An Active Contour Model for Segmenting and Measuring Retinal Vessels

Bashir Al-Diri, Andrew Hunter*, and David Steel

Abstract—**This paper presents an algorithm for segmenting and measuring retinal vessels, by growing a "Ribbon of Twins" active contour model, which uses two pairs of contours to capture each vessel edge, while maintaining width consistency. The algorithm is initialized using a generalized morphological order filter to identify approximate vessels centerlines. Once the vessel segments are identified the network topology is determined using an implicit neural cost function to resolve junction configurations. The algorithm is robust, and can accurately locate vessel edges under difficult conditions, including noisy blurred edges, closely parallel vessels, light reflex phenomena, and very fine vessels. It yields precise vessel width measurements, with subpixel average width errors. We compare the algorithm with several benchmarks from the literature, demonstrating higher segmentation sensitivity and more accurate width measurement.**

Index Terms—**Parametric active contour, retinal vessel segmentation.**

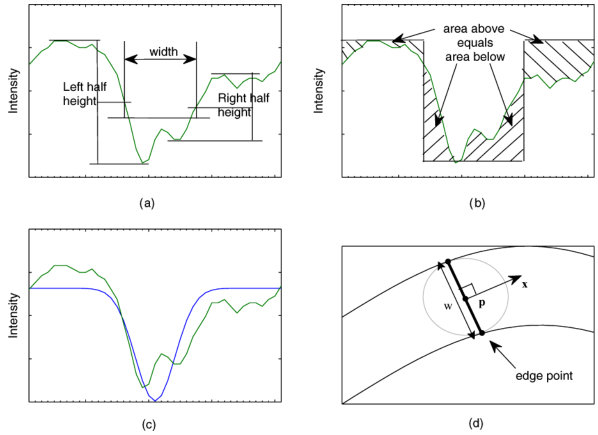


Fig. 1. (a)–(c) 1-D Profile measurement algorithms and (d) profile definition. (a) HHFW, (b) Gregson rectangular fit, (c) Gaussian model fit, (d) Profile: center, width, and direction.

I. Introduction

THIS paper presents a new algorithm for the segmentation and measurement of retinal vessels, the *extraction of segment profiles* (ESP) algorithm. In contrast to many previous algorithms from the literature, ESP integrates vessel segmentation and width measurement. The algorithm may be of use in the analysis of vascular diseases that cause measurable changes to the geometry of retinal vessels, including proliferative diabetic retinopathy (venous beading, tortuosity) [1], hypertension (arterial narrowing) [2], and arteriosclerosis (focal arterial narrowing) [3]. To support such analysis the retinal vasculature needs to be segmented with very high reliability. Given that retinal vessels are often only a few pixels across, width measurements must as precise as possible to detect symptomatic changes. The ESP algorithm combines reliable segmentation with subpixel average width measurement error.

A number of vessel segmentation techniques have been suggested in the literature, although they typically concentrate on segmenting a pixel map of the vasculature, leaving the issue of measurement to be treated separately. All exploit the characteristic local appearance of a vessel; some additionally exploit the global connectivity. Local detection methods include adaptive local thresholding [4] and the use of 2-D "vessel-shaped" matched filters [5], [6]. The use of pattern recognition techniques [7], [8] allows for relatively sophisticated local appearance modelling. Vascular connectivity is often exploited by "growing" algorithms, which progressively extend a model along the vessel. For example, [9] uses simple region-growing in conjunction with scale–space analysis. Hoover's seminal work combined local appearance modelling with vessel growing and pruning [10].

Algorithms to estimate the vessel diameter have largely been developed independently of vessel segmentation algorithms. The literature contains a number of width-measurement algorithms which the comparisons in [11] and [12] show to have subpixel average measurement error. Several of these are defined on 1-D cross-sectional intensity profiles of the vessel; see Fig. 1(a)–(c). The half-height full-width (HHFW) algorithm [13] defines the width as the distance between the points on the intensity curve at which the function reaches half its maximum/height value to either side of the estimated center point. The Gregson algorithm [14] fits a rectangle to the profile, setting the width so that the area under the rectangle is equal to the area under the profile. The intensity profile has an approximately Gaussian shape, and so 1-D gaussian model-fitting has also been used to measure the diameter [15]. This may be extended to an extruded 2-D Gaussian model [12], which is more robust and can also optimize the profile direction.

Parametric active contours have been explored for retinal vessel segmentation [16], and more widely for vascular segmentation [17], [18]. Outside the vascular literature, the

Manuscript received November 03, 2008; revised February 08, 2009. First published March 24, 2009; current version published August 26, 2009. *Asterisk indicates corresponding author*.

B. Al-Diri is with the Department of Computing and Informatics, University of Lincoln, LN6 7TS Lincoln, U.K.

*A. Hunter is with the Department of Computing and Informatics, University of Lincoln, LN6 7TS Lincoln, U.K. (e-mail: ahunter@lincoln.ac.uk).

D. Steel is with the Department of Ophthalmology, Sunderland Eye Infirmary, SR2 9HP Sunderland, U.K.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMI.2009.2017941

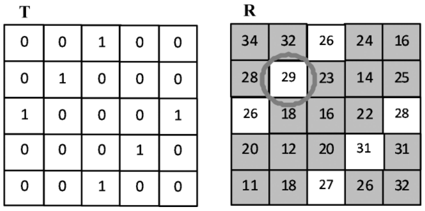
concept of linking two contours with a separating width has been proposed for linear object segmentation as the "ribbon snake" [19]. Separately, the use of twin contours converging from either side of a boundary, to identify blurred or difficult edges, and to detect convergence failure, have been presented as dual [20] and sandwich [21] snakes. The *Ribbon of Twins* (ROT) model, presented below, combines these two concepts to provide a robust retinal vessel segmentation algorithm, and is the first active contour algorithm to apply the width consistency principle to measurement of retinal vessels.

II. Extraction of Segment Profiles Algorithm

This section describes the *extraction of segment profiles* (ESP) algorithm. The algorithm has a number of stages. First, the *tramline* algorithm is used to locate an initial set of potential vessel segment centerline pixels. Second, the *segment growing* algorithm converts the tramline pixel map into a set of segments, each consisting of a series of *profiles*, while discarding false positive pixels. A profile is defined as lying across the vessel joining the two edges with the shortest possible distance, and orthogonal to the vessel direction; see Fig. 1(d). It may alternatively be described by its *center point* $\mathbf{p}$, *width* w, and *vessel direction vector* $\mathbf{x}$, or by its two *edge points*. The growing algorithm uses the ROT active contour model, described below, to progressively explore the vessel segment and add profiles. Third, the *junction resolution* algorithm extends the discrete segments resulting from the segment growing stage, and resolves various crossings, junctions and joinings.

A. Tramline Algorithm

The tramline algorithm [22] produces an approximate map of vessel centerline pixels using a fast, robust procedure, which is designed to produce very few false positives and to avoid identifying junctions. It uses a pair of oriented structuring elements—the inner element a straight line, seven pixels long and one pixel wide, the outer element a pair of length seven "tramlines" displaced five pixels to either side of the inner element, rotated to 12 orientations. It is a generalized morphological filter, related to the standard grey-scale morphological top-hat filter, but using order filtering rather than maximum and minimum operations to enhance robustness. The output of the filter is defined as follows. Let $\mathbf{T}$ be a binary filter mask of size $M \times M$, with $\#\mathbf{T}$ the number of nonzero elements, and let $\mathbf{R}$ be an $M \times M$ region of the image, centered at a given pixel. Let $S(i, \mathbf{R}, \mathbf{T})$ be the ith largest element of $\mathbf{R}$ masked by $\mathbf{T}$; see Fig. 2. Let $\mathbf{T}_j^I$ and $\mathbf{T}_j^O$ be the jth inner and outer tramline filters respectively. Then, the filter output is defined as $\bigwedge_j (S(\#\mathbf{T}_j^O - 2, \mathbf{R}, \mathbf{T}_j^O) - S(3, \mathbf{R}, \mathbf{T}_j^I) < \theta)$, where $\theta = 0$ is the filtering threshold. The underlying concept is that if the inner element lies inside a vessel and the outer elements outside, then all the pixels of the inner element should be darker than all the pixels of the outer element; the order filter discounts the two lightest and darkest pixels respectively for robustness. If any of the oriented filter differences passes the threshold, a vessel pixel has been found. The resulting pixel vessel map is morphologically cleaned up by speckle removal, skeletonization (to identify the approximate centerlines), spur-removal and removal of junction pixels. See Fig. 3 for sample output.



0	0	1	0	0
0	1	0	0	0
1	0	0	0	1
0	0	0	1	0
0	0	1	0	0

34	32	26	24	16
28	29	23	14	25
26	18	16	22	28
20	12	20	31	31
11	18	27	26	32

Fig. 2. Definition of the Tramline filter. A simplified outer structuring element, $\mathbf{T}$, and the corresponding pixels selected in region $\mathbf{R}$; the value selected by $S(2, \mathbf{R}, \mathbf{T})$, is circled—the second largest value in the area of $\mathbf{R}$ masked by $\mathbf{T}$.

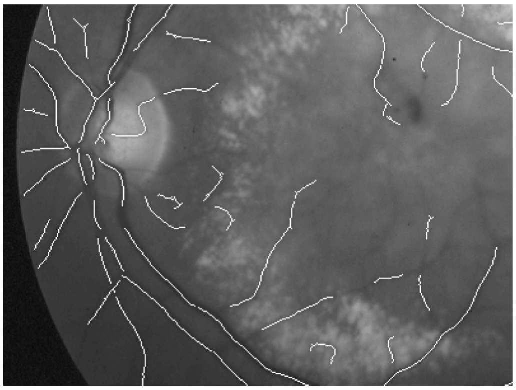


Fig. 3. Results of the tramline algorithm on a challenging section of the second image of the STARE database. Junctions, close overlaps and some fine, blurred segments are missed; false positives are very rare.

The tramline algorithm is robust, and particularly good at avoiding false positive segments. We tested it on the STARE database [23], which features many challenging and noisy images. To quantify performance, we morphologically thinned the STARE ground truth images and the output of Hoover's [10] algorithm. We dilated the ground truth with a 3 × 3 filter, and then compared the outputs of the tramline algorithm and Hoover's algorithm with the thickened ground truth centerlines (the thickening allows for slight misplacement of the filter outputs). The tramline algorithm achieved 57.2% sensitivity and precision rate 97.2%, against 57% sensitivity and precision rate 92.2% for Hoover's algorithm, where the sensitivity and precision rate are defined as $tp/(tp + fn)$ and $tp/(tp + fp)$, where tp, fp, and fn are true positive, false positive, and false negative pixel counts, respectively.

B. Ribbon of Twins

The ROT is an active contour based model to segment retinal vessels; we previously briefly introduced the ROT model in [24]. It is used as a base algorithm for the growing algorithm, described in the next section.

Each ROT contains four linked active contours, $\mathbf{v}_c(s, t) = (x_c(s, t), y_c(s, t)), c \in \{-2, -1, 1, 2\}$, where x and y are coordinate functions of the parameter $s \in [0, 1]$, and t is time. There are two contours inside the vessel, that move outwards toward the internal side of the edges ($\mathbf{v}_{\pm 1}$) and are linked together to maintain vessel width consistency (the "ribbon"); and two located outside the edges and moving toward the internal contours

($\mathbf{v}_{\pm 2}$), so that each edge is sandwiched within a "twin" of external and internal contours. The energy of the four contours are defined as

$$E_c(t) = \int_0^1 \left(E_c^{\mathrm{int}}(\mathbf{v}_c(s,t)) + E_c^{\mathrm{pho}}(\mathbf{v}_c(s,t)) + E_c^{\mathrm{rot}}(\mathbf{v}_c(s,t))\right) ds \quad (1)$$

where $E_c^{\mathrm{int}}, E_c^{\mathrm{pho}}, E_c^{\mathrm{rot}}$ are the internal, photometric, and ROT model energy functions, respectively.

The internal energy function E_c^{int} is defined [25] as

$$E_c^{\mathrm{int}}(\mathbf{v}_c(s,t)) = \frac{\alpha\|\mathbf{v}_c^{'}(s,t)\|^2 + \beta\|\mathbf{v}_c''(s,t)\|^2}{2} \quad (2)$$

where $\mathbf{v}^{'}(s,t)$ and $\mathbf{v}''(s,t)$ denote the first and second derivatives, and $\alpha = 0.3$ and $\beta = 0.3$ represent tension and rigidity.

The photometric energy E_c^{pho} is derived from the image to attract the contour towards edges. Considering the intensity image as a 2-D continuous function, $I(x,y)$, the photometric energy is defined [25] as

$$E_c^{\mathrm{pho}}(\mathbf{v}_c(s,t)) = -\frac{\kappa}{2}\|\mathrm{GVF}(G_\sigma * I(\mathbf{v}_c(s,t)))\|^2 \quad (3)$$

where G_σ is a 2-D Gaussian filter with standard deviation $\sigma = 0.8$, GVF is the gradient vector flow (as described in [26]), $*$ is the 2-D image convolution operator, and $\kappa = 0.6$ is a weighting coefficient. The ROT model energy E_c^{rot} links the four contours, attempting to capture the vessel's two edges from either side, while maintaining a consistent width; see Fig. 4. The internal contours are pushed outwards towards the estimated vessel width, $\tilde{w}$ (a parameter of the algorithm), using the term

$$E_{\pm 1}^{\mathrm{rot}}(\mathbf{v}_{\pm 1}(s,t)) = \frac{\delta}{2}(\|\mathbf{v}_1(s,t) - \mathbf{v}_{-1}(s,t)\| - \tilde{w})^2 \quad (4)$$

where $\delta = 0.6$ is a weighting coefficient. The external contours are pulled inwards towards the internal contours, using the term

$$E_{\pm 2}^{\mathrm{rot}}(\mathbf{v}_{\pm 2}(s,t)) = \frac{\delta}{2}\|\mathbf{v}_{\pm 2}(s,t) - \mathbf{v}_{\pm 1}(s,t)\|^2. \quad (5)$$

The contour that minimizes the energy satisfies the Euler–Lagrange equation [25]

$$\frac{\partial}{\partial s}\left(\alpha\frac{\partial \mathbf{v}_c}{\partial s}\right) - \frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2 \mathbf{v}_c}{\partial s^2}\right) - \nabla E_c^{\mathrm{pho}} - \nabla E_c^{\mathrm{rot}} = 0. \quad (6)$$

The model is discretized, with each contour represented by a number (N) of *control points*, $\mathbf{v}_{c,j}, j \in [1,N]$; these points are initialized so that the control points of each contour lie on a line parallel to the estimated vessel direction, $\mathbf{x}$, and each set of four corresponding control points lies on a line orthogonal to $\mathbf{x}$, with direction denoted $\mathbf{n}$; see Fig. 4.

The numerical scheme proposed by [25] approximates the derivatives in (6) with finite differences. The partial derivative of $\mathbf{v}$ with respect to t using the backward difference schema is set equal the left-hand side of (6) as

$$\gamma\frac{\partial \mathbf{v}_c}{\partial t} = \frac{\partial}{\partial s}\left(\alpha\frac{\partial \mathbf{v}_c}{\partial s}\right) - \frac{\partial^2}{\partial s^2}\left(\beta\frac{\partial^2 \mathbf{v}_c}{\partial s^2}\right) - \nabla E_c^{\mathrm{pho}} - \nabla E_c^{\mathrm{rot}} \quad (7)$$

where $\gamma = 1.0$ is a time step weight.

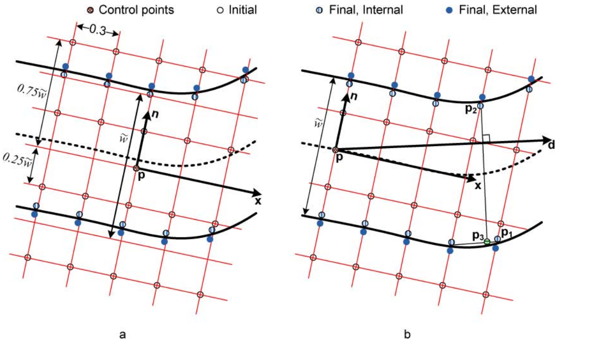


Fig. 4. The ROT model. (a) Initialization and convergence of the model for the first profile. The model is centered at the initial profile point, $\mathbf{p}$, aligned with estimated direction $\mathbf{x}$, based on estimated width $\tilde{w}$. (b) Convergence of the model during growing. The model is projected along the vessel direction $\mathbf{x}$; $\mathbf{d}$ denotes the new vessel direction, $\mathbf{p_2}$ and $\mathbf{p_3}$ the new profile edge points, on a line constructed orthogonal to $\mathbf{d}$.

The Euler equation (7) can be solved by iteration using central differences in space with respect to h (the discretized space plane representing contour resolution), and using the backward difference schema in time with respect to t (the discretized time step), so the resulting equation becomes (with the subscript c dropped for clarity)

$$\begin{aligned}\gamma\frac{\mathbf{v}_j^t - \mathbf{v}_j^{t-1}}{\Delta t} &= \alpha\frac{(\mathbf{v}_{j-1}^t - 2\mathbf{v}_j^t + \mathbf{v}_{j+1}^t)}{\Delta h^2} \\ &\quad - \beta\frac{(\mathbf{v}_{j-2}^t - 4\mathbf{v}_{j-1}^t + 6\mathbf{v}_j^t - 4\mathbf{v}_{j+1}^t + \mathbf{v}_{j+2}^t)}{\Delta h^4} \\ &\quad - \nabla E^{\mathrm{pho}} - \nabla E^{\mathrm{rot}}. \end{aligned} \quad (8)$$

Rearranging (8) and rewriting in matrix form yields:

$$\mathbf{v}_j^t(A + \gamma I) = \gamma\mathbf{v}_j^{t-1} - \Delta t\nabla E^{\mathrm{pho}} - \Delta t\nabla E^{\mathrm{rot}} \quad (9)$$

where $(A + \gamma I)$ is a pentadiagonal banded matrix; then

$$\mathbf{v}_j^t = (A + \gamma I)^{-1}\left(\gamma\mathbf{v}_j^{t-1} - \Delta t\nabla E^{\mathrm{pho}} - \Delta t\nabla E^{\mathrm{rot}}\right). \quad (10)$$

The pentadiagonal banded matrix A is given by

$$A = \begin{bmatrix} c+b & d+a & e & 0 & 0 & 0 & 0 & 0 \\ b+a & c & d & e & 0 & 0 & 0 & 0 \\ a & b & c & d & e & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a & b & c & d & e \\ 0 & 0 & 0 & 0 & a & b & c & d+e \\ 0 & 0 & 0 & 0 & 0 & a & b+e & c+d \end{bmatrix}. \quad (11)$$

where the matrix coefficients are

$$\begin{aligned} a &= +\frac{\Delta t\beta}{\Delta h^4} \\ b &= -\frac{\Delta t(\Delta h^2\alpha + 4\beta)}{\Delta h^4} \\ c &= +\frac{\Delta t(2\Delta h^2\alpha + 6\beta)}{\Delta h^4} \\ d &= -\frac{\Delta t(\Delta h^2\alpha + 4\beta)}{\Delta h^4} \quad (12) \\ e &= +\frac{\Delta t\beta}{\Delta h^4} \quad (13) \end{aligned}$$

where $\Delta t = 1, \gamma = 1$, and $\Delta h = 0.3$. This model is designed for an open contour, estimating central differences at end points using a symmetrical boundary condition. The photometric and ROT model forces are normalized, and in discretized form are therefore given by

$$\nabla E_c^{\text{pho}}(\mathbf{v}_{c,j}^t) = -\kappa \frac{\nabla \left\|\text{GVF}\left(\mathbf{G}_\sigma\left(\mathbf{v}_{c,j}^t\right) * \mathbf{I}\left(\mathbf{v}_{c,j}^t\right)\right)\right\|}{\left\|\nabla \left\|\text{GVF}\left(\mathbf{G}_\sigma\left(\mathbf{v}_{c,j}^t\right) * \mathbf{I}\left(\mathbf{v}_{c,j}^t\right)\right)\right\|\right\|} \tag{14}$$

$$\nabla E_{\pm 1}^{\text{rot}} = -\delta \mathbf{n} \frac{\tilde{w} - \left\|\mathbf{v}_{1,j}^t - \mathbf{v}_{-1,j}^t\right\|}{\tilde{w} - \left\|\mathbf{v}_{1,j}^0 - \mathbf{v}_{-1,j}^0\right\|} \tag{15}$$

$$\nabla E_{\pm 2}^{\text{rot}} = -\delta \frac{\mathbf{v}_{\pm 2,j}^t - \mathbf{v}_{\pm 1,j}^t}{\left\|\mathbf{v}_{\pm 2,j}^0 - \mathbf{v}_{\pm 1,j}^0\right\|}. \tag{16}$$

The control points are constrained to remain on the profile lines. Setting $\mathbf{v}_j^*$ equal to the right-hand side of (10), using the gradient calculations from (14)–(16), the control points are projected back onto the line using

$$\mathbf{v}_j^t = \mathbf{v}_j^{t-1} + (\mathbf{n} \cdot \mathbf{v}_j^*)\mathbf{n}. \tag{17}$$

We use a short ROT model, with five control points per contour, to effectively capture a local profile. It is initialized with the internal and external contours offset $0.25\tilde{w}$ and $0.75\tilde{w}$, respectively, from the center point, parallel to the vessel direction $\mathbf{x}$ (i.e., to either side of the expected edge locations), with the control points at 0.3 pixels spacing; see Fig. 4(a). The model is also designed so that the absence of a vessel may be detected by failure to converge. Model convergence is detected using the contour energy and the maximum separation between corresponding internal and external contour control points—the *twin separation*. If the energy drops below a threshold (0.3) while the twin separation still exceeds 1.0 pixel, the model has failed to converge to vessel edges. This sometimes happens close to distracting features such as the edge of another, parallel vessel. We then temporarily cancel the photometric energy, setting $\kappa = 0$ in (14) for two consecutive iterations. If the model does not then recover within 20 iterations, we conclude that no edge exists, and convergence has failed. In ideal circumstances, successful convergence is indicated by the twin separation dropping below a threshold. However, on some edges convergence is very slow, or may never occur fully; we therefore use two additional criteria, stopping if the twin separation drops below 1.0 and either the energy drops below 0.003, or 10 iterations pass.

After convergence, we assign direction vectors to each control point. For middle points, these are the normalized vectors from the previous to next control points; for end points, the normalized vector between the given point and its single neighbor; see (18). The direction of each edge is then estimated as the average of the ten control point direction vectors of its internal and external contours. If the edge directions diverge by more than $\pi/4$ (as may occur very close to a junction, or if the model has captured a pathology such as a haemorrhage), the model is deemed not to have converged successfully.

The combination of the twin convergence criteria, with an "escape procedure" to deal with photometric distractions, and the edge direction criterion, makes the ROT very effective at locating vessels even in noisy and difficult situations, while also being able to detect the absence of the vessel

$$\begin{aligned}
\mathbf{d}_1 &= (\mathbf{v}_2 - \mathbf{v}_1)/\|\mathbf{v}_2 - \mathbf{v}_1\| \\
\mathbf{d}_j &= (\mathbf{v}_{j+1} - \mathbf{v}_{j-1})/\|\mathbf{v}_{j+1} - \mathbf{v}_{j-1}\| \quad j \in \{2, 3, 4\} \\
\mathbf{d}_5 &= (\mathbf{v}_5 - \mathbf{v}_4)/\|\mathbf{v}_5 - \mathbf{v}_4\|.
\end{aligned} \tag{18}$$

C. Segment Growing

The second stage of the algorithm is to convert the tramline pixel map into segment profiles. This is achieved using an ROT-based segment growing algorithm. Initialized at a single estimated profile, the algorithm progressively grows along a segment, appending profiles, until the vessel ends or reaches a junction. It also detects and rejects nonvessel artifacts. Typically a single starting profile is enough to segment an entire segment (between consecutive junctions), so the algorithm uses only a fraction of the vessel map identified by the tramline algorithm, and is likely to fail to find a segment only if the tramline algorithm misses the segment in its entirety. To avoid repeatedly segmenting the same vessels, the results of the segment growing algorithm are converted into a pixel map, and each tramline pixel is checked to see if it lies within this pixel map before segment growing starts; if so, it is discarded. As the algorithm grows it is initialized using the previous profile, and so particular care is taken to initialize the first profile effectively.

The first profile is derived from the tramline pixel map as follows. A pixel is selected to provide the initial profile center point estimate. The profile direction is estimated using neighboring centerline pixels in an analogous manner to (18). The initial width estimate, w_g, is then determined using Gregson's [14] robust algorithm.

Next, an ROT model is initialized using the parameters determined above, centered on the profile center point. The ribbon force (15) is turned off, and the twins are separately optimized to each edge, each repeated at a number of proposed width offsets, $\tilde{w}$, ranging from $0.5w_g$ to $2w_g$ in steps of $0.125w_g$, with the twin contours initialized at $0.25\tilde{w}$ and $0.75\tilde{w}$ from the center point. This gives the algorithm a wide capture range, compensating for any inaccuracy in the initial profile definition. If the ROT fails at all proposed widths, the location is classified as "nonvessel." After convergence, the pair of control points, one on each of the two internal contours, that are nearest to each other are selected as a new estimate of the profile edge points, based on the definition of the profile as running across the shortest distance between edges. The profile center point, width and direction are recalculated accordingly.

The whole ROT model is then used iteratively to reestimate the profile edge points, and hence the profile, reinitializing the contours at $0.25\tilde{w}$ and $0.75\tilde{w}$ to either side of the center point. The ROT model is optimized, and the profile edge points again identified as the nearest internal contour points. The algorithm repeats until the change in width estimate drops below a stopping threshold (0.1), or a maximum number of iterations (five) is exceeded. Typically the algorithm iterates just one or two times.

Once the first segment profile has been established, the growing algorithm progressively extends the segment by searching for a vessel profile adjacent to the current segment end profile. The ROT model is initialized as shown in Fig. 4(b), with the vessel direction estimated as the mean of the previous three profile directions. Growing continues until the ROT model fails to converge, and is conducted in both directions from the initial profile point. After convergence, the ROT internal contour end point nearest to the segment end profile center point is selected to provide one profile edge point, $\mathbf{p}_1$. The nearest control point to $\mathbf{p}_1$ on the other internal contour, $\mathbf{p}_2$, is identified. Any control points after $\mathbf{p}_2$ on this contour are considered superfluous, and are trimmed. The vessel direction vector is then set to the average of the remaining control point direction vectors on both internal contours. This gives a robust direction estimate. The profile is set orthogonal to this direction, through $\mathbf{p}_2$, with the second edge point, $\mathbf{p}_3$, lying at the point on the profile where it intersects $\mathbf{p}_1$'s direction vector.

D. Segment Assessment Algorithm

The tramline and growing algorithms may produce some false positive segments. These are typically short, having low contrast with the background and varying width. We therefore define the following two features. The *background contrast*, Ψ, is derived by considering the intensity values along each profile line. On each profile, the ratio is calculated between the median of the pixel intensities on the profile line within the vessel, and the median of the intensities of the pixels 2–4 pixels outside each edge. The segment contrast measure Ψ is the mean of these. The *width consistency*, Ω, is given by

$$\Omega = \frac{\sum |w_i - \bar{w}|}{\sum w_i} \tag{19}$$

where w_i is the width of profile i. A true segment is identified using empirically-defined thresholds ($\Psi < 0.99, \Omega < 0.16$). False positive segments are discarded.

E. Joining Segments

The growing algorithm identifies vessel segments with some accuracy, but it does not join up the network—in particular, it actually stops segmentation at junctions, where the edge parallelism in a normal vessel segment fails. It may also sometimes fail due to noisy, low contrast or pathological vessels. We therefore use a subsequent processing stage to join segments (briefly introduced in [27]), by assigning the segment ends (SEs) to one of several *joining forms*: *leaves* (if they do not join to another segment), *bridges* (where two segments are part of the same unbroken vessel—a special case of this occurs at vessel crossings, where there are two bridges), and *junctions* (where a vessel bifurcates, or forms a side-branch).

The joining algorithm involves several stages. First, it uses a ridge following technique to extend all segments, to ensure that bridges and junctions are not missed through poor performance of the segment growing stage—this algorithm can track vessels that the ROT algorithm cannot. Any SEs that terminate close to another segment are classified as *joinable segment ends* (JSEs). Second, these JSEs are assigned to local *JSE sets*, each of which contains a number of JSEs which might plausibly be connected together, by considering how they intersect if projected forwards. Third, all possible combinations of joining forms within each JSE set are enumerated, and the most plausible combination is chosen using an *implicit neural cost function*.

In the first stage, ridge following is applied to all SEs simultaneously. Each SE is assigned a state, initialized to “unresolved.” The centerlines of all unresolved SEs are extended one step, the nearest profile center point on another segment found, and an attempt to resolve the SEs is made as described below. When all SEs are resolved, the algorithms stops.

The SEs are resolved as follows. If the growing centerline reaches the image edge, extends further than $R/2$ (where R is the Optic Nerve Head radius, estimated using the algorithm reported in [28]), or there is no segment profile center point within the radius $R/2$, then the SE is classified as a “leaf.” If a segment center point is found within a radius equal to half the width of the SE profile, then the SE is classified as a JSE. In this case, if the detected segment center point is within the body of another segment, rather than at the end, that point is removed and that segment split into two.

The vessel ridge is tracked by using a twin contour, which converges onto the ridge from either side. This is analogous to using a single twin from the ROT model, with the internal energy defined as in (2), the model energy as (5) (i.e., retaining the attractive twin force, but discarding the ribbon force), and the photometric energy redefined to seek an intensity minimum (rather than an edge), as below

$$\nabla E_{\text{pho}}(\mathbf{v}(s,t)) = \kappa \frac{\text{GVF}(G_\sigma * I(\mathbf{v}(s,t))}{\|\text{GVF}(G_\sigma * I(\mathbf{v}(s,t))\|}. \tag{20}$$

The ridge-following model otherwise is used similarly to the ROT growing algorithm, with a five control point twin initialized along the vessel direction, and converging, to identify the next growing point. The ridge is present and can be followed even in vessels that have very noisy edges, closely parallel neighbors, and may be traced within junctions, even though it is not possible to measure the vessel width in these cases.

At the end of this stage, the vessels have been resolved into a number of segments, with some SEs classified as leaves, and some classified as JSEs and within close proximity (when the central ridge is extended) to another JSE. However, the JSEs do not necessarily join to the nearest neighboring JSE, particularly at crossings, and in complex cases where a number of junctions and/or crossings are close together. We therefore assign all JSEs to local JSE sets, each of which can be resolved separately, as follows. If two JSEs should be joined in either a bridge or junction, we expect that when their segments are projected forwards, they will intersect. We consider each JSE against its nearest neighbor, and calculate a *projective intersection score*, c_{ij}, which characterizes the extent to which they are mutually aligned. If $c_{ij} > 0$, the two JSEs should belong to the same set, and we accordingly create a new set and add them to it, assign one of them to a preexisting set, or merge their existing sets depending on whether neither, one or both are already assigned to sets. Having considered all the nearest neighbors in this fashion, we repeat for the second nearest neighbors, assigning any remaining JSEs (which had $c_{ij} = 0$ on the previous iteration), and iterate until all are assigned. Three iterations have always proved sufficient to date.

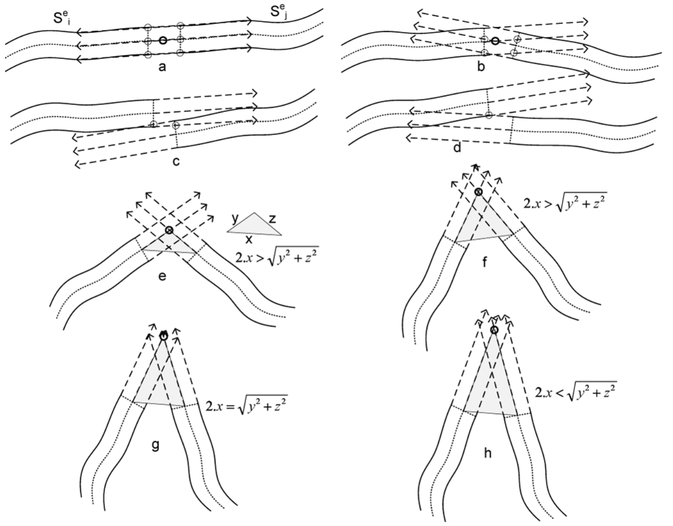


Fig. 5. Projective intersections between segment ends. Segments are shown with a solid line, projections are dotted. (a)–(d) show intersection counts 7, 5, 2, and 1, respectively. (e)–(h) show oblique centerline projective intersections; (e) and (f) are accepted, (g) marginally accepted, (h) is rejected.

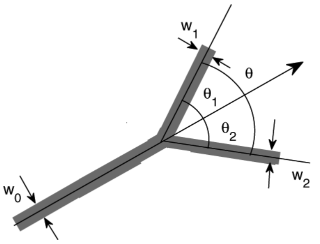


Fig. 6. Definition of junction geometric features.

The projective intersection score is calculated by projecting the edges and centerlines of both JSEs forwards, and counting intersections between each of the three projected lines and the segment end profile of the other JSE, plus the mutual intersection of the two centerlines; see Fig. 5(a)–(d). The latter intersection must not be at too oblique an angle, satisfying the condition in (21) with reference to Fig. 5(e)–(h). There are a total of seven possible intersections, giving an intersection score, $c_{ij} = m/7$, where m is the number of projective intersections

$$2x \geq \sqrt{y^2 + z^2}. \tag{21}$$

The use of projective intersection, rather than mere proximity, in constructing JSE sets is particularly important in keeping the size of the sets under control; for example, at a near-orthogonal vessel crossing each JSE is likely to be nearer to the JSEs of the other vessel than its proper partner on the other side of the crossing. However, it is unlikely to have a projective intersection with these crossing JSEs, and so the algorithm will usually assign two sets, each with just two members.

Once formed, the configurations of the JSE sets are resolved independently of each other. The JSEs within a particular set may potentially be assigned to any possible combination of junctions, bridges and leaf junction forms (involving three, two, and one JSEs each, respectively). We enumerate all possible configurations within a set, and choose the lowest cost configuration, where the cost is defined as the sum of the costs of the constituent joining forms. Hence, we choose the most plausible configuration of local joining forms.

The costs of the individual joining forms are derived by considering the expected geometry of bridges and junctions. Bridges represent a joining of a gap in a vessel, and so we expect the characteristics of the vessel—intensity, width and direction—to be approximately equal on either side. Junctions have a much more complex geometry, for when a vessel bifurcates the angles subtended by the branches are related to their relative widths. If the two branches are equal in width, the branch will be approximately symmetrical ("Y-shaped"). If they differ greatly in width, the larger branch will have minimal deflection from the parent vessel's direction, and the smaller one branches off at close to 90°.

To capture these complex interrelated characteristics, we define implicit cost functions using self-organizing feature maps (SOFMs) [29]. The output of these neural networks is a "novelty signal" indicating whether the input vector is similar to vectors presented in a training stage; they are thus ideal to provide an implicit cost function characterizing whether a configuration is consistent with normal expectations. We trained two SOFMs, the Junction SOFM and the Bridge SOFM, both with topological map dimensions 8×6 nodes using a specially-assembled and hand-labelled data set of junctions and bridges (using 21 images taken from the Sunderland Eye Infirmary database). After training, the activation level of the *winning* neuron represents the *novelty* of the input—J_{sofm} or B_{sofm}. The overall junction form cost is $J_{\text{cost}} = (1 - c_{12}c_{23}c_{13})J_{\text{sofm}}$, taking into account both the SOFM output and the degree of projective intersection between JSEs; the bridge cost is given by $B_{\text{cost}} = (1 - c_{12})B_{\text{sofm}}$.

The inputs to the SOFMs are feature vectors that capture the key modes of variation in bridges and junctions. For the Junction SOFM, the input vector is $(a, b, \lambda, \lambda_1, \lambda_2, \theta, \theta_1, \theta_2)$, where the angles θ, θ_1, and θ_2 are as defined in Fig. 6, and the other parameters, derived from the segment end profile widths are: the child area ratio $a = w_2^2/w_1^2$, the branch area ratio $b = (w_1^2 + w_2^2)/w_0^2$, and the diameter ratios $\lambda = w_2/w_1, \lambda_i = w_i/w_0$. Note that these measures rely on identifying the parent, first branch and second branch. Since these are not known, all six possible permutations are enumerated and the lowest-cost configuration taken.

For the Bridge SOFM, the input vector is $(\lambda_1, \varphi, \eta)$, where λ_1 is the diameter ratio, φ the direction consistency and η the intensity consistency. Defining ρ_i as the mean intensity of the ith segment end (of pixels within the last three profiles), and $\mathbf{x}_i^e$ as the average of the last three profile directions of the ith segment, the parameters are defined as

$$\lambda_1 = \frac{\min(w_1, w_2)}{\max(w_1, w_2)} \tag{22}$$

$$\varphi = \frac{\cosh(\mathbf{x}_1^e \cdot \mathbf{x}_2^e)}{\pi} \tag{23}$$

$$\eta = \frac{\min(\rho_1, \rho_2)}{\max(\rho_1, \rho_2)}. \tag{24}$$

A leaf JSE is identifiable as is does not form a viable bridge or junction (or, if falsely assigned to a bridge or junction

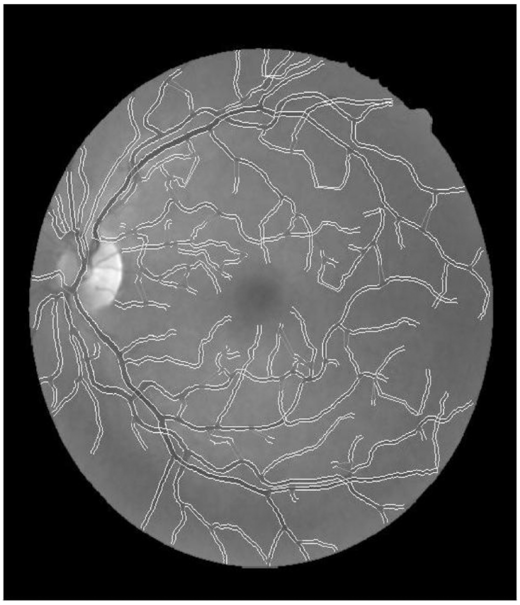


Fig. 7. Sample image showing the final results of the growing stage.

joining form, displaces another JSE and forms a worse bridge or junction). The leaf cost is therefore defined as one minus the minimum cost junction or bridge form that includes that JSE. The implication is that a segment end will be left as a leaf where so-doing allows a sufficiently low cost alternative to be found—in effect, the leaf form is modelled as an "opportunity cost."

Fig. 7 gives an example of the final results of the ESP algorithm.

F. Algorithm Stability and Parameters

The algorithm has a number of adjustable parameters, for which values have been empirically chosen. Within the ROT model, values of α and β in (2) have a significant effect, with curvature increasing as α is increased and as β is decreased; equal values $\alpha = \beta = 0.3$ are appropriate for retinal vessels. The value $\Delta h = 0.3$ is chosen to achieve accurate edge location, balanced against acceptable convergence speed. Five control points are required to allow the pentadiagonal banded matrix (11) to be constructed. The coefficients κ and δ are set to 0.6.

Of the parameters for identifying false positive segments, the threshold for Ψ (0.99) requires careful tuning, with changes as small as 0.01 making an observable difference on blurred vessels; the threshold for Ω (0.16) on the other hand is quite insensitive.

Other parameters are relatively insensitive.

III. Evaluation

In this section we report the segmentation and measurement performance of our algorithm against publicly available databases: for segmentation, DRIVE [30] and STARE [10]; and for measurement, REVIEW [31].

TABLE I
Segmentation Performance on the DRIVE Database, Algorithms, and Benchmark Observer Versus Ground Truth Observer

	Sensitivity	**Specificity**
Human observer	77.61	97.25
ESP	72.82	95.51
Staal	71.94	97.73
Martnez-Prez	70.87	94.97
Niemeijer	67.93	98.01

TABLE II
Segmentation Performance on the STARE Database, Algorithms, and Benchmark Observer Versus Ground Truth Observer

	Sensitivity	**Specificity**
B vs. A	89.51	95.62
ThP vs. A	67.33	96.92
ESP vs. A	75.21	96.81
A vs. B	64.24	99.18
ThP vs. B	54.59	97.60
ESP vs. B	61.62	97.74

A. Segmentation

To evaluate the algorithm's segmentation performance, we transform the results to a pixel map—any pixel whose center point lies inside a segment is regarded as a vessel pixel—and compare with pixel maps generated by a ground truth observer. The opinion of a second observer is also compared to the ground truth observer, to provide an observer benchmark. Results are given as the pixel-wise sensitivity and specificity, defined as $tp/(tp+fn)$, and $tn/(tn+fp)$, where tp, tn, fp and fn are true positive, true negative, false positive, and false negative pixel counts, respectively.

The DRIVE database [30] contains pixel maps for 20 retinal images, including three pathologies. Table I compares the results of ESP with three leading methods from the literature [7]–[9], and the benchmark observer. ESP has higher sensitivity than any of the other algorithms at 73.19%, but lower specificity than two of them—in particular, lower than Staal's algorithm, which has the second highest sensitivity. We conclude that segmentation performance is close to Staal's, trading off higher sensitivity against lower specificity.

The STARE database [23] contains pixel maps for 20 very challenging images, including various pathologies. We have used it to compare results with Hoover's threshold probing (ThP) algorithm [10]. In STARE, Observer B segments significantly more of the finer vessels than Observer A; we have therefore conducted the assessment twice, with Observers A and B swapping the roles of ground truth and human benchmark. Table II shows that ESP has noticeably higher sensitivity than threshold probing, with roughly equal specificity.

B. Junctions

We evaluated the performance of the junction resolution algorithm on the first five images of DRIVE. The algorithm detected 281 bifurcations, 435 bridges and 67 leaves, of which 251, 415, and 60, respectively, were correct, corresponding to precision rates of 89%, 95%, and 90%.

C. Width Measurement

We used four datasets from the REVIEW database [31] (http://ReviewDB.lincoln.ac.uk) to assess the diameter measurement performance. These contain a number of retinal profiles (including center points, width and direction) marked up by three observers, with the mean width used as a ground truth. The *high resolution image set* (HRIS) contains 2368 profiles from 90 segments from four images, resolution 2438 × 3584; these are down-sampled by a factor of four before submission to the measurement algorithms, so that the vessels widths are known to ±0.25 of a pixel, discounting human error. *The vascular disease image set* (VDIS) contains 2249 profiles from 79 segments from eight images, resolution 1024 × 1360. Both HRIS and VDIS images contain a range of normal and diseased retinae, including diabetic retinopathies and an arteriosclerotic. The *central light reflex image set* (CLRIS) contains 285 profiles from 21 segments from two images, at resolution 1440 × 2169, representing early atherosclerotic changes with an exaggerated vascular light reflex. The *Kick Point Image Set* (KPIS) contains 164 profiles from three segments on two images, resolution 3300 × 2600. They are taken from clean, large vessel segments, and are down-sampled like those in HRIS.

We compared the performance of ESP with the HHFW, Gregson, 1-D gaussian and 2-D gaussian algorithms, described in Section I. The REVIEW profiles were used to set the profile directions for the 1-D cross-sectional algorithms, and to initialize the 2-D Gaussian. The ESP estimates were taken from the profiles with center points nearest to the REVIEW profile center point.

Sometimes algorithms fail entirely to detect the vessel width at a given point (e.g., do not converge). We therefore report results as a success rate (i.e., a meaningful measurement was returned), together with the error mean and variance of the successful measurements. The different algorithms may yield consistently different mean widths to one another, and observers. This *bias* is due to the different implicit width definition in the algorithms, and can easily be compensated for by subtraction of a bias constant. It is also worth noting that the key physiological feature of interest is a *change* in the width along a segment, and consistent bias is irrelevant in the detection of changes. In contrast, *variance* in the estimation error cannot be compensated for; consequently, we primarily report algorithm performance using the standard deviations (σ_χ) of the width differences (χ_i), given by

$$\chi_i = \omega_i - \psi_i \tag{25}$$

where ω_i is the estimated width and ψ_i is the ground truth width of the ith profile. Table III lists the success rate and mean widths for each algorithm and observer on each dataset. The success rates show that HHFW and 2-D Gaussian are unstable on more challenging vessels; the very poor means of 1DG and 2DG on the CLRIS dataset occur as those algorithms fit to one side of the vessel.

The disparate means also lend support to the hypothesis that different algorithms have different biases. We, therefore, conducted a one-way ANOVA procedure on VDIS; using the Tukey HSD method, the resulting homogeneity subsets of means are listed in Table IV; the last row gives the significance. This verifies that the algorithms have significantly different means. Interestingly, the test also shows significant differences in the means of the *observers*. This indicates that the observers are consistently marking up to different perceptions of the edge location—albeit with subpixel differences. The test also shows that, in VDIS, the reference standard, observer two, and ESP algorithms belong to a group with no significant differences in the mean. However, this result is not reproduced in ANOVA tests on the other data sets (details omitted), and may be treated as spurious. We do however observe, loosely, that ESP appears to have lower bias than the other algorithms.

TABLE III
PERFORMANCE OF THE MEASUREMENT METHODS, SUCCESS RATE, AND MEAN WIDTH

	HRIS		VDIS		CLRIS		KPIS	
	%	Mean	%	Mean	%	Mean	%	Mean
Standard	100	4.35	100	8.85	100	13.8	100	7.52
O_1	100	4.58	100	9.15	100	14.5	100	7.97
O_2	100	4.35	100	8.91	100	13.7	100	7.60
O_3	100	4.12	100	8.50	100	13.2	100	7.00
Gregson	100	7.64	100	10.07	100	12.8	100	7.29
HHFW	88.3	4.97	78.4	7.94	0	-	96.3	6.47
1DG	99.6	3.81	99.9	5.78	98.6	6.3	100	4.95
2DG	98.9	4.18	77.2	6.59	26.7	7.0	100	5.87
ESP	99.7	4.63	99.6	8.80	93.0	15.7	100	6.56

TABLE IV
TUKEY HSD HOMOGENEOUS SUBSETS OF MEANS, WITH HARMONIC MEAN SAMPLE Size $= 2118.329$

Observer	N	Subset for alpha = 0.05						
		1	2	3	4	5	6	7
1DG	2247	5.783						
2DG	1735		6.494					
HHFW	1764			7.939				
O_3	2249				8.499			
ESP	2241					8.796		
Standard	2249					8.850		
O_2	2249					8.906		
O_1	2249						9.146	
Gregson	2249							10.07
Sig.		1.000	1.000	1.000	0.153	0.137	1.000	1.000

TABLE V
DIFFERENCE STANDARD DEVIATIONS, σ_χ

	HRIS	VDIS	CLRIS	KPIS
O_1	0.285	0.669	0.566	0.233
O_2	0.256	0.621	0.698	0.213
O_3	0.288	0.543	0.567	0.234
Gregson	1.479	1.494	2.841	0.602
HHFW	0.926	0.879		0.389
1DG	0.896	2.110	4.137	0.399
2DG	0.703	1.328	6.019	0.337
ESP	0.420	0.766	1.469	0.328

Table V gives the standard deviations of the differences of the algorithms and observers to the reference standard. ESP exhibits very good performance, achieving an accuracy of 0.420 pixels on the HRIS set (1.52 times the accuracy of the observers), and 0.766 on the lower resolution VDIS set (1.26 times that of the observers). The latter in particular is quite close to human levels

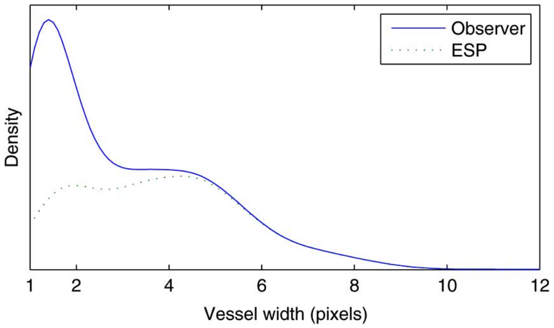


Fig. 8. Distribution of detected vessel widths on the STARE database.

of performance; for the former, it is worth noting that the observers had access to images at four times the resolution of the algorithm. Performance on the CLRIS set is poorer—by observation, these vessels have very blurred edges, and it is this rather than the central light reflex that causes a deterioration in algorithm performance. On the very clean KPIS images, all the algorithms bar Gregson perform very well, although ESP has marginally the best results.

D. Segmentation of Fine Vessels

The ROT algorithm is able to successfully segment quite fine vessels. Fig. 8 shows the distribution of actual and detected vessel widths from the DRIVE database. The actual widths are based on the human observer's pixel map, and consequently do not drop below one pixel, as the markup technique is limited to at least this width; the actual limit of vessel visibility is at about the same level. This shows that the algorithm successfully detects virtually all vessels down to about four pixels in width, with the detection rate dropping off rapidly below three pixels. The failure to detect very fine vessels largely occurs in the tramline stage—the ROT model is able to capture and measure vessels down to 1.5 pixels width, which suggests that future work concentrated on improving the initial segmentation may bear fruit. The difficulty lies in balancing detection of fine vessels against the danger of adding false positives.

E. Performance Limitations

The ESP algorithm is usually effective, as illustrated by the performance figures presented above. In does, however, has some limitations. These largely arise from the need to clearly identify two parallel edges. It may fail to properly identify overlaps or closely parallel vessels. It may also sometimes fail in the ONH, where a steep background intensity gradient may obscure one edge of the vessel. It may fail for similar reasons at the edge of the image. However, these limitations are compensated for by high performance on fine vessels.

F. Computational Performance

We have implemented the system in unoptimized MATLAB code, running on a 1.2 GHz Pentium system. The average runtime on DRIVE is 11 min per image (standard deviation 3 min), although this could probably be significantly reduced if the system were recoded for efficiency using a faster programming language. Memory requirements are for four double precision floating point numbers per image pixel, plus some much smaller data structures.

IV. Conclusion

We have introduced a robust, accurate algorithm for segmenting and measuring retinal vascular segment profiles. The algorithm uses a simple "tramline" vessel pixel detection algorithm, a "Ribbon of Twins" active contour model grown along vessels for detection and integrated measurement, and an auxiliary algorithm to identify the network topology.

The algorithm's segmentation performance has been evaluated against the DRIVE and STARE databases. It has slightly higher sensitivity than benchmark algorithms that have reported results in the literature, with slightly lower specificity than some algorithms on the DRIVE database. The width measurement performance has been evaluated on the REVIEW database, compared with a variety of algorithms from the literature, and found to be superior to these.

It thus provides a valuable combination of good segmentation and superior measurement performance, and so is well-suited to act as a basis for research into the diagnosis of disease using automatically extracted vessel measurements. In future work we will evaluate the algorithm for its ability to aid in screening for or diagnosis of retinal vascular diseases, including arteriosclerosis, hypertension and diabetic retinopathy; and improve the segmentation of fine vessels.

References

[1] F. K. Sutter and H. Helbig, "Familial retinal arteriolar tortuosity: A review," *Survey Ophthalmol.*, vol. 48, no. 3, pp. 245–255, May 2003.

[2] R. Klein, B. E. K. Klein, and S. E. Moss, "The relation of systemic hypertension to changes in the retinal vasculature: The beaver dam eye study," *Trans. Amer. Ophthalmological Soc.*, vol. 95, pp. 329–350, 1997.

[3] R. Klein, B. E. K. Klein, S. E. Moss, and Q. Wang, "Hypertension and retinopathy, arteriolar narrowing, and arteriovenous nicking in a population," *Arch. Ophthalmol.*, vol. 112, no. 1, pp. 92–98, 1994.

[4] 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. Mach. Intell.*, vol. 25, no. 1, pp. 131–137, Jan. 2003.

[5] 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, no. 3, pp. 263–269, Sep. 1989.

[6] O. Chutatape, L. Zheng, and S. M. Krishman, "Retinal blood vessel detection and tracking by matched gaussian and kalman filters," in *Proc. 20th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, 1998, vol. 6, pp. 3144–3149.

[7] M. Niemeijer, J. Staal, B. V. Ginneken, M. Loog, M. D. Abramoff, J. M. Fitzpatrick, and M. Sonka, "Comparative study of retinal vessel segmentation methods on a new publicly available database," *SPIE Med. Imag.*, vol. 5370, pp. 648–656, May 2004.

[8] J. Staal, M. D. Abrmoff, M. Niemeijer, M. A. Viergever, and B. V. Ginneken, "Ridge-based vessel segmentation in color images of the retina," *IEEE Trans. Med. Imag.*, vol. 23, no. 4, pp. 501–509, Apr. 2004.

[9] M. Martinez-Perez, A. Hughes, A. Stanton, S. Thorn, A. Bharath, and K. Parker, "Scale-space analysis for the characterisation of retinal blood vessels," in *Proc. Med. Image Understand. Anal. (MIUA)*, 1999, pp. 57–60.

[10] 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, no. 3, pp. 203–210, Mar. 2000.

[11] N. Chapman, N. Witt, X. Gao, A. A. Bharath, A. V. Stanton, S. A. Thom, and A. D. Hughes, "Computer algorithms for the automated measurement of retinal arteriolar diameters," *Br. J. Ophthalmol.*, vol. 85, pp. 74–79, 2001.

[12] J. A. Lowell, A. Hunter, D. Steel, A. Basu, R. Ryder, and R. L. Kennedy, "Measurement of retinal vessel widths from fundus images based on 2-D modeling," *IEEE Trans. Med. Imag.*, vol. 23, no. 10, pp. 1196–1204, Oct. 2004.

[13] O. Brinchmann-Hansen and H. Heier, "Theoretical relations between light streak characteristics and optical properties of retinal vessels," *Acta Ophthalmologica, Supplement*, vol. 179, pp. 33–37, 1986.

[14] P. Gregson, Z. Shen, R. Scott, and V. Kozousek, "Automated grading of venous beading," *Comput. Biomed. Res.*, vol. 28, no. 4, pp. 291–304, 1995.

[15] L. Zhou, M. S. Rzeszotarski, L. J. Singerman, and J. M. Chokreff, "The detection and quantification of retinopathy using digital angiograms," *IEEE Trans. Med. Imag.*, vol. 13, no. 4, pp. 619–626, Dec. 1994.

[16] T. Chanwimaluang and G. Fan, "An efficient algorithm for extraction of anatomical structures in retinal images," *Image Process.*, vol. 1, pp. 1093–1096, Sep. 2003.

[17] H. Zhu and M. H. Friedman, "Vessel tracking by template string in angiography," *Med. Image Acquisition Process.*, vol. 4549, pp. 29–33, 2001.

[18] A. Klein, T. K. Egglin, J. S. Pollak, F. Lee, and A. A. Amini, "Identifying vascular features with orientation specific filters and b-spline snakes," *IEEE Comput. Cardiol.*, pp. 113–116, Sep. 1994.

[19] I. Laptev, H. Mayer, T. Lindeberg, W. Eckstein, C. Steger, and A. Baum-gartner, "Automatic extraction of roads from aerial images based on scale space and snakes," *Mach. Vis. Appl.*, vol. 12, no. 1, pp. 23–31, 2000.

[20] S. R. Gunn and M. S. Nixon, "A robust snake implementation; A dual active contour," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 1, pp. 63–68, Jan. 1997.

[21] F. A. Velasco and J. L. Marroquin, "Robust parametric active contours: The sandwich snakes," *Mach. Vis. Appl.*, vol. 12, no. 5, pp. 238–242, 2001.

[22] A. Hunter, J. Lowell, and D. Steel, "Tram-line filtering for retinal vessel segmentation," in *IFMBE Proc. 3rd Eur. Med. Biol. Eng. Conf. EMBEC'05*, 2005, vol. 11, no. 1.

[23] Structured analysis of the retina STARE, 2007 [Online]. Available: http://www.parl.clemson.edu/stare/

[24] B. Al-Diri and A. Hunter, "A ribbon of twins for extracting vessel boundaries," in *IFMBE Proc. 3rd Eur. Med. Biol. Eng. Conf. EMBEC'05*, 2005, vol. 11, no. 1.

[25] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, 1988.

[26] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 359–369, 1998.

[27] B. Al-Diri, A. Hunter, D. Steel, and M. Habib, "Joining retinal vessel segments," in *Proc. 8th IEEE Int. Conf. BioInformatics BioEngineering (BIBE)*, Oct. 2008, pp. 1–6.

[28] J. A. Lowell, A. Hunter, D. Steel, R. Ryder, and E. Fletcher, "Optic nerve head segmentation," *IEEE Trans. Med. Imag.*, vol. 23, no. 2, pp. 256–264, Feb. 2004.

[29] J. Humphreys and A. Hunter, "Multiple object tracking using a neural cost function," *Image Vis. Comput.*, vol. 4, no. 3, pp. 417424–, 2009.

[30] Digital retinal images for vessel extraction 2007 [Online]. Available: http://www.isi.uu.nl/Research/Databases/DRIVE/

[31] B. Al-Diri, A. Hunter, D. Steel, M. Habib, T. Hudaib, and S. Berry, "Review—A reference data set for retinal vessel profiles," in *Proc. 30th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2008, pp. 2262–2265.