Robust segmentation of intraretinal layers in the normal human fovea using a novel statistical model based on texture and shape analysis

Vedran Kajić,[1,*] Boris Považay,[3] Boris Hermann,[3] Bernd Hofer,[3] David Marshall,[2] Paul L. Rosin,[2] Wolfgang Drexler[3, 1]

[1] *School of Optometry and Vision Sciences, Cardiff University, Maindy Road, Cardiff, CF24 4LU, UK*
[2] *School of Computer Science, Cardiff University, 5 The Parade, Cardiff, CF24 3AA, UK*
[3] *Center for Medical Physics and Biomedical Engineering, Medical University Vienna,*
General Hospital Vienna 4L, Waehringer Guertel 18-20, A-1090 Vienna, Austria
**kajicv1@cf.ac.uk*

Abstract: A novel statistical model based on texture and shape for fully automatic intraretinal layer segmentation of normal retinal tomograms obtained by a commercial 800nm optical coherence tomography (OCT) system is developed. While existing algorithms often fail dramatically due to strong speckle noise, non-optimal imaging conditions, shadows and other artefacts, the novel algorithm's accuracy only slowly deteriorates when progressively increasing segmentation task difficulty. Evaluation against a large set of manual segmentations shows unprecedented robustness, even in the presence of additional strong speckle noise, with dynamic range tested down to 12dB, enabling segmentation of almost all intraretinal layers in cases previously inaccessible to the existing algorithms. For the first time, an error measure is computed from a large, representative manually segmented data set (466 B-scans from 17 eyes, segmented twice by different operators) and compared to the automatic segmentation with a difference of only 2.6% against the inter-observer variability.

OCIS codes: (170.4500) Optical coherence tomography; (100.0100) Image processing; (100.3008) Image recognition, algorithms and filters; (170.4580) Optical diagnostics for medicine.

References and links

1. W. Drexler, and J. G. Fujimoto, *Optical Coherence Tomography: Technology and Applications* (Springer, 2008).
2. T. Fabritius, S. Makita, M. Miura, R. Myllylä, and Y. Yasuno, "Automated segmentation of the macula by optical coherence tomography," Opt. Express **17**(18), 15659–15669 (2009).
3. R. J. Zawadzki, S. S. Choi, S. M. Jones, S. S. Oliver, and J. S. Werner, "Adaptive optics-optical coherence tomography: optimizing visualization of microscopic retinal structures in three dimensions," J. Opt. Soc. Am. A **24**(5), 1373 (2007).
4. M. M. K. Garvin, M. M. D. Abramoff, R. R. Kardon, S. S. R. Russell, X. X. Wu, and M. M. Sonka, "Intraretinal Layer Segmentation of Macular Optical Coherence Tomography Images Using Optimal 3-D Graph Search," IEEE Trans. Med. Imaging **27**(10), 1495–1505 (2008).
5. D. Cabrera Fernández, H. M. Salinas, and C. A. Puliafito, "Automated detection of retinal layer structures on optical coherence tomography images," Opt. Express **13**(25), 10200–10216 (2005).
6. M. Mujat, R. Chan, B. Cense, B. Park, C. Joo, T. Akkin, T. Chen, and J. de Boer, "Retinal nerve fiber layer thickness map determined from optical coherence tomography images," Opt. Express **13**(23), 9480–9491 (2005).
7. D. Koozekanani, K. Boyer, and C. Roberts, "Retinal thickness measurements from optical coherence tomography using a Markov boundary model," IEEE Trans. Med. Imaging **20**(9), 900–916 (2001).
8. D. Tolliver, Y. Koutis, H. Ishikawa, J. S. Schuman, and G. L. Miller, "Unassisted Segmentation of Multiple Retinal Layers via Spectral Rounding," in *ARVO*(2008).
9. A. Mishra, A. Wong, K. Bizheva, and D. A. Clausi, "Intra-retinal layer segmentation in optical coherence tomography images," Opt. Express **17**(26), 23719–23728 (2009).
10. I. W. Selesnick, R. G. Baraniuk, and N. G. Kingsbury, "The Dual-Tree Complex Wavelet Transform," IEEE Signal Process. Mag. **22**(6), 123–151 (2005).
11. A. Mishra, A. Wong, D. A. Clausi, and P. W. Fieguth, "Quasi-random nonlinear scale space," Pattern Recognit. Lett. In Press. (Corrected Proof.).

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

12. A. Wong, A. Mishra, K. Bizheva, and D. A. Clausi, "General Bayesian estimation for speckle noise reduction in optical coherence tomography retinal imagery," Opt. Express **18**(8), 8338–8352 (2010).
13. P. Thevenaz, and M. Unser, "A pyramid approach to sub-pixel image fusion based on mutual information," in *Image Processing, 1996. Proceedings., International Conference on*(1996), p. 265.
14. C. O. S. Sorzano, P. Thevenaz, and M. Unser, "Elastic registration of biological images using vector-spline regularization," BIEEE Biomed. Eng. **52**(4), 652–663 (2005).
15. A. K. Mishra, P. W. Fieguth, and D. A. Clausi, "Decoupled Active Contour (DAC) for Boundary Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence **99**.
16. T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active Appearance Models," IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 681–685 (2001).
17. M. Scholz, M. Fraunholz, and J. Selbig, "Nonlinear Principal Component Analysis: Neural Network Models and Applications," in *Principal Manifolds for Data Visualization and Dimension Reduction*(2007), pp. 44–67.
18. M. Scholz, F. Kaplan, C. L. Guy, J. Kopka, and J. Selbig, "Non-linear PCA: a missing data approach," Bioinformatics **21**(20), 3887–3895 (2005).
19. A. A. Efros, and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*(ACM, 2001), pp. 341–346.

1. Introduction

Optical Coherence Tomography (OCT) [1] is a biomedical imaging method using infrared light that gives high resolution, three-dimensional (3D) sub-surface insight into living tissue utilizing white-light interferometry. The simple optical access to the light sensitive retina, which is hard to reach by other high resolution methods lead to a wide clinical acceptance of this imaging technique. Currently available OCT systems have increased in speed towards hundred thousand depth scans per second and axial resolution of less than 3 µm, enabling imaging of the retinal microstructure. Result of a typically less than 10 second retinal OCT scan is a large volumetric data set consisting of a stack (typically 128-512) of high resolution cross-sectional images (B-scans, typically 1024x512 pixels).

In order to make these large retinal 3D OCT data sets clinically useful it is necessary to analyze the structure by segmentation of layers, as they correspond to patches of similar cellular components that can be used to establish a potential early disease diagnosis or perform therapy monitoring. However, due to the sheer amount of data, it is inconvenient or even impossible for a human operator to manually perform the segmentation in a high throughput clinical environment. Therefore it is necessary to develop effective computer algorithms for automated segmentation of relevant layers of the investigated tissue.

Existing published approaches to retinal OCT data segmentation vary depending on the number of layers to be segmented and on their robustness in the presence of strong speckle noise, shadows, irregularities (i.e. vessels, structural changes at the fovea and optic nerve head) and pathological changes in the tissue. In general they tend to be very sensitive to noisy data or are limited to only segment a small number of layers.

Fabritius et al. [2] presented a fast, efficient algorithm for finding only the internal limiting membrane (ILM) and retinal pigment epithelium (RPE) boundaries that utilizes 3D information and performs simple filtering. This rather simple step is typically the first one performed before a more detailed analysis of the intraretinal structure.

Zawadzki et al. [3] used a semi-automatic algorithm for OCT segmentation where the user would have to paint the areas of interest in any slice of the volume. For segmentation a support vector machine (SVM) was used with a feature vector that contained intensity, location, mean of the neighbourhood, standard deviation and gradient magnitude.

A 3D graph search approach to OCT retinal layer segmentation was presented by Garvin et al. [4]. The algorithm first aligned all the slices and straightened the RPE layer. Then the optimal graph cut was performed with weights describing both edge and regional information. Good results were obtained but only for high quality data. Due to its computational complexity this approach is unlikely to be applicable to less ideal foveae because, necessarily, more complex constraints would disproportionally increase the computation time.

Fernandez et al. [5] presented segmentation results using a peak finding algorithm. Since it is an iterative thresholding algorithm it is sensitive to noise and deviation from the normal retinal data. Extension to any non-typical case might prove to be difficult since the algorithm's parameters are manually selected, rather than learned from a set of segmentation

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

examples. Even though some good results were obtained it is prone to failure and it allows detected boundaries to overlap.

Mujat et al. [6] used the deformable spline algorithm (active contour) to determine nerve fibre layer (NFL) thickness only. Blood vessels could be detected by using intensity holes in the RPE layer.

A Markov boundary model was applied to connect the extracted boundary edge primitives by Koozekanani et al. [7]. Even though more robust than standard column-wise thresholding methods, it still relies on connecting 1D points. That makes it sensitive to noise and thus detected layer boundaries can easily drift off from the real ones. Special rules have to be applied to correct for such cases which makes the whole approach less general.

An elegant approach to retinal segmentation based on spectral rounding was introduced by Tolliver et al. [8]. It is a graph partitioning algorithm based on the eigenvector calculation to determine the oscillation steps that represent the retinal edges. It performed very well since no a priori information was available to the algorithm, simply dividing an image iteratively along the oscillation boundary - different regions of the image correspond to different modes of oscillation. Although the accuracy was good, the number of extracted layers was low and it is very unlikely that layers with weaker signal could be extracted without using additional structural information.

Mishra et al. [9] presented a promising two-step algorithm based on a kernel optimization scheme. Initially, approximate positions of the boundaries are found, followed by the second, refinement step. Very good segmentation results were obtained; however no quantitative evaluation on a large data set was given, nor was any result given on the actual dynamic range of the presented images. Additionally, it is unclear how the algorithm performs in cases with more variability in boundary distances, such as the foveal pit region. Only images of the flat part of the retina were shown and the algorithm imposes some fixed constraints on the shape of layers.

Thus, all of the aforementioned methods suffer from one or more of the following disadvantages: they distinguish only the most prominent layers, do not exhibit robustness in noisy and varied cases and/or require manual intervention of the operator.

The proposed method of the present paper uses training data obtained from manual segmentations by human operators as input to a statistical model which is able to actively learn and determine the plausible solutions in a noisy environment. During the learning stage parameters of a statistical model are extracted so that it best fits the training data. That includes the possible variation of layer boundaries as well as texture information within the layers. This approach offers greater flexibility over the fixed constraints on layer smoothness, since it learns from the data what amount of variability is possible and in what regions, while on the other hand constrains data to a plausible space of states.

Our model based approach uses the variation obtained from the training set and imposes those constraints when segmenting an unseen image. This guarantees that the segmentation will be close to the ground truth and less sensitive to noise. However, it is extremely important to have a large, representative training set that includes all possible variation. We solved this issue by applying a novel approach for obtaining manual segmentations of the OCT data via an Amazon service called The Mechanical Turk, designed to offer a large international human work force for completion of user defined tasks.

Overall the novel algorithm segments eight layers: NFL, ganglion cell layer and inner plexiform layer (GCL + IPL), inner nuclear layer (INL), outer plexiform layer (OPL), outer nuclear layer (ONL), connecting cilia (CL), outer segment (OS) and RPE.

2. Materials and methods

We have used a three-dimensional OCT system for imaging. It uses a superluminescent light source, with 840nm central wavelength and 50nm optical bandwidth. Axial resolution is 5-6 microns, while transverse resolution is 15-20 microns. Data acquisition speed was 27 klines/sec. Optical power was 500 μW and SNR was 96dB with a sensitivity roll off −6dB/mm. Depth range was 3.5mm and axial sampling 2.3 μm/vx.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

2.1 The algorithm overview

As seen in the overview of the algorithm (Fig. 1), one can be observe that the pre-processing stage is performed for both the training step and the segmentation of the unseen data. Once the variation parameters have been learned from the manually segmented training data, they can be used to drive the model to perform segmentation of unseen data. The actual segmentation process is essentially an optimization run that changes the model parameters in order to minimize the objective function which defines the difference between the model and a given unseen image that is to be segmented.

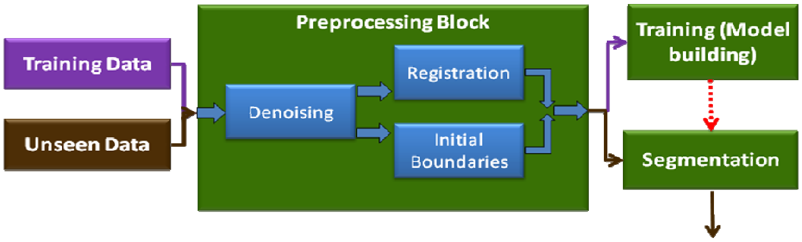

Fig. 1. Algorithm overview: manually segmented data is used as the input to the training phase of the algorithm. After passing the pre-processing block a statistical model is constructed that captures the variance in the training data, which can be then used to segment unseen data.

2.2 Pre-processing

Before the segmentation process, dual-tree complex wavelet (DTCW) denoising is applied to the data. The denoising algorithm exhibits very good performance, while being computationally efficient [10]. This reduces the speckle noise present and thus makes the subsequent segmentation tasks easier.

Denoising based on quasi-random nonlinear scale space described in [11] and applied to OCT speckle reduction in [12] would likely be more effective. It is an effective and fast method based on formulating the denoising problem as a general Bayesian least-squares estimation problem. A quasi-random density estimation approach is introduced for estimating the posterior distribution between consecutive scale space realizations. However, the relatively small performance difference (larger speed difference) in not significant for the performance of the statistical model, thus we have used a well tested and freely available DTCW code.

After that, registration of the stack and segmentation of the three initial well defined boundaries (ILM, connecting cilia (CL) and end of RPE) is performed. Registration and initial boundary location finding are currently independent since detection of the initial boundary location operates on each B-scan independently.

A stack registration algorithm has been developed based on B-spline multi-resolution pyramid registration approach [13] and [14]. The basic algorithm for translation and rotation is used to register source to target image.

ILM, CL and end of RPE boundaries are found using an adaptive thresholding algorithm (auto adjusts to appropriate power) that converges to a close strong edge after the first estimate, additionally using constraints on distances between the boundaries. Robust polynomial fitting is afterwards used to eliminate outliers, followed by interpolation along the remaining points. ILM boundary is found first by starting the thresholding process from the top of the image, while RPE boundary is found next by starting from the bottom. The CL boundary is determined the last and depends on the positions of the already found ILM and RPE boundaries. It is found starting from the top after eliminating the pixels in the neighbourhood of the already found ILM boundary and imposing constraints on the distance from the RPE boundary. An example with a large shadowed area is shown in Fig. 2.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

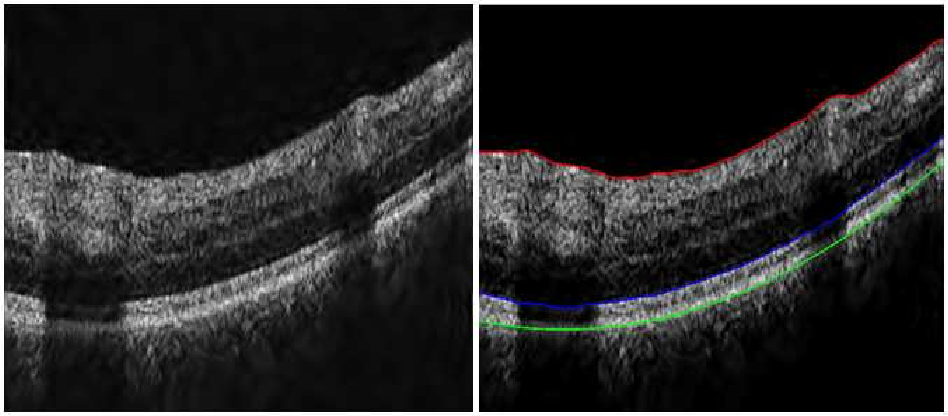

Fig. 2. Initial segmentation step of a despeckled OCT frame (on the left) after adaptive thresholding boundary detection demarking (on the right): internal limiting membrane (ILM, red), connecting cilia (CL, blue), retinal pigment epithelium (RPE, green).

Initial boundary detection would also be possible based on a decoupled active contour (DAC) approach as presented in [15]. We have tested the level set active contour approach and discarded it for its slow convergence. However, DAC is both robust and fast, as it decouples the measurement (solved by using Hidden Markov Model (HMM) and Viterbi search) and prior active contour energy terms. As we have found our initial boundary estimation approach sufficient for the current application we have not experimented with all other available methods. For future work, however, algorithms such as DAC could prove valuable.

2.3 Model building

After the pre-processing stage the statistical model is first trained on a set of manually segmented images and can be then applied to the unseen data. Using a statistical model based on the training data is a potentially effective tool for both segmentation and registration [16]. Its main advantage is that knowledge of the problem can be used to resolve the confusion caused by structural complexity, provide tolerance to noisy or missing data, and provide a means of labelling the recovered structures. The idea is to perform supervised learning by applying knowledge of the expected shapes of structures, their spatial relationships, and their textural appearance to restrict the automated system to plausible interpretations. Supervised learning is a type of machine learning for learning a function based on training data, which consists of pairs of input objects, and desired outputs. The task of the supervised learner is to predict the value of the function for any valid input object after having seen a number of training examples. To be useful, a model needs to be specific, capable of representing only legal examples of the modelled object.

From the manually segmented images we extract the shape and texture features and for each image we put all the extracted shape features into one vector and all the texture features into another vector. Separate models for shape and texture are constructed similarly, so only the shape model construction will be explained. If we have m training images, for each layer (n layers) we get one vector of offsets $\mathbf{v}$ per layer, per image of width w, which stacked together for all the layers define $\mathbf{x}$. All of the manual segmentations then comprise the matrix $\mathbf{X}$ Eq. (1).

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_m \end{pmatrix} = \begin{pmatrix} \mathbf{v}_{11} & \cdots & \mathbf{v}_{1n} \\ \vdots & \ddots & \vdots \\ \mathbf{v}_{m1} & \cdots & \mathbf{v}_{mn} \end{pmatrix}$$

$$\mathbf{v}_{ij} = [\mathit{off}_1 \ldots \mathit{off}_W] \tag{1}$$

Shape features that are used are sparsely sampled distances of the boundaries from the top boundary (ILM). Texture features that are currently used are simple, although it is trivial to include additional features if needed to further increase performance in case of vessels, large shadows and pathological tissue; currently used features are the mean of all the pixels for each of the layers in the original image, standard deviation and mean of all the pixels for each of the layers in the median filtered image, as well as the multiple-scale (a pyramid of Gaussian filtered versions of the image) edges sampled along the boundaries. In practice, for an image of width 512, we sampled each boundary at 26 positions. Thus we have 26 spatial features and 4 texture features per each layer, and for eight layers, we obtain 208 spatial and 32 texture features.

Statistical models can reproduce specific patterns of variability in shape and texture by analyzing the variations in shape across the training set. It is difficult to achieve this selectivity, whilst allowing for natural variability, without using very large descriptors and thus it is essential to select good features from the training set for the model building phase. The key step of the statistical model training phase is the dimensionality reduction of the large set of features from the training data set. The reason for dimensionality reduction is to reduce the computational cost of the optimization method that is used to fit the model to the real data later on. The idea behind this concept is to find statistical dependencies between the produced features and reduce the dimensionality of the space by identifying only a certain number of the most prominent properties in the data set, represented by the most important eigenvectors.

Principal component analysis (PCA) is the standard vector space transform technique used to reduce multidimensional data sets to lower dimensions for analysis. It works by calculating the eigenvalue decomposition of a data covariance matrix or singular value decomposition of a data matrix. Usually a relatively small number of eigenvectors with greatest eigenvalues can describe the original data well. If $\mathbf{X}$ is the original data matrix, as defined in Eq. (1), after the decomposition we can select only L principal components and in that way project the data into a reduced dimensionality space to get $\mathbf{Y}$ Eq. (2).

$$\begin{gathered} \mathbf{X} = \mathbf{W}\Sigma\mathbf{V}^T \\ \mathbf{Y} = \mathbf{W}_{\mathrm{L}}^T\mathbf{X} \end{gathered} \tag{2}$$

However, rather than PCA, we used neural network based dimensionality reduction since it offers nonlinear eigenvectors and therefore can reduce the space more compactly if the data is nonlinearly distributed than the linear representation obtained by PCA [17]. The shape features proved to be nonlinear and thus we obtained a more compact representation using nonlinear dimensionality reduction, rather than PCA. A Neural network (NN) is a mathematical or computational model based on principles found in biological neural networks. It consists of an interconnected group of artificial neurons and processes information where each connection between neurons has a weight, with the weights modulating the value across the connection. The training phase is performed to modify the weights until the network implements a desired function. Once training has completed, the network can be applied to data that was not part of the training set. It is useful to note that a special type of neural network (inverse) [18] can be used to perform dimensionality reduction on the training feature set that is produced which contains missing values. Missing values occur when no data value is stored for the variable in the current observation. The generating

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

function is used to produce larger dimensionality data $\mathbf{X}$ from the parameters $\mathbf{z}$ (equivalent to $\mathbf{W}_L$ in PCA) Eq. (3). The extraction function does the reverse.

$$\begin{aligned} \Phi_{gen} &: \mathbf{z} -> \hat{\mathbf{X}} \\ \Phi_{extr} &: \mathbf{X} -> \mathbf{z} \end{aligned} \tag{3}$$

We encounter the problem of missing data because during the registration process slices are moved, and since input to the dimensionality reduction step has to be a rectangular matrix, it is necessary to fill the missing values. In practice we can set them as "not a number" (NaN) and perform the nonlinear PCA (Fig. 3). After that, we end up with a reduced number of variables which can reasonably well describe any variation observed in the training data. We reduced dimensionality of the original spatial feature space from 208 to 12, and the texture feature space from 32 to 2. This number of eigenvectors allowed for an efficient optimization in the subsequent steps, while still preserving the original data variation well.

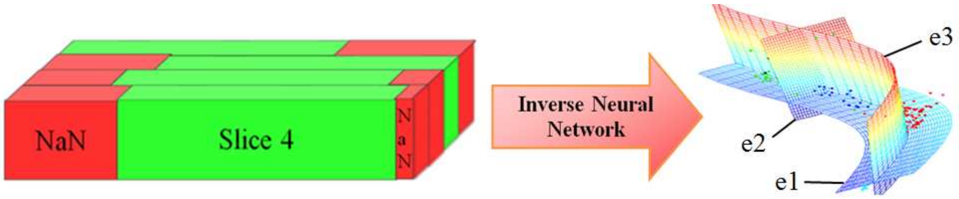

Fig. 3. Filling the gaps after the registration with NaNs and applying inverse neural network nonlinear PCA dimensionality reduction. In the case of the example data shown on the right, we can see that already the first eigenvector (e1) captures most of the variance in the original data set. This illustrates the idea behind the dimensionality reduction.

Our approach is based on a similar concept to the Active Appearance Model (AAM). For completeness, it will be first explained how the basic AAM model works, followed by an explanation of how the proposed statistical model differs from that concept. An Active Appearance Model (AAM) manipulates a model capable of synthesising new images of the object of interest by finding the model parameters which generate a synthetic image as close as possible to the target image [16]. An AAM will, based on learned shape deformation, generate a new image with a texture learned from the texture variation and then compute the distance between the synthesized and the given image that is to be segmented. $\mathbf{x}$ is the shape vector (which is normalized by subtracting the mean shape and rescaling, Eq. (4)) and $\mathbf{g}$ is the texture vector obtained from an image $\mathbf{I}$ and the shape vector (it is also normalized) Eq. (5).

$$\mathbf{x} \longrightarrow (\mathbf{x} \text{ - } \mu(\mathbf{x})\mathbf{1})/\sigma(\mathbf{x}) \tag{4}$$

$$\mathbf{g} = G(\mathbf{x}, \mathbf{I}) \tag{5}$$

Function $S(\mathbf{s})$ produces new shape vectors by adding the shape parameters $\mathbf{s}$ multiplied by the shape matrix $\mathbf{Q}_s$ (a matrix of sorted eigenvectors learned from the training set, usually produced by PCA decomposition) to the mean shape vector $\overline{\mathbf{x}}$ Eq. (6). The same procedure is used to generate new texture vectors.

$$\begin{aligned} \mathbf{x} &= S(\mathbf{s}) = \overline{\mathbf{x}} + \mathbf{Q}_s\mathbf{s} \\ \mathbf{g} &= T(\mathbf{t}) = \overline{\mathbf{g}} + \mathbf{Q}_g\mathbf{t} \end{aligned} \tag{6}$$

However, unlike the AAM which compares pixelwise synthesized images, we use the layer boundaries produced by the model during the optimization to compute texture features of the bounded area and compare it to the expected texture properties of each layer learned from the training set. This approach is used since unlike the areas in which AAMs are usually applied, the texture of retinal OCT scans varies so much within one layer that the direct comparison with a synthesized image is unusable. The objective function (Eq. (7)) evaluates how well the model matches real data and is minimized during the optimization.

$$f(\mathbf{s},\mathbf{t}) = \left| T^{-1}(G(S(\mathbf{s}),\mathbf{I})) - \mathbf{t} \right| + \frac{b * \sqrt{\sum (S(\mathbf{s})_b - \mathbf{b}_{init})^2}}{w} \tag{7}$$

b is the number of boundaries, w is image width and $\mathbf{b}_{init}$ defines the initial three boundaries positions found by the adaptive thresholding algorithm. T^{-1} is the inverse of T; T is defined in Eq. (6). T^{-1} returns the model texture parameters **t** that are most likely to generate a given vector of texture features **g**. The first term of the objective function defines the main measure for evaluation of the model fitting, determined by the difference of the model texture parameters and the texture parameters extracted from the image regions defined by the model shape parameters. The second term penalizes deviations from the initial boundary as found by the initial three boundaries algorithm and the one produced by running the optimization function for the statistical model. This is an important novelty, when compared to the standard AAM, which helps to constrain the optimization process to valid solutions. Additionally, we do not start the optimization process from the mean of the model, but rather we determine the median distance between ILM and RPE boundaries found by the adaptive thresholding algorithm, as well as the ratio of the foveal pit distance to the greatest thickness found in the image. Using these values we pick the closest example from the training set and use these parameters for the initial model position. This way we ensure a faster and more robust convergence.

Another novelty is introduced in the second stage of the algorithm based on fitting a model for each independently used A-scan (depth-scan) to further improve the accuracy. This stage starts from the position defined by the result of the first stage B-scan fitting. We have chosen to divide the image area into four segments and built an A-scan model for each segment since different types of variation can be expected at different offsets from the foveal depression. The A-scan model is trained on offsets produced by back projecting the manual segmentation data using the main B-scan model and computing the boundary offsets between the back projections and the original segmentations Eq. (8) (n is the number of layers and u is the number of A-scans from all the images in the given segment).

$$\mathbf{A} = \begin{pmatrix} aOff_{11} & \cdots & aOff_{1n} \\ \vdots & \ddots & \vdots \\ aOff_{u1} & \cdots & aOff_{un} \end{pmatrix} \tag{8}$$

In Fig. 4 it can be seen how the second refinement stage of the algorithm improves precise tracking of the layer boundaries.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

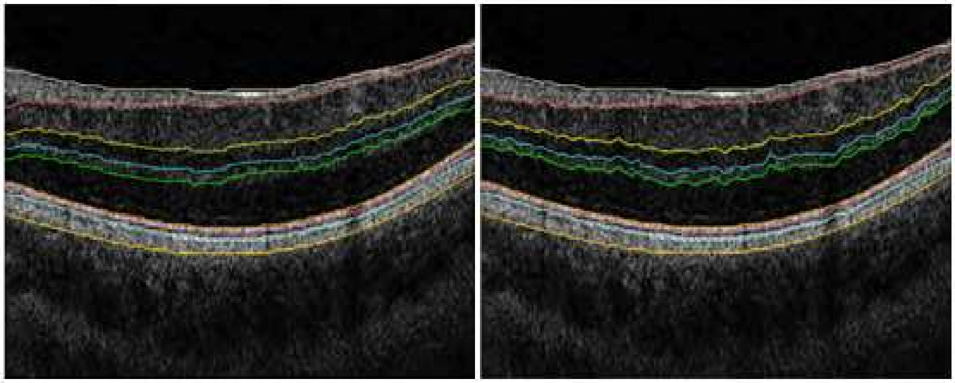

Fig. 4. On the left is the result after the global low-res optimisation followed by, on the right, the refined result by the A-scan optimization.

2.4 The Mechanical Turk

A large training data set has been efficiently obtained via an Amazon service called the Mechanical Turk (AMT), designed to offer a large international work force for completion of user defined tasks (Human Intelligence Task or HIT). In principle, the task of segmenting was divided to the detailed description of the task by a skilled person, manual delineation of the interfaces by a large number of less skilled workers, the comparison of multiple results for the same task and the supervision of the whole process by the skilled operator. Two account types are used: worker and requester. The worker account type is used for performing the tasks, while the requester type is used for submitting them. Submitted tasks are usually simple but it is possible to define criteria for the workers and in that way use skilled workers, for larger payments, of course. In our case no testing was performed for selection of the workforce apart from the general ranking of a worker based on previous performance recorded by the AMT-system. However, it was necessary to supervise the work relatively often and update the instructions based on the input from workers and give bonuses for good work to stimulate reliable workers to continue doing the provided tasks.

The architecture of the whole system is comprised of a web page with JavaScript to handle the user input that was designed through the AMT interface and inside we have embedded a Java applet through which the workers perform the segmentation. For storage of the B-scans, example images and results to be saved, Amazon S3 storage service was used. We have submitted 505 B-scans and each image was set to be segmented twice by different workers respectively. That way we can compute the inter worker variability, as well as leave out inaccurate results, while still having another one which is usually good. Inter worker variability was computed only on the images for which both results were deemed to be accurate. We have also paid out bonuses for good work, approximately equivalent to the initial payment. In case of inaccurate or inappropriate results it is not necessary to pay the worker. Since the behaviour of the AMT system can be better described by the rules of sociology than simple mathematical relations, the processing speed is nonlinear. It is important to note that while we obtained half of the results in just a few days; it usually takes significantly longer to get all the tasks completed. That is not a problem since it is possible to use results as they are produced without having to wait for the completion of the whole batch. One most likely reason for the reduced speed of work completed is that workers use the default sorting for viewing the available tasks, which sorts based on the number of available tasks. We have also used it for the segmentation of the choroid (four boundaries). It took four weeks to complete the segmentation of about 2700 images. Workers seemed to be more interested in the task once the purpose of the work was given in the introduction and it was pointed out that it serves a valuable medical goal. We included a few questions in the form of a web form so that workers can give us feedback on the work that they are doing.

3. Results and discussion

For evaluation purposes we have used 466 manually segmented B-scans, almost (in some cases we had to discard the manual segmentation) uniformly sampled from 17 eyes (each stack contains 128 B-scans). We have tested the performance of our algorithm on this data set using the leave-one-out test; we iteratively left out all data from one person, trained the model on the remaining data and then tested the performance on the data from the person left out. This procedure is performed for each person in the training set. This way we make sure that we are testing the performance of our algorithm on the "unseen" data.

For evaluation, automatic segmentation results were compared to manual segmentation done by the AMT workers. Two types of error measures were used, computed for each boundary *i* separately and from these we compute error measures for an entire B-scan or for an individual layer, Eq. (9).

$$ {}^{i}E_{B} = \sum_{j=1}^{j=w} \left| yAut_{ij} - yRef_{ij} \right|, \; {}^{i}E_{LDEV} = \sqrt{w * \sum_{j=1}^{j=w} (yAut_{ij} - yRef_{ij})^{2}} \qquad (9) $$

E_B (Basic) is the basic error measure that defines the number of misclassified pixels. E_{LDEV} (Layer DEViation) uses the $\sqrt{w}$ term for normalization so that for the special case when the two boundaries are equally distant from each other along their whole length ($yAut_{ij} - yRef_{ij} = d$ for all *j*), it is equal to E_B (proved in Eq. (10)).

$$ \begin{aligned} {}^{i}E_{B} &= \sum_{j=1}^{j=w} \left| yAut_{ij} - yRef_{ij} \right| = \sum_{j=1}^{j=w} |d| = w * |d| \\ {}^{i}E_{LDEV} &= \sqrt{w * \sum_{j=1}^{j=w} (yAut_{ij} - yRef_{ij})^{2}} = \sqrt{w * \sum_{j=1}^{j=w} d^{2}} = \sqrt{w * w * d^{2}} = w * |d| = {}^{i}E_{B} \end{aligned} \qquad (10) $$

For all other cases E_{LDEV} is larger than E_B. Thus E_{LDEV} will penalize large deviations from the reference position of a boundary, unlike E_B which only measures the number of misclassified pixels. E_{LDEV} is therefore useful for penalizing specific types of poor algorithm performance which could show as, for example, a large jump in a boundary position that could be narrow and thus not affect E_B significantly since the misclassified area would be relatively small.

The error for a whole image (this refers to both E_B and E_{LDEV}) is defined in Eq. (11).

$$ E = \frac{\sum_{i=1}^{i=b} {}^{i}E}{A} \quad (11) $$

${}^{i}E$ is the error for each boundary and A is the area between top (ILM) and bottom boundaries (RPE/CH).

In the case when we express error for layer *k* separately, instead of summing up across all boundary errors, only the two boundaries that define a layer are added and divided by the sum of the layer area as given by the automatic segmentation (A_A) and the layer area as given by the reference segmentation (A_R), Eq. (12). This is used to normalize for double counting of misclassified pixels, as each layer is bounded by two boundaries.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

$$E = \frac{\sum_{i=k}^{i=k+1} {}^{i}E}{A_A + A_R}, 0 < k < b \qquad (12)$$

A confidence measure could be introduced based on the values returned by the objective function after the optimization step. Large values are proportional to the low confidence in the boundary positions determined by the model fitting. This would be useful for the operator to decide whether the obtained results are reliable.

In Table 1 the inter-worker variability of the manual segmentations used in training is presented for each boundary and in total, while in Table 2 and Table 3 results are presented for both the initial segmentation and after the second step refinement.

Table 1. Variability of manual segmentations on 75 B-scans in percent (the data has been previously examined and "bad" results left out)

Error Type	NFL	GCL + IPL	INL	OPL	ONL	CL	OS	RPE	Total
E_B	13.6	11.4	22.8	25.0	6.0	28.0	23.3	18.7	16.1
E_{LDEV}	17.9	14.4	28.4	31.3	7.4	35.4	28.6	22.5	19.9

Table 2. Error values on 466 B-scans at various positions from 17 eyes in percent before the A-scan optimization

Error Type	NFL	GCL + IPL	INL	OPL	ONL	CL	OS	RPE	Total
E_B	23.2	14.3	31.6	41.9	8.6	35.2	32.1	22.1	22.4
E_{LDEV}	31.9	17.4	39.7	55.4	10.7	47.3	41.0	27.0	27.8

Table 3. Error values on 466 B-scans at various positions from 17 eyes in percent after the A-scan optimization

Error Type	NFL	GCL + IPL	INL	OPL	ONL	CL	OS	RPE	Total
E_B	20.0	10.1	22.1	31.6	7.1	34.9	30.8	21.6	18.7
E_{LDEV}	29.2	13.2	30.4	46.4	9.3	47.1	39.5	26.5	24.2

It can be seen that the total error rates (especially E_B which is the main measure) are close to the inter-operator variability (18.2% compared to inter-operator's 16.1%). E_{LDEV} difference is somewhat larger. Thus, we can conclude that the algorithm performance is almost the same as ground truth.

Our algorithm performs well even when artefacts are present, such as strong shadows, which can cause problems for less robust algorithms (Fig. 5).

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

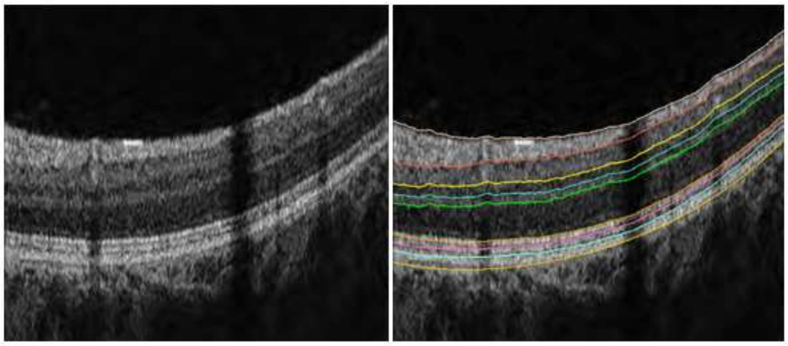

Fig. 5. Robust performance for all the layers is achieved even in presence of shadowing. A despeckled image is shown on the left; the segmented image is on the right.

In Fig. 6 thickness maps are shown for 17 different eyes after registering them and computing median and coefficient of variation (expressed as absolute variation in pixels), since it would take too much space to present the results for each eye individually. It can be seen that despite the data being affected by artefacts, the results are accurate and show larger variation only around the foveal pit region, as can be expected.

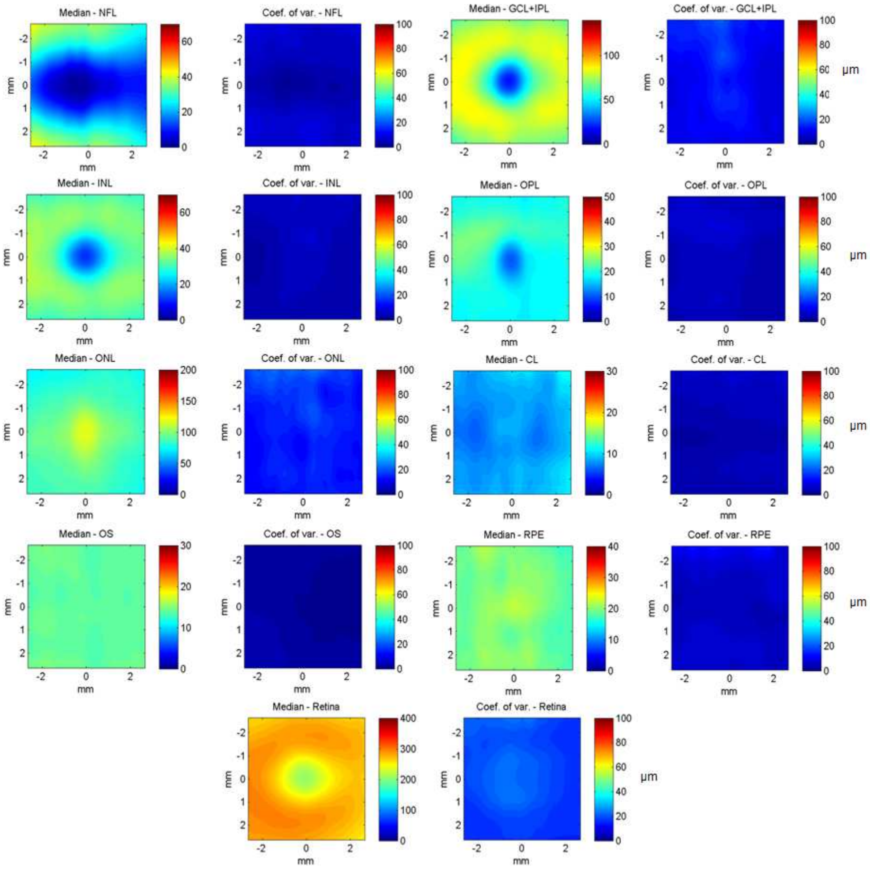

Fig. 6. Median and coefficient of variation computed on thickness maps of all the individual layers (nerve fibre layer (NFL), ganglion cell layer and inner plexiform layer (GCL + IPL), inner nuclear layer (INL), outer plexiform layer (OPL), outer nuclear layer (ONL), connecting cilia (CL), outer segment (OS), retinal pigment epithelium (RPE)), as well as the retina, obtained from 17 eyes.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

To evaluate performance of the algorithm in conditions of increased noise (reduced dynamic range) that frequently occurs in clinical measurements for a number of reasons (opaque cornea of cataract lens, residue in vitreous humour, non optimal imaging conditions, etc) background noise (speckle, multiplicative random noise) has been added to tomograms (Fig. 7) and results plotted on a graph. The background was generated using a texture synthesis approach [19]. This enables us to efficiently produce a different speckle noise pattern for each image even though they are all based on the same physical speckle template, which is only one image of background noise with the typical spatial frequency distribution. Using this approach we can generate an arbitrary number of synthetic, but uncorrelated and realistic, images of background noise that we add subsequently to each given image to simulate low dynamic range. The algorithm shows robust performance under such conditions shown by a gentle rise of the error/dynamic range curve.

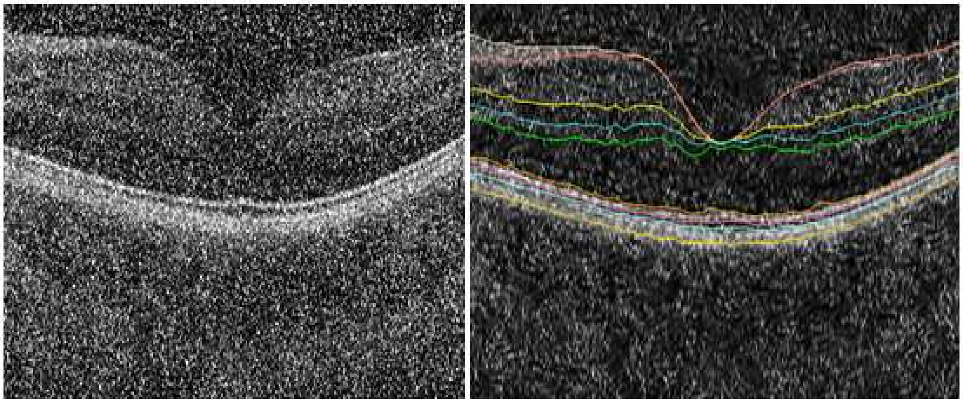

Fig. 7. Segmentation in a case of added strong noise. Left original image. Right filtered, denoised image with segmentation results superimposed.

This can be seen in two graphs showing error rates E_B and E_{LDEV} plotted versus the dynamic range for a set of images for all the layers combined and with the confidence interval (1.96 std. dev.) plotted as dashed lines (Fig. 8), as well as two graphs showing the error rates for each individual layer (Fig. 9). The individual boundaries most affected by decreasing dynamic range are those defining INL and OPL, as could be expected since these layers exhibit normally significant variation and have weak boundaries which are affected early by the noise increase. Also, the boundaries between CL, OS and RPE are difficult to determine.

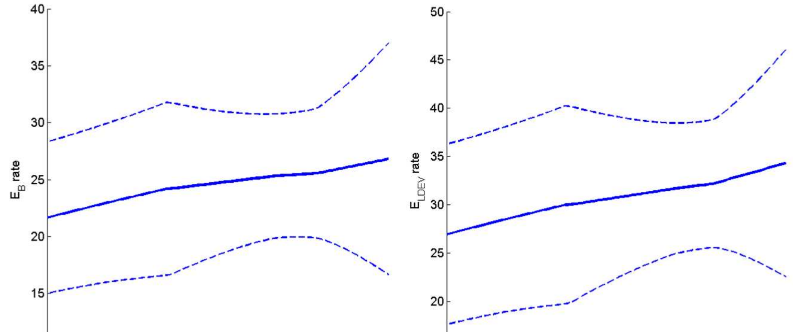

Fig. 8. Error rates E_B (Basic) and E_{LDEV} (Layer DEViation) with decreasing dynamic range for all the data sets, with confidence interval (1.96 * standard deviation) marked by the dashed lines. For both error measures a slow rise in the error values can be observed, which guarantees robust performance with noisy data.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

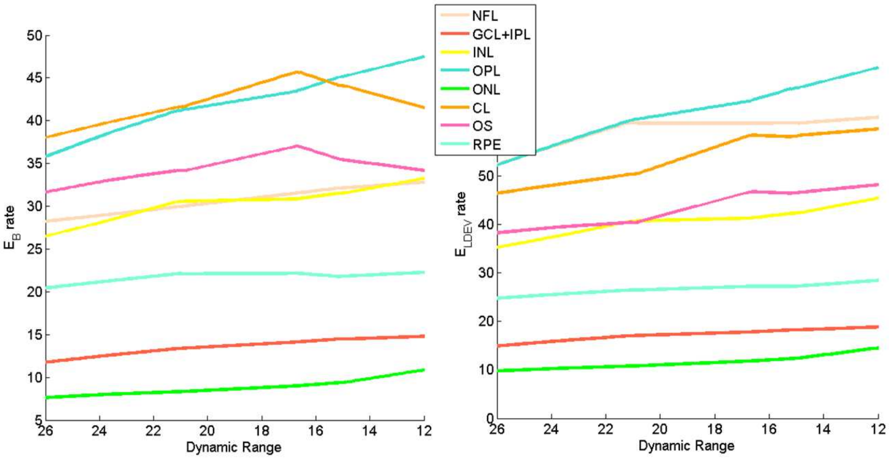

Fig. 9. Error rates for the individual layers E_B (Basic) and E_{LDEV} (Layer DEViation) with decreasing dynamic range for all the data sets. For all the individual layers (nerve fibre layer (NFL), ganglion cell layer and inner plexiform layer (GCL + IPL), inner nuclear layer (INL), outer plexiform layer (OPL), outer nuclear layer (ONL), connecting cilia (CL), outer segment (OS), retinal pigment epithelium (RPE)) a slow rise in the error values can be observed. Thin layers inherently exhibit greater error values, as both errors are normalized by the layer area.

4. Conclusion

We have proposed an algorithm for automatically segmenting all major retinal layers based on a novel statistical model. We have introduced two important novelties with respect to the standard Active Appearance Model (AAM): a second term in the optimization function that penalizes large deviations from the three boundaries found by the adaptive thresholding algorithm and the second algorithm stage that refines the model fit for each A-scan independently, giving increased accuracy.

It has been thoroughly tested and evaluated against the manually segmented large data set from a 800nm OCT system and proved highly robust in full foveal scans even in the presence of artefacts and added strong background noise that reduces dynamic range down to 12dB. It is the first time that a large, representative data set (466 B-scans from 17 eyes) has been used for evaluation of an OCT segmentation algorithm. We have used manual segmentations of large data set as ground truth, rather than the frequently used error computed between the results of the algorithm on inter-visit measurements, as it is susceptible to underdetermine the real error value as it is susceptible to ignore systematic error of the algorithm. Apart from the basic error measure that counts the number of the misclassified pixels, we have also used a second error measure to penalize large deviations from the ground truth.

Thus, we can conclude that our algorithm successfully demonstrated reliable performance under conditions which prove extremely challenging for the pre-existing methods. Additionally, it has the potential of being used in other areas where boundaries are not well defined, such as segmentation of choroid layers, which is an important open problem in OCT data analysis. It would be also possible to extend the proposed algorithm to segmentation of pathological cases, as well as segmentation of ONH (optic nerve head) scans which contain discontinuous boundaries. In case that the stack registration is very precise, the initial ILM and RPE boundary finding step could be replaced by the algorithm proposed by Fabritius et al. [2] that relies on full 3D information present in the stack, since it is very efficient. Clinically, fully automated segmentation of all major layers is essential in making medically useful the possibilities given by the method of high resolution, high speed OCT of large portions of the human retina at microscopic detail.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010

Acknowledgements

The authors would like to thank Marieh Esmaeelpour for advice on biological interpretation of the data. This research was supported in part by the BBSRC, Cardiff University; FP6-IST-NMP-2 STREPT (017128, NanoUB); DTI grant (OMICRON); AMR grant (AP1110); European Union project FUN OCT (FP7 HEALTH, contract no. 201880) & CARL ZEISS Meditec Inc.

#128452 - $15.00 USD Received 14 May 2010; revised 20 Jun 2010; accepted 21 Jun 2010; published 24 Jun 2010