Ridge-Based Vessel Segmentation in Color Images of the Retina

Joes Staal*, *Associate Member, IEEE*, Michael D. Abràmoff, *Member, IEEE*, Meindert Niemeijer,
Max A. Viergever, *Member, IEEE*, and Bram van Ginneken, *Associate Member, IEEE*

Abstract—A method is presented for automated segmentation of vessels in two-dimensional color images of the retina. This method can be used in computer analyses of retinal images, e.g., in automated screening for diabetic retinopathy. The system is based on extraction of image ridges, which coincide approximately with vessel centerlines. The ridges are used to compose primitives in the form of line elements. With the line elements an image is partitioned into patches by assigning each image pixel to the closest line element. Every line element constitutes a local coordinate frame for its corresponding patch. For every pixel, feature vectors are computed that make use of properties of the patches and the line elements. The feature vectors are classified using a kNN-classifier and sequential forward feature selection. The algorithm was tested on a database consisting of 40 manually labeled images. The method achieves an area under the receiver operating characteristic curve of 0.952. The method is compared with two recently published rule-based methods of Hoover *et al.* [1] and Jiang *et al.* [2]. The results show that our method is significantly better than the two rule-based methods ($p < 0.01$). The accuracy of our method is 0.944 versus 0.947 for a second observer.

Index Terms—Classification, convex sets, feature selection, fundus, image primitives, retina, ridges, vessel segmentation.

I. INTRODUCTION

ASSESSMENT of the characteristics of vessels plays an important role in a variety of medical diagnoses. For these tasks measurements are needed of e.g., vessel width, color, reflectivity, tortuosity, abnormal branching, or the occurrence of vessels of a certain width. When the number of vessels in an image is large, or when a large number of images is acquired, manual delineation of the vessels becomes tedious or even impossible.

The focus of this paper is on the automated segmentation of vessels in color images of the retina. These images, also known as fundus images, are acquired by making photographs of the back of the eye. We are interested in vessel segmentation for screening of diabetic retinopathy. Diabetes is a disease that affects about 5.5% of the population worldwide, a number that can

be expected to increase significantly in the coming years. About 10% of all diabetic patients have diabetic retinopathy, which is the primary cause of blindness in the Western World. Since this type of blindness can be prevented with treatment at an early stage, the WHO advises yearly ocular screening of patients. Automation will facilitate this screening [3].

Knowledge about the location of the vessels can aid in screening of diabetic retinopathy, e.g., to reduce the number of false positives in the detection of microaneurysms [4]–[6], to serve as a means for registration of images taken at different time instants or at different locations of the retina [7], or to find the location of the optic disc and the fovea.

Previous methods for vessel segmentation in images of the retina can be divided into two groups. The first group consists of rule-based methods and comprises vessel tracking [8]–[11], matched filter responses [1], [12], [13], grouping of edge pixels [14], model based locally adaptive thresholding [2], topology adaptive snakes [15] and morphology-based techniques [16]–[18]. The second group consists of supervised methods, which require manually labeled images for training. To the best of our knowledge, the only published method in this category is the neural network scheme for pixel classification by Sinthanayothin *et al.* [19], [20]. Our method belongs to the last category.

In our opinion, a pixel representation is not optimal for vessel structure. Therefore, our approach is based on the intrinsic property that vessels are elongated structures. This observation leads to a primitive-based method, which we refer to as PBM. Our algorithm uses image primitives formed from image ridges that are grouped into sets that approximate straight line elements. The sets are used for two purposes. First, features are computed which together with a classifier give a probability that the line element is part of a vessel. Second, the sets divide the image into patches by assigning every pixel of the image to its nearest primitive. Within each patch, the line element defines a local coordinate frame in which local features can be extracted for every pixel. The probability that the line element is part of a vessel is one of these features. The features are used to classify the pixels in the patch into vessel and nonvessel.

Many of the published methods have not been evaluated on large datasets or fail to give good results for large numbers of images as encountered in a screening process. In [9] and [19], evaluation is done on vessel segments and bifurcations. Only in [1] and [2] is an evaluation on complete manually labeled images presented.

We have constructed a database of manually labeled images for training and evaluation of our method. The database consists

of 40 images taken from a screening programme for diabetic retinopathy in the Netherlands. We compare our method with two rule-based methods. The first one is the method of Hoover *et al.* [1], the second one the method of Jiang *et al.* [2]. Hoover *et al.* have collected a database of manually labeled images, which is publicly available together with the results of their method. For comparison, our system is evaluated on their database too.

The paper is organized as follows. In Section II, a method for extracting ridges and line elements is developed. The line elements are used to subdivide the image into patches. Section III describes the classifier used and the features that are extracted. A description of the material used is given in Section IV. Section V presents the results, which are summarized and discussed in Section VI.

II. REPRESENTATION OF VESSELS

A. Ridge Detection

Since image ridges are natural indicators of vessels, we start our analysis with a short overview of ridge detection for two-dimensional gray value images. For a more extensive discussion on this subject, see [21] and [22]. The ridge detection method used in this paper is described in full detail in [22]. Because the green channel of color fundus images formatted as an RGB image gives the highest contrast between vessel and background [1], this channel is used for extraction of the image ridges.

Ridges are defined as points where the image has an extremum in the direction of the largest surface curvature. Stated otherwise, we search for the points in the image $L(\mathbf{x})$, with $\mathbf{x} = (x_1, x_2)^T$, where the first derivative of the intensity in the direction of the largest surface curvature changes sign.

The direction of largest surface curvature is the eigenvector $\hat{\mathbf{v}}$ of the matrix of second order derivatives of the image corresponding to the largest absolute eigenvalue λ. This matrix is referred to as the Hessian matrix $\mathbf{H}$. The sign of λ determines whether a local minimum ($\lambda > 0$) or a local maximum ($\lambda < 0$) is found.

Because taking derivatives of discrete images is an ill-posed operation, they are taken at a scale σ using the Gaussian scale-space technique [23], [24]. The main idea is that the image derivatives can be taken by convolving the image with derivatives of a Gaussian

$$
\begin{aligned}
L_{x_j} &= \frac{\partial L(\mathbf{x}, \sigma)}{\partial x_j} \\
&= \frac{1}{2\pi\sigma^2} \int_{\mathbf{x}' \in \mathbb{R}^2} \frac{\partial e^{-\|\mathbf{x}-\mathbf{x}'\|^2/2\sigma^2}}{\partial x_j} L(\mathbf{x}') \mathrm{d}\mathbf{x}'
\end{aligned}
\tag{1}
$$

where x_j is the image coordinate with respect to which the derivative is taken. Mixed and higher order derivatives are computed by taking mixed and higher order derivatives of the Gaussian kernel.

It is now possible to define a scalar field $\rho(\mathbf{x}, \sigma)$ over the image that takes value -1 for ridges of local minima, 1 for ridges of local maxima and 0 elsewhere as follows:

$$
\begin{aligned}
\rho(\mathbf{x}, \sigma) = -\frac{1}{2}\mathrm{sign}(\lambda)|\mathrm{sign}(\boldsymbol{\nabla}L(\mathbf{x}+\epsilon\hat{\mathbf{v}}, \sigma) \cdot \hat{\mathbf{v}}) \\
-\mathrm{sign}(\boldsymbol{\nabla}L(\mathbf{x}-\epsilon\hat{\mathbf{v}}, \sigma) \cdot \hat{\mathbf{v}})|
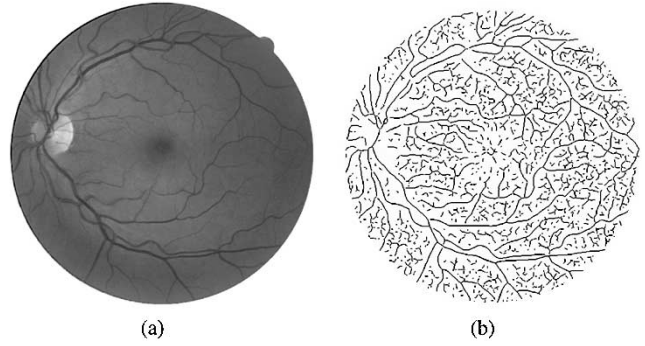\end{aligned}
\tag{2}
$$

Fig. 1. (a) Green channel of a fundus image obtained from a digital fundus camera. The diameter of the FOV is 540 pixels. (b) The local minima ridges of (a), $\sigma = 2.0$ pixel. A subset of the ridges coincide with the vessels.

where the gradient operator $\boldsymbol{\nabla}$ is defined as $(\partial/\partial x_1, \partial/\partial x_2)^T$ and $\lambda(\mathbf{x}, \sigma)$ is the largest eigenvalue by absolute value of $\mathbf{H} = \boldsymbol{\nabla}\boldsymbol{\nabla}^T L(\mathbf{x}, \sigma)$. In (2), λ and $\hat{\mathbf{v}}$ are evaluated at $(\mathbf{x}, \sigma)$. The parameter ϵ is the spatial accuracy with which the point-sets are detected. In the continuous case, the limit $\epsilon \to 0$ is taken, but in the discrete pixel case $\epsilon = 1.0$ pixel is a natural choice. The locations at which the gradients are evaluated are in general not on a grid point and linear interpolation is used to obtain the gradient values.

Fig. 1 shows an example of ridge detection in a fundus image. Since the vessels are dark structures, only local minima ridges are shown. In fluorescein angiography, where the vessels are the brighter structures, the local maxima ridges should be used.

B. Affine Convex Sets: Grouping Ridge Pixels

The next step in forming primitives for the vessels is a grouping of ridge pixels which belong to the same ridge. The aim is to obtain primitives which represent approximately straight line elements.

The grouping method is a simple region growing algorithm which compares an already grouped ridge pixel with ungrouped pixels in a neighborhood of radius ϵ_c, where the subscript "c" stands for connectivity. If no grouped pixel is available, a new one is selected randomly as seed from the remaining ungrouped ridge pixels. The comparison between the grouped and a candidate pixel within the neighborhood is based on two conditions: 1) The eigenvector directions of the ridge pixels should be similar and 2) If condition 1) is met, the pixels should be on the same ridge (and not on parallel ridges). The first condition can be checked by taking the scalar product of the eigenvectors $\hat{\mathbf{v}}$ at the location of the pixels. If the pixels have similar orientation the scalar product will be close to 1. The second condition can be checked by computing the unit-length normalized vector $\hat{\mathbf{r}}$ between the locations of the two pixels under consideration and taking the vector product between $\hat{\mathbf{r}}$ and $\hat{\mathbf{v}}$ of the grouped pixel. If the pixels are on the same segment, the vector product will be close to 1.

Mathematically, we check the following inequalities:

$$
\|\mathbf{x}_g - \mathbf{x}_u\| \le \epsilon_c
\tag{3}
$$

$$
|\hat{\mathbf{v}}(\mathbf{x}_g, \sigma) \cdot \hat{\mathbf{v}}(\mathbf{x}_u, \sigma)| \ge \epsilon_o
\tag{4}
$$

$$
\|\hat{\mathbf{v}}(\mathbf{x}_g, \sigma) \wedge \hat{\mathbf{r}}\| \ge \epsilon_P
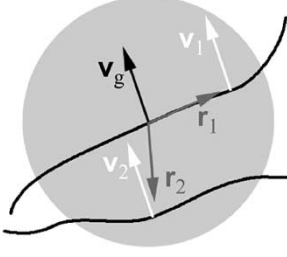\tag{5}
$$

Fig. 2. The dark curved lines are two ridges. The diameter of the disk is ϵ_c. $\mathbf{v}_g$ is the eigenvector belonging to a grouped pixel, $\mathbf{v}_1$ and $\mathbf{v}_2$ are the eigenvectors of still ungrouped pixels. The vectors $\mathbf{r}_1$ and $\mathbf{r}_2$ are unit vectors pointing from the grouped pixels to the ungrouped pixels. The pixel that belongs to the same ridge will be added to the group, because it satisfies the conditions in (3)–(5). The pixel on the parallel ridge does not satisfy condition (5) and will not be grouped.
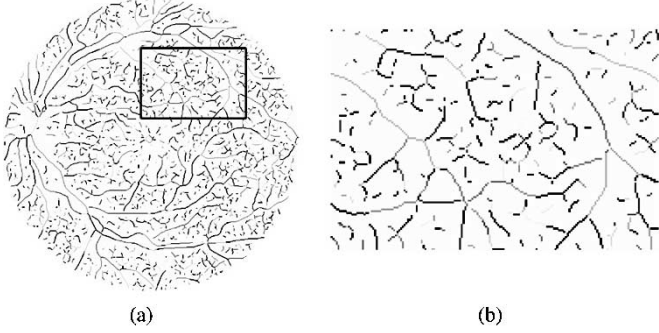

Fig. 3. (a) The convex sets of the ridges of Fig. 1(b). Every grouped set has its own color. (b) Blow up of (a). Note that the number of ridge pixels is equal to the number of ridge pixels in Fig. 1.

where the subscript "g" stands for grouped, "u" for ungrouped, "o" for orientation and "p" for parallelism. The ϵ's determine the measure for similarity, where $0 \leq \epsilon_c < \infty$ and $0 \leq \epsilon_o, \epsilon_p \leq 1$.

We refer to these sets of (x_1, x_2) coordinates as affine convex sets: convex because they approximate straight line elements and affine because of the geodesic convexity instead of straight line (Euclidean) convexity.

If the ridge detection is not perfect, ϵ_c can overcome the discontinuity caused by gaps in the ridges. The parameter ϵ_o controls the amount of curvature between two pixels. If it is set close to zero, highly curved convex sets are formed. If it is set close to one, straighter convex sets are obtained. The parameter ϵ_p prevents the grouping process to jump to parallel ridges. A value close to one is recommended. In this paper, $\epsilon_c = 3.0$ pixels, $\epsilon_o = 0.95$, and $\epsilon_p = 0.95$ is used. Fig. 2 illustrates the construction of the affine convex sets. In Fig. 3, the convex sets of the ridges of Fig. 1(b) are shown.

The kth element in convex set number i, consisting of K_i points, will be denoted by $\mathbf{c}_i(k), k = 1, \ldots, K_i$. The vector $\mathbf{c}(k)$ is the location $\mathbf{x}$ of the kth element in the set. The subscript i is omitted when no particular convex set is referred to. For every point in a convex set i there is a corresponding direction $\hat{\mathbf{v}}_i(k)$, the direction in which the ridge is detected.

Fig. 4. Convex set regions of the convex sets of Fig. 3(a).

C. Convex Set Regions

The image can now be partitioned into patches based on the convex sets. Every pixel is assigned to the convex set to which it is closest. Fig. 4 shows the result of this operation on the convex sets of Fig. 3(a). The patches are referred to as convex set regions (CSRs).

A fitted straight line through a convex set can be used as the main axis of a local coordinate frame for the pixels in the corresponding CSR.

III. FEATURES AND CLASSIFIERS

The goal of this work is to classify every pixel in an image as vessel or nonvessel. For this purpose labeled examples or training sets, features and a classifier are needed.

From the training sets feature vectors are constructed that can be labeled as vessel or nonvessel, so every feature vector belongs to one of two classes. The idea is that feature vectors from a particular class cluster together in the feature space and that a classifier can be designed that determines a decision boundary between the different classes. After the training, a nonlabeled feature vector can be classified by determining on which side of the decision boundary it is situated. With some classifiers it is possible to approximate the chance, given the features, that a pixel is vessel or not. This is called soft classification.

A. Classification and Performance

In initial experiments, three classifiers have been compared, a kNN-classifier, a linear classifier and a quadratic classifier [25]. Performance of the kNN-classifier was superior for all experiments, so this classifier has been selected. We use the optimized implementation for kNN-classifiers that is made available by Arya *et al.* [26].

Using k neighbors of which n are labeled as vessel the *a posteriori* probability for being part of a vessel is approximated as

$$p(\text{vessel}) = \frac{n}{k}. \qquad (6)$$

This converges to the true probability in the limit of an infinite number of examples [25].

The performance of the system is measured with receiver operating characteristic (ROC) curves [27]. An ROC curve plots the fraction of pixels that is falsely classified as vessel against the fraction that is correctly classified as vessel. The fractions are determined by setting a threshold on the posterior probability. The closer a curve approaches the top left corner, the

better the performance of the system. A single measure to quantify this behavior is the area under the curve, A_z, which is 1 for a perfect system. A system that makes random classifications has an ROC curve that is a straight line through the origin with slope 1 and $A_z = 0.5$.

B. Feature Selection

In the following sections, many features will be defined and it is not known beforehand which ones will give good classification results and which ones will not. Therefore, feature selection is applied. The scheme used in this paper is the sequential forward selection method [28]. This algorithm starts with a null feature set and, for each step, the best feature that satisfies some criterion function is included with the current feature set. In this paper, A_z is taken as the criterion function. After all features have been included, the set that gives the best performance is chosen.

C. Convex Sets Features

In Fig. 3, the convex sets that are part of a vessel can be distinguished from the nonvessels by their local appearance. For that reason, features based on the profiles perpendicular to the convex sets seem to be a good choice. To extract profile information, a profile is sampled for every point k in a convex set from the image's green plane in the direction of the vector $\hat{\mathbf{v}}(k)$ with $\mathbf{c}(k)$ as midpoint. The obtained profiles are averaged and the mean profile $\psi(n)$, with $n = -N, \ldots, N$, is taken as the profile of the convex set. The averaging operation performs some smoothing, so blurring of the profile is not considered. Features that are extracted from the profile are as follows.

1) The height of the profile: $h = \psi(0)$.
2) The width of the profile defined as the distance between the strongest right and left edge of the profile: $w = n_{\text{re}} - n_{\text{le}}(n_{\text{re}} = \arg\max_{n>0} \psi'(n))$, with ψ' the first derivative of the profile. n_{le} is defined similar for $n < 0$).
3) The height divided by the width: h/w.
4) The edge strength, defined as: $s_{\text{e}} = \psi'(n_{\text{le}}) + \psi'(n_{\text{re}})$.
5) The edge strength divided by the width: s_{e}/w.
6) The edge height: $h_{\text{e}} = 1/2(\psi(n_{\text{le}}) + \psi(n_{\text{re}}))$.
7) The height minus the edge height: $h - h_{\text{e}}$.
8) The height divided by the edge height: h/h_{e}.

There are also features extracted that are not profile related, but might give useful information. The following are computed.

9) The distance between the first and last point of a convex set: $d = \|\mathbf{c}(1) - \mathbf{c}(K)\|$.
10) The length of a convex set: $l = \sum_{k=2}^{K} \|\mathbf{c}(k) - \mathbf{c}(k-1)\|$.
11) The curvature of a convex set, approximated by: $\kappa = \sum_{k=2}^{K} \hat{\mathbf{v}}(k) \cdot \hat{\mathbf{v}}(k-1)$.
12) A rectangular image patch of size $1.5 \times w$ by K is sampled in the green plane around a convex set. The mean μ_{g} for this patch is computed.
13) The standard deviation σ_{g} for the green patch.
14) The mean value of the green plane at the locations of the convex set divided by the the mean value of the red plane.
15) At different scales σ the mean value of $\lambda(\sigma)$ at the locations of the convex set: $\bar{\lambda}(\sigma) = 1/K \sum_{k=1}^{K} \lambda(\mathbf{c}(k), \sigma)$, this is a measure of ridge strength (see Section II-A).

D. CSR Pixel Features

Fitting a straight line through a convex set, a local coordinate system can be established for every CSR. For the origin of the coordinate the center of mass of the convex set is chosen. The first axis is along the direction of the fitted line, the second axis is perpendicular to the first axis and its direction is chosen so that a right hand oriented coordinate system is established. The features computed for the pixels in the CSR can be subdivided into features that take no information of the convex set into account [number 1)–3) in the list below], features that take convex set information into account [features 4)–9) below] and features that exploit the use of the local coordinate system [features 10)–12)].

1) The value of the red plane of the image at the pixel location: $r(\mathbf{x})$.
2) The value of the green plane of the image at the pixel location: $g(\mathbf{x})$.
3) The ratio of the green values and red values of the pixel: $g(\mathbf{x})/r(\mathbf{x})$.
4) The chance that the corresponding convex set belongs to a vessel: $p(\mathbf{c} = \text{vessel})$.
5) The distance between the pixel and the closest point on the convex set: $d_{\text{closest}} = \|\mathbf{x} - \mathbf{c}_{\text{closest}}\|$.
6) The difference in the red values of the pixel and the closest point on the convex set: $r(\mathbf{x}) - r(\mathbf{c}_{\text{closest}})$.
7) The ratio of the red values of the pixel and the closest point on the convex set: $r(\mathbf{x})/r(\mathbf{c}_{\text{closest}})$.
8) The difference in the green values of the pixel and the closest point on the convex set: $g(\mathbf{x}) - g(\mathbf{c}_{\text{closest}})$.
9) The ratio of the green values of the pixel and the closest point on the convex set: $g(\mathbf{x})/g(\mathbf{c}_{\text{closest}})$.
10) The coordinate of the pixel with respect to the first axis: x_1'.
11) The coordinate of the pixel with respect to the second axis: x_2'.
12) $L_{x_1'}, L_{x_2'}, L_{x_1'x_1'}$, and $L_{x_2'x_2'}$ at different scales σ. (Note that these are the number of scales $\times$ 4 features).

Not all of the features defined in this and the previous subsection are independent, but this is not a concern for the kNN-classifier used here.

In the databases we use, the blue channel is often empty or contains a lot of noise. Therefore, no features are extracted from that channel.

IV. MATERIAL

Two databases with images are used. The first one is obtained from a screening programme in the Netherlands and will be referred to as the Utrecht database. From this database 40 images are taken, containing 7 images with pathology (exudates, hemors, pigment epithelium changes). The images are captured in digital form from a Canon CR5 nonmydriatic 3CCD camera at $45°$ field of view. The images are of size 768×584 pixels, 8 bits per color channel and have a field of view (FOV) of approximately 540 pixels in diameter. The images are in compressed JPEG-format, which is unfortunate for image processing but is commonly used in screening practice. Fig. 1(a) shows an example of the green channel of such a fundus image.

The images were manually segmented by three observers, a computer science student, the third and the first author.[1] They were asked to mark all pixels for which they were for at least 70% certain that they were vessel. The observers were trained by an ophthalmologist, the second author.

The 40 images have been divided into a train and test set, each containing 20 images. The images in the train set were segmented by the first (14 images) and second observer (6 images). The images in the test set were segmented twice, resulting in a set A and a set B. The images in set A were labeled by the first (13 images) and second observer (7 images). Set B was segmented by the last observer.

The train set contains 3 images with pathology. Performance is computed with respect to the test set (the segmentations of set A are used as ground truth). The observers of set A marked 577 649 pixels as vessel and 3 960 494 as background (12.7% vessel), for set B these numbers are 556 532 and 3 981 611, respectively (12.3% vessel).

The second database has been collected by Hoover *et al.* [1] and consists of 19 images. This database will be referred to as the Hoover database. The Hoover images are digitized slides captured by a TopCon TRV-50 fundus camera at $35°$ field of view. The slides were digitized to 700×605 pixels, 8 bits per color channel. The FOV in the images are approximately 650×550 pixels in diameter. Two observers manually segmented all images. The first observer segmented 615 726 pixels as vessel and 5 293 034 as background (10.4% vessel), the second observer marked 879 695 pixels as vessel and 5 029 065 as background (14.9% vessel). Nine images contain pathology. The first observer segmented far less (small) vessels than the second observer and there is a large variability between the observers. Performance is computed with the segmentations of the first observer as ground truth.

Hoover *et al.* used the complete image for measuring the performance. Since the dark background outside the FOV is easily extracted, in this paper all experiments are done on the FOV only.

We compare the performance of our algorithm to that of Hoover *et al.* [1] using their publicly available results. To compare our system with the method of Jiang *et al.* [2] we implemented that method.

V. RESULTS

All experiments for the PBM are carried out with the following settings, which were found after a pilot study.

The ridges in the images are extracted from the green channel at scale $\sigma = 1.5$ pixel. To obtain approximately straight lines, the maximum size of the convex sets is set to 25 pixels.

Training sets for the convex sets are constructed by counting how many of their pixels intersect with the vessel pixels in the manually labeled ground truth images. If more than 50% of the pixels in a convex set intersect they are labeled as vessel, else as nonvessel.

The profile features for the convex sets are computed with a half profile width $N = 15$, i.e., the profile consists of 31 pixels.

[1]The images and their manual segmentations are made publicly available at http://www.isi.uu.nl/Research/Databases/DRIVE/.

To compensate for the lighting variations and to enhance local contrast, the pixels of every color channel C_i of the images are locally normalized to zero mean and unit variance

$$N_i(\mathbf{x}, \sigma) = \frac{C_i(\mathbf{x}) - \mathcal{E}_\sigma\{C_i\}(\mathbf{x})}{\sqrt{\mathcal{E}_\sigma\{C_i^2\}(\mathbf{x}) - \mathcal{E}_\sigma^2\{C_i\}(\mathbf{x})}} \tag{7}$$

with

$$\mathcal{E}_\sigma\{C\}(\mathbf{x}) = \frac{1}{2\pi\sigma^2} \int_{\mathbf{x}' \in \mathbb{R}^2} C(\mathbf{x}') \mathrm{e}^{-\|\mathbf{x}-\mathbf{x}'\|^2/2\sigma^2} \mathrm{d}\mathbf{x}' \tag{8}$$

acting as a local averaging operator. A value of $\sigma = 8.0$ pixels is used. In [19], a similar filtering is performed with a square filter.

A. Settings

The ridge measures are extracted from the green channel and scales $\sigma = 0.5, 1.0, 2.0$, and 4.0 pixels are used.

With these settings a total 18 features per convex set is extracted. For the training of the classifier, only every fourth convex set is taken. This reduces computation time and memory resources.

For the computation of the features for the CSR the following settings are used. In the training phase, the *a posteriori* probabilities for the convex sets are computed using (6) in a leave-one-out fashion, i.e., the convex sets of one image are classified with a classifier trained on the convex sets of the other images in the training set.

The derivatives with respect to the local coordinate systems are taken at scales $\sigma = 0.5, 1.0, 2.0$, and 4.0, resulting in 27 features. For the training, only every fourth pixel in the x_1 and x_2-directions is used.

The feature selection is also done on a leave-one-out basis. This is done for every image and the A_z value of all images is averaged to obtain a criterion upon which it is decided to include a feature or not.

The kNN-classifiers for the classification of the convex sets and the CSR use $k = 101$.

Because kNN-classifiers are sensitive to scaling between different features, in all experiments each feature is normalized independently to zero mean and unit variance.

B. The Utrecht Database

The convex sets detection on the train set gives 12 648 vessel segments and 48 838 nonvessel segments (20.6% vessel).

The results of feature selection on the train set are presented in Table I for the convex sets and in Table II for the CSR. Every row in the table gives the performance of the best feature that is added to the feature set (so including the previously selected features). In the last line, the performance is shown when all features are included.

Fig. 5 shows the ROC curves for the segmentations obtained by the PBM and the method of Jiang *et al.* on the images of the test set. The performance of the observers of set A versus B is also plotted, together with the result of thresholding the classified images at $p = 0.5$. This is equivalent to a hard classification in two classes.

TABLE I
SELECTED FEATURES FOR THE CONVEX SETS. FOR EVERY ADDED FEATURE
THE OBTAINED AREA UNDER THE ROC CURVE IS GIVEN

Utrecht database		Hoover database	
	A_z		A_z
λ, $\sigma = 2.0$	0.843	λ, $\sigma = 2.0$	0.934
h	0.864	d	0.945
κ	0.870	μ_g	0.948
σ_g	0.871	λ, $\sigma = 4.0$	0.947
λ, $\sigma = 1.0$	0.872	λ, $\sigma = 1.0$	0.948
μ_g	0.873	h_e	0.950
λ, $\sigma = 4.0$	0.874	all features	0.940
all features	0.864		

TABLE II
SELECTED FEATURES FOR THE CONVEX SET REGIONS. FOR EVERY ADDED
FEATURE THE OBTAINED AREA UNDER THE ROC CURVE IS GIVEN

Utrecht database		Hoover database	
	A_z		A_z
$p(\mathbf{c} = \text{vessel})$	0.8755	$p(\mathbf{c} = \text{vessel})$	0.8955
$g(\mathbf{x}) - g(\mathbf{c}_{\text{closest}})$	0.9314	$g(\mathbf{x})/g(\mathbf{c}_{\text{closest}})$	0.9653
d_{closest}	0.9372	d_{closest}	0.9677
$L_{x_1'}$, $\sigma = 1.0$	0.9437	$L_{x_1'x_1'}$, $\sigma = 2.0$	0.9683
$L_{x_2'}$, $\sigma = 2.0$	0.9463	$L_{x_2'x_2'}$, $\sigma = 2.0$	0.9680
$L_{x_2'x_2'}$, $\sigma = 1.0$	0.9472	$r(\mathbf{x})/r(\mathbf{c}_{\text{closest}})$	0.9682
$L_{x_1'x_1'}$, $\sigma = 2.0$	0.9485	$L_{x_1'}$, $\sigma = 1.0$	0.9684
$r(\mathbf{x})/g(\mathbf{x})$	0.9490	all features	0.9589
$L_{x_2'x_2'}$, $\sigma = 2.0$	0.9497		
$L_{x_2'}$, $\sigma = 1.0$	0.9498		
$g(\mathbf{x})$	0.9500		
all features	0.9493		

Fig. 6 shows for the PBM the images with the highest and lowest A_z, the results of hard classification of these images and the corresponding manually segmented images for the observers of set A and set B.

In Table III, an overview is given of the results of the different methods. The upper part of the table shows the A_z values. The A_z value for the method of Jiang *et al.* is approximated by adding the points $(0, 0)$ and $(1, 1)$ to the curve. The lower part of the table lists the accuracies for the different methods.

For comparison, the last row shows the results when all pixels are segmented as the most likely class, in this case the background.

Paired t-tests on the A_z values for the individual images of the database, show that the PBM performs significantly better than Jiang *et al.* with $p < 0.01$. Using the accuracies of each image, all values in Table III are significantly different from each other with $p < 0.01$, except for Jiang *et al.* versus the most likely class, where $p < 0.02$.

A t-test for the PBM shows no significant differences when the results of the images with pathology are compared to those without pathology.

C. The Hoover Database

Because no independent test set is available for the Hoover database, leave-one-out experiments are performed, i.e., every image is classified using the other 18 images as training set. However, the computation time required for feature selection is
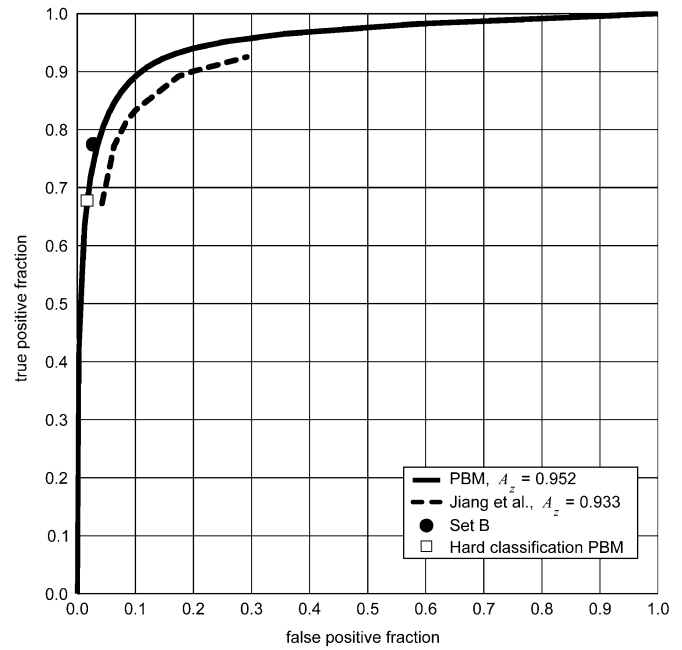

Fig. 5. Results for the Utrecht database. The PBM gives $A_z = 0.952$ and the method of Jiang *et al.* [2] $A_z = 0.933$. Comparing set B to set A, false and true positive fractions of $(0.0275, 0.775)$ are found. Performing a hard classification on the results of the PBM gives $(0.017, 0.678)$ for the false and true positive fractions.

very large and is, therefore, only done once for both the convex sets and the CSR, using 9 of the 19 images (4 with and 5 without pathology).

Tables I and II list the features that have been selected. In the train set 8615 convex sets are vessel, and 57 996 nonvessel (12.9% vessel).

After the feature selection, the images are segmented in leave-one-out experiments.

The ROC curves for the PBM and for the methods of Jiang *et al.* and Hoover *et al.* are shown in Fig. 7. The results of observer 1 versus observer 2 and of hard classification are plotted too.

Table III shows the A_z values and the accuracies for the different methods. Paired t-tests on the A_z values show that the PBM performs significantly better than the two other methods and that the method of Jiang *et al.* is significantly better than the method of Hoover *et al.*, all with $p < 0.01$. For the accuracies, the PBM is significantly better than the other methods (including the second observer) with $p < 0.01$. The method of Hoover *et al.* is significantly better than Jiang *et al.* and than classifying every pixel to its most likely class with $p < 0.01$, but differs only significantly with the second observer with $p < 0.1$. The method by Jiang *et al.* is not significantly better than classifying the pixels to their most likely class.

The PBM performs significantly better on the images without pathology than on those with pathology ($A_z = 0.969$ versus $A_z = 0.950$).

VI. DISCUSSION AND CONCLUSION

The results show that the method proposed in this paper (the PBM) outperforms the previously published rule-based methods

(a) PBM, $A_z = 0.972$ (b) Hard classification, acc. = 0.962   (c) Observer set A   (d) Observer set B

(e) PBM, $A_z = 0.937$ (f) Hard classification, acc. = 0.934   (g) Observer set A   (h) Observer set B
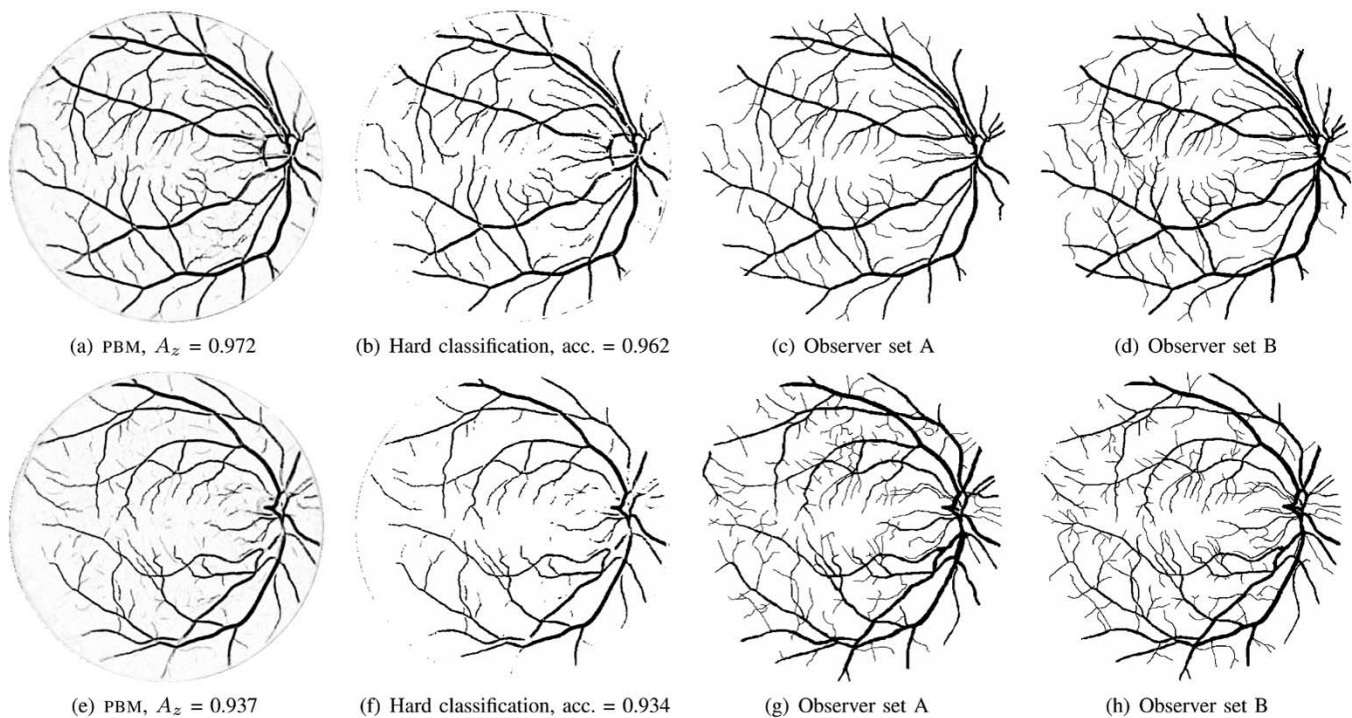
Fig. 6. First column: Best and worst result of the PBM. The grey value denotes the prior probability of the pixel being vessel (bright for the lower values and dark for the higher values). Second column: hard classification. Third column: observer from set A. Fourth column: observer from set B.

TABLE III
RESULTS FOR THE DIFFERENT DATABASES AND METHODS. ROWS 1–3 GIVE THE AREA UNDER THE ROC-CURVES, ROWS 4–8 THE ACCURACY (THE SUM OF THE NUMBER OF CORRECTLY CLASSIFIED FOREGROUND AND BACKGROUND PIXELS, DIVIDED BY THE TOTAL NUMBER OF PIXELS)

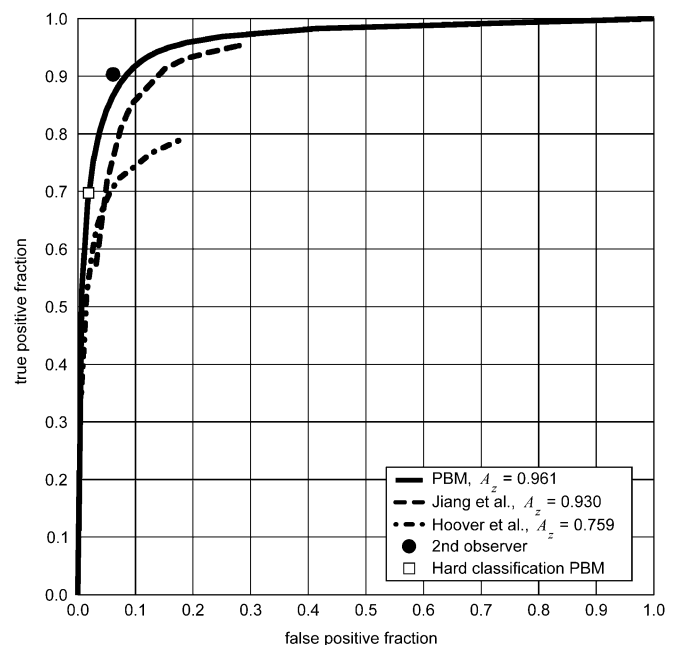| Criterion	Method	Database	
		Utrecht	Hoover
A_z	Hoover		0.7590
	Jiang	0.9327	0.9298
	PBM	0.9520	0.9614
Accuracy	2nd obs.	0.9473	0.9351
	Hoover		0.9275
	Jiang	0.8911	0.9009
	PBM	0.9441	0.9516
	Most likely class	0.8727	0.8958

Fig. 7. Results for the Hoover database. $A_z = 0.961$ for the PBM, $A_z = 0.930$ for the algorithm by Jiang *et al.* [2] and $A_z = 0.759$ for the algorithm by Hoover *et al.* [1]. Comparing the second observer to the first observer, false and true positive fractions of (0.061, 0.903) are found. Performing a hard classification on the results of the PBM gives (0.019, 0.697) for the false and true positive fractions.

by Hoover *et al.* [1] and by Jiang *et al.* [2]. A possible explanation is that it is very difficult to come up with good rules for problems where many features interact in a complex manner. Supervised methods with feature selection like the PBM seem to be better equipped for these tasks. In the Hoover database, this is illustrated by the fact that the accuracy of the PBM is higher than that of an independent second observer. The PBM is able to adapt to the first observer, who disagrees especially on the number of small vessels with the second observer.

A disadvantage of supervised methods is the need for (manually) labeled training data. For the Utrecht database it took an observer 2 hours on average to label a single image.

Figs. 5 and 7 and Table III demonstrate that it is possible to design a system that approaches the performance of human observers. It can be noticed from these figures that the second observer does not achieve an accuracy of 1, because the first and second observer disagree in their manual segmentations. One

reason is that due to JPEG-artifacts it is hard to discern vessels of pixel or subpixel width. Another reason is the subjective decision that the human observers must make regarding pixels at the border of the vessels. Do they belong to the vessel or not?

(a) Image with pathology

(b) Ground truth

(c) Hard classification of the PBM

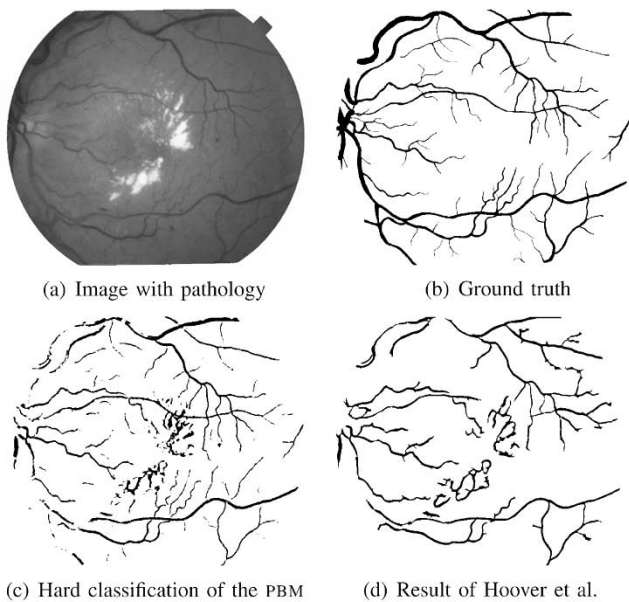(d) Result of Hoover et al.

Fig. 8.   Results on an image with pathology from the Hoover database.

Asking the observers to rate the probability for such pixels being vessel would increase the time required for the segmentation prohibitively.

Two types of errors can be distinguished. The first type is over- and undersegmentation of the vessels. This is important in applications where determination of the vessel width is needed. The second type of error is the missing or erroneous detection of vessel branches. Suppose for example that two observers both label a small vessel, but in one of the segmentations the vessel is slightly shifted so that both vessels do not overlap. This will degrade accuracy more than if one of the observers had labeled no vessel at all. Type 2 errors are expected to be encountered with small vessels mostly. Removing the small vessels by a morphological opening, the type 1 error for large vessels can be approximated. The accuracy between first and second observer increased from 0.947 to 0.964 in the Utrecht database and from 0.935 to 0.948 in the Hoover database.

A perfect system would yield $A_z = 1$, which is not reached by our method. There are a few reasons. First, in all images high probability for being vessel is found around the boundaries of the FOV, while there are no vessels; see Fig. 6(a) and (e) for an example. This is an artifact of the method, introduced by the blurring that is needed for detecting the ridges of the image. Second, in some images part of the boundary of the optic disc is marked as vessel, cf. Fig. 6(a), while a human observer can clearly discern between the two. This is an indication that the amount of training data might be insufficient. Third, related to the last observation, is the appearance of (severe) pathology of which Fig. 8 gives an example. The method selects with high probability some vessels between bright areas where there are clearly no vessels. More training data that includes the various types of pathology encountered in screening practice might overcome this problem. No experiments were done to investigate the influence of the amount of training data. However, certain types of pathology might be detected and labeled in a preprocessing step. A fourth reason for imperfect performance

of the method can be the incorrect labeling of the convex sets in the training sets. Or even worse, no ridge is detected at the location of a vessel, which can happen for very small vessels. This can occur because the locations of the ridges are perturbed by the blurring that is needed for their detection. For large vessels, the detected ridge pixels are still within the vessel, but for small vessels they can be a little off. A possible solution might be to do ridge detection with scale selection, like in [24]. Another interesting reason for scale selection is the use of scale as a feature.

For the PBM it takes about 15 min to segment an image in the Utrecht database on a Pentium-III PC, running at 1.0 GHz with 1-GB memory. Most time is spent in the classification. However, our implementation is experimental and could be optimized. For example, Table II shows that for both databases, the most important feature is the probability that a convex set is part of a vessel. It can be expected that for a set with low probability of being vessel the pixels in its CSR will have low probability. This measure can be used to speed up the segmentation by not processing CSR belonging to convex sets with a low chance of being vessel. In the Utrecht database, we observed that only about 10% of the convex sets have a probability higher than 0.5. In [2], processing times of about half a minute are reported. Computation times for [1] are not available.

In this paper, evaluation has been done using accuracy of hard classifications and A_z values of soft classifications. Other evaluation measures might be more appropriate, depending on the application at hand. For example, if one is interested in examining the tortuosity of the vessels, the width of the vessels might not be important, only the centerlines. The measures used do not take into account the number of branches, the connectedness of the vessels or the number of branching points, which all might be relevant in specific applications. Another good evaluation measure might be the performance of a computer-aided diagnosis system for retinal images that uses the results of a vessel segmentation algorithm in its analysis.

REFERENCES

[1] A. Hoover, V. Kouznetsova, and M. Goldbaum, "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response," *IEEE Trans. Med. Imag.*, vol. 19, pp. 203–210, Mar. 2000.

[2] X. Jiang and D. Mojon, "Adaptive local thresholding by verification-based multithreshold probing with application to vessel detection in retinal images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, pp. 131–137, Jan. 2003.

[3] D. C. Klonoff and D. M. Schwartz, "An economic analysis of interventions for diabetes," *Diabetes Care*, vol. 23, no. 3, pp. 390–404, 2000.

[4] T. Spencer, J. A. Olson, K. C. McHardy, P. F. Sharp, and J. V. Forrester, "An image-processing strategy for the segmentation and quantification of microaneurysms in fluorescein angiograms of the ocular fundus," *Comput. Biomed. Res.*, vol. 29, no. 4, pp. 284–302, 1996.

[5] A. J. Frame, P. E. Undrill, M. J. Cree, J. A. Olson, K. C. McHardy, P. F. Sharp, and J. V. Forrester, "A comparison of computer based classification methods applied to the detection of microaneurysms in ophthalmic fluorescein angiograms," *Comput. Biol. Med.*, vol. 28, no. 3, pp. 225–238, 1998.

[6] M. Larsen, J. Godt, N. Larsen, H. Lund-Andersen, A. K. Sjølie, E. Agardh, H. Kalm, M. Grunkin, and D. R. Owens, "Automated detection of fundus photographic red lesions in diabetic retinopathy," *Investigat. Opht. Vis. Sci.*, vol. 44, no. 2, pp. 761–766, 2003.

[7] F. Zana and J. C. Klein, "A multimodal registration algorithm of eye fundus images using vessels detection and Hough transform," *IEEE Trans. Med. Imag.*, vol. 18, pp. 419–428, May 1999.

[8] O. Chutatape, L. Zheng, and S. Krishman, "Retinal blood vessel detection and tracking by matched Gaussian and Kalman filters," in *Proc. IEEE Int. Conf. Eng. Biol. Soc.*, vol. 20, 1998, pp. 3144–3149.

[9] Y. A. Tolias and S. M. Panas, "A fuzzy vessel tracking algorithm for retinal images based on fuzzy clustering," *IEEE Trans. Med. Imag.*, vol. 17, pp. 263–273, Apr. 1998.

[10] A. Can, H. Shen, J. N. Turner, H. L. Tanenbaum, and B. Roysam, "Rapid automated tracing and feature extraction from retinal fundus images using direct exploratory algorithms," *IEEE Trans. Inform. Technol. Biomed.*, vol. 3, pp. 125–138, June 1999.

[11] L. Gagnon, M. Lalonde, M. Beaulieu, and M.-C. Boucher, "Procedure to detect anatomical structures in optical fundus images," *Proc. SPIE Med. Imaging: Image Processing*, vol. 4322, pp. 1218–1225, 2001.

[12] S. Chaudhuri, S. Chatterjee, N. Katz, M. Nelson, and M. Goldbaum, "Detection of blood vessels in retinal images using two-dimensional matched filters," *IEEE Trans. Med. Imag.*, vol. 8, pp. 263–269, Sept. 1989.

[13] L. Gang, O. Chutatape, and S. M. Krishnan, "Detection and measurement of retinal vessels in fundus images using amplitude modified second-order Gaussian filter," *IEEE Trans. Biomed. Eng.*, vol. 49, pp. 168–172, Feb. 2002.

[14] A. Pinz, S. Bernögger, P. Datlinger, and A. Kruger, "Mapping the human retina," *IEEE Trans. Med. Imag.*, vol. 17, pp. 606–619, Aug. 1998.

[15] T. McInerney and D. Terzopoulos, "T-snakes: Topology adaptive snakes," *Med. Imag. Anal.*, vol. 4, no. 2, pp. 73–91, 2000.

[16] M. E. Martínez-Pérez, A. D. Hughes, A. V. Stanton, S. A. Thom, A. A. Bharath, and K. H. Parker, "Scale-space analysis for the characterization of retinal blood vessels," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI'99*, C. Taylor and A. Colchester, Eds. Berlin, Germany : Springer-Verlag, 1999, pp. 90–97. ser. Lecture Notes in Computer Science.

[17] T. Walter and J. C. Klein, "Segmentation of color fundus images of the human retina: Detection of the optic disc and the vascular tree using morphological techniques," in *Medical Data Analysis*, J. Crespo, V. Maojo, and F. Martin, Eds. Berlin, Germany: Springer-Verlag, 2001, pp. 282–287. ser. Lecture Notes in Computer Science.

[18] F. Zana and J. C. Klein, "Segmentation of vessel-like patterns using mathematical morphology and curvature evaluation," *IEEE Trans. Image Processing*, vol. 10, no. 7, pp. 1010–1019, 2001.

[19] C. Sinthanayothin, J. F. Boyce, H. L. Cook, and T. H. Williamson, "Automated localization of the optic disc, fovea, and retinal blood vessels from digital color fundus images," *Br. J. Ophthalmol.*, vol. 83, no. 8, pp. 902–910, 1999.

[20] C. Sinthanayothin, J. F. Boyce, T. H. Williamson, H. L. Cook, E. Mensah, S. Lal, and D. Usher, "Automated detection of diabetic retinopathy on digital fundus images," *Diabetic Med.*, vol. 19, no. 2, pp. 105–112, 2002.

[21] D. Eberly, *Ridges in Image and Data Analysis*, Dordrecht, The Netherlands: Kluwer Academic, 1996.

[22] S. N. Kalitzin, J. J. Staal, B. M. ter Haar Romeny, and M. A. Viergever, "A computational method for segmenting topological point sets and application to image analysis," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, pp. 447–459, May 2001.

[23] L. M. J. Florack, *Image Structure*, Dordrecht, The Netherlands: Kluwer Academic, 1997.

[24] T. Lindeberg, *Scale-Space Theory in Computer Vision*, Dordrecht, The Netherlands: Kluwer Academic, 1994.

[25] R. O. Duda, P. E. Hart, and H. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley-Interscience, 2001.

[26] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions," *J. ACM*, vol. 45, no. 6, pp. 891–923, 1998.

[27] C. E. Metz, "Basic principles of ROC analysis," *Seminars Nucl. Med.*, vol. 8, no. 4, pp. 283–298, 1978.

[28] A. W. Whitney, "A direct method of non parametric measurement selection," *IEEE Trans. Comput.*, vol. C-20, pp. 1100–1103, 1971.