ordenador ha inspirado nuevas investigaciones en base a él, que han dado lugar a diversas variantes. A ello ha contribuido la flexibilidad del método LBP, ya que es fácilmente modificable para adaptarlo a las necesidades del problema que se intenta resolver. Vamos a ver a continuación dos de estas variantes del operador LBP, que son el LTP

(*Local Ternary Pattern*) y el CLBP (*Complete Local Binary Pattern*), y como se han combinado para extraer las características de las imágenes de retina, en un nuevo operador que hemos denominado CLBPLTP.

3.1.2.1. Local Ternary Pattern. LTP.

Se sabe que el operador LBP es sensible al ruido en regiones de las imágenes que son prácticamente uniformes. Tan y Triggs, [Tan&Triggs-2010], introducen para tratar de evitar esto, un operador al que denominan, patrón ternario local, *Local Ternary Pattern* (LTP), que es más discriminante y menos sensible al ruido que el original LBP. Fundamentan que esta sensibilidad al ruido se debe principalmente a que en el LBP, el umbral es justo el valor del píxel central g_c , e introducen un umbral δ , que fijará el usuario, de tal manera que a los píxeles alrededor del píxel central que tengan un valor de $g_c \pm \delta$ se les asigna el valor 0, a los que estén por encima el valor 1, y a los que estén por debajo el valor -1. Es decir, la ecuación (6), en la que $s(x)$ representaba a $s(g_p - g_c)$ se transforma en:

$$s'(g_p, g_c, \delta) = \begin{cases} 1, & g_p \geq g_c + \delta \\ 0, & |g_p - g_c| < \delta \\ -1, & g_p \leq g_c - \delta \end{cases} \quad (11)$$

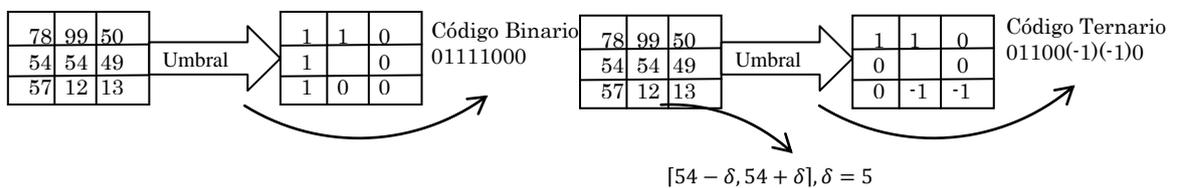


Figura 3.5. Ejemplo del cálculo de los operadores LBP y LTP con $\delta = 5$. (Tan & Triggs. 2010)

Una vez asignados los valores correspondientes a cada uno de los píxeles alrededor del píxel central, en base a (11), ver figura 3.5, podemos generar

dos conjuntos de píxeles, geoméricamente iguales al conjunto original, asignando el valor 1 en el primero de ellos a los píxeles que tenían valor 1 y al resto el valor cero, y en el segundo asignando el valor 1 a los píxeles que tuvieran valor -1 y al resto el valor 0. Así disponemos de dos conjuntos de píxeles a partir de los cuales calculamos el valor LBP de la misma forma que la descrita en la figura 3.2. Es decir que para cada píxel de la imagen, y en función del umbral elegido δ , obtenemos dos valores de LBP, como se aprecia en la figura 3.6.

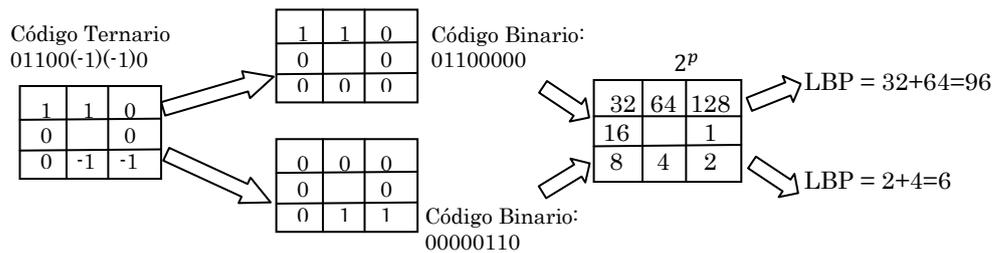


Figura 3.6. División de un código LTP en dos códigos LBP, positivo y negativo.

A modo de ejemplo, en la figura 3.7., se muestran los dos componentes LTP positivo y LTP negativo resultantes de aplicar el operador LTP al canal verde de una imagen de fondo de ojo.

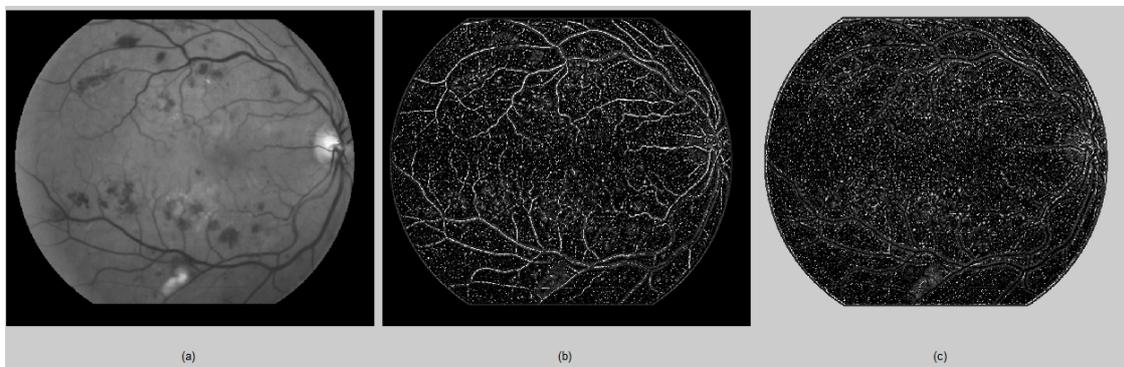


Figura 3.7. Resultado de aplicar el operador LTP al canal verde de una imagen de fondo de ojo. (a) Imagen original; (b) LTP signo positivo; (c) LTP signo negativo. $P=8$, $R=2$, $\delta=2$. (Imagen 10 de la base de datos de imágenes de retina DRIVE)

3.1.2.2. LBP Completo. (CLBP)

En [Guo et al-2010] se propone un nuevo operador con el fin de generalizar y completar el LBP, al que denomina LBP Completo (CLBP).

En la ecuación (7) correspondiente al LBP, calculábamos para cada punto alrededor del píxel central, el valor $s(g_p - g_c)$, considerando el resultado como 1 si la diferencia era positiva o cero en caso contrario. Esa diferencia, $d_p = g_p - g_c$, entre el nivel de gris del píxel central y los píxeles vecinos que le rodean se puede también expresar como una combinación de signo y magnitud:

$$d_p = s_p * m_p, \text{ siendo } \begin{cases} s_p = \text{signo}(d_p) \\ m_p = |d_p| \end{cases} \quad (12)$$

donde $s_p = \begin{cases} 1, & \text{si } d_p \geq 0 \\ -1, & \text{si } d_p < 0 \end{cases}$ es el signo de d_p y m_p es la magnitud en valor absoluto de d_p . En la figura 3.8., se muestra un ejemplo del cálculo de las componentes de signo y magnitud para un bloque de píxeles de tamaño 3x3.

9	12	34	-16	-13	9
10	25	28	-15		3
99	64	56	74	39	31
(a)			(b)		
-1	-1	1	16	13	9
-1		1	15		3
1	1	1	74	39	31
(c)			(d)		

Figura 3.8. (a) bloque ejemplo; (b) Diferencias locales; (c) Las componentes de signo y (d) magnitud. [Guo et al-2010]

A la expresión (12) en [Guo et al, 2010] la han denominado la transformada local de diferencias de signo-magnitud (LDSMT), y proponen representar cada bloque de píxeles de la imagen a partir del valor de su píxel central y la LDSMT. Para ello definen tres operadores a los que denomina CLBP_Signo (CLBP_S), CLBP_Magnitud (CLBP_M) y CLBP_Centro (CLBP_C). Vamos a ver a continuación como se calculan.

El operador de signo, CLBP_S, coincide con el LBP, y por tanto se obtiene al igual que éste mediante la ecuación (7). Es decir:

$$CLBP_S_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p, \quad \text{donde } s(x) = \begin{cases} 1, & \text{si } x \geq 0 \\ 0, & \text{si } x < 0 \end{cases} \quad (13)$$

Inspirados en el modo de calcular el LBP, para calcular el segundo componente, las diferencias de magnitud m_p (CLBP_M), definen el siguiente operador:

$$CLBP_M_{P,R} = \sum_{p=0}^{P-1} t(m_p, c)2^p, \quad \text{donde } t(x, c) = \begin{cases} 1, & \text{si } x \geq c \\ 0, & \text{si } x < c \end{cases} \quad (14)$$

en la que c se determina adaptativamente. En su artículo, [Guo et al-2010], la fijan como el valor medio de los valores de m_p calculados para todos los píxeles de los bloques de p píxeles de la imagen.

El tercer operador, basado en el valor de los píxeles centrales de cada bloque de la imagen, g_c , representa el nivel de gris de la imagen, y también aporta información discriminante. Para hacerle consistente con las componentes CLBP_S y CLBP_M, se codifica como:

$$CLBP_C_{P,R} = t(g_c, C_I), \quad \text{donde } t(g_c, C_I) = \begin{cases} 1, & \text{si } g_c \geq C_I \\ 0, & \text{si } g_c < C_I \end{cases} \quad (15)$$

donde el umbral C_I se fija como la media del nivel de gris de la imagen completa.

A modo de ejemplo, la figura 3.9, muestra los tres componentes CPBP_S, CLBP_M y CLBP_C, resultantes de aplicar el operador CLBP al canal verde de una imagen de fondo de ojo.

3.1.2.3. CLBPLTP.

A partir de los operadores definidos anteriormente y teniendo en cuenta que algunos de sus componentes coinciden, hemos definido un nuevo operador, CLBPLTP, que tiene cuatro componentes. Los dos primeras están formados por el CLBP_S (que expresa lo mismo que el LBP), y el CLBP_M. Las otros dos los hemos derivado aplicando los conceptos de terna y umbral

δ del LTP al cálculo del CLBP_M. Así partiendo de (14) y aplicando (11) tenemos dos nuevos operadores CLBP_LTP_PM, magnitud positiva, y CLBP_LTP_NM, magnitud negativa.

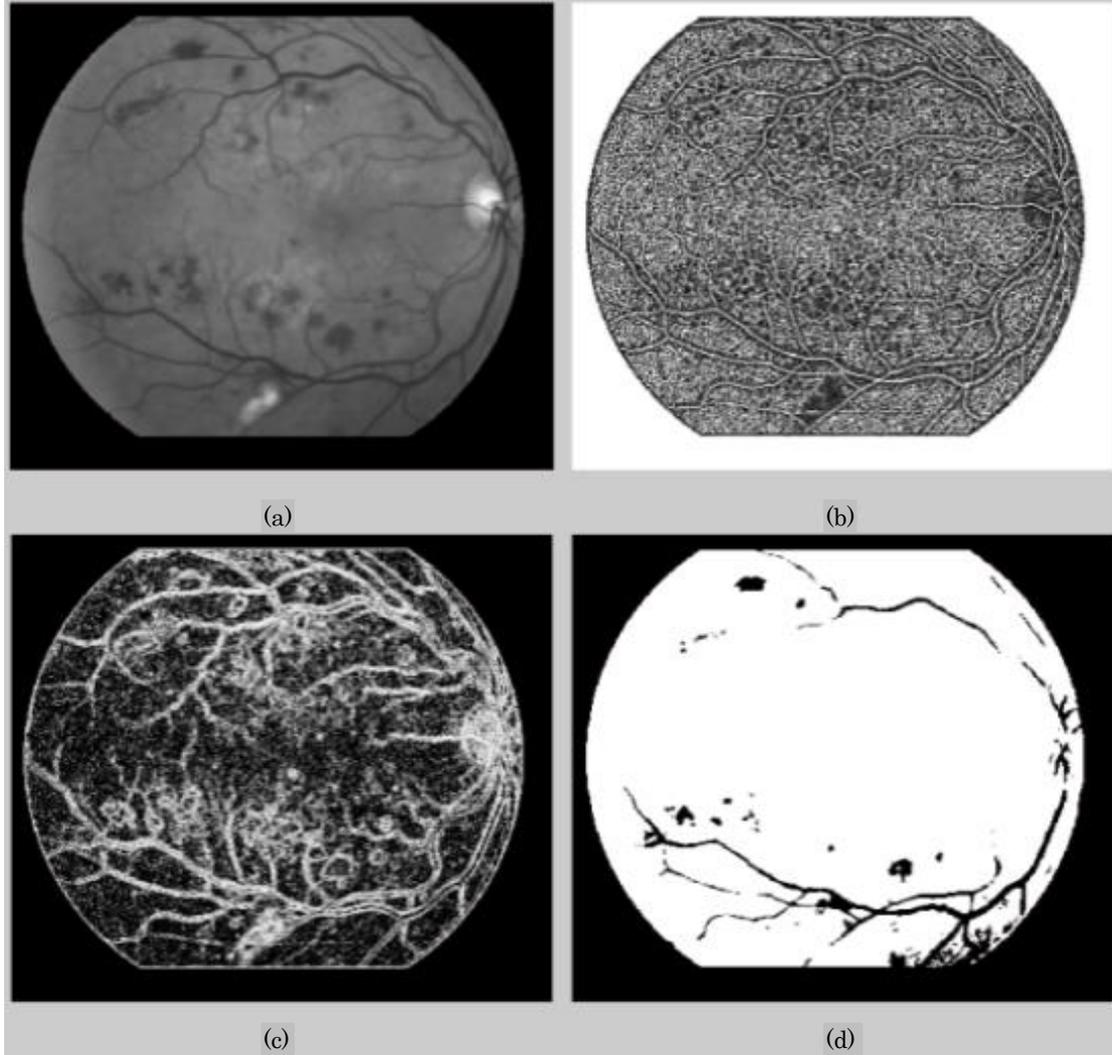


Figura 3.9. Resultado de aplicar CLBP al canal verde de una imagen de fondo de ojo. (a) Imagen original; (b) CLBP_S; (c) CLBP_M; (d) CLBP_C. P=8, R=2. (Imagen 10 de la base de datos de imágenes de retina DRIVE)

A modo de ejemplo, la figura 3.10, muestra el resultado de aplicar estos dos últimos operadores a la misma imagen de partida de la figura 3.9.

$$CLBP_{LTP_PM_{P,R}} = \sum_{p=0}^{P-1} t(m_p, c) 2^p, \text{ donde } t(x, c) = \begin{cases} 1, & \text{si } x \geq c + \delta \\ 0, & \text{si } x < c + \delta \end{cases} \quad (16)$$

$$CLBP_{LTP_NM_{P,R}} = \sum_{p=0}^{P-1} t(m_p, c) 2^p, \text{ donde } t(x, c) = \begin{cases} 1, & \text{si } x \leq c - \delta \\ 0, & \text{si } x > c - \delta \end{cases} \quad (17)$$

donde c se ha tomado finalmente como la media de las diferencias absolutas, $m_p = |d_p|$, entre cada píxel y el píxel central del bloque al que pertenece, calculadas para todos los píxeles, de todos los bloques de p píxeles de la imagen, y a δ se le ha asignado el valor 2.

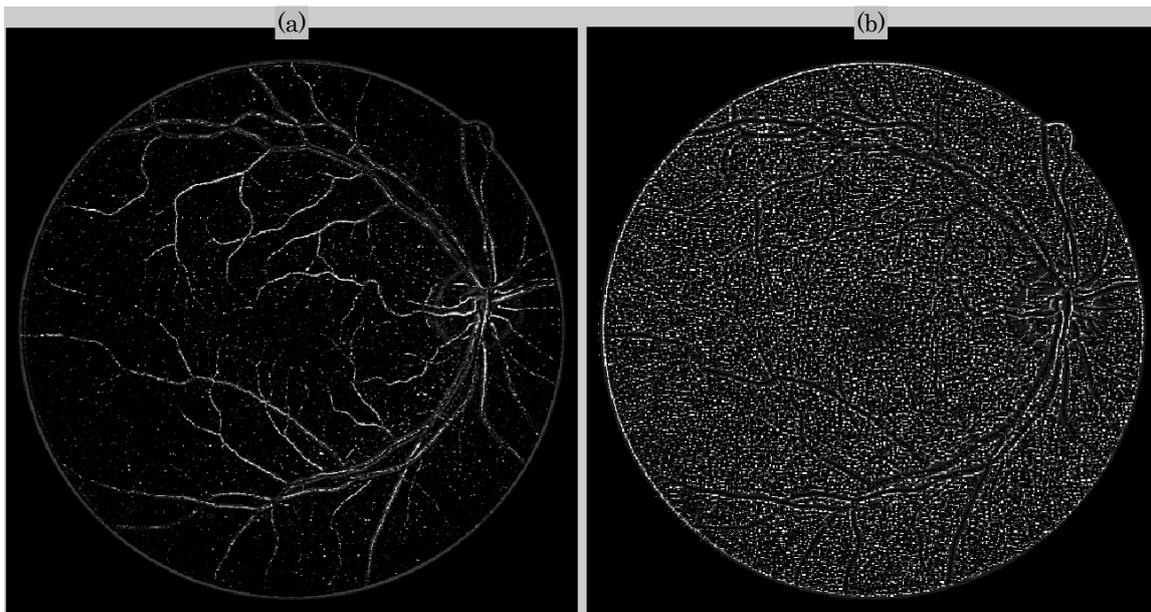


Figura 3.10. Los dos nuevos componentes de CLBPLBP obtenidos al aplicar el CLBPLTP al canal verde de una imagen de fondo de ojo. (a) CLBPLTP_PM ; (b) CLBPLTP_NM; P=8, R=2. (Imagen 10 de la base de datos de imágenes de retina DRIVE)

3.1.3. Aprendizaje de redes neuronales con Evolución Gramatical.

En esta sección vamos a ver en primer lugar en que consiste la Evolución Gramatical, para posteriormente centrarnos en su aplicación en la construcción y entrenamiento de una red neuronal.

3.1.3.1. Evolución Gramatical.

O'Neill y Ryan en su artículo *Grammatical Evolution*, [O'Neill&Ryan-2001], presentan la Evolución Gramatical (GE) como un algoritmo evolutivo, con el que es posible desarrollar un programa completo en cualquier lenguaje, utilizando un proceso de decodificación definido a partir de una gramática libre de contexto (CFG), expresada en la forma de Backus-Naur (BNF).

Todo algoritmo evolutivo es un algoritmo de optimización estocástico y se basa en la teoría de la evolución de los seres vivos. Se parte de una población inicial, que va evolucionando de generación en generación por medio de operadores de variación como la mutación y el cruce. La probabilidad de que un individuo de la población pase a la siguiente generación dependerá generalmente de su aptitud (*fitness*).

O'Neill y Ryan definen a cada individuo de la población como una cadena de bits aleatoria de una determinada longitud, inspirándose en el ADN de los seres vivos, que es la molécula que tiene codificada toda la información genética. Al igual que en el ADN, en el que las subunidades básicas son los nucleótidos, los cuales se agrupan para formar la unidad del código genético denominada codón, en GE se denomina codón a cada grupo de 8 bits. Así, cada codón podemos definirlo por un número entero $n \in \{0, 1, 2 \dots 255\}$ y se decodificará de acuerdo a alguna de las reglas definida por la gramática. Más adelante, en la figura 3.11, se muestra un ejemplo de cómo se realiza la decodificación de un individuo (cromosoma) de acuerdo a una gramática expresada en BNF. En base a esto la población estará formada por individuos, o cromosomas, que se representan por una secuencia de números enteros aleatorios, $i = (n_1, n_2, \dots, n_l)$, codones, con valores entre 0 y 255.

Como en todo algoritmo evolutivo, en GE el algoritmo que describe el proceso se puede escribir como:

1. Inicializar una población aleatoria de cromosomas,
2. Evaluar cada uno de los cromosomas,
3. Seleccionar el grupo de cromosomas que será incluido en la siguiente generación,
4. Crear la siguiente generación a partir del grupo seleccionado, aplicando los operadores genéticos de cruce y mutación
5. Si se ha alcanzado la condición de finalización (por ejemplo, el número límite de generaciones), terminar. En otro caso, volver al paso 3.

La selección de padres se realiza por el procedimiento denominado selección por torneo, en el cual se toman aleatoriamente N individuos de la población, de los cuales se eligen solamente 2 de acuerdo a su *fitness* que actuarán como padres en el proceso de cruce. Dado que cada individuo representará una RNA, la evaluación del *fitness* de cada individuo se realiza calculando la diferencia cuadrática entre la salida que produce la red que con él se genera y la salida deseada, es decir:

$$f = \sum_{i=1}^M (N_g(x_i) - y_i)^2 \quad (18)$$

siendo y_i la salida deseada para el píxel considerado, x_i el vector de características asociado a dicho píxel, $N_g(x_i)$ la salida de la red y M el número total de píxeles en el conjunto de entrenamiento.

El cruce se realiza de la forma más simple, que consiste en que una vez seleccionados dos cromosomas como padres, $i_1 = (n_1^1, n_2^1, \dots, n_l^1)$ y $i_2 = (n_1^2, n_2^2, \dots, n_l^2)$, si se supera la probabilidad de cruce establecida, se selecciona un número aleatorio $j \in \{1, 2, \dots, l-1\}$ y se generan dos nuevos individuos hi_1 y hi_2 :

$$hi_1 = (n_1^1, n_2^1, \dots, n_j^1, n_{j+1}^2, \dots, n_l^2)$$

$$hi_2 = (n_1^2, n_2^2, \dots, n_j^2, n_{j+1}^1, \dots, n_l^1)$$

El procedimiento para realizar la mutación puede ser simplemente cambiar cada uno de los elementos de un cromosoma a otro número aleatorio en el rango de $[0,255]$, siempre que se supere la tasa de mutación preestablecida, al seleccionar un número aleatorio entre $[0,1]$.

Para resolver un problema con GE, como se mencionó en el primer párrafo de esta sección, es necesario expresarlo mediante la notación Backus-Naur (BNF). En el siguiente apartado, 3.1.3.2, veremos cómo se lleva a cabo este proceso en el caso de utilizar la GE para la construcción y el entrenamiento de redes neuronales.

3.1.3.2. Construcción y entrenamiento de redes neuronales con Evolución Gramatical.

Se utilizará el paradigma, basado en GE expuesto en [Tsoulos et al-2005, Tsoulos et al-2008] para construir la topología de una red neuronal y aprender los pesos de la misma.

Una red neuronal, está formada por un conjunto de procesos elementales interconectados, conocidos como neuronas o nodos, de los cuales unos forman parte de la capa de entrada, y reciben las señales procedentes del exterior, y otros forman la capa de salida de la red. Una red de este tipo se puede expresar como un grafo dirigido, [Yao-1999], en el cual cada nodo i realiza una función de transferencia de la forma:

$$y_i = f_i \left(\sum_{j=1}^n w_{ij} x_j - \theta_i \right) \quad (19)$$

donde y_i es la salida que produce el nodo i , x_j es la j -ésima entrada al nodo, w_{ij} representa al peso sináptico entre los nodos i y j , y θ_i es el umbral o bias, del nodo i . En nuestro caso utilizaremos como función de transferencia la función sigmoidea $\sigma(x) = 1/(1 + e^{-x})$.

Si la red neuronal tiene una capa de entrada, una capa oculta y una de salida, por extensión de la expresión (19), se puede expresar como: [Tsoulos et al- 2008]

$$N(\vec{x}, \vec{w}) = \sum_{i=1}^H w_{(d+2)i-(d+1)} \sigma \left(\sum_{j=1}^d (w_{(d+2)i-(d+1)+j} x_j) + w_{(d+2)i} \right) \quad (20)$$

donde, d denota la dimensión del vector de entrada x y H es el número de neuronas de la capa oculta.

En la figura 3.11., se muestra una gramática libre de contexto expresada en BNF a partir de la cual se puede crear una red neuronal como la enunciada en (20). El símbolo no terminal <sigexpr> representa el resultado de la red y cada neurona está simbolizada por <Nodo>. La función sigmoidea se expresa con sig(x), y las entradas a la red son el símbolo no terminal <listaxx>. Los pesos sinápticos y el *bias* están representados por <número>.

Regla	Nº Regla
S ::= <sigexpr>	(0)
<sigexpr> ::= <Nodo> <Nodo> + <sigexpr>	(0) (1)
<Nodo> ::= <peso> * sig(<sum> + <bias>)	(0)
<sum> ::= <peso>*<listaxx> <sum>+<sum>	(0) (1)
<listaxx> ::= x1 x2 ... xn	(0) (1) ... (n-1)
<peso> ::= <número>	(0)
<bias> ::= <número>	(0)
<número> ::= (<lista_dígito>.<lista_dígito>) (-<lista_dígito>.<lista_dígito>)	(0) (1)
<lista_dígito> ::= <dígito> <dígito><lista_dígito>	(0) (1)
<dígito> ::= 0 1 ... 9	(0) (1) ... (9)

Figura 3.11. Gramática expresada en BNF para construir una red neuronal de dos capas. [Tsoulos et al-2008]

El proceso de generación de una red neuronal a partir de la gramática definida con BNF, sería el siguiente:

- Definir una cadena $x = (n_1, n_2, \dots, n_l)$ de l números enteros aleatorios (codones), formados por 8 bits, es decir $n_x \in \{0, 1, 2 \dots 255\}$
- Seleccionar el primer codón y calcular:

Regla = (valor codón) mod (número de reglas del no terminal actual)

- Continuar con el siguiente codón de x hasta obtener una expresión compuesta únicamente por símbolos terminales.

Supongamos que hemos generado una cadena compuesta por 200 codones. Puede suceder que antes de llegar al último, ya hayamos obtenido una expresión correspondiente a una red neuronal o que, por el contrario, estemos en medio de la resolución de una regla de la gramática. En este caso, también inspirándose en el solapamiento de genes que se produce en virus, bacterias y la mitocondria, y que les permite a estos organismos reutilizar el mismo material genético en la expresión de genes diferentes, O'neill y Ryan [O'neill&Ryan,2001] proponen una técnica denominada *wrapping* consistente en volver al principio de la cadena si al llegar al final de la misma, el proceso de decodificación no ha finalizado. El número de veces que se realiza el *wrapping* se fijará de antemano, y si se alcanza el límite sin que se haya conseguido tener una cadena completa compuesta sólo de símbolos terminales, marcaremos el individuo como inválido.

En la figura 3.12., se muestra un ejemplo del proceso de decodificación seguido por la Evolución Gramatical para la gramática definida en la figura 3.11.

La utilización de una red neuronal, desarrollada a partir de la Evolución Gramatical ya ha sido utilizada con éxito en trabajos como [Motsinger et al-2006, Motsinger et al-2008, Smart et al-2011, Tsoulos et al-2009, Hoover et al-2012].

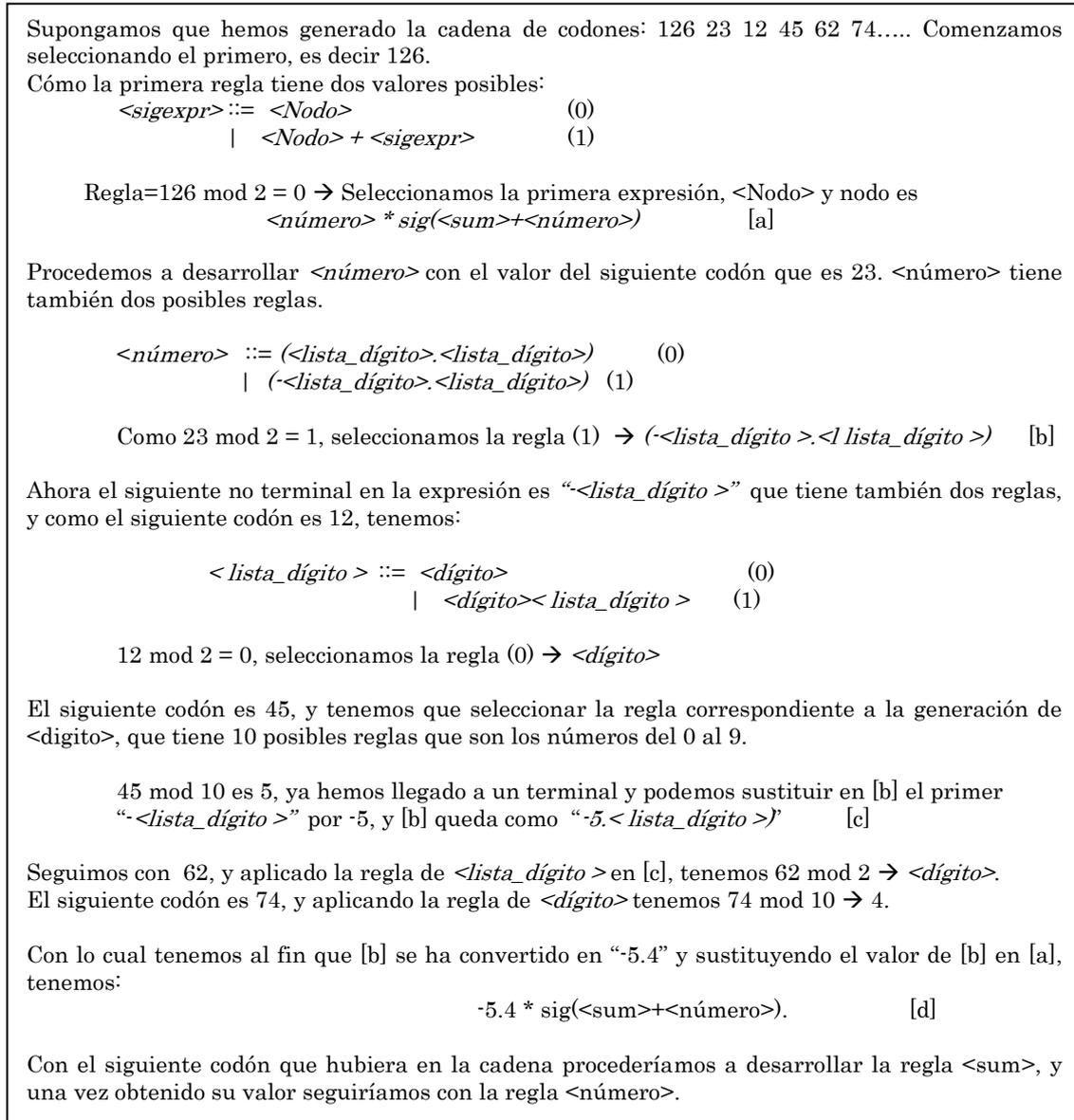


Figura 3.12. Ejemplo del procedimiento de decodificación seguido por la Evolución Gramatical para el BNF de una red Neuronal

3.2. Descripción del Método

La solución al problema, se divide en dos fases. La primera de ellas consiste en la obtención de dos redes neuronales artificiales, diseñadas y entrenadas por un generador de RNAs, desarrollado mediante Evolución Gramatical, que se van a encargar de clasificar los píxeles de las imágenes de fondo de ojo como pertenecientes o no a vasos sanguíneos. La primera

red, denominada RNA_V (red neuronal artificial de vasos), se centrará en la localización de píxeles de vasos sin imponer ninguna restricción en el grosor de los vasos a detectar y, la segunda red, denominada RNA_VF (red neuronal de vasos finos), se centrará únicamente en la detección de vasos cuyo grosor esté por debajo de un umbral. La segunda fase consiste en el modo de fusionar la salida de estas dos RNAs, para conseguir finalmente la segmentación de la red vascular completa de la retina, correspondiente a la imagen de fondo de ojo en cuestión. Todo el desarrollo se ha llevado a cabo con Matlab.

En las figuras 3.13., y 3.14., se presentan los esquemas de ambas fases y a continuación veremos cómo se implementa cada uno de sus pasos.

3.2.1. Fase I. Obtención de las redes neuronales: RNA_V y RNA_VF.

Describimos a continuación los diferentes pasos implicados en la obtención de las dos redes neuronales, RNA_V y RNA_VF, que utilizaremos posteriormente en la Fase II, para conseguir la segmentación de la red vascular de la retina.

1. Separación del canal G y eliminación de ruido.

Utilizamos el canal verde (G) para obtener las características CLBPLTP y LTP, ya que en las imágenes de fondo de ojo en formato RGB es el canal que proporciona el mayor contraste entre los vasos sanguíneos y el fondo. [A.Hoover et al., 2000]

Los dos primeros pasos, partiendo de las imágenes de fondo de ojo, en formato RGB, son separar los canales del espacio de color, y eliminar el ruido de la imagen. La obtención del canal verde (G), es muy sencilla, ya que una imagen en formato RGB no es más que una matriz de tres dimensiones en la cual cada elemento representa a un

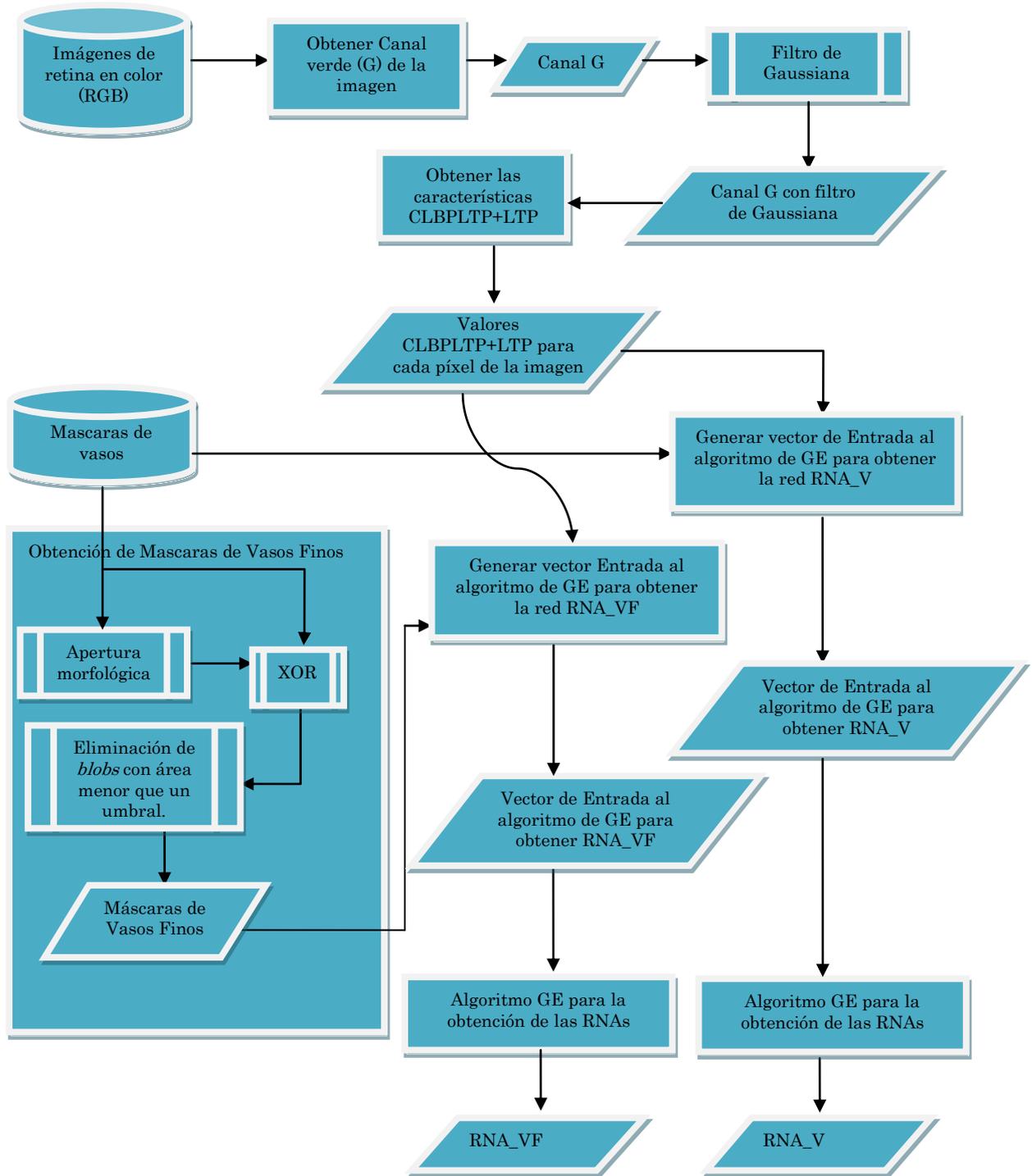


Figura 3.13. Fase I. Arquitectura para la construcción y aprendizaje de las dos redes neuronales (RNA_V y RNA_VF)

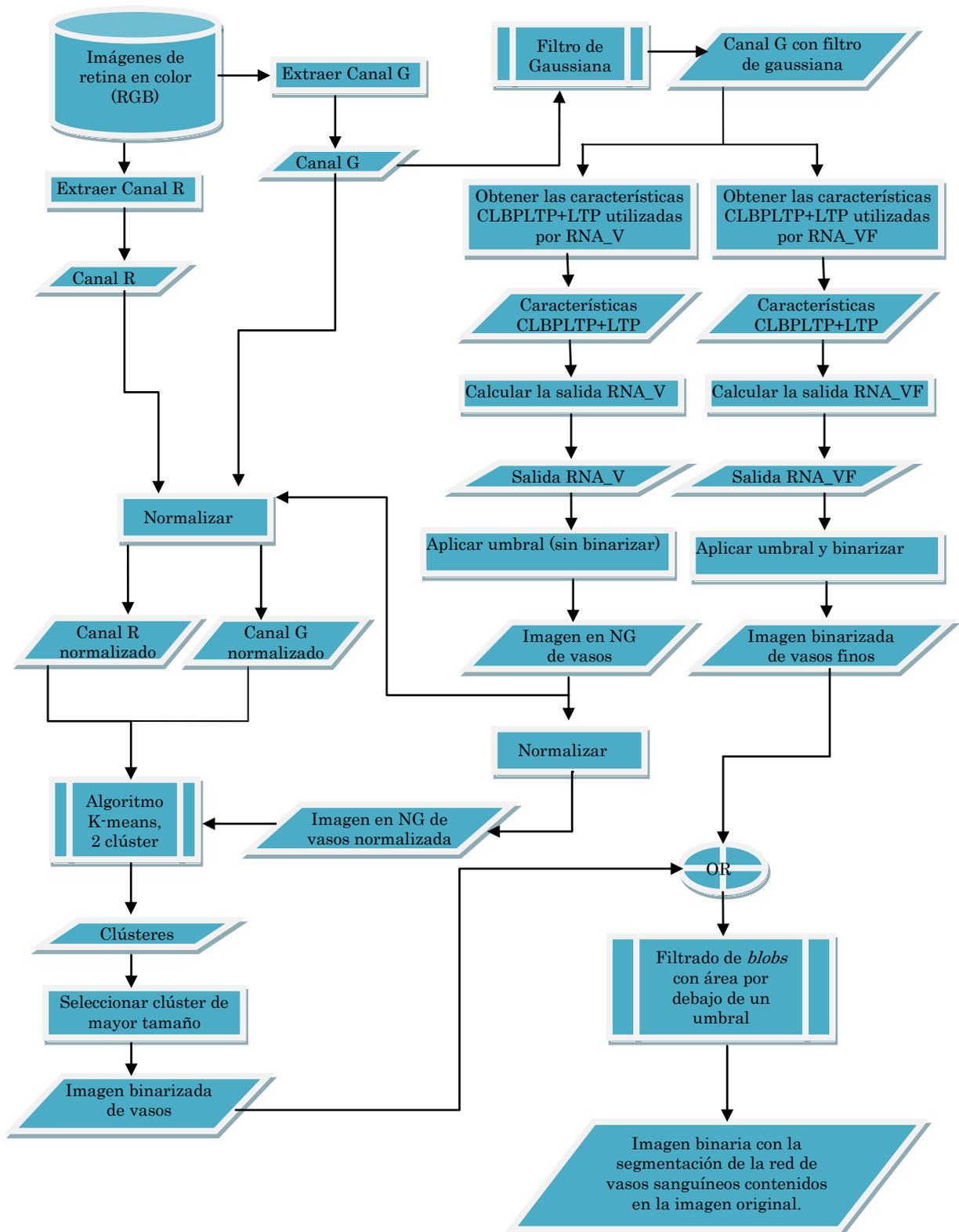


Figura 3.14. Fase II. Arquitectura para la fusión de las dos redes, RNA_V y RNA_VF, y obtener así la segmentación final (imagen binaria) de la red de vasos sanguíneos.

píxel en una fila y columna con los valores de cada canal o componente del espacio de color RGB.

La dimensión de las imágenes de la base de datos de imágenes de retina DRIVE es de $565 \times 584 \times 3$, y el canal G es la componente número dos de la matriz tridimensional, que es en definitiva una imagen en escala de grises.

Una vez separado el canal G, aplicamos un filtro de paso bajo, de tipo gaussiano, que sirve tanto para eliminar ruido de la imagen, producido por ejemplo en el proceso de digitalización de la misma, como para resaltar la información correspondiente a una determinada escala, en este caso el tamaño de la matriz de filtrado.

El filtro gaussiano consiste básicamente en que a cada píxel se le asigna el valor resultante de promediar con distintos pesos los valores de sus píxeles vecinos. El valor máximo aparece en el píxel central y disminuye hacia los extremos tanto más rápido cuanto menor sea el parámetro de desviación típica.

En nuestro caso hemos utilizado un filtro paso bajo gaussiano con simetría de rotación y tamaño de matriz de filtrado de 3×3 con un valor de desviación típica de 0.5.

2. Obtención de las máscaras de vasos finos.

En la base de datos de imágenes de retina DRIVE, como vimos en el apartado 2.2.1., están disponibles las máscaras denominadas “*gold standard*”. En principio el método se desarrolló en base a ellas, pero para intentar aumentar la precisión en la detección de vasos finos, se decidió entrenar una segunda red neuronal con máscaras generadas solamente a partir de los vasos con diámetro máximo de dos píxeles en cada imagen. Vamos a ver a continuación como se obtienen estas máscaras.

En primer lugar aplicamos a las máscaras “*gold standard*”, una apertura morfológica. La apertura morfológica consiste en que todos los objetos de la imagen, que no estén completamente contenidos en el elemento estructurante utilizado, son eliminados. En nuestro caso el elemento estructurante utilizado es un disco de 2 píxeles de diámetro. Una vez aplicada la apertura morfológica, aplicamos el operador lógico XOR entre la imagen resultante y la imagen original, y así obtenemos justamente todos aquellos vasos cuyo diámetro no sea mayor de dos píxeles. Pero, tal y como se aprecia en la figura 3.15., después de aplicar la apertura y realizar el XOR, pueden obtenerse algunos grupos de píxeles conectados entre sí y aislados (*blobs*), que no pertenecían en origen a un vaso fino, sino que forman parte de un vaso grueso. Obsérvese, por ejemplo, los grupos de puntos en el cuadrante inferior izquierdo de la imagen etiquetada como XOR. Estos grupos de píxeles se pueden eliminar de la imagen aplicando una función de apertura de zona, que consiste en eliminar todos aquellos objetos cuya área esté por debajo de un cierto umbral, y que además tengan menos de un determinado número de conexiones con sus píxeles vecinos. En nuestro caso se ha aplicado la apertura de zona para eliminar aquellos objetos que contienen menos de dos píxeles y tienen menos de 4 conexiones con sus píxeles vecinos. A continuación, en la figura 3.15 se muestra el proceso de obtención de las máscaras de vasos finos a partir de la máscara “*gold standard*”.

3. Cálculo de los operadores LBP y sus variantes para generar el vector de características.

Después de las pruebas realizadas con las diversas variantes del LBP, finalmente se han elegido para el método, al haber producido los mejores resultados en la segmentación de vasos, la variante LTP y la nueva variante que hemos desarrollado, denominada CLBPLTP, descritas ambas en el apartado 3.1.

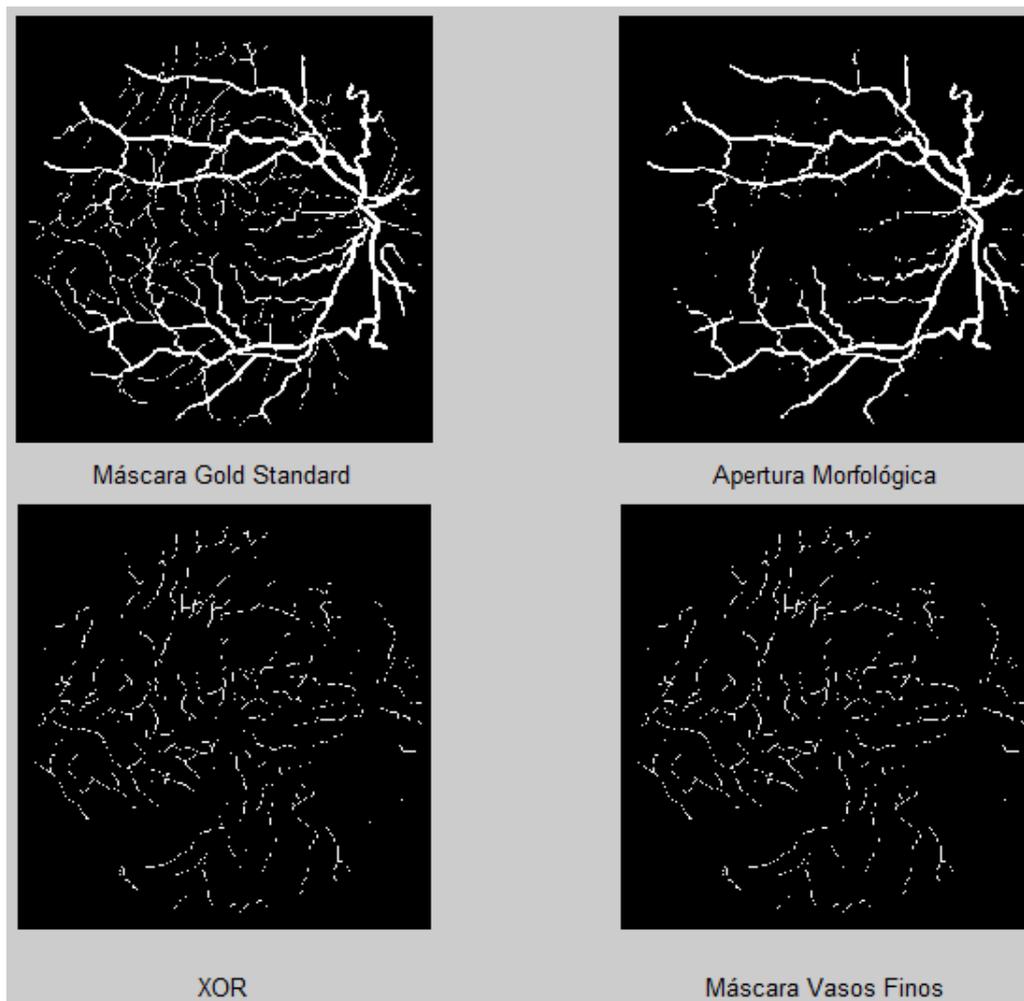


Figura 3.15. Proceso obtención de máscara de vasos finos para la imagen 22 de la base de datos de imágenes de retina DRIVE.

Teniendo en cuenta que el grosor de los vasos sanguíneos oscila entre un píxel y nueve píxeles, para la obtención de la primera red neuronal, RNA_V, se experimentó con las siguientes combinaciones de radios y puntos:

$$(R,P)=\{(1,8) (2,8) (3,12) (4,12) (5,16) (6,16) (7,24) (8,24), (9,24)\}.$$

Pero después de varios ensayos, se decidió utilizar en todos los casos un número de puntos igual a 24, para que, en el caso de radios pequeños, las direcciones del haz de muestreo fueran lo suficientemente numerosas como para poder muestrear, con cierta garantía, todas las direcciones en las que pudiera existir un vaso fino

en la imagen. Por ello finalmente, las características utilizadas para definir el vector de entrada al algoritmo de evolución gramatical, que genera y entrena la red RNA_V, han sido:

- a. LTP, con valor de $\delta=2$, para las combinaciones de puntos y radios:

$$(R,P)=\{(1,24) (2,24) (3,24) (4,24) (5,24) (6,24) (7,24) (8,24), (9,24)\}.$$

Lo que nos proporciona un total de 18 características para cada píxel, al obtener dos valores en cada combinación de operador LTP.

- b. CLBPLTP [CLBP_S, CLBP_M, CLBPLTP_PM ($\delta=2$) y CLBPLTP_NM ($\delta=2$)], también para las combinaciones de puntos y radios:

$$(R,P)=\{(1,24) (2,24) (3,24) (4,24) (5,24) (6,24) (7,24) (8,24), (9,24)\}.$$

En este caso obtenemos un total de 36 características para cada píxel, al obtener cuatro valores por cada una de las combinaciones de radio y puntos para el operador CLBPLTP.

Así pues, disponemos de 54 características ($2 * 9 = 18$ de tipo LTP más $4 * 9 = 36$ de tipo CLBPLTP) de cada uno de los píxeles de las imágenes, a partir de las cuales generaremos el vector de entrada a la RNA. Para generar dicho vector, se toman todos los píxeles etiquetados como vasos en las máscaras “*gold standard*”, del conjunto de imágenes de entrenamiento, de la base de datos de imágenes de retina DRIVE, el cual está compuesto por 20 imágenes, en concreto las etiquetadas con los números del 21 al 40 y un número igual de píxeles, distribuidos aleatoriamente en la retina y que no estén etiquetados como vasos. Con ambos conjuntos de píxeles, generamos una matriz bidimensional, cuyo número de filas es el número de

píxeles seleccionados y con 55 columnas. Las primeras 54 columnas son las características LBP obtenidas, y a la columna número 55 se le asigna el valor 1, si el píxel pertenece a la red de vasos sanguíneos, o el valor cero en caso contrario.

El vector de entrada al algoritmo para la obtención de la red que segmenta los vasos finos, RED_VF, se obtiene del mismo modo, pero ahora las máscaras utilizadas son las de vasos finos, generadas en el paso número 2. En este caso, solamente utilizamos las 18 características ($2 * 3 = 6$ de tipo LTP más $4 * 3 = 12$ de tipo CLBPLTP) obtenidas de calcular LTP y CLBPLTP para las combinaciones de puntos y radios siguientes:

$$(R,P)=\{(1,24) (2,24),(3,24)\}$$

Utilizamos solamente radios de tamaño 1, 2 y 3, ya que ahora estamos trabajando con vasos de un grosor máximo de 2 píxeles.

4. Implementación del algoritmo de Evolución Gramatical para la obtención de la red neuronal artificial de detección de vasos.

Se ha implementado un algoritmo de generación de redes neuronales artificiales, basado en Evolución Gramatical, en base a lo expuesto en el punto 3.1.3.

Los parámetros de ejecución del algoritmo evolutivo de generación de las redes neuronales han sido los siguientes:

Número de codones: 200

Número de individuos: 500

Número máximo de generaciones: 1500

Tipo de selección de Padres: Torneo de tres individuos.

Tipo de selección de supervivientes: Modelo *Steady State*

Tasa de mutación: 0.08.

Tasa de cruce: 1.00

Elitismo: Si

Porcentaje de la población que se conserva para la siguiente generación: 0.002

Límite de *wrapping*: 2

3.2.2. Fase II. Fusión de la salida de las RNAs (RNA_V y RNA_VF)

En esta fase vamos a ver como se fusiona la salida de las dos RNAs obtenidas en la fase anterior para, finalmente, segmentar la red vascular en cualquier imagen de fondo de ojo. Los pasos son los siguientes:

1. **Separamos los canales G y R de la imagen de entrada y eliminamos el ruido en el canal G.**

La separación de canales y la eliminación de ruido aplicando un filtro de gaussiana al canal G, se realizan aquí del mismo modo que en el punto 1 de la fase I.

El canal R es la primera de las tres dimensiones de la matriz que conforma el espacio de color RGB, y como ya vimos el G es la segunda.

2. **Obtención de las características CLBPLTP y LTP.**

Con el canal G obtenido en el paso precedente, se calculan las características de los operadores CLBPLTP y LTP que utilizan las dos redes neuronales. Veremos en el capítulo de evaluación de los resultados, que la RNA_V utilizará sólo cuatro características (de las 54 disponibles) y la RNA_VF utilizará tan solo 2 (de las 18 posibles).

3. Calcular las salidas de las dos redes, RNA_V y RNA_VF.

Obtenemos la salida de las dos redes neuronales, utilizando como entrada para cada una de ellas, las características calculadas en el paso 2. El resultado son dos imágenes de niveles de gris (NG) como, por ejemplo, las que se muestran en la figura 3.16.

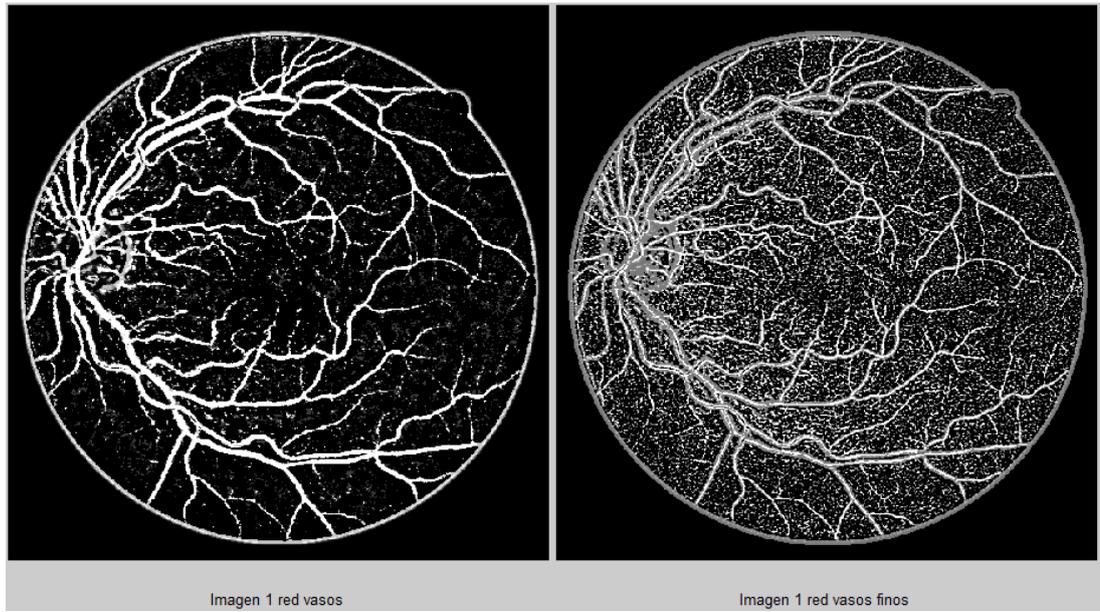


Figura 3.16. Resultados de aplicar las dos redes neuronales de segmentación de vasos, RNA_V y RNA_VF, al canal G de la imagen 1 de la base de datos de imágenes de retina DRIVE.

4. Aplicar umbrales de discriminación

Como las redes han clasificado cada píxel de la imagen asignándoles un valor comprendido entre 0 y 1, tenemos que fijar un umbral en cada una de ellas, a partir del cual un píxel será considerado como vaso. En primera aproximación, en la red RNA_V, los píxeles con valor ≥ 0.8 , y en la red RNA_VF, los píxeles con valor ≥ 0.76 , serán considerados píxeles pertenecientes a vasos y el resto no. Así pues, asignamos el valor cero a todos los píxeles en ambas máscaras que no superen estos umbrales. Además, en el caso de la red de vasos finos, a todos aquellos puntos que sí superan el umbral les asignamos el valor 1, con lo cual tenemos una máscara binaria. Este proceso de “binarizado”, no lo hacemos con la salida obtenida con

la RNA_V, porque se va a utilizar todo el rango de valores disponible, por encima del umbral, en los procesos siguientes. A modo de ejemplo, en la figura 3.17, se muestran los resultados de aplicar estos umbrales a la misma imagen de la figura 3.16.

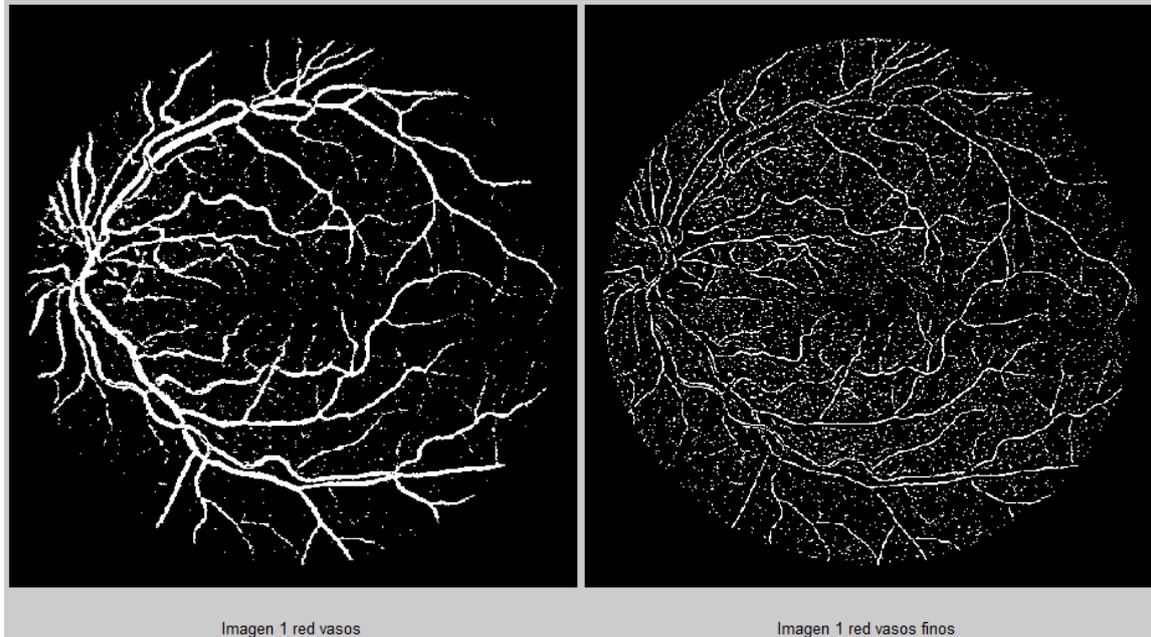


Figura 3.17. Máscaras obtenidas al aplicar los valores umbral, a los resultados obtenidos de aplicar las dos redes neuronales de segmentación de vasos, al canal G de la imagen 1 de la base de datos de imágenes de retina DRIVE.

5. Normalización Canales R, G, y la salida de la RNA_V.

Ahora, normalizamos la salida de la red RNA_V umbralizada, a la que vamos a denominar A_{vasos} . Para ello, calculamos para cada píxel x_i el valor:

$$\hat{x}_i = \frac{x_i - \bar{x}}{\sigma} \quad (21)$$

En el cálculo de los valores de la media (\bar{x}) y de la desviación típica (σ), en la ecuación (21), sólo se tienen en cuenta los píxeles que tengan un valor distinto de cero en A_{vasos} . $\hat{x}_{A_{vasos}}$ es la salida normalizada de la red.

Seguidamente, vamos a seleccionar en los canales R y G, el conjunto de todos los píxeles que en A_{vasos} tienen un valor distinto de

cero, y normalizar estos dos conjuntos por medio de la ecuación (21). Así, obtenemos \hat{x}_R para el canal R, y \hat{x}_G para el canal G. Como los valores de los píxeles de A_{vasos} oscilan entre el umbral aplicado y 1, y los de los canales R y G están comprendidos en el intervalo [0,255], normalizamos de nuevo $\hat{x}_{A_{vasos}}$, \hat{x}_R y \hat{x}_G , ahora en el intervalo [0,255], por medio de la siguiente ecuación:

$$\hat{x}_i^{[0\ 255]} = \frac{(\hat{x}_i - \min(\hat{x})) \times 255}{\max(\hat{x}) - \min(\hat{x})} \quad (22)$$

6. Eliminación de ruido de salida de la RNA_V mediante K-medias.

Con los canales R y G, y A_{vasos} normalizados en el intervalo [0 255], construimos una variable de tres dimensiones, que utilizaremos como entrada al algoritmo k medias. Este algoritmo es un algoritmo de aprendizaje no supervisado, que agrupa en un número prefijado de clústeres, k, el conjunto de datos de entrada, por aproximaciones sucesivas en función, por ejemplo, del cuadrado de las distancias euclídeas de cada objeto a su centroide más cercano. En principio los k centroides los elige aleatoriamente y va recalculando su posición en cada aproximación, hasta que ésta se mantenga estable. El número de clústeres que hemos utilizado es de dos. A modo de ejemplo, la figura 3.18 muestra los dos clústeres que nos devuelve el algoritmo k medias, para la variable que hemos construido con las tres componentes normalizadas, generada en el proceso de tratamiento de la imagen número 7 de la base de datos de imágenes de retina DRIVE. Al aplicar el algoritmo k-medias conseguimos eliminar algo del ruido asociado generalmente a la parte brillante de la papila, que en este caso vemos en la imagen de la derecha de la figura y que se corresponde con el clúster con menor número de puntos de los dos generados.



Figura 3.18. Clústeres producidos por el algoritmo K medias para la variable compuesta por los canales G y R, y la salida de la red umbralizada, normalizados en el intervalo [0 255] para la imagen 7 de la base de datos de imágenes de retina DRIVE.

7. Aplicar operador OR a la salida de RNA_VF y al clúster con más píxeles generado por K-medias, obtenido en el paso anterior.

Finalmente, aplicamos el operador OR al clúster de salida que tiene más puntos del algoritmo k medias y a la salida binarizada que obtuvimos al aplicar la red neuronal de vasos finos, RNA_VF, a la imagen, descrita en el punto 5 anterior. Con ello tenemos una máscara binaria como la de la figura 3.19, en la que se aprecia ya claramente la red de vasos sanguíneos, si bien se observa una gran cantidad de ruido en forma de grupos de puntos aislados, *blobs*, que procedemos a eliminar en el siguiente y último paso.

8. Eliminación de Blobs

Por último, sólo nos resta eliminar todos aquellos *blobs* cuya área no supere un determinado umbral. Para ello, al igual que en la

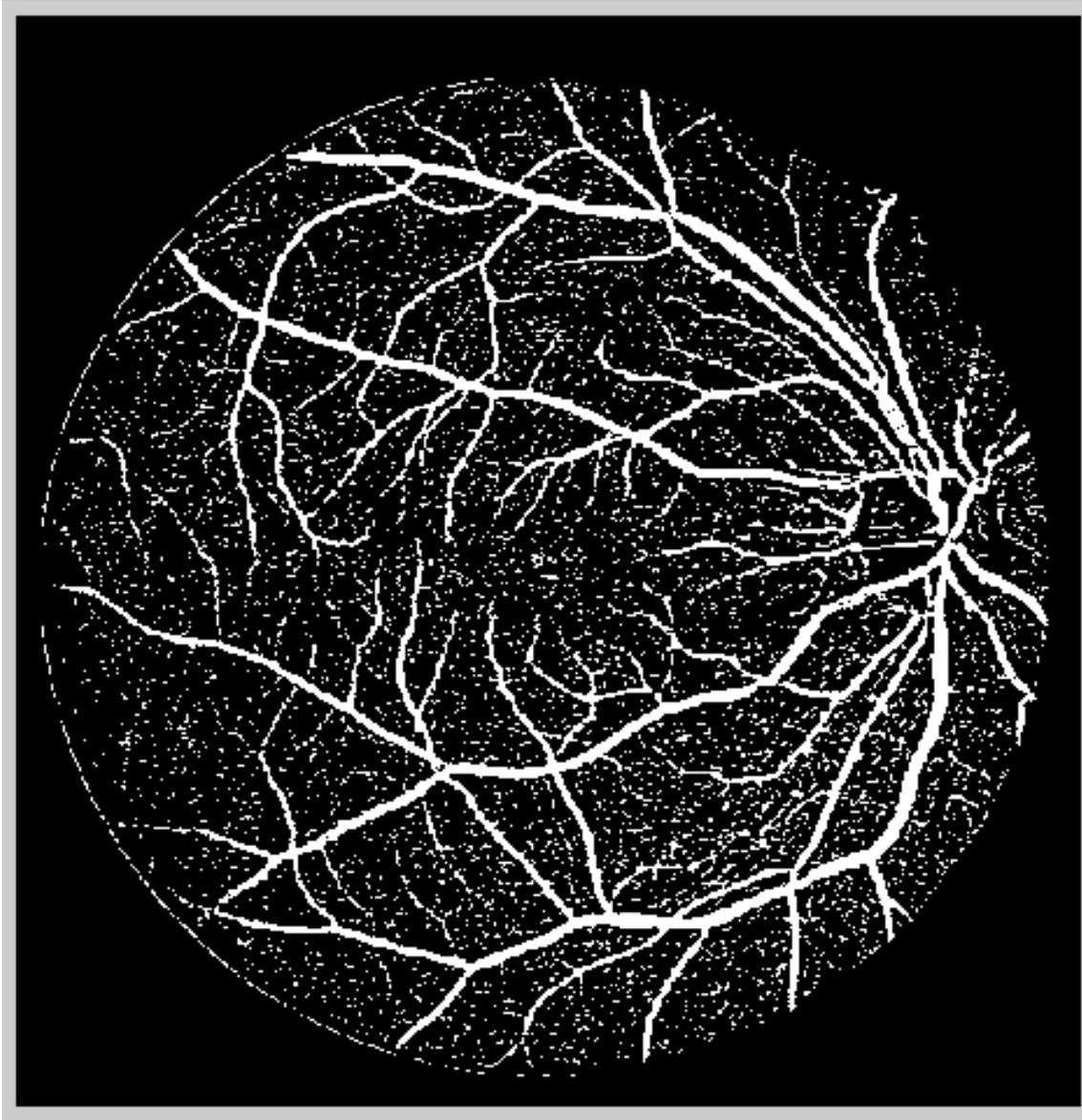


Figura 3.19. Resultado de aplicar el operador OR entre la salida del algoritmo k medias y la salida generada al aplicar la red de vasos finos a la imagen número 19 de la base de datos de imágenes de retina DRIVE.

obtención de la máscara de vasos finos, recurrimos a una apertura de zona. El umbral que ha producido mejores resultados es 30, y con él obtenemos la segmentación final. A modo de ejemplo, la figura 3.20 muestra la segmentación de la imagen 19 de la base de datos de imágenes de retina DRIVE.

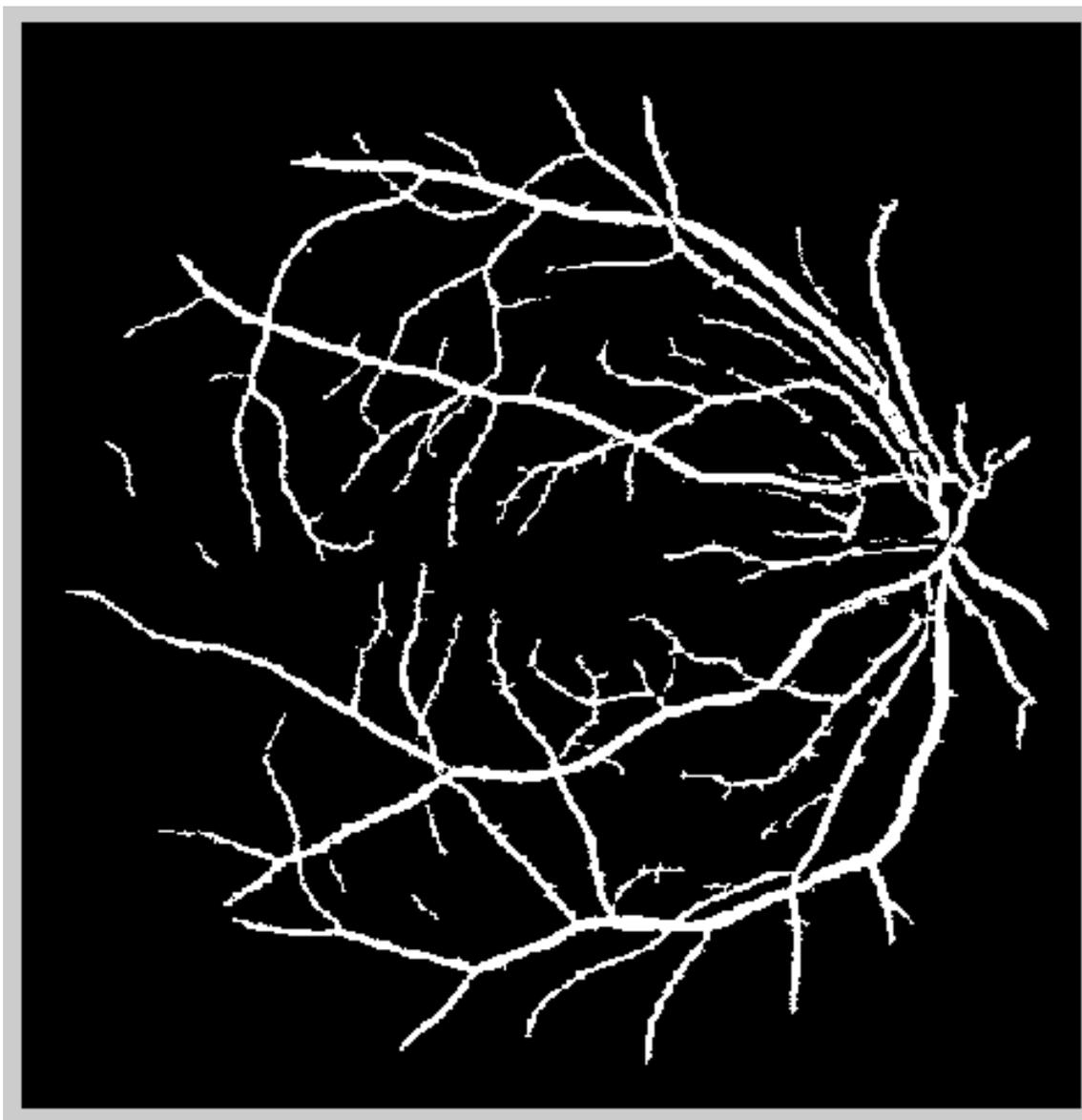


Figura 3.20. Segmentación resultante de aplicar el método propuesto a la imagen 19 de la base de datos de imágenes de retina DRIVE.

Capítulo 4.

Evaluación

4.1. Evaluación preliminar.

El objetivo de este capítulo es exponer la evaluación del método presentado en este trabajo, explicando en primer lugar, cómo se apreció la validez de los operadores LBP, para extraer las características de los píxeles de las imágenes de retina, que nos van a permitir clasificar estos píxeles como pertenecientes o no a vasos sanguíneos, y a continuación, cómo se validó el poder de clasificación y discriminación de estas características por parte del algoritmo generador de RNAs basado en Evolución Gramatical, para conseguir el objetivo final de segmentar la red vascular completa de la retina.

4.1.1. Estudio preliminar de la aplicabilidad de los operadores LBP a la segmentación de vasos en imágenes de retina.

En el apartado 3.1.1., vimos que los operadores LBP son operadores de texturas con una gran potencia discriminativa. Estos operadores han demostrado su eficacia en la identificación de texturas, incluso en la identificación de rostros, pero los vasos sanguíneos son estructuras geométricas complejas de las que necesitamos clasificar cada uno de sus píxeles con la mayor precisión posible. Por ello, decidimos realizar un estudio previo para verificar el nivel de discriminación que podía aportarnos el operador LBP. Para lo cual, se compararon los histogramas obtenidos al aplicar el operador $LBP_{P=8,R=6}^{riu2}$ de dos de las estructuras anatómicas más

características de las imágenes de retina, como son la red de vasos sanguíneos y el contorno papilar, con el resto de píxeles pertenecientes a la retina, representados por coronas elípticas de 4 píxeles de grosor, trazados alrededor del contorno papilar a distancias de 40 píxeles y 80 píxeles respectivamente. Ver figura 4.1. Para obtener estos histogramas se han utilizado las bases de datos de imágenes de retina DRIVE, para los vasos, y DRIONS (Digital Retinal Images for Optic Nerve Segmentation Database) para el contorno papilar y las coronas elípticas a su alrededor. La base de datos DRIONS está realizada por la UNED, el Hospital Clínico San Carlos y el Hospital Universitario Miguel Servet. [Carmona et al., 2008].

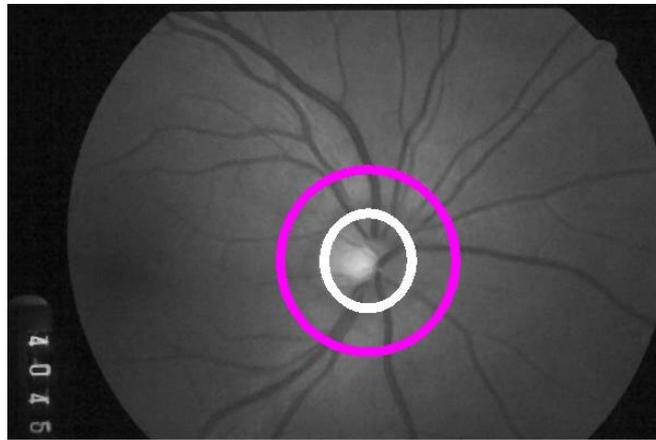


Figura 4.1. Corona elíptica alrededor del contorno papilar de 4 píxeles de grosor (blanca), y corona alejada 40 píxeles de la anterior (rosa). (Imagen tomada de la base de datos DRIONS)

El resultado se muestra en la figura 4.2. En ella vemos como se diferencian claramente los histogramas del contorno papilar y de los vasos sanguíneos, mientras que los de las dos coronas elípticas prácticamente coinciden, y así debe de ser, puesto que ambas coronas pertenecen a píxeles de la retina, alejados del contorno papilar, y que sólo podrían contener algún píxel espurio de vasos, por lo que no hay razón para que fueran diferentes.

El histograma para la corona que contiene el contorno papilar muestra dos patrones discriminantes diferentes, que son el #3 y el #4. Cabía esperar el patrón #4, ya que el contorno papilar separa una zona más oscura, la retina,

de una zona más clara, la papila. El que el segundo patrón más discriminante sea el #3 y no el #5, también es lógico dado que los puntos del contorno están situados en una elipse y la papila corresponde al área de dicha elipse, para un punto situado en el contorno papilar, el número de vecinos pertenecientes a la retina (zona oscura), siempre será igual (patrón #4), o mayor (patrón #3) que el número de vecinos perteneciente a la papila (zona clara). Sin embargo, en el histograma correspondiente a los vasos sanguíneos, se aprecia que los patrones LBP más discriminantes, con respecto a los demás histogramas, son el #7 y el #8. Esto se explica porque los vasos sanguíneos son estructuras más oscuras que el resto de la retina, y los puntos interiores de los vasos están rodeados por puntos con igual o menor intensidad de gris que ellos, (patrón #8), o tienen un único vecino que pertenece a la retina y por ello más claro, (patrón #7). No obstante, vemos que el histograma de vasos en todos los canales de color presenta el mayor número de ocurrencias en el patrón #9. Esto es debido a que por la estructura morfológica de los vasos, formas alargadas estrechas, bifurcaciones, etc., lo normal es que predominen los patrones con más de dos transiciones espaciales, (cambios bit a bit 0/1), lo que nos conduce a elegir, para nuestro trabajo, en lugar de $LBP_{P,R}^{riu2}$ que nos proporciona los 9 patrones “uniformes”, (#0 a #8), con sólo 2 transiciones espaciales, y engloba en el patrón #9 el resto, el operador $LBP_{P,R}^i$, que nos aporta los 36 patrones presentados en la figura 3.3.

Por otro lado, los histogramas de la figura 4.2, también nos facilitan información valiosa en cuanto a cuáles son los canales más discriminantes y en los que más se reflejan las diferencias entre los tres tipos de estructuras. Vemos que en este sentido destacan el canal Y, del espacio de color YCbCr, el G del espacio de color RGB y el H del espacio de color HSV.

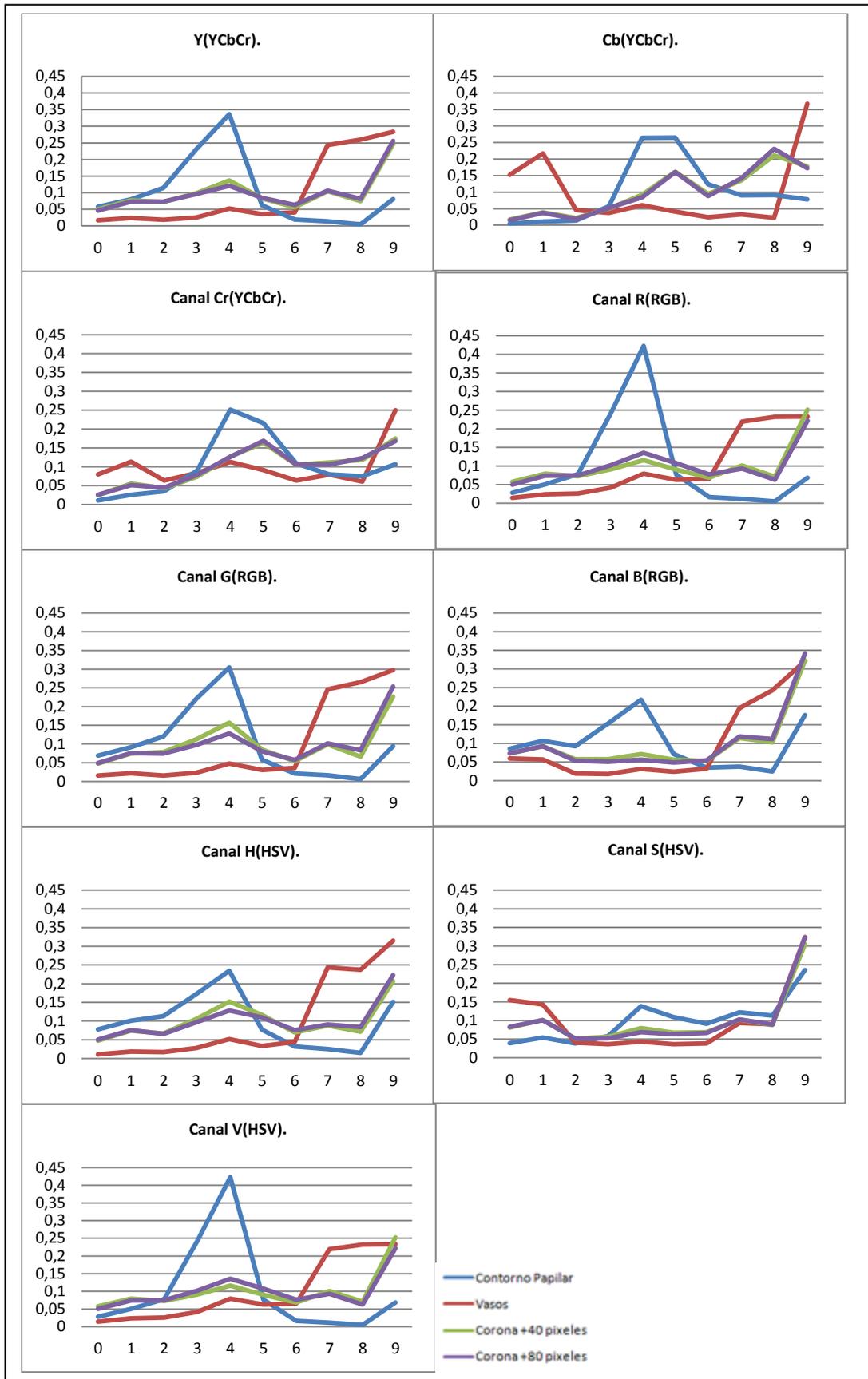


Figura 4.2. Histogramas de la red de vasos, el contorno papilar y coronas elípticas situadas alrededor del contorno papilar a una distancia de 40 y 80 píxeles, calculados para el operador $LBP_{P=8,R=6}^{riu2}$.

4.1.2. Estudio preliminar de la validez del algoritmo evolutivo de generación de RNAs mediante Evolución Gramatical.

En su artículo, [Gavrilis et al-2008], presentan los resultados obtenidos con su algoritmo de generación de redes neuronales con Evolución Gramatical frente a otros métodos sobre algunas de las bases de datos más utilizadas para medir el rendimiento de los clasificadores, como son la WINE, la PIMA, la WDBC o la Ionosphere, todas ellas disponibles en la página del UCI Machine Learning Repository².

Para probar la efectividad de nuestro algoritmo, también realizamos pruebas con estas bases de datos. Los mejores resultados para cada una de ellas después de 50 ejecuciones del algoritmo fueron los siguientes:

Para el WDBC (Winconsin Diagnostic Breast Cancer dataset), que contiene datos 32 atributos, para cada uno de los 569 pacientes que sirven para diagnosticar tumores de mama como M (malignos), o B (benignos), la mejor red neuronal obtenida está compuesta de cuatro neuronas, y ha sido la siguiente:

$$N(x) = 0.90 * \text{sigmoide}(3.1 * x_{11} + 88.21 * x_{27} - 5.782 * x_0 + 0.1 * x_{23} - 9.99966) - 994.3 * \text{sigmoide}(-8.9 * x_{11} - 5.1) + 0.0653 * \text{sigmoide}(718.1 * x_{24} - 98.108) + 0.0794 * \text{sigmoide}(5.84 * x_{13} - 8.8 * x_{26} - 283.5)$$

Esta red proporciona un error del 2,63%, frente al 4,56% obtenido por [Tsoulos et al-2008], y al 3,51% del mejor de los métodos comparados en [Gavrilis et al-2008]

Para la base de datos Ionosphere, que contiene datos de radar sobre la ionosfera, captados por un conjunto de antenas, compuesto por 351 ejemplos cada uno de ellos con 34 atributos, repartidos en dos clases, buenos y malos.

$$N(x) = 0.0009 * \text{sigmoide}(-6.0 * x_7 - 0.9 * x_6 + 0.501) + 77.9 * \text{sigmoide}(8.5 * x_6 + 52.8 * x_7 - 0.658) + 0.299 * \text{sigmoide}(-28.8 * x_{23} + 69.997518 * x_{13} + 9.427 * x_0 - 30.2)$$

² <http://archive.ics.uci.edu/ml/about.html>

Esta red proporciona una tasa de error del 8,47% frente al 9,66% obtenido por [Tsoulos et al-2008] y al 9,89% del mejor de los métodos comparados en [Gavrilis et al-2008]

Para la base de datos PIMA (Pima Indians Diabetes) que contiene datos de 768 pacientes y 8 atributos para cada uno de ellos, clasificados como diabéticos o sanos, se obtiene igualmente una tasa de error ligeramente mejor que la obtenida por [Tsoulos et al-2008] y [Gavrilis et al-2008], del 23,82% frente al 25,00%, y al 25,10%, mientras que para la base de datos WINE que contiene datos sobre el análisis químico de 178 vinos, con 13 atributos para cada uno de ellos y los clasifica en 3 categorías, el resultado que hemos obtenido es ligeramente peor a las otras dos, un 7,05% de error frente al 4,44%, y al 6,7%.

Estos resultados avalan la validez de nuestro algoritmo para la clasificación de diversos conjuntos de datos, y el diseño de la red neuronal más adecuada para ello, que es lo que necesitamos para clasificar los píxeles de las imágenes de fondo de ojo como pertenecientes a vasos o no, en función de las características extraídas de las mismas con los operadores LBP

4.2. Evaluación experimental.

Para llevar a cabo las pruebas experimentales del método, se ha elegido la base de datos de imágenes de retina DRIVE, del Instituto de Ciencias de la Imagen de la Universidad de Utrecht. Esta base de datos se ha creado con el objetivo de permitir el estudio comparativo sobre la segmentación de los vasos sanguíneos en imágenes de retina. Las imágenes de la base de datos se han obtenido a partir de un estudio sobre la retinopatía diabética realizado en los Países Bajos. La población para el estudio está formada por 400 individuos diabéticos de entre 25 a 90 años de

edad. De todas las fotografías obtenidas se han seleccionado aleatoriamente 40, de las cuales 33 no muestran ningún signo de la enfermedad y 7 muestran signos leves del principio de la misma. Las imágenes están en formato JPEG, y tienen un tamaño de 768 por 584 píxeles con una resolución de 8 bits por plano de color. El campo de visión, FOV, (*Field Of View*), de cada imagen es circular y de aproximadamente 540 píxeles de diámetro. Para cada una de las imágenes, en la base de datos, se proporciona una máscara que delinea el FOV. Con estas 40 imágenes se forman dos subconjuntos, de 20 imágenes cada uno, que se utilizan uno en la fase de entrenamiento y otro en la fase de test. Para las imágenes del conjunto de entrenamiento está disponible una única máscara de segmentación de los vasos sanguíneos realizada manualmente, mientras que para las imágenes del conjunto de test, están disponibles dos conjuntos de máscaras de segmentación de vasos, una utilizada como “*gold standard*” y la otra designada como “2º observador”, que se puede utilizar para comparar segmentaciones realizadas por computadora con las realizadas por un observador humano independiente.

Vimos en el punto 3 del apartado 3.2.1, que para generar el vector de entrada al algoritmo encargado de generar las redes RNA_V y RNA_VF, se calculan para cada píxel, de los marcados como vasos en las máscaras de referencia, y para un número igual de píxeles, elegidos aleatoriamente entre los que no son vasos y pertenecen al FOV, un total de 54 características para la primera y 18 para la segunda. El número total de píxeles marcados como vasos, correspondientes al conjunto de imágenes de entrenamiento, son 569.493 en las máscaras *gold standard*, y 93.381 en las máscaras de vasos finos.

Debido a la naturaleza estocástica de los algoritmos evolutivos, se realizaron 50 ejecuciones del algoritmo para la construcción y entrenamiento de cada una de las dos redes, siendo finalmente las que obtuvieron los mejores resultados las presentadas a continuación.

La red RNA_V, está formada por dos neuronas y utiliza solamente 4 de las 54 características de entrada. Su formulación es la siguiente:

$$RED_V = 0.8 * \text{sig}(0.01 * x_{40} - 1.28497956) + 0.18 * \text{sig}(-0.22 * x_{34} - 9.984 * x_{49} + 31.807 * x_{16} + 13.8)$$

Las 4 características que utiliza, todas ellas con P=24, se corresponden con:

X16 → Componente positivo del LTP ($\delta = 2$), para R=9.

X34 → Componente CLBP_M de CLBPLTP, para R=8.

X40 → Componente CLBPLTP_PM ($\delta = 2$) de CLBPLTP, para R=5.

X49 → Componente CLBPLTP_NM ($\delta = 2$) de CLBPLTP, para R=5.

Por otro lado, la red para la clasificación de vasos finos, RNA_VF, también está formada por dos neuronas y utiliza, tan solo, dos características de las 18 que se habían utilizado en el vector de entrada. Su formulación es:

$$RED_VF = 0.77 * \text{sig}(2.212 * x_{14} + 0.76) - 2.9 * \text{sig}(-0.0001 * x_2 - 2.2194)$$

Las dos características, ambas con P=24, que utiliza son:

X2 → Componente positivo del LTP ($\delta=2$), para R=2.

X14 → Componente CLBPLTP_PM ($\delta = 2$) de CLBPLTP, para R=3.

4.2.1. Método de evaluación.

Para analizar la bondad del método y tener una referencia comparativa con las soluciones propuestas por otros autores, se han evaluado, en primer lugar, la *precisión*, que es la relación entre el número de píxeles correctamente clasificados y el número total de puntos en el FOV de la imagen, la *sensibilidad*, que representa el porcentaje de píxeles correctamente clasificados por el método y la *especificidad* que nos da el

porcentaje de píxeles que no son vasos correctamente clasificados como tales por el método.

$$\textit{Precisión} = \frac{T_n + T_p}{T} \quad \textit{Sensibilidad} = \frac{T_p}{T_p + F_n} \quad \textit{Especificidad} = \frac{T_n}{T_n + F_p}$$

donde T_p y T_n son, respectivamente, el número total de verdaderos positivos y verdaderos negativos, es decir, T_p sería el número de píxeles de vasos clasificados correctamente como vasos, y T_n el número de píxeles de no vasos clasificados correctamente como no vasos. F_p y F_n son falsos positivos y falsos negativos, es decir píxeles erróneamente clasificados, bien como vasos no siéndolo o viceversa. Y T son el número total de píxeles en el FOV de la imagen. A la sensibilidad también se la conoce como la fracción de verdaderos positivos, mientras que a la especificidad se la asocia con la fracción de verdaderos negativos.

Otra medida que se utiliza para analizar el rendimiento de un clasificador en general y a la que recurren muchos autores, para realizar la comparativa entre los métodos de segmentación de vasos sanguíneos, en las imágenes de retina, es el área bajo la curva ROC (*Receiver Operating Characteristic*). La curva ROC es una representación gráfica de la *sensibilidad* frente a ($1 - \textit{especificidad}$), o lo que es lo mismo de la fracción de verdaderos positivos frente a la fracción de falsos positivos, según se varía el umbral de discriminación. El área bajo la curva ROC (A_z) determina la capacidad discriminatoria del método. Así, si clasificáramos todos los píxeles aleatoriamente obtendríamos un área $A_z = 0.5$, mientras que si el método fuera perfecto A_z valdría 1.

En los artículos [Staal et al-2004] y [Niemeijer et al-2004], distinguen entre métodos en los que se obtienen máscaras binarias de segmentación de los vasos sanguíneos, cuya clasificación de los píxeles denominan dura y aquellos métodos, que denominan de clasificación suave, en los que los píxeles de la imagen segmentada tienen diferentes intensidades, y en

función del umbral de discriminación que se aplique, se obtiene una pareja de valores para sensibilidad y especificidad diferente, lo cual nos permite trazar la curva ROC correspondiente.

En nuestro caso, la máscara final de segmentación obtenida es una máscara binaria, pero se aplica en el proceso un umbral de discriminación óptimo, que fija en cada una de las salidas de las dos RNAs, el valor que ha de tener un píxel para considerarlo vaso o no vaso.

Recordemos que a la salida de las RNAs, el valor asignado a cada píxel está comprendido entre 0 y 1, por lo cual este umbral de discriminación óptimo, se obtiene variando el umbral de discriminación desde 0 hasta 1, en intervalos de 0,01, y calculando para cada valor asignado al umbral, el valor de *precisión* obtenido. Finalmente, los umbrales de discriminación para cada RNA, se han ajustado a aquellos valores en los que se consigue un valor de *precisión* más elevado. En la red RNA_V el umbral óptimo ha resultado ser de 0.8. Recordemos que como esta red no se “*binariza*”, ponemos a cero los píxeles con valor inferior al umbral y el resto conserva su valor. Por otro lado, para la red RNA_VF el valor óptimo que se obtiene es de 0.76 y a todos los píxeles con valor inferior se les asigna el valor de cero, pero en este caso al resto se les aplica el valor de 1, obteniendo de este modo una máscara binaria.

Este proceso de variación del umbral de discriminación para conseguir los umbrales óptimos, también nos permite trazar las curvas ROC para ambas RNAs, como se muestran en la figura 4.3. El valor de A_z para la RNA_V es de 0.9024, mientras que para la RNA_VF es de 0.8969.

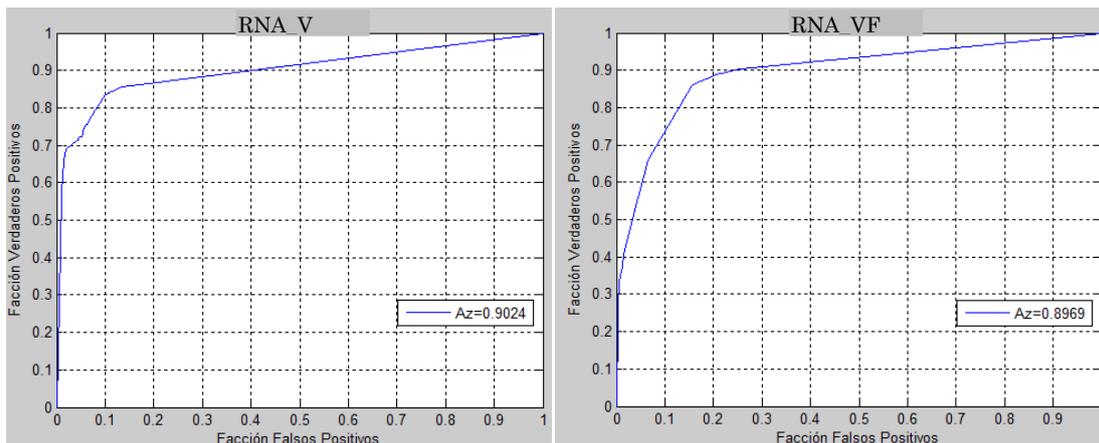


Figura 4.3. Curvas ROC para las redes RNA_V ($A_z=0.9024$) y RNA_VF. ($A_z=0.8969$)

Si ahora aplicamos, a medida que vamos variando el umbral, el operador OR a la salida producida por ambas redes, vemos en la figura 4.4, como el área de la curva ROC obtenida aumenta hasta un valor de 0.9208.

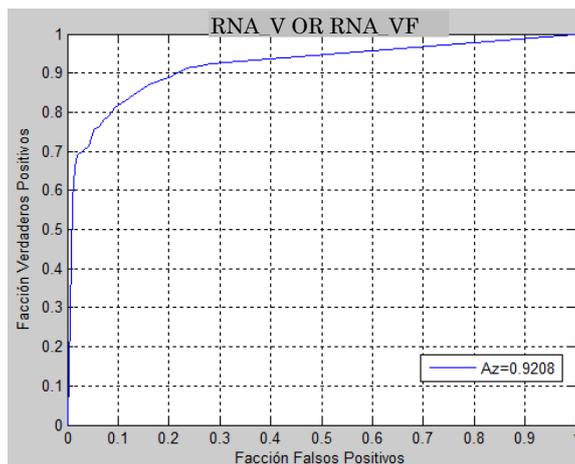


Figura 4.4. Curva ROC RNA_V OR RNA_VF. ($A_z=0.9208$)

Capítulo 5.

Comparación y Discusión de Resultados

En la Tabla 2.1., del apartado 2.2.1., vimos la comparativa presentada en el artículo [Staal et al-2004], entre su método y el de otros autores. Antes de 2004 y posteriormente han surgido otros métodos basados en diferentes técnicas como vimos en el capítulo sobre la Revisión de la Bibliografía, pero no en todos ellos se publican los resultados obtenidos con las imágenes de bases de datos públicas como la DRIVE o la Hoover o se publican sólo resultados parciales, por ejemplo se ciñen solamente a la sensibilidad obtenida.

En base a los valores de *precisión*, *sensibilidad*, *especificidad*, y al área bajo la curva ROC, en la tabla 5.1, se presentan los resultados publicados por otros ocho autores y el que se ha obtenido por el método presentado en este trabajo, para los umbrales que optimizan la precisión en ambas redes neuronales, descritos en el apartado 4.2.1, para el conjunto de imágenes de test de la base de datos de imágenes de retina DRIVE. Los datos representan los valores medios obtenidos para cada uno de los valores, y junto al dato de *precisión* figura la desviación típica.

A Continuación, en la figura 5.1, se muestra la gráfica de la curva ROC para nuestro método y la gráfica de las curvas ROC de los métodos de otros de autores.

En la tabla 5.1, vemos que nuestro método supera al resto en cuanto a precisión, y por otro lado sólo es superado por el método de Staal, en cuanto al área bajo la curva ROC, A_z .

Método	Precisión	Sensibilidad	Especificidad	A_z
2º Observador	0.9473000±0.0048	0.7761000	0.9725000	-
RNAE+LBP	0.9443581±0.0064	0.7075470	0.9792190	0.9319
Staal	0.9441500±0.0065	0.7193500	0.9773000	0.9520
Fraz (2012)	0.9430000±0.0072	0.7152000	0.9768000	-
Niemeijer	0.9416500±0.0065	0.6793000	0.9801000	0.9294
Zana	0.9377000±0.0077	0.6453450	0.9769000	0.8984
Fraz (2008)	0.9303000±0.0079	0.7114000	0.9680000	-
Jiang	0.9212000±0.0076	0.6478000	0.9625500	0.9114
Pérez	0.9181500±0.0240	0.7086500	0.9501000	-
Chaudhuri	0.8894500±0.0232	0.2716000	0.9794500	0.7878

Tabla 5.1. Comparativa de resultados obtenidos por nuestro método (RNAE+LBP) con diferentes métodos de segmentación, en el conjunto de test de la base de datos de imágenes de retina DRIVE. (Niemeijer et al. 2004, Staal et al. 2004 y Fraz et al 2012)

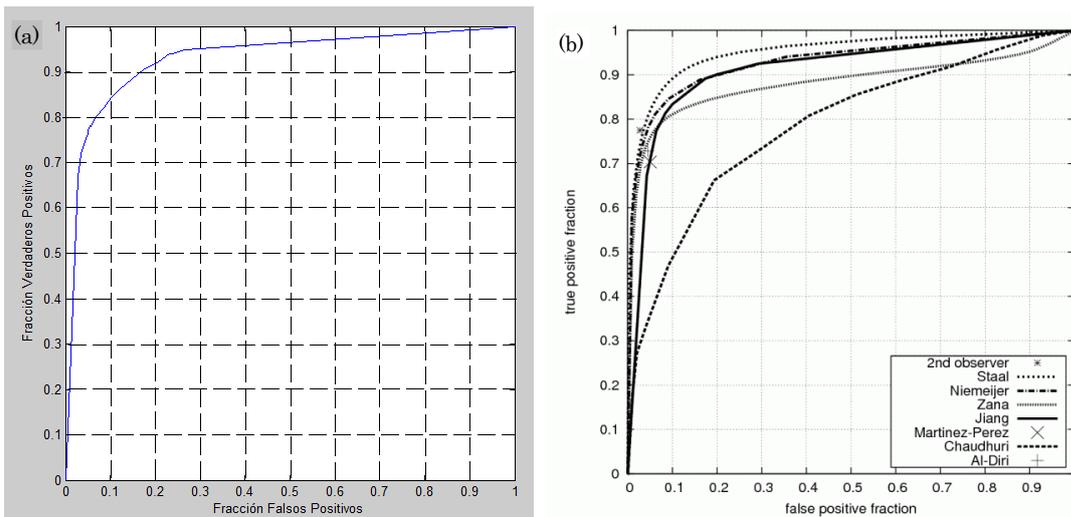


Figura 5.1. Curvas ROC obtenida para (a) nuestro método ($A_z=0.9319$) y (b) curvas ROC para los métodos de otros autores, [Staal et al-2004].

Dado que las redes neuronales que utiliza el método, han sido obtenidas a partir de las imágenes de entrenamiento de la base de datos de imágenes de retina DRIVE, se ha querido comprobar también su eficacia en otra base de datos de imágenes de retina. Para este fin, se ha contado con las 20 imágenes disponibles en la base de imágenes de retina STARE (*Structured Analysis of Retina*), de la Universidad de California, también

conocida como Hoover, y los resultados se muestran en la tabla 5.2. Hay que reseñar que los autores que utilizan métodos supervisados, han efectuado el entrenamiento de su método con las imágenes de STARE, mientras que nosotros hemos aplicado las mismas RNAs, obtenidas en el entrenamiento de la base de datos de imágenes DRIVE, a las imágenes de STARE. Los datos presentados por [Staal et al-2004], para esta base de imágenes, se refieren únicamente a 19 de las 20 imágenes utilizables y, para segmentar los vasos sanguíneos de cada imagen, utilizaron como conjunto de entrenamiento las 18 imágenes restantes.

Observamos que, incluso sin entrenamiento específico en esta base de imágenes, se ha conseguido una precisión cercana a la obtenida para la base de datos de imágenes de retina DRIVE, por lo que, con estos resultados, se aporta evidencia de la robustez del método. Además, hay que tener en cuenta que las imágenes de la base de datos Hoover se han obtenido a partir del escaneado de película, y no directamente con una cámara digital de fondo de ojo (como es el caso de la base de datos de imágenes de retina DRIVE), lo que hace que la calidad de las mismas sea bastante más baja.

Método	Precisión	Sensibilidad	Especificidad
Staal	0,9516	0.6970	0.9810
Fraz (2012)	0.9442	0.7311	0.9680
RNAE+LBP	0,9395	0,7173	0,9652
Fraz (2008)	0.9367	0.6849	0.9710
2 Observador	0,9351	0.8951	0.9384
Hoover	0,9275	0.7500	0.9562
Jiang	0,9009	0.8340	0.9562

Tabla 5.2. Comparativa de resultados obtenidos por nuestro método (RNAE+LBP) y diferentes métodos de segmentación, en el conjunto de test de la base de datos STARE. [Niemeijer et al-2004, Staal et al-2004 y Fraz et al-2012]

Como comentan en su artículo [Niemeijer et al-2004], la diferencia entre el primer observador (*gold standard*) y el segundo observador nos da idea de que la segmentación de los vasos sanguíneos en las imágenes de retina no es una tarea fácil, ya que existen multitud de factores que la complican. Por ejemplo los vasos finos son muy difíciles de ver, ya que

algunos no tienen ni siquiera la anchura de un píxel. La compresión que realiza el formato de imágenes JPEG también dificulta la segmentación, ya que con ella es más difícil establecer cuáles son los bordes de los vasos más grandes, y por ello algunos píxeles que el primer observador incluye como bordes de vasos, el segundo observador puede no haberlos incluido. Un último factor que afecta a la segmentación manual es el tiempo, ya que de media se necesitan aproximadamente 2 horas para segmentar una sola imagen, lo que puede dar lugar a la aparición de fatiga y con ello a errores.

Igualmente, elegir un buen criterio para comparar el rendimiento de los diferentes métodos de segmentación, tampoco es trivial. En este trabajo, al igual que en otros, se ha elegido como criterio la precisión media máxima, que nos indica cual es el método capaz de segmentar los vasos de forma más exacta, en comparación con la segmentación marcada como *gold standard*. Pero esta medida, presenta el inconveniente de que los vasos grandes tienen mucho más peso, en el resultado final que los vasos finos, lo que hace que en la mayor parte de las segmentaciones automáticas se pierdan una gran parte de este tipo de vasos. Por ello, si en una aplicación determinada fuera vital la detección de los vasos más finos, podría ser mejor un método con una menor precisión, pero con una mayor sensibilidad que detectara mayor cantidad de vasos finos. Igualmente si lo importante fuera determinar la tortuosidad de la red vascular, lo más importante sería la precisión en la determinación de las líneas centrales de los vasos.

En nuestro método, otro parámetro que es necesario ajustar es el área de filtrado de *blobs* en el último paso de la fase II. Ya comentamos que se había ajustado a 30. En la figura 5.2, a modo de ejemplo, se muestra una gráfica de la variación que tiene lugar en los valores de *precisión*, *sensibilidad* y *especificidad* a la hora de efectuar el último paso de eliminación de *blobs*, en función del tamaño del área elegida.

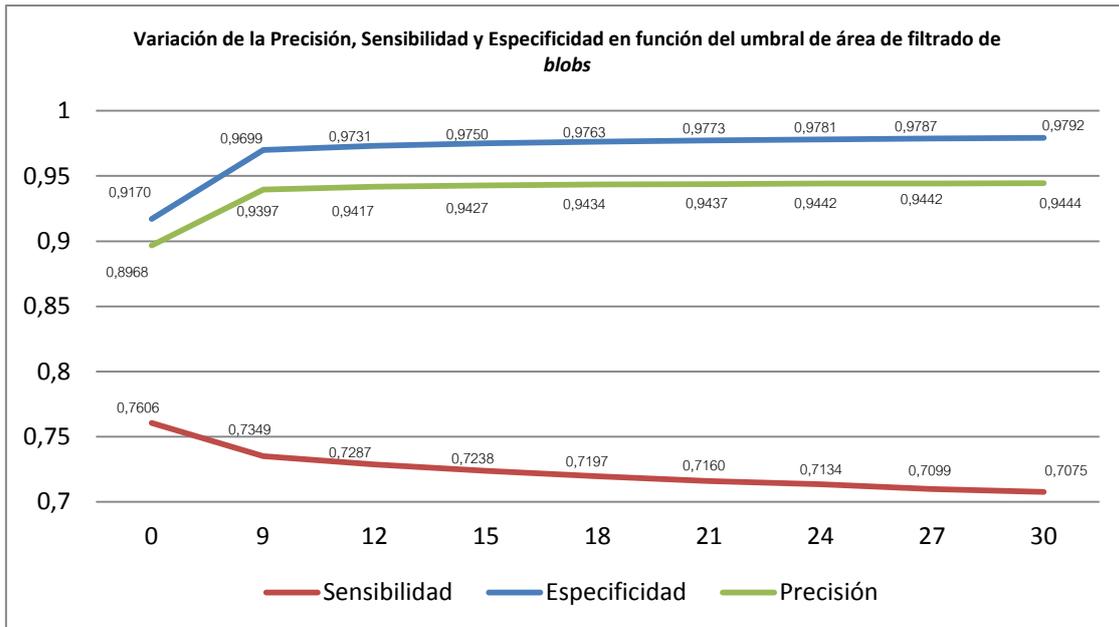


Figura 5.2. Variación de la Precisión, Sensibilidad y Especificidad en función del área de filtrado de blobs, para el conjunto de imágenes de test de la base de datos de imágenes de retina DRIVE.

Se aprecia como a medida que se aumenta el área de filtrado de blobs, eje de abscisas, disminuye la *sensibilidad* en mayor proporción de lo que aumenta la precisión. Es decir a medida que se elimina ruido generado en la segmentación, también se eliminan vasos etiquetados correctamente y de ahí la disminución de sensibilidad. Esto se debe a que en muchos vasos finos existen píxeles que no están etiquetados como vasos, lo que da lugar a discontinuidades en los mismos que originan, al aplicar el filtrado por área, la pérdida de estos segmentos no conectados.

Vemos que si aplicamos un área de filtrado igual a 24, la precisión obtenida es de 0.94415 que iguala a la mejor obtenida por el resto de métodos en la base de imágenes DRIVE, y la sensibilidad aumenta hasta un valor de 0.71340, que si bien no supera a la de [Staal et al-2004], 0.71935, se convierte en la segunda mejor.

Este efecto se puede ver claramente en la figura 5.3, donde se muestra la segmentación obtenida para la figura 20 del conjunto de test de la base de datos de imágenes de retina DRIVE. A pesar de que el valor medio más alto para la *precisión* en el conjunto de imágenes de test, se ha

obtenido para un área de filtrado igual a 30, apreciamos en dicha figura como en este caso se obtiene una precisión más alta para un valor de 24, esto también es una consecuencia de la discontinuidad en la segmentación de vasos delgados.

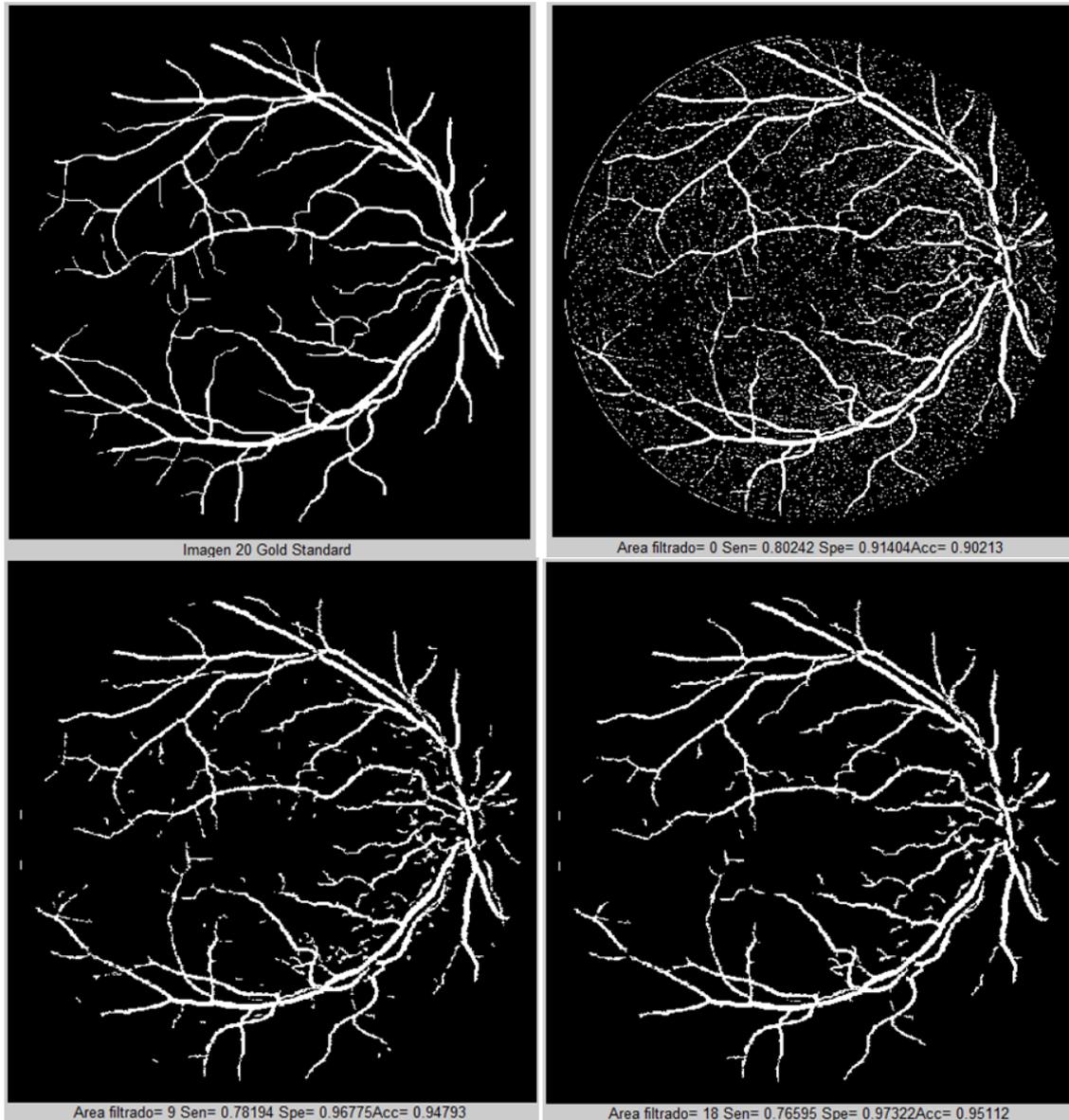


Figura 5.3.i. Influencia del área de filtrado de eliminación de blobs en los valores de Sensibilidad, Especificidad y Precisión. Imagen número 20 de la base de datos de imágenes de retina DRIVE.

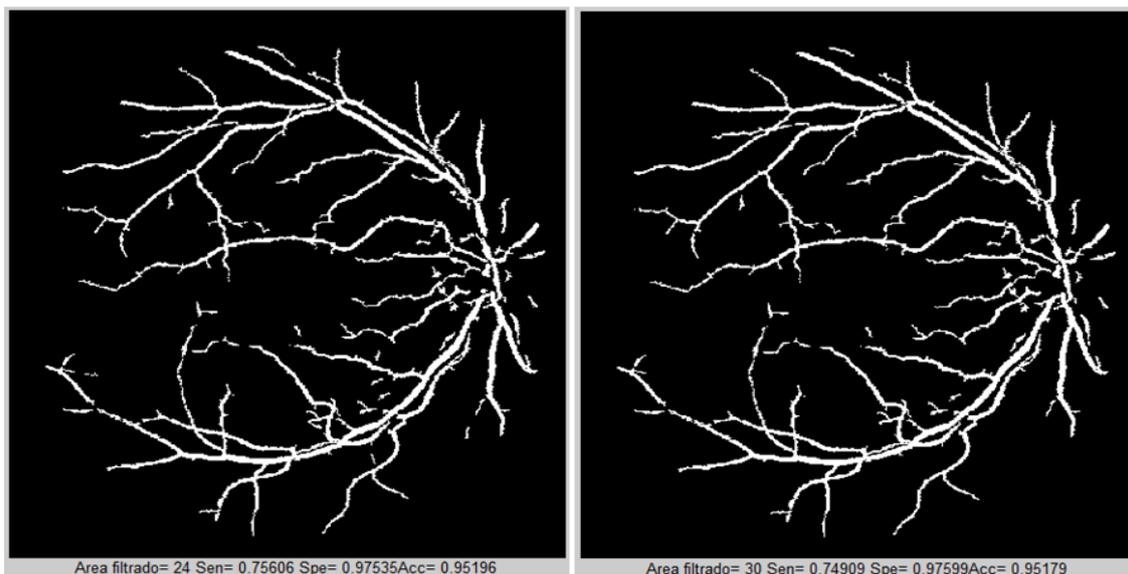


Figura 5.3.ii. Influencia del área de filtrado de eliminación de blobs en los valores de Sensibilidad, Especificidad y Precisión. Imagen número 20 de la base de datos de imágenes de retina DRIVE.

La figura 5.4, a modo de ejemplo, muestra la segmentación obtenida por el resto de métodos³, para la misma imagen número 20 de la base de datos de imágenes de retina DRIVE, junto con la imagen original y la segmentación gold standard.

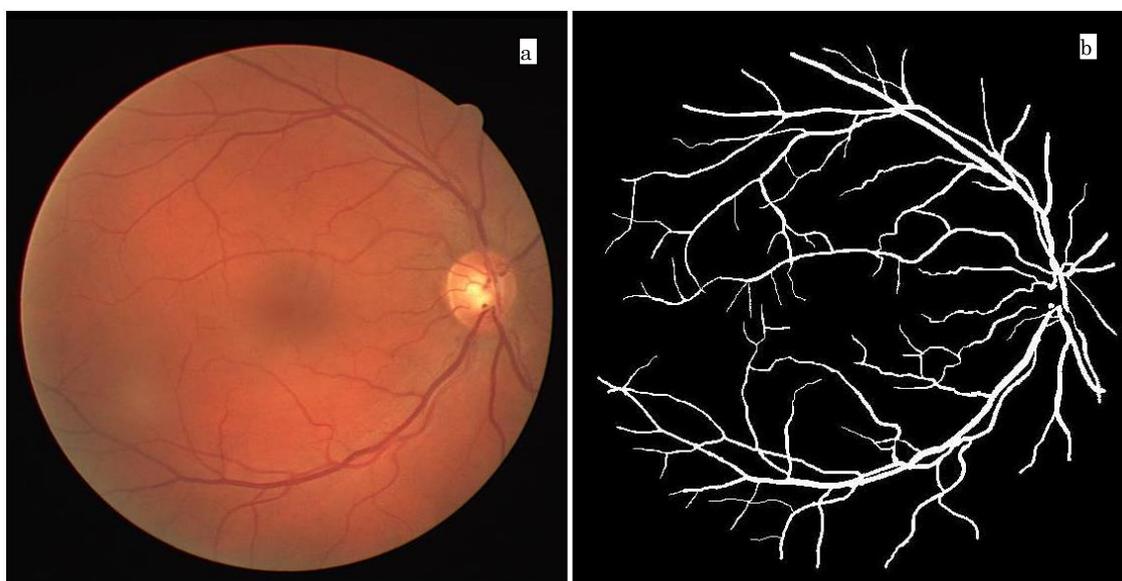


Figura 5.4.i. Segmentación de la imagen 20 de la base de datos de imágenes de retina DRIVE. a) Imagen original. b) Gold Standard.

³ <http://www.isi.uu.nl/Research/Databases/DRIVE/browser.php>

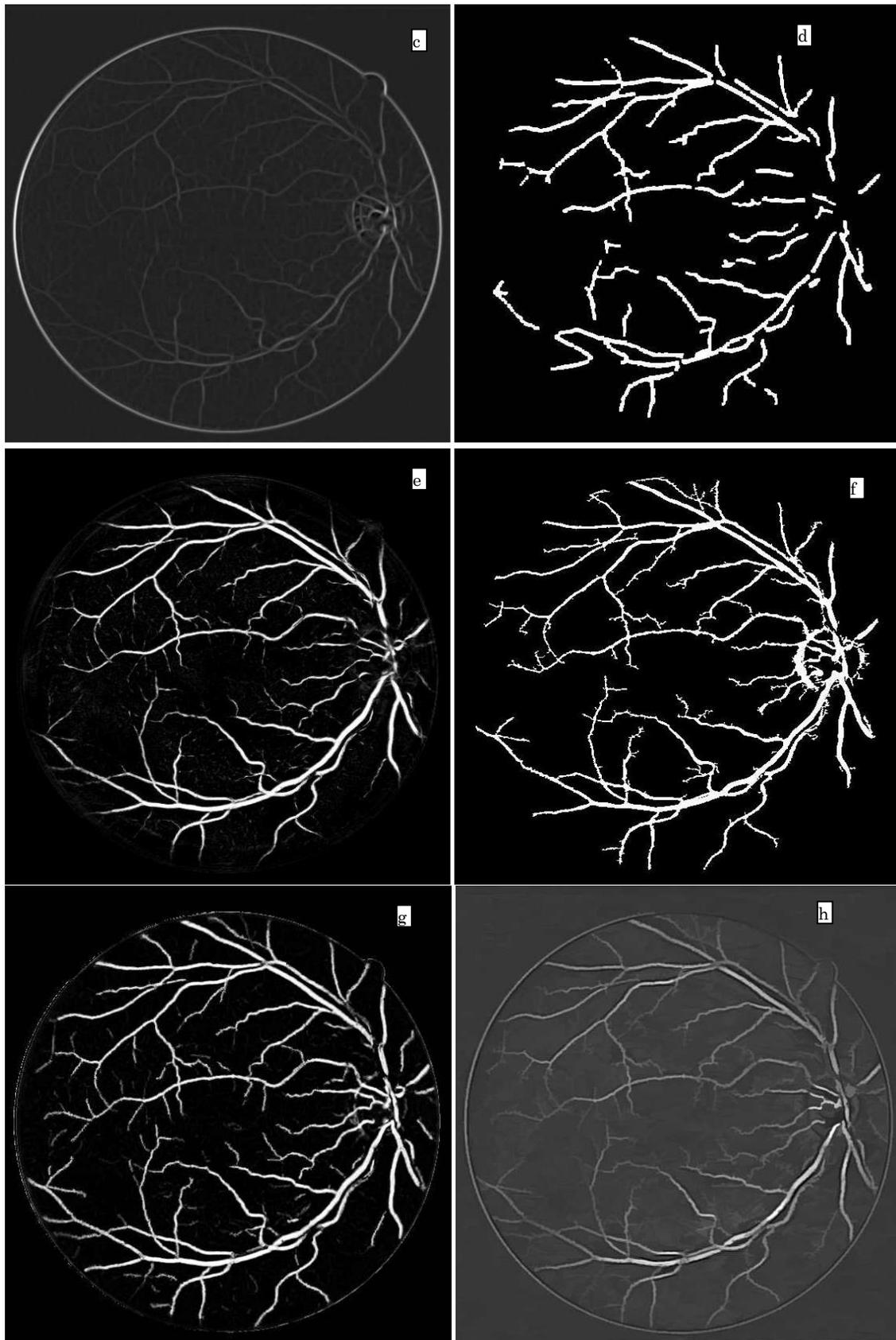


Figura 5.4.ii. Segmentación de la imagen 20 de la base de datos de imágenes de retina DRIVE. c) Chaudhuri. d) Jiang. e) Niemeijer. f) Pérez. g) Staal. h) Zana.

Los datos de sensibilidad, especificidad y precisión para esta imagen son los que se muestran en la tabla 5.3, en la que vemos como el valor máximo de precisión en este caso se obtiene con el método de Niemeijer, con 0.953, seguido del nuestro con 0.952 y del de Staal con 0.951. Todos ellos valores prácticamente iguales, sin embargo, en cuanto a sensibilidad, se sitúa por delante el método de Staal, con 0.755, seguido del nuestro con 0.749 y por último el de Niemeijer con 0.726.

Método	Precisión	Sensibilidad	Especificidad
2º Observador	0.945	0.870	0.954
Niemeijer	0.953	0.726	0.980
RNAE+LBP	0.952	0.749	0.976
Staal	0.951	0.755	0.974
Zana	0.946	0.658	0.980
Jiang	0.928	0.659	0.960
Pérez	0.932	0.785	0.949
Chaudhuri	0.911	0.306	0.983

Tabla 5.3. Comparativa de resultados obtenidos por diferentes métodos de segmentación, para la imagen número 20 del conjunto de test de la base de datos de imágenes de retina DRIVE.

En la tabla 5.4., se presentan los datos para la imagen 15. En este caso el valor más alto de precisión lo obtiene Zana, un 0.941, seguido de nuestro método y del de Staal con 0.938 y detrás el de Niemeijer con 0.937. En cuanto a sensibilidad destaca de nuevo el de Staal con un 0.825, seguido del nuestro con 0.795, del de Niemeijer con 0.788 y por último el de Zana con 0.743.

Con todos estos datos, podemos apreciar que la diferencia entre los métodos que obtienen un valor más elevado en cuanto a precisión es ligerísima, nuestro método supera al de [Staal et al-2004] en tan sólo un 0.02%, este al de [Fraz et al-2012] en 0.115% y este al de [Niemeijer et al.2004] en un 0.135% que a su vez supera al de [Zana&Klein-2001] en un 0.395%.

Método	Precisión	Sensibilidad	Especificidad
2º Observador	0.954	0.800	0.972
RNAE+LBP	0,938	0.795	0,956
Staal	0.938	0.825	0.951
Niemeijer	0.937	0.788	0.954
Zana	0.941	0.743	0.964
Jiang	0.909	0.777	0.925
Pérez	0.854	0.884	0.850
Chaudhuri	0.923	0.357	0.988

Tabla 5.4. Comparativa de resultados obtenidos por diferentes métodos de segmentación, para la imagen número 15 del conjunto de test de la base de datos de imágenes de retina DRIVE.

En cuanto al valor de sensibilidad que obtienen estos cuatro métodos, sí que se aprecia una diferencia algo más significativa, superando el método de [Staal et al-2004] al de [Fraz et al-2012] en un 0,415%, el de Fraz al nuestro en un 0.76%, el nuestro al de [Niemeijer et al- 2004] en un 2,82%, y este al de [Zana&Klein-2001] en un 3,39%.

Igualmente es de destacar, como no hay un método que obtenga siempre el mejor resultado en todas las imágenes. Esto se puede deber, por ejemplo, a la influencia que tenga en cada método la presencia en la imagen de determinadas patologías como por ejemplo lesiones rojas, exudados, drusas, etc. A modo de ejemplo en la figura 5.5, se muestra como varían la precisión, la sensibilidad y la especificidad obtenidas por nuestro método, para cada una de las imágenes del conjunto de test de la base de imágenes DRIVE.

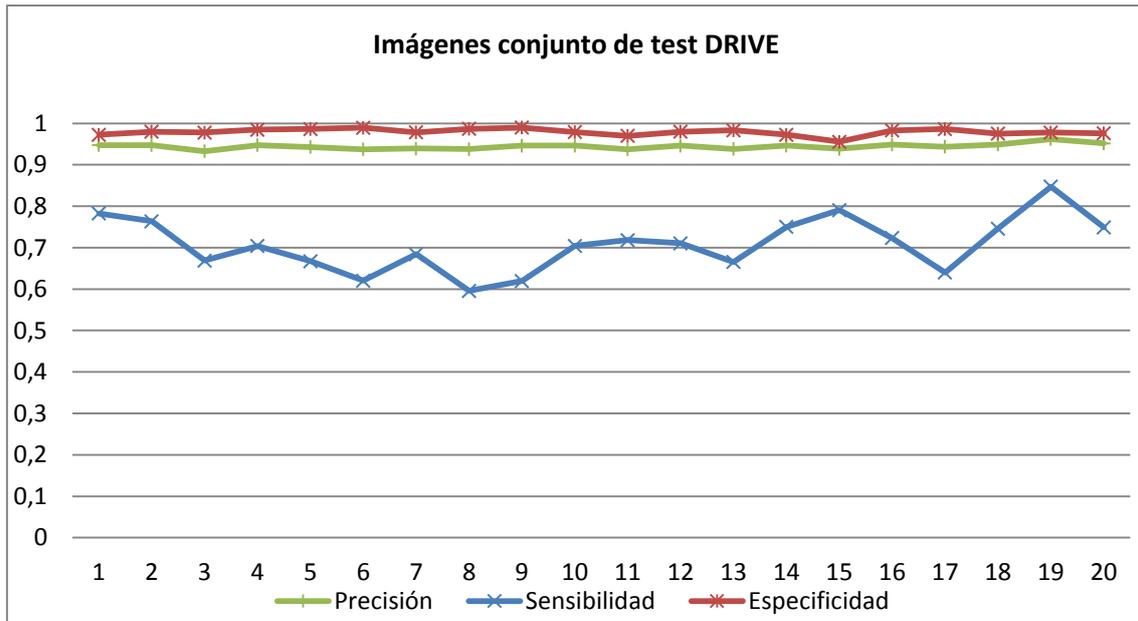


Figura 5.5. Valores de Precisión, Sensibilidad y Especificidad obtenidos con el método RNA+LBP, para cada una de las imágenes del conjunto de test de la base de datos de imágenes DRIVE.

En algunos estudios, se tiene en cuenta el tiempo de segmentación del método. Así en el trabajo de [M. M. Fraz et al. (2008)], toman como criterios de evaluación la sensibilidad y el tiempo de segmentación. De los cuatro métodos que exponen en su artículo, el que presenta menor tiempo de segmentación medio para las imágenes de la base de datos de imágenes de retina DRIVE lo hace en 35.4708 segundos, y los otros tres oscilan alrededor de los 53 segundos. En [Kaur&Sinha-2012], el tiempo medio de proceso por imagen es de 30 segundos, mientras que en [Staal et -2004], comentan que su método emplea del orden de 15 minutos en realizar la segmentación de una imagen de la base de datos de imágenes DRIVE.

En nuestro método, para el conjunto de imágenes de test de la base de imágenes DRIVE, el tiempo medio que se tarda en la fase I, de extracción de las características que utilizan las dos RNAs, es de 2.2975 ± 0.0617 segundos, y el tiempo medio una vez obtenidas las características para realizar la segmentación de la imagen, correspondiente a la Fase II del método, es de 0.23645 ± 0.029 segundos, lo que nos da un tiempo medio total de

2.5340±0.0726 segundos. -La CPU utilizada ha sido un Intel i5-2400 a 3.1Ghz y 8Gb RAM-

En la tabla 5.5, se muestra una comparativa de los tiempos de segmentación empleado por los diferentes métodos.

Método	Tiempo de Segmentación (segundos)
RNAE+LBP	2.534
Jaspreet Kaur (2012)	≈30
Fraz(2008) Bit plane slicing	35.47
Fraz(2008) Multi scale reconstruction	53.71
Staal (2004)	≈900

Tabla 5.5. Comparativa de tiempos medios de segmentación, empleados por los distintos métodos, para el conjunto de imágenes de test de la base de imágenes DRIVE.

En la mayoría de los trabajos examinados no se han encontrado los tiempos de segmentación para cada una de las imágenes de DRIVE, y solamente en los mencionados anteriormente existen referencias al respecto. A pesar de que todos ellos son anteriores al nuestro y que algunos son de hace bastantes años, como el de [Staal et al-2004], los tiempos que hemos obtenido creemos que son muy competitivos y en cualquier caso el método se puede considerar como bastante rápido.

Capítulo 6.

Conclusiones y Trabajo Futuro

El método de detección de vasos sanguíneos en imágenes de retina basado en Redes Neuronales Artificiales Evolutivas y operadores basados en LBP, presentado en este trabajo obtiene unos valores de precisión, sensibilidad y especificidad en la segmentación competitivos con los mejores métodos existentes en la literatura, relacionados con el campo, siendo incluso el que obtiene un valor más alto en precisión. Igualmente obtiene el segundo mejor valor en la comparativa respecto al área bajo la curva ROC.

Por otra parte con este trabajo, se aporta evidencia de la utilidad del uso de operadores LBP en la detección de patrones geométricos, además de su ya conocida utilidad en la detección de texturas.

Queda también demostrada la gran ventaja de utilizar GE para el aprendizaje de RNA, al evitar el esfuerzo que supone el diseño de la topología de la red (número de neuronas de la capa oculta) y el tener que elegir las características de entrada más discriminativas, decisiones que sí deben ser tomadas con los métodos de entrenamiento clásico de RNAs. Los dos aspectos anteriores, junto al aprendizaje de una RNA mediante una evolución de una población de RNAs permiten obtener finalmente una RNA bastante simple, compacta y con gran poder de generalización.

Además aunque no disponemos de datos suficientes para juzgar el tiempo de proceso de los métodos analizados, por los pocos datos publicados al respecto, creemos que el método aquí presentado obtiene unos tiempos de segmentación excelentes, debido a que sólo hay que obtener el valor de los

operadores LBP, cuyo cálculo es muy simple, y aplicarlos a dos RNAs ya entrenadas.

Como trabajo futuro creemos que utilizando las mismas técnicas se podrían detectar, igualmente, otras estructuras importantes de la retina, lesiones rojas y lesiones brillantes e, incluso, discernir en la segmentación de la red vascular entre venas y arterias. En definitiva, se podría obtener un sistema completo de ayuda al diagnóstico (CAD) y seguimiento de las enfermedades oculares eficaz y preciso con un coste computacional muy bajo, lo que hace muy posible su aplicabilidad en el tratamiento diario de pacientes en las consultas oftalmológicas.

La importancia de un sistema completo CAD de este tipo se evidencia al descubrir, que solo en los Estados Unidos de América existían, en 2007, aproximadamente unos 18 millones de personas con diabetes [Niemeijer et al-2007], y que una de las consecuencias de esta dolencia es la retinopatía diabética, la cual es de vital importancia diagnosticar en los primeros estadios y, una vez diagnosticada, efectuar un seguimiento, eficaz y continuo de la enfermedad, ya que es la principal causa de ceguera en la población en el mundo occidental.

Bibliografía

E.J. Carmona, M.Rincón, J.García-Feijoó, J.M. Martínez de la Casa. (2008). "Identification of the optic nerve head with genetic algorithms". *Artificial Intelligence in Medicine*, vol. 43, pp. 243-259.

S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, M. Goldbaum. (1989). "Detection of blood vessels in retinal images using two dimensional matched filters". *IEEE Transactions on medical imaging*, vol 8, no 3, pp. 263- 269.

D. Eberly. (1996). "Ridges in Image and Data Analysis". Dordrecht, The Netherlands: Kluwer Academic.

M.M. Fraz, M.Y. Javed, A. Basit. (2008). "Evaluation of retinal vessel segmentation methodologies based on combination of vessel centerlines and morphological processing". 4th International Conference on Emerging Technologies. *Emerging Technologies*, pp. 232-236

M. M. Fraz, S. A. Barman, P. Remagnino, A. Hoppe, A. Basit, B. Uyyanonvara, A. R. Rudnicka, C. G. Owen. (2012). "An approach to localize the retinal blood vessels using bit planes and centerline detection". *Comput Methods Programs Biomed*, vol 108, pp. 600-616.

D. Gavrilis, I.G. Tsoulos, E.Dermatas. (2008). "Selecting and constructing features using gramatical evolution". Elsevier Science. *Pattern recognition letters*, vol 29, no. 9, pp. 1358-1365.

X.W. Gao, A. Bharath, A. Hughes, A. Staton, N. Chapman, S. Thom. (1997). "Towards retinal vessel parametrisation". *SPIE Conference on Medical Imaging. SPIE 3034, Medical Imaging 1997: Imaging Processing*, 734, pp. 734-744.

- X.W. Gao, A. Bharath, A. Hughes, A. Staton, N. Chapman, S. Thom (2000). "Quantification and characterization of arteries in retinal images". *Computer Methods and Programs in Biomedicine*, vol: 63, pp. 133-146.
- Z. Guo, L. Zhang, D.Zhang. (2010). "A completed modeling of Local Binary Pattern Operator for texture classification". *IEEE Transaction on image processing*, vol. 19, no 6, pp. 1657-1663.
- A.Hoover, V. Kouznetsova, M. Goldbaum, (2000). "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response". *IEEE Transactions on medical imaging*, vol 19, no 3, pp. 203-210.
- K.Hoover, R. Marceau, T. Harris, D. Reif, A. Motsinger-Reif. (2012). "A comparison of GE optimized neural networks and decision trees". *GECCO Companion, proceedings of the Fourteenth international conference on Genetic and evolutionary computation conference companion*, pp. 611-614.
- X. Jiang, D. Mojong. (2003). "Adaptative local thresholding by verification-based multithreshold probing with application to vessel detection in retinal images". *IEEE Transaction on Pattern Analysis and Machine Intelligence*. vol 25, no 1, pp. 131-137.
- S.N. Kalitzin, J. Staal, B.M. Romeny, M.A. Viergever. (2001). "A computational method for segmenting topological point-sets and application to image analysis". *IEEE transactions on pattern analysis and machine intelligence*, vol 23, no 5, pp. 447-459.
- J. Kaur, H. P. Sinha. (2012). "An efficient blood vessel detection algorithm for retinal images using local entropy thersholding". *International Journal of Engineering Research & Technology*, vol 2, no 4, pp. 924-931
- B. Kochner, D. Schulmann , M. Michaelis, G. Mann, K.H. Engle-Meier. (1998). *Course tracking and contour extraction of retinal vessels from colour fundus photographs: Most efficient use of steerable filters for model based*

image analysis, SPIE Proceedings of Medical Imaging, vol 3328(2), pp. 755–761.

M.E. Martinez-Perez, A.D. Hughes, S.A. Thom, A.A. Bharath, K.H. Parker. (1999). “Scale-space analysis for the characterization of retinal blood vessels”. Medical Image Computing and Computer Assisted Intervention. Springer-Verlang, pp.90-97

M.E. Martinez-Perez, A.D. Hughes, S.A. Thom, A.A. Bharath, K.H. Parker. (2007) “Segmentation of blood vessels from red-free and fluorescein retinal images”. Elsevier. Medical Image Analysis, vol 11, pp. 47-61

A.M. Mendoca, A.Campilho. (2006). “Segmentation of retinal blood vessels by combining the detection of centerlines and morphological reconstruction”. IEEE transactions on medical imaging, vol. 25, no 9, pp. 1200-1213.

A.A. Motsinger-Reif, D.M. Reif, S.M. Dudek, M.D. Ritchie (2006). “Understanding the evolutionary process of grammatical evolution neural networks for feature selection in genetic epidemiology”. IEEE Symposium on Computational Intelligence and Bioinformatics Computational Biology, pp. 1-8

A.A. Motsinger-Reif, T.J. Fanelli, A.C. Davis, M.D. Ritchie. (2008). “Power of grammatical evolution neural networks to detect gene-gene interactions in the presence of error”. BMC Research Notes, pp. 1-8

M. Niemeijer, J. Staal, B. Ginneken, M. Loog, M.D. Abràmoff, (2004).“Comparative study of retinal vessel segmentation methods on a new publicly available database”. Proceedings of SPIE Medical Imaging. vol 5370, pp. 648-656

M. Niemeijer, M.D. Abràmoff, B. Ginneken, (2007).“Segmentation of the Optic Disc, Macula and vascular arch in fundus photographs”. IEEE transactions on medical imaging, vol 26, no.1, pp. 116-127

M. O'Neill, C. Ryan. (2001). Grammatical Evolution. IEEE transactions on evolutionary computation, vol 5, no 4, pp. 349-358.

T. Ojala, M. Pietikäinen, T. Mäenpää, (2000). "Gray-scale and rotation invariant texture classification with local binary patterns". 6th European Conference on Computer Vision. Part I, pp. 404-420.

T. Ojala, M. Pietikäinen, T. Mäenpää, (2002). "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns." IEEE transactions on pattern analysis and machine intelligence. vol. 24, no 7, pp. 971-987.

S.H. Park, J.M. Goo, C.H. Jo, (2004). "Receiver Operating Characteristic (ROC) Curve: Practical review for radiologists". Korean J Radiol 5(1), pp. 11-18.

M. Pietikäinen, G. Zhao, A. Hadid, T. Ahonen. Computer Vision Using Local Binary Patterns. Ed. Springer. 2011

S.A. Salem, N.M. Salem, A.K. Nandi, 2007 "Segmentation of retinal blood vessels using a novel clustering algorithm (RACAL) with a partial supervision strategy". Medical Biology Engineering and Computing. Vol 45, pp. 261-273.

C. Sinthanayothin, J.F. Boyce, H.L. Cook, T.H. Williamson, 1999. "Automated localization of the optic disc, fovea, and retinal blood vessels from digital colour fundus images". Br J Ophthalmol. Vol 83, pp. 902-910.

O. Smart, I.G. Tsoulos, D. Gavrilis, G. Georgoulas. 2011. "Grammatical Evolution for Features of Epileptic Oscillations in Clinical Intracranial Electroencephalograms". Expert System with Application. Vol 38, No 8, pp: 9991-9999.

- J. Staal, M.D. Abràmoff, M. Niemeijer, M.A. Viergever, B.Ginneken, 2004 “Ridge-based vessel segmentation in color images of the retina”. IEEE transactions on medical imaging, vol 23, No.4, pp. 501-509.
- S. Tamura, Y.Okamoto, K. Yanashima, 1988. “Zero-crossing interval correction in tracing eye-fundus blood vessels”. IEEE Pattern Recognition, Vol. 21, No3, pp. 227-233,
- X. Tan, B. Triggs, 2010. “Enhanced local texture feature sets for face recognition under difficult lighting conditions”. IEEE transactions on image processing, Vol 19, No 6. Pp. 1635-1650.
- Y.A. Tolias, S.M. Pannas, 1998. “A fuzzy vessel tracking algorithm for retinal images based on fuzzy clustering”. IEEE transactions on medical imaging, vol 17, no 2, pp. 263-273.
- I.G. Tsoulos, D. Gavrilis, E. Glavas, 2005. “Neural network construction using grammatical evolution.” Signal Processing and Information Technology. Proceedings of fifth IEEE International symposium on signal processing and information technology, pp. 827-831.
- I.G. Tsoulos, D. Gavrilis, E. Glavas, 2008. “Neural network construction and training using grammatical evolution”. Neurocomputing 72, pp. 269-277.
- I.G. Tsoulos, D. Gavrilis, E. Glavas, 2009. “Solving differential equations with constructed neural networks”. Neurocomputing 72, pp. 2385-2391.
- A.W. Whitney, 1971. “A direct method of non parametric measurement selection”. IEEE Trans. Comput., Vol C-20, pp. 1100-1103.
- X. Yao. “Evolving Artificial Neural networks”, 1999. Proceedings of the IEEE IEEE, Vol 87, issue 9, pp. 1423-1447.

F. Zana, J. C. Klein. 1999. "A multimodal Registration Algorithm of eye fundus images using vessels detection and Hough transform". IEEE Transactions on medical imaging, Vol. 18, No 5. Pp. 419-428.

F. Zana, J. C. Klein, 2001. Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation. IEEE Transactions on Image Processing, vol 10, No 7. Pp..1010-1019.

G. Zhao, G. Wu, Y. Liu, J. Chen, 2011 "Texture classification based on completed modeling of local binary pattern." International Conference on Computational and Information Sciences, pp. 268-271.

L. Zhou, M. Rzeszotarski, L., Singerman, J. Chokreff, 1994. "The detection and quantification of retinopathy using digital angiograms". IEEE Transactions on medical imaging, Vol 13, pp. 619-626.