Graduate School Form 30
(Revised 08/14)

# PURDUE UNIVERSITY
## GRADUATE SCHOOL
## Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Nathan Hammes

Entitled

Segmentation of Human Retinal Layers from Optical Coherence Tomography Scans

For the degree of Master of Science

Is approved by the final examining committee:

Gavriil Tsechpenakis

Mihran Tuceryan

Shiaofen Fang

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Gavriil Tsechpenakis

Approved by Major Professor(s): _____

_____

Approved by: Shiaofen Fang 02/05/2014

Head of the Department Graduate Program Date

# SEGMENTATION OF HUMAN RETINAL LAYERS FROM OPTICAL COHERENCE TOMOGRAPHY SCANS

A Thesis

Submitted to the Faculty

of

Purdue University

by

Nathan M. Hammes

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science

May 2015

Purdue University

Indianapolis, Indiana

This thesis is dedicated to those who read its dedication.

ACKNOWLEDGMENTS

I may have written the entirety of this work, but it is not without the support of a good many others that I would have succeeded. I must thank my advisor, Dr. Gavriil Tsechpenakis, for his guidance when I needed it most, and for the freedom to explore the realm of possibilities while researching and developing solutions for this application. Thanks also to Sarun Gulyanon for graciously being a sounding board for ideas throughout. Also, I must mention my family, whose support throughout my degree has been pivotal in my success. Finally, I must acknowledge Dr. Brian Samuels and Dr. Lyne Racette, whose research ideas and support gave me the chance to develop this thesis.

TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# ABBREVIATIONS

| | |
|---|---|
| FFD | Free-Form Deformations |
| HCF | Highest Confidence First |
| HEYEX | Heidelberg Eye Explorer |
| ILM | Inner-Limiting Membrane |
| OCT | Optical Coherence Tomography |
| ONH | Optic Nerve Head |
| RNFL | Retinal Nerve Fiber Layer |
| RPE | Retinal Pigmented Epithelium |
| SD-OCT | Spectral Domain-Optical Coherence Tomography |
| 2D | 2-dimensional |
| 3D | 3-dimensional |

# NOMENCLATURE

B-scan In an OCT image volume, one high-resolution image. Image volumes used for the purposes of this thesis each consisted of 61 B-scans.

A-scan In an OCT image volume, one column of voxels. Each B-scan consists of many A-scans.

# ABSTRACT

Hammes, Nathan M. M.S., Purdue University, May 2015. Segmentation of Human Retinal Layers from Optical Coherence Tomography Scans. Major Professor: Gavriil Tsechpenakis.

An algorithm was developed in to efficiently segment the inner-limiting membrane (ILM) and retinal pigmented epithelium (RPE) from spectral domain-optical coherence tomography image volumes. A deformable model framework is described and implemented in which free-form deformations (FFD) are used to direct two deformable sheets to the two high-contrast layers of interest. Improved accuracy in determining retinal thickness (the distance between the ILM and the RPE) is demonstrated against the commercial state-of-the-art Spectralis OCT native segmentation software. A novel adaptation of the highest confidence first (HCF) algorithm is utilized to improve upon the initial results. The proposed adaptation of HCF provides distance-based clique potentials and an efficient solution to layer-based segmentation, reducing a 3D problem to 2D inference.

# 1 INTRODUCTION

Glaucoma is one of the leading causes of significant vision loss and blindness throughout the world [1, 2]. The disease is characteristically defined as a chronic optic neuropathy that results in the loss of retinal ganglion cells and their axons (i.e. the retinal nerve fiber layer; RNFL), with increased intraocular pressure being the primary risk factor. It is the cumulative loss of these retinal ganglion cells that leads to permanent visual field defects and eventual blindness. Thus, the goal of clinicians is to detect glaucoma as early as possible in the disease process in order to preserve visual function.

New advances in technology have resulted in the development of quicker, high-definition spectral-domain optical coherence tomography (SD-OCT) imaging with a retinal image resolution of 3.9$\mu$m [3]. Glaucoma analysis software has been developed to examine for glaucomatous retinal defects by identifying macular retinal thickness and asymmetry between the superior and inferior hemifields [4]. Total retinal thickness is calculated as the distance between the inner-limiting membrane (ILM), lying on the interface between the dark vitreous environment and the bright RNFL, and the highly reflective retinal pigmented epithelium (RPE), the last clear boundary between the retina and the vessels of the choroid (Fig. 1.1(a)).

## 1.1 Related Work

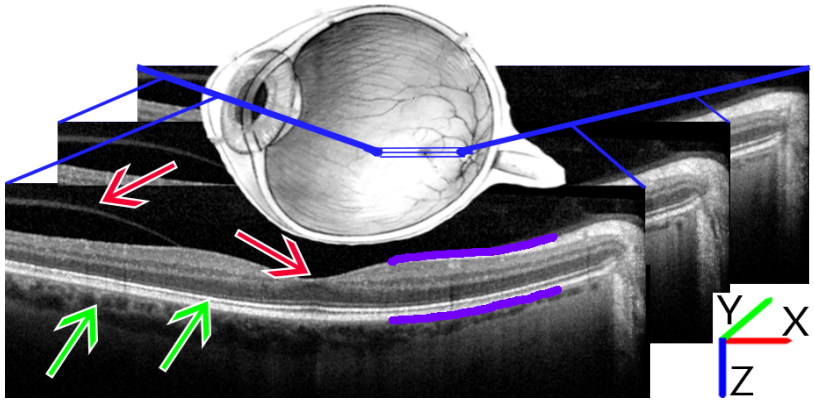
Since the advent of SD-OCT, various automatic segmentation algorithms have been employed to determine the layers of the human retina and the corresponding thicknesses between layers. A recent review article by DeBuc categorizes these methods into classification, deformable models, global optimization using graph cuts, and model-fitting and registration methods [5]. Supervised classification methods,

especially support vector machine methods, are effective in obtaining desired boundaries [6]. For analysis of 3D image stacks, however, the computational complexity of machine-learning-based approaches is beyond the maximum allowable segmentation time in clinical settings. Model-fitting and registration methods use a predetermined shape to fit to a given image feature. If instead a match must be found for an entire image, an atlas is warped to achieve the mapping. These methods are computationally efficient, but lack robustness when dealing with shape variations beyond their predetermined model [7]. Lastly global optimization methods mainly arise from the utilization of the max-flow min-cut algorithm commonly paired with graph models [8]. Additionally, segmentation based upon graph cuts is a fast and robust method for achieving many-layer segmentation in this scenario. See the preliminary paper by Li and Wu and the subsequent application of graph theory to layered segmentation of retinal scans by Garvin et al. [9, 10].

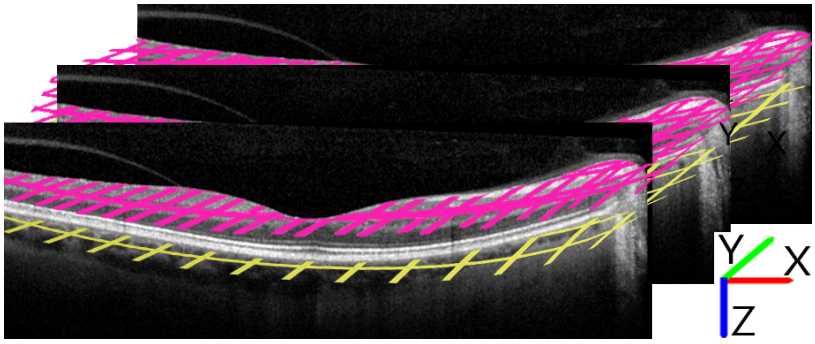
The history of deformable models, the central focus of the model employed by the author, points to a common source. Kass et al., in 1988, published *Snakes: Active Contour Models* [11]. In this paper, an entity with both an internal and external energy was described. These two energies compete: the internal energy pressures the object to maintain a simplistic shape, while the external (data) energy deforms the snake to match an object of interest - an edge, for example. When the overall energy function is minimized, the model is at an optimal configuration for the given combination of internal and external energies:

$$E = E_{int} + E_{ext} \tag{1.1}$$

In the subsequent decades, the area expanded into the new ideas offered by a model whose energy minimization could be applied to the complex task of object segmentation. The task itself is similar to clustering in terms of complexity. In an image, for example, it may be desired to segment three separate objects from background, for a total of $k = 4$ clusters. Given a $100 \cdot 100$-pixel image, the total number of possible clusterings for it is $O(4^{10000}/4!)$. For $k$ clusters of $n$ points, $O(k^n/k!)$ possible




(a) ILM




(b) RPE

Figure 1.1.: Three sequential images taken from within an SD-OCT image stack of the human retina. Note Cartesian axes for future reference. (a) Purple lines denote the two layers of interest to be segmented: inner-limiting membrane (ILM) above, retinal pigmented epithelium (RPE) below. In a completely segmented image, the lines would extend along the layers in both directions. Red arrows indicate potential difficulties for the ILM: vitreous artifact at left presents an area of continuous contrast similar to the ILM; topological dip at the foveola is often accompanied by a reduction in absolute contrast, making a concrete measure of contrast impractical. Green arrows indicate potential issues for the RPE: both the choroid (left) and inner/outer photoreceptor segment junction provide areas of contrast similar to that of the RPE. (b) Two lightly-colored grids demonstrate the final desired result for a 3D segmentation of the two layers.

clusterings exist [12]. It is easy to see why establishing a model that is able to control its own complexity while still allowing flexibility for external information is desirable. It simplifies the problem drastically and provides an elegant objective function to be minimized.

### 1.1.1 Segmentation of OCT Scans using Deformable Models

The realm of deformable models can be divided into two classes: the implicit/geometric models and the parametric models. Currently, only a few deformable model approaches have been presented to segment various aspects of OCT images, and all belong to the parametric model category, including that explained in this article. In 2005, Cabrera Fernández first demonstrated the utility of parametric deformable models through demonstration of the accurate segmentation of the fluid-filled regions common in the OCT's of patients with age-related macular degeneration [13].

That same year, Mujat investigated a deformable model using splines for retinal layer segmentation, but limited their analysis of a 3D SD-OCT image stack to sequential 2D analysis of the images [14]. Few details of the proposed method were published, and it is difficult to determine the relatedness of the algorithm to that demonstrated here due to a dearth of information. Total segmentation time for a stack using Mujat's method was given as 62 seconds. In 2009, Mishra et al developed an algorithm with a basis in active contours to segment multiple layers in the results of time-domain OCT, a precursor of the significantly higher resolution SD-OCT technology [15]. Reported segmentation time was considered "highly efficient" at five seconds per 2D image, also known technically in the world of ophthalmology as a B-scan.

The parametric deformable models can be subdivided into region-based and edge-based methods. The most recent example of deformable model-based segmentation is that of Yazdanpanah et al. In a region-based approach, standalone murine SD-OCT B-scans were segmented for six retinal layers. Their model self-reportedly is easily

corrupted by noise and suffers from areas of low contrast [16]. A segmentation time for the OCT images was also not provided. These previous deformable-model approaches, while effective with respect to a single slice of an OCT image, lose useful information that could be harnessed if an integrated 3D volume analysis was performed. No deformable-model approach has yet been presented that either expands to analyze the image stack as a whole or integrates edge-based energy into its objective function.

The algorithm developed for this thesis introduces a fast dual-model deformable spline framework designed to evolve to the ILM and RPE for the purpose of retinal thickness calculation. This framework is designed with intrinsic smoothness constraints both within and between each sheet-like model, and is driven by the minimization of the energy of an overall objective function, similar to most parametric deformable models. Inclusion of a data energy term based on areas of high relative edge contrast allows for accurate segmentation despite considerable amounts of noise. The details of this simple methodology are explained in Chapter 2. Results are provided in Chapter 3 that demonstrate the effectiveness of this approach. See Fig. 1.1(a) for a clear representation of the OCT image stack as it relates to this framework. Despite more accurate segmentation with respect to commercial software, initialization of the distance transforms associated with the data energy for the above approach required improvement. An adaptation of the Highest Confidence First (HCF) algorithm was recruited for this task; a brief background of the algorithm is presented here.

### 1.1.2 Highest Confidence First

The HCF algorithm was originally presented by Chou et al. in 1990 as an "efficient", "predictable", and "robust" edge-detection algorithm for 2D images using a Markov Random Field (MRF) [17, 18]. It has since been used across the computer vision community for various segmentation tasks including object segmentation in both 2D and 3D images, object tracking in video sequences, and text and handwrit-

ing identification [19–22]. HCF operates on an arbitrarily shaped MRF, a structure generally composed of a layer of interconnected nodes. Each of these is connected to an additional node that is a representation of a random variable at the connected node's location. This is the label for the connected node.

In the field of computer vision, an image is generally represented by the connected layer of the MRF, where each pixel is a single node at that location. The labels to be assigned to each pixel are the random variables associated with each node in the connected layer. Specifically for the HCF algorithm, we consider a set of labels $X = x_1, x_2, \ldots, x_n$ to be assigned to the initially uncommitted node labels $x_0$ of the MRF. The specific value $\omega_s$ of a random variable at site $s \in S$ may be any $x_i$ in $X$.

The assignment of labels to all variables in the field is called a configuration $\Omega$, with $\Omega_0$ the initial configuration in which all $\omega_s$ are set to $x_0$. Given a set of observations $O$ (an image, for example), the prior distribution using clique potentials $P(\Omega) = \sum_{c \in C} e_c(\Omega)$ over an MRF, an energy function $E$ of the posterior probability distribution $P(\Omega|O)$ can be derived using Bayes's Rule:

$$E(\Omega|O) = \sum_{c \in C} e_c(\Omega) - \sum_{s \in S} \log P(O_s|\omega_s) \tag{1.2}$$

Minimization of this energy function through maximum *a posteriori* probability (MAP) *estimation* leads to a global optimum for $P(\Omega|O)$. HCF seeks to approximate this inference. The methods chapter explains in detail the proposed adjustments and improvements to the HCF algorithm, and how they can be used in a 2D or 3D surface segmentation problem to achieve accurate results.

We incorporate the HCF algorithm as an intermediate step in our approach to determining the initialization of the data energy $E_{data}$ of the abovementioned deformable models. After, we compare our results to those of expert graders and those of the Spectralis to assess the validity of our approach, identical to the comparison performed for the method without the addition of HCF. Finally, the 2 methods are compared, showing that despite considerable accuracy for the deformable framework

alone, HCF improves both mean and standard deviation of errors in determining ILM location, RPE location, and overall retinal thickness.

# 2 METHODS

The two open surfaces that evolve as part of the model are targeted to two retinal layers, the ILM and the RPE, whose vertical difference in distance, the retinal thickness, is the value of interest for this clinical application. Corresponding mathematical terms will be denoted with a subscript of ILM and I or RPE and R respectively. An initial flat surface extending through the $768 \cdot 61 \cdot 496$ $(x \cdot y \cdot z)$ stack of OCT images will be said to occupy the $x$ and $y$ axes of the stack. The images themselves each lie on the $x$-$z$ plane with dimensions $768 \cdot 496$. Evolution of the surface $S$ will propagate through the $z$ dimension, initially "downward" for $S_{\mathrm{I}}$ and "upward" for $S_{\mathrm{R}}$ (see Fig. 1.1(b)). The two models, $S_{\mathrm{I}}$ and $S_{\mathrm{R}}$, are interconnected through their signed distance transforms. These transforms can be considered to have the following conditional breakdown:

$$\Phi_{\mathrm{ILM}} = \left\{ \begin{array}{ll} 0 & z \in S_{\mathrm{I}} \\ +\min_{z_{S_{\mathrm{I}}} \in S_{\mathrm{I}}} ||z - z_{S_{\mathrm{I}}}|| & z \in S_{\mathrm{I}+} \\ -\min_{z_{S_{\mathrm{I}}} \in S_{\mathrm{I}}} ||z - z_{S_{\mathrm{I}}}|| & z \in S_{\mathrm{I}-} \end{array} \right\} \tag{2.1}$$

where $S_{\mathrm{I}}$ represents the surface meant to deform from above to the ILM, and $S_{\mathrm{I}+}$ and $S_{\mathrm{I}-}$ are the regions above and below that evolving front, respectively. For comparison, $S_{\mathrm{R}}$, which evolves from the bottom of the image stack, has regions $S_{\mathrm{R}+}$ below and $S_{\mathrm{R}-}$ above it on the distance volume $\Phi_{\mathrm{RPE}}$. $z$ represents all z-values in a Cartesian coordinate system of the points $\mathbf{C}$ in the given image stack. $\mathbf{C} = (x, y, z)$.

The deformable model used for this experiment is comprised of two separate surfaces whose iterative action is the result of a minimization of an objective function consisting of two energy-based terms representing topology and data.

$$E_{\mathrm{total}} = E_{\mathrm{data}}(\Phi^G_{\mathrm{ILM}}(z), \Phi^G_{\mathrm{RPE}}(z)) + E_{\mathrm{topology}}(\Phi^t(z)) \tag{2.2}$$

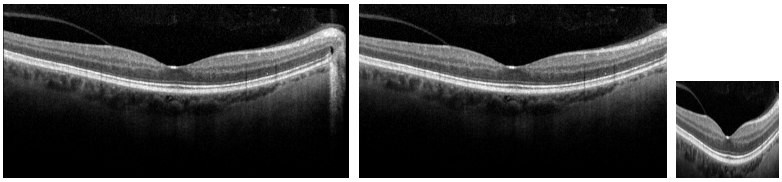
where $E_{\text{data}}((\Phi^G_{\text{ILM}}(z), \Phi^G_{\text{RPE}}(z))$ represents the data-driven external energy, $E_{\text{topology}}(\Phi^t(z))$ can be characterized as the classical $E_{\text{int}}$ of deformable models.

## 2.1 Preprocessing and Energy Initialization

As important, if not more important, than the specification of the deformable model framework is the development of the features to which the models will evolve. In order to facilitate accurate segmentation, specific steps were taken to preprocess the image volumes. Each step is explained here with an accompanying figure. For clarity, each figure uses the same image as in each of the other preprocessing steps. These steps end with the initialization of the energies for the two deformable sheets to minimize.

Image volumes exported from the Spectralis include, in addition to the image stacks analyzed here, a reference brightfield image showing the location of a specific B-scan with respect to the retina itself. Scales for each area of the exported volume are included, along with the date and time of the examination. These much larger image stacks are trimmed to the mentioned dimensions of $768 \cdot 496 \cdot 61$ voxels. An image volume with these dimensions contains a total of 61 images (again, known as B-scans in the world of ophthalmology). These images, for an unknown reason, contain sets of "blank" columns of voxels at the extreme ends of the B-scans. The number of blank A-scans/columns at either end of a B-scan varies.

Unfortunately, the methods involved in this experiment require that information be present in each column of a volume and that each B-scan be of equal length. To allow for segmentation, a volume's B-scans were initially trimmed from both sides by the **maximum** largest set of "blank" A-scans in any image. This process reduced the number of A-scans per B-scans by at most 100 (from the original 768) and was improved upon in the second iteration of the algorithm to prevent such a loss of information. Compare Fig. 2.1(a) with Fig. 2.1(b) to see the amount of cropping this would typically entail.



(a) (b) (c)

Figure 2.1.: Effects of initial preprocessing steps on an image in a volume. Original image (a) is cropped to become (b) by 39 A-scans on the left and 45 on the right due to "blank" A-scans in other images of the volume. Subfigure (b) is compressed 3-fold to (c) for efficiency and for reduction in the overall presence of feature ambiguities. Additional rows of pixels in (c) have been removed in this figure and subsequent ones only to improve the clarity of the figures; these rows are used for the segmentation algorithm. Images are to scale.

Because the retinal layers themselves at the provided B-scans' resolution smoothly vary, it was decided that an improvement in both segmentation accuracy and efficiency could be achieved by compressing each image by a factor of three along the $x$ dimension, per the recommendation of Yang et al. [23] (see Fig. 2.1(c)). Following this "A-scan reduction" technique and the subsequent segmentation, the results are interpolated back using splines [24] (Fig. 2.2).

Spline interpolation was a natural choice, considering the deformable model framework itself evolves given a carefully selected basis of spline equations. This will be described in detail in section 2.2.

### 2.1.1 Contrast Determination

Starting at the top of an image stack and moving downward along its A-scans (columns of pixels), the first instance of high negative contrast is the ILM, and the last instance of high positive contrast is the RPE (Fig. 2.3(a,b)). Negative contrast indicates a transition from a low-intensity area to one of higher intensity. Specifically,

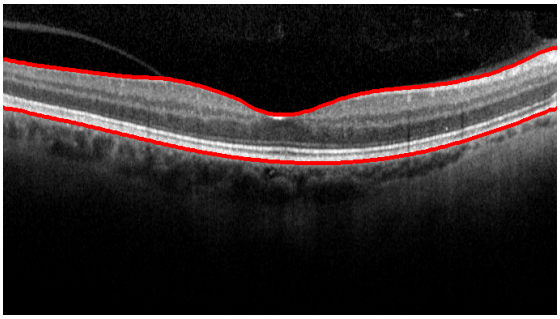


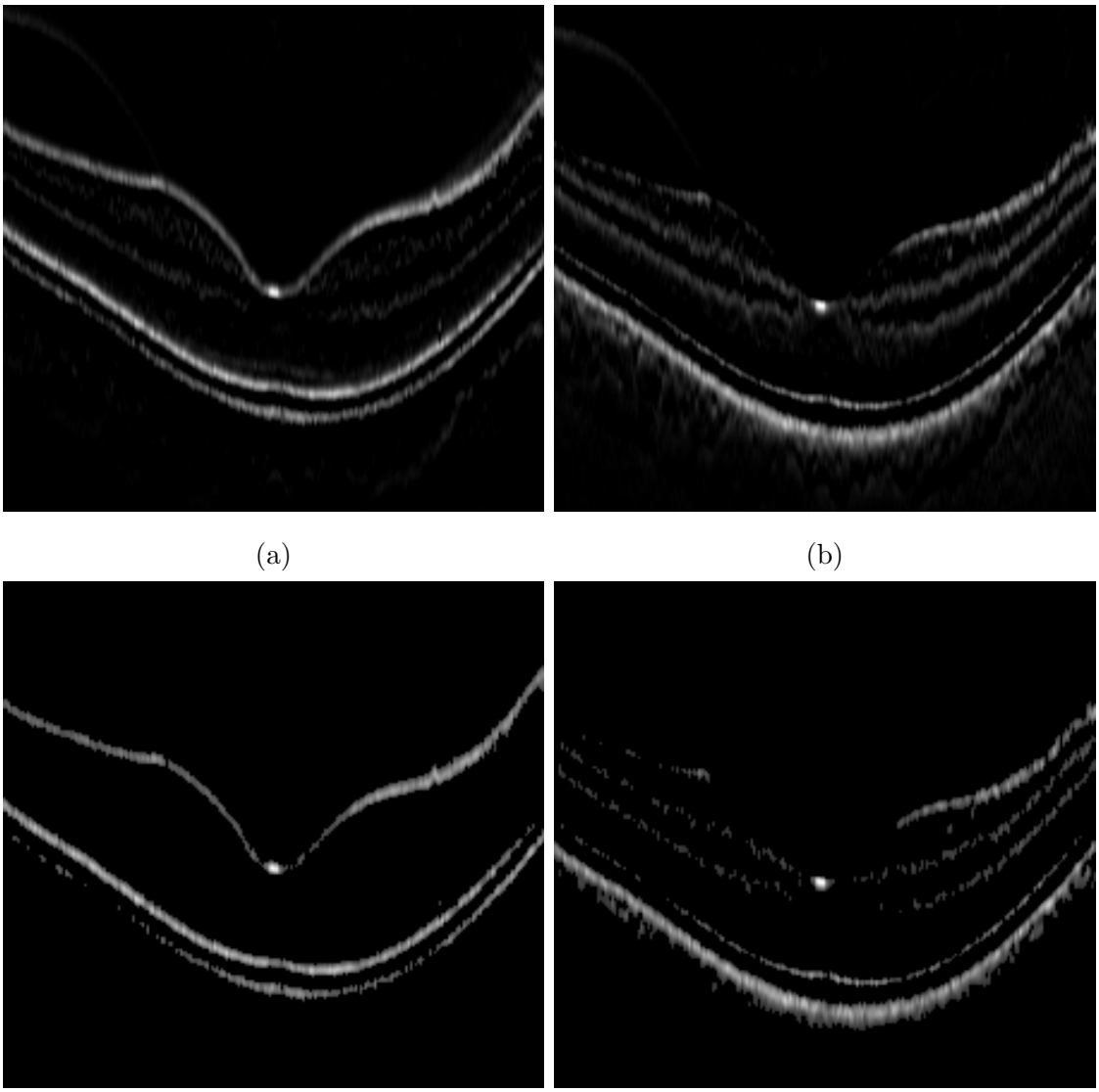
Figure 2.2.: Final segmentation result for the sample image used for the preprocessing figures. Results are interpolated back from smaller images created using A-scan reduction using spline interpolation. Again, the ILM is the top layer, while the bottom layer is the RPE.

contrast was determined for each location $a_j$ (depth $j$ of A-scan $a$) by the following equation:

$$(\sum_{i=0}^{4} a_{j-i})^2 - (\sum_{i=1}^{5} a_{j+i})^2 \tag{2.3}$$

This equation gives a smooth representation of the vertical contrast at an area. One can compare the artifact in the vitreous layer in Fig. 2.1 with the corresponding contrast values in the same area in Fig. 2.3 to see why such a formulation for contrast would be beneficial: including a few values both above and below a specific site in the calculation for contrast eliminates the chance that an artifact will be later mistaken for one of these surfaces of interest. High values of this contrast are useful for finding the RPE (Fig. 2.3(b)), while low values (negative values) are used for the ILM (Fig. 2.3(a)). For each surface, points of interest are isolated from the rest. For the RPE, only the top 25% of contrast values are kept, and only the lower 25% of values are kept for the ILM. This results in points of interest (with "interest" quantified by intensity) seen in Fig. 2.3(c,d).

Finally, as shown in Fig. 2.4(a,b), these points of interest are skeletonized into initial binary estimates of surface location using the MATLAB function *imregionalmax()* [25]. If, at this point, distance transforms were to be made of these volumes,



(a) (b)

(c) (d)

Figure 2.3.: First preprocessing steps in determination of $\Phi^G_{\mathrm{ILM}}(z))$ (for the ILM, left column) and $\Phi^G_{\mathrm{RPE}}(z))$ for the RPE (right column), the two distance transform volumes determined from layer-specific contrast. (a,b) Single contrast images from each volume. (c,d) Contrast images reduced to the most relevant 25% of points for the ILM and RPE, respectively.

and the deformable sheets were evolved using their guidance, accuracy would be substantially compromised. For example, some small specks of high contrast are still present below the RPE. The model will successfully pass through a small number of these specks; however, a large amount can significantly diminish segmentation accuracy. The specks are removed to give Fig. 2.4(c). Additionally, it is only desired to direct the two deformable model sheets to the contrast that represents the ILM and the RPE, while other layers within the retina exhibit extreme contrast (positive or negative).

One final preprocessing step is used to remove this extraneous contrast from consideration. Focus will first be placed on the RPE. The steps to isolate its contrast are as follows (and are performed per image):

1. Remove from consideration all points of interest except those that are the first to be encountered from the bottom of the image.

2. Find all connected components (given a standard 8-voxel neighborhood).

3. Determine the size of each component.

4. Using the largest component as a starting point, grow it outward from both ends, picking the points of highest contrast (one per A-scan) in a preselected neighborhood above and below the current depth until the ends of the image are reached.

The method used to select the contrast was identical for the ILM, except for choice of neighborhood during the growth phase of this algorithm. The result of this step is shown in Fig. 2.5. The volumes obtained are termed $G_{\text{ILM}}$ and $G_{\text{RPE}}$. The result looks to be quite accurate already; however, the inherent smoothness of the deformable model ameliorates some of the roughness introduced by the above four-step approach. Unfortunately, small errors in localization of some layer-specific contrast persist when using the above approach. An algorithm using HCF was developed to take $G_{\text{ILM}}$ and

(a) (b) (c)

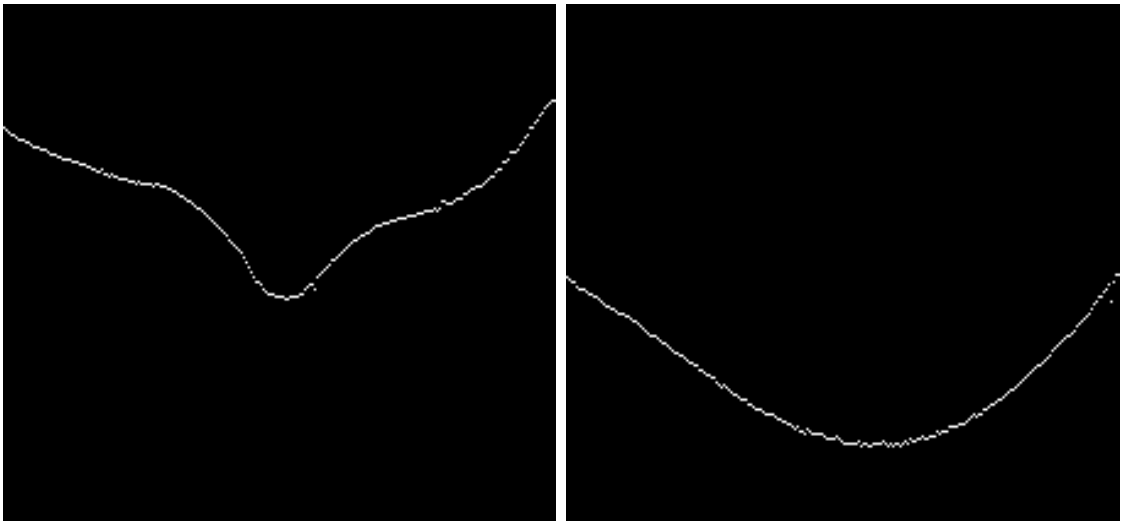
Figure 2.4.: (a,b) Binary images resulting from the skeletonization of information from 2.3(c,d) Nearly fully connected lines are present for both the ILM and the RPE, demonstrating the utility of this contrast method in detecting these layers. In (c), (b) has been processed to remove all connected components smaller than five voxels. Such a step helps to ensure detection of contrast specific to the RPE in the subsequent step shown in Fig. 2.5.

$G_{\text{RPE}}$ as input and output more accurate volumes from which to derive the data energy.

### 2.1.2 Distance-based HCF Algorithm

The distance-based HCF algorithm ("the algorithm") allows for the case of an MRF in which neighbors in the connected layer impart clique potentials to a site $s$ with consideration for their distance in the **image** domain from $s$. The traditional notion of an MRF assumes no idea of spatial distance in calculating the interactions of neighbors, indeed graphs traditionally assume no coordinate space, only containing a set of nodes and a set of edges. Images, and image stacks, however, are often modeled by graphs, and it is natural to use information from the image (including spatial resolutions) as prior knowledge before performing inference.

The principal use of the following algorithm is its ability to reduce the multilayer segmentation problem to a 2D graph, where each column of the voxels in the image

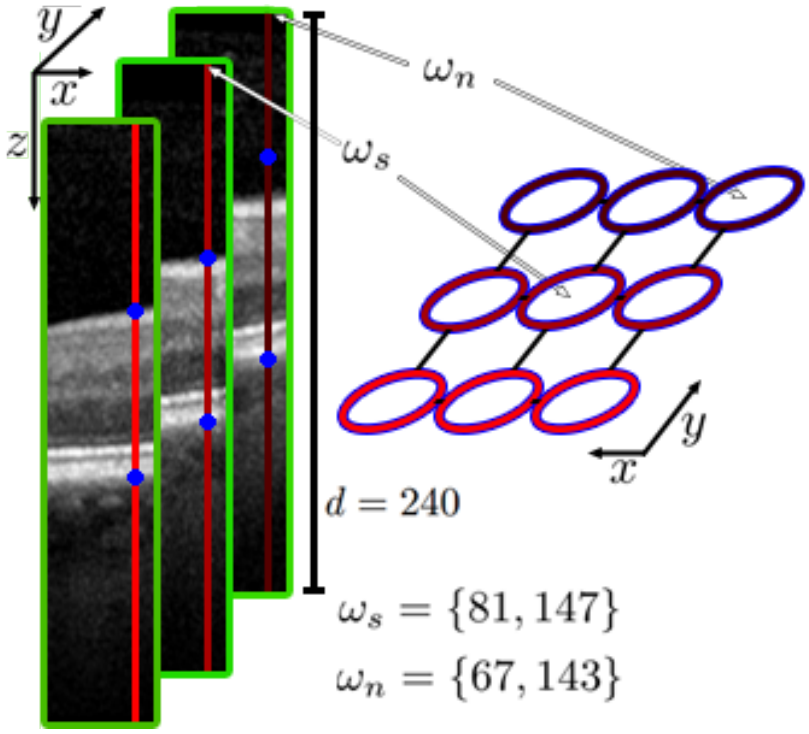


(a) (b)

Figure 2.5.: Using the algorithm mentioned in the text and the skeletonized contrast in 2.4, binary volumes $G_{\text{ILM}}$ and $G_{\text{RPE}}$ are obtained. A representative image pulled from $G_{\text{ILM}}$ is seen at left, and one from $G_{\text{RPE}}$ is seen at right.

volume represents a graph node. An illustration of this conversion is shown in Fig. 2.6. If we can consider $d$ to be the depth/height in voxels of an A-scan and $a$ to be the number of crossing or non-crossing surfaces to segment from the image, the classes to which an A-scan may belong could be as high as $d^a$. This is the primary driver of the complexity of the algorithm. Each time the stability of a site is changed using a heap implementation, the energetic states of all classes of the site's neighbors must be checked, and stability recalculated. For this problem, we segment two non-crossing surfaces. The total number of possible classes per A-scan in this case is $\frac{d(d+1)}{2}$.

Recall equation 1.2, where $\sum_{c \in C} e_c(\Omega)$ is representative of the prior information present in the graph. In the original HCF formulation, the prior information is present in edges (object boundaries/edges, not graph edges) between pixels, not pixels themselves. Specifically, potentials associated with both vertical and horizontal edges and non-edges are given values that encourage certain neighborhoods of edges, e.g. those that are straight and continuous. These potentials (especially the smaller one-clique




(a)

Figure 2.6.: Demonstration of the mathematical concepts as they relate to the example application. A set of nine neighboring A-scans all of depth $d$, shown in various shades of red, are represented as nodes in an MRF. Nodes of different shades are from separate images and are spatially much farther away than their neighbors from the same image. Their clique potential and resulting influence during the algorithm can be chosen to be modulated by these varying distances. The blue dots represent the current labels $\omega = \mathbf{x}_s \in \mathbf{X}_s$. The first value in a label represents the voxel depth along an A-scan of estimated top of the RNFL, the second represents that of the RPE. Here, the central site $s$'s current label $\omega_s$ has an estimated depth for the top of the RNFL that is much higher than that for the neighbor $n$, with label $\omega_n$.

and two-clique configurations) must be carefully selected for each specific segmentation problem. Furthermore, such potentials do not carry as much meaning when assigned to pixels, which have no inherent directionality.

Finally, for our particular application the algorithm operates on an initialization of the labels on the 2D map, specifically, the binary contrast volumes $G_{\text{ILM}}$ and $G_{\text{RPE}}$. Each A-scan (site) $s$ is assigned a class, an $a$-tuple of depths for each of the surfaces to be segmented:

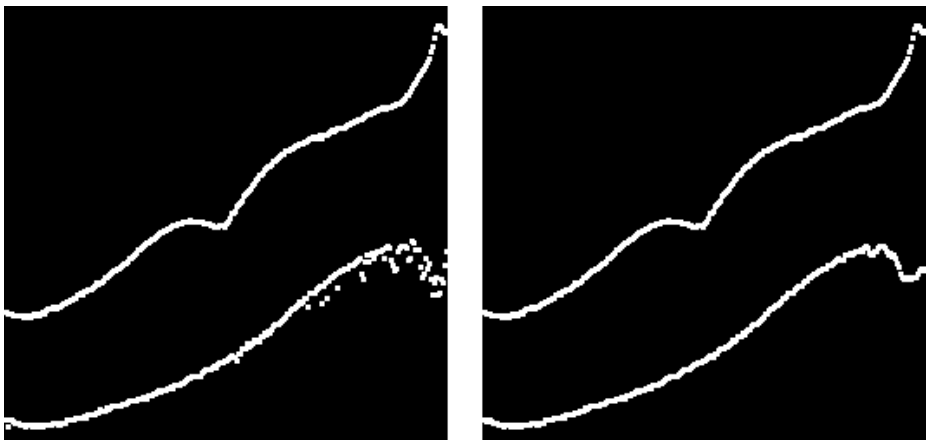
$$\mathbf{x}_s = \{x_{s1}, x_{s2}, ..., x_{sa}\}, x_{s1} \leq x_{s2} \leq ... \leq x_{sa} \leq d \tag{2.4}$$

The set of classes to which $\omega_s$ can be assigned is the set of possible tuple labels $\mathbf{X}$. This initialization of labels eliminates the initial phase of HCF, and all $s$ begin in the committed state. Another application may find it suitable to not use a separate automatic initialization; the algorithm as it is presented here accommodates this option. An example initialization is shown in Fig. 2.7(a), and corresponds to an image from the binary volume $G_{\text{ILM}}|G_{\text{RPE}}$ (the binary OR operation between the two volumes).

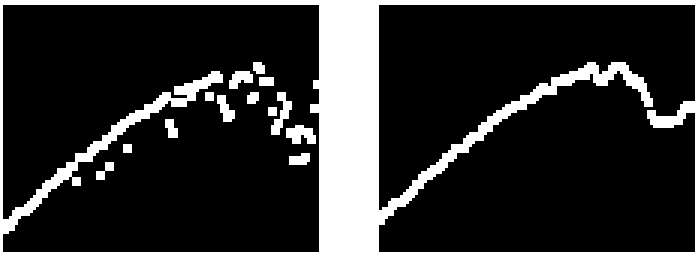
The clique potentials that we describe here are to be used when pixels/voxels are the focus of the segmentation. Potentials among neighbors are chosen to encourage smoothness, but the influence each potential has in the energy function is mediated by the distance between those neighbors in the image domain. For reference, an image volume from the Spectralis machine is 61 images total, and neighboring voxels from separate images are ∼ten times farther from each other than neighbors in the same image. It is natural to include this distance in the clique potential calculation. The clique potential for two neighbors $n_l, n_r \in N_s$ equidistant from and on opposite sides of $s$ at a distance of $dist(n, s)$ is

$$e_c(s, n_l, n_r|\Omega) = \frac{1}{dist(n, s)} \sum_{i=1}^{a} abs(x_{n_l i} + x_{n_r i} - 2x_{si}) \tag{2.5}$$

In practice, the spatial resolutions are constant along each dimension, and this vector of resolutions $\mathbf{r}$ only need be passed to an energy function. The sum of these clique potentials for all pairs $n_l, n_r$ given class state $\mathbf{x}_s$ combined with the likelihood of each class provides the equation for the energy $E_s(\mathbf{x}_s)$ for $\mathbf{x}_s$ at any given time during the algorithm's progress:



(a)

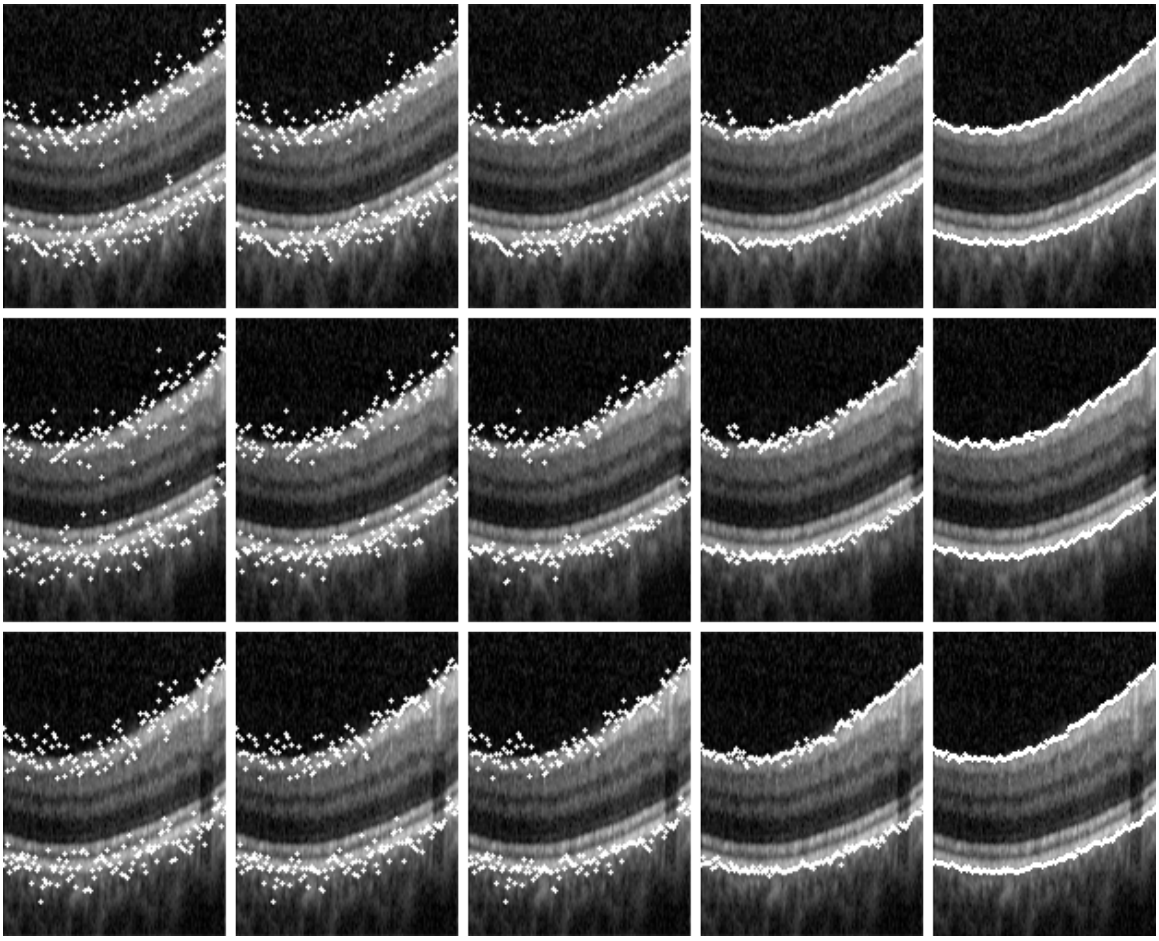


(b)

Figure 2.7.: Comparison of an initialization of a retinal scan (left) and the result after processing using HCF (right). (b) included to show a close-up of the changes.

$$E_s(\mathbf{x}_s) = \sum_{n_l, n_r \in N_s} e_c(s, n_l, n_r|\Omega) - \lambda(P(\mathbf{x}_s|\mathbf{f}_s)) \tag{2.6}$$

As before, a configuration of class labels is denoted $\Omega$. Given the above equation, it can be seen that assigning a class label $\mathbf{x}_s$ that will result in surfaces who more smoothly vary, i.e. have more constant derivatives in all directions, will have more stability, provided these labels also have salient features $\mathbf{f}_s$, whose influence is controlled by a positive parameter $\lambda$. HCF proceeds by changing the site whose current labeled state is least stable. Stability ranges from negative values, signaling instability, to $\geq 0$ when a site is completely stable. Our formulation does not change from the original equation for committed sites:

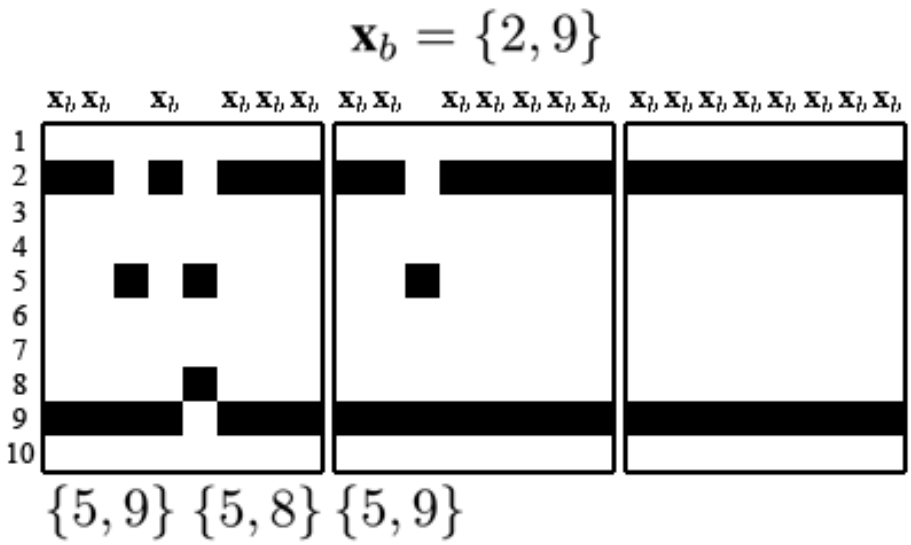


(a)

Figure 2.8.: Demonstration of the results of the algorithm on sections of three consecutive B-scans. Labels are shown in white. Each column of images gives a snapshot of the algorithm's progress: initialization of the images is seen at left, and each column of images shows a progression until convergence in column 5. The distance between voxels along an image was set to 1, that between images was set to 2, and $\lambda = 30$.

$$G_s(\mathbf{x}_s) = \min_{\mathbf{x}_s \in \mathbf{X}_s} (E_s(\mathbf{x}_s) - E_s(\omega_s)) \tag{2.7}$$

For the committed site, stability is the difference between the lowest non-selected energy state and the current one, $\omega_s$. For uncommitted sites, stability is the negative difference between the label $l$ that gives the $2^{nd}$ smallest energy for a site and that




(a)

Figure 2.9.: Toy demonstration of the algorithm in a 2D scenario, where all likelihoods are equivalent. The left image represents algorithm initialization, where most site labels $\omega_s = \mathbf{x}_b = \{2, 9\}$. As a result, the 3rd and 5th columns/sites are unstable, with nonnegative stability achieved with site label $\mathbf{x}_b$. The algorithm adjusts the label of the 5th column first due to its more negative stability.

which gives the smallest energy. When stability is $\geq 0$ for a site, it will not be changed, as only an increase in energy can come from changing its state. As previously explained, almost all applications of HCF in the past have been limited to 2D images of isotropic dimensions. See Fig. 2.8 for an example of the segmentation progression of a poor initialization on a subset of three consecutive images of an OCT scan. This example demonstrates how the algorithm effectively pulls the initialization into a proper segmentation as unstable sites are converted to more stable ones.

The conversion from an originally 3D problem to 2D inference is a one benefit of our solution; the second comes from the flexibility provided by distance-based potentials. For a simple demonstration, please see Fig. 2.9. A heap is desirable for

the algorithm, as suggested by Chou et al. [18]. We provide here the pseudocode for the algorithm for reference:

**Algorithm 1** Distance-Based HCF for Surface Segmentation

$\mathbf{r}$ = spatialResolutions
**for** voxelColumn $\in$ Image **do**
  $s = voxelColumn$
  $L(s) = calculateLikelihoods(s)$
  **if** $initialization = automatic$ **then**
    $\omega_s = initializeSite(s)$
    $\Omega(s) = \omega_s$
  **else**
    $\Omega(s) = \mathbf{x}_0$
**for** $s \in \Omega$ **do**
  **for** $\mathbf{x}_s \in \mathbf{X}$ **do**
    $E_s(\mathbf{x}_s) = Energy(\mathbf{x}_s, L(s), \Omega(\{s, N_s\}), \mathbf{r})$
  $G(s) = Stability(E_s)$
$Heapify(G)$
**while** $(G_{min}) < 0$ **do**
  $s = site(G_{min})$
  $\Omega = changeLabel(s, \Omega)$
  **for** $\mathbf{x}_s \in \mathbf{X}$ **do**
    $E_s(\mathbf{x}_s) = Energy(\mathbf{x}_s, L(s), \Omega(\{s, N_s\}), \mathbf{r})$
  $s.stability = Stability(E(s))$
  $delete(G, G_{min})$
  $insert(G, s)$
  **for** $n \in N_s$ **do**
    $delete(G, n)$
    **for** $\mathbf{x}_n \in \mathbf{X}$ **do**
      $E_n(\mathbf{x}_n) = Energy(\mathbf{x}_n, L(n), \Omega(\{n, N_n\}), \mathbf{r})$
    $n.stability = Stability(E_n)$
    $insert(G, n)$

The remedied output of the HCF algorithm is an adjusted version of the binary volumes $G_{\text{ILM}}$ and $G_{\text{RPE}}$. The distance maps of these, calculated for each point in the image stack using only values directly above and below i.e. based only in the $z$-direction, are represented by the unsigned distance maps

$$\Phi^G_{\text{ILM}}(z) = \min_{z \in G_I} ||z - z_{G_I}||, \forall z \in \Omega \tag{2.8}$$

$$\Phi^G_{\text{RPE}}(z) = \min_{z \in G_R} ||z - z_{G_R}||, \forall z \in \Omega \tag{2.9}$$

which are the argument of the data term of the overall energy functional presented here. These final unsigned distance maps can be seen in Fig. 2.10.

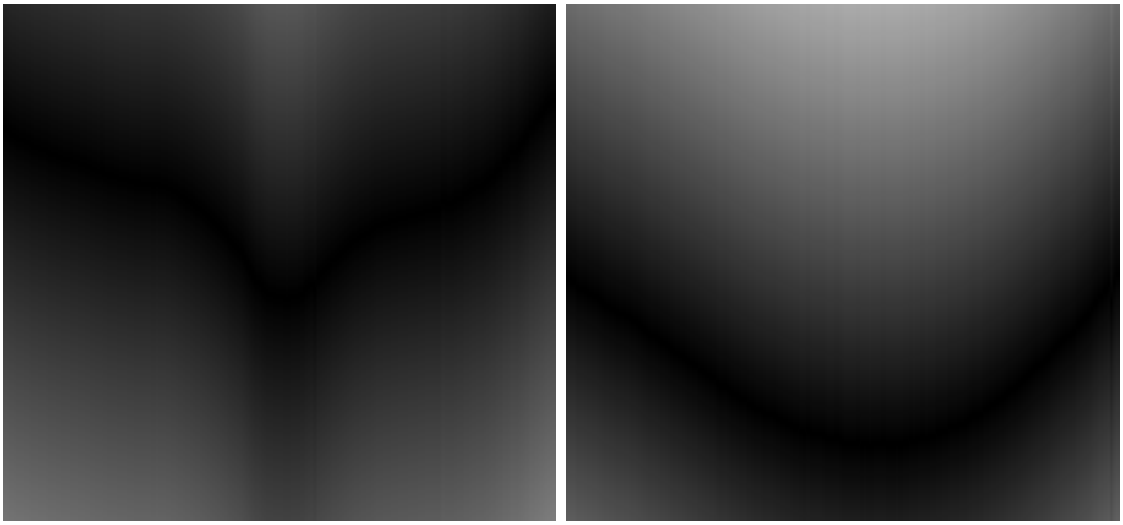
With the data term explained, we can move on to the topology term of the energy functional. The topology distance map $\Phi^t$ is derived from the two transforms $\Phi_{\text{ILM}}(z)$ and $\Phi_{\text{RPE}}(z)$.

$$\Phi^t(z) = \left\{ \begin{array}{ll} \frac{|\Phi_{\text{ILM}}(z)+\Phi_{\text{RPE}}(z)|+\beta}{4} & |\Phi_{\text{ILM}}(z) + \Phi_{\text{RPE}}(z)| + \beta > 0 \\ 0 & \text{otherwise} \end{array} \right\} \tag{2.10}$$

Essentially when the models are at a large distance apart, this term will provide information to each model to pull the two sheets each 1/4 of the distance between them with $\beta$ a distance buffer to be kept between them. For this experiment, with vertical resolution of 3.8717$\mu$m/voxel and a maximum allowable retinal thickness of 400$\mu$m, $\beta$ was chosen to be 103. With a description for each distance map to be used in model evolution in place, the model's deformations can be described.

## 2.2 Model Evolution

Free-form deformations (FFD) uses a grid of control points to manipulate a solid in space [26]. The idea was originally presented as an application to the field of 3D animation, where a simple adjustment in control points would allow for intuitive adjustment to the solid in areas around those control points. Imagine a user interface that presents an egg-shaped solid, with a grid overlaid on its surface. The vertices of the grid are termed control points. Pulling on one of the points stretches the



(a) (b)

Figure 2.10.: A single image from each of the distance volumes $\Phi^G(z)$. These volumes serve as the argument of the data energy: $E_{\text{data}}(\Phi^G_{\text{ILM}}(z), \Phi^G_{\text{RPE}}(z))$.

area around it in a predetermined fashion (based on spline equations), and the result requires much less work than moving each part of the solid separately. This is a type of internal smoothness, and is useful for this segmentation application [27]. The grid overlaid on each of the models for each step of deformations is described as:

$$L = \{(L^x, L^y)|a, b\}; \tag{2.11}$$

where $a$ and $b$ represent integer-valued grid spacing in the $x$ and $y$ directions, respectively. As a practical example of model deformations, we consider the first deformations of a given iteration. $S_{\text{I}}$, deforms according to the voxel values $\mathbf{z}_{\text{ILM}}$ of $\Phi_{\text{ILM}}$ to be used in calculating control-point deformations $\mathbf{d}_{\text{ILM}}$. A single control-point deformation is calculated using a weighted map derived using cubic B-spline functions (Fig. 2.11). This map gives more weight to the voxels in close proximity to the control point, and varies with the chosen spacing of the grid $L$ overlaid to each model. Specifically, for a given spacing $a, b$ for a surface, the corresponding map is derived from the following basis equations

$$\mathbf{m} = (\frac{1}{a}, \frac{2}{a}, \ldots, \frac{a}{a})^T \tag{2.12}$$

$$\mathbf{n} = (\frac{1}{b}, \frac{2}{b}, \ldots, \frac{b}{b})^T \tag{2.13}$$

$$\mathbf{m}_1 = (\mathbf{1} - \mathbf{m}).^3 \tag{2.14}$$

$$\mathbf{m}_2 = \mathbf{3} \odot \mathbf{m}.^3 - \mathbf{6} \odot \mathbf{m}.^2 + \mathbf{4}) \tag{2.15}$$

Boldface numbers indicate integral vectors the same length as other vectors in an equation whose lengths are predefined. $\odot$ indicates the Hadamard product (entrywise multiplication), and the operation $x.^y$ indicates an entrywise power $y$ of the vector $x$. The same operations are used with the vector $n$ to form the vectors $\mathbf{n}_1$ and $\mathbf{n}_2$. Then to form a quadrant of the deformation map, matrix multiplication is required:

$$M_1 = \begin{pmatrix} \mathbf{m}_1 \cdot \mathbf{n}_1^T & \mathbf{m}_1 \cdot \mathbf{n}_2^T \\ \mathbf{m}_2 \cdot \mathbf{n}_1^T & \mathbf{m}_2 \cdot \mathbf{n}_2^T \end{pmatrix} \tag{2.16}$$

Finally, the other quadrants can be made by reversing the order of the elements of $M_1$ along the first dimension ($M_2$), the second dimension ($M_3$), or both ($M_4$). The final result is the final deformation map (Fig. 2.11):

$$M = \begin{pmatrix} M_1 & M_3 \\ M_2 & M_4 \end{pmatrix} \tag{2.17}$$

The map is normalized so that its elements sum to 1 after this step. Evolution using these maps is an iterative process. Deformations of the control-point grid $L$ overlaid to a deformable sheet are calculated using information present around the model. Once the grid has deformed, however, the reverse process occurs: each voxel that is a part of the deformable model is imparted a specific deformation by each of the control points on which it originally had an effect. This second step of the process uses an inverse version of the deformation maps. In more technical terms, through this second map, the $4 \cdot 4$ grid of control points around each voxel sector of the current model imparts weighted deformations to each of $\mathbf{z}_{\mathrm{ILM}}$. The model $S_{\mathrm{I}}$ is moved to the new voxel locations $\hat{\mathbf{z}}_{\mathrm{ILM}}$.

Returning to Fig. 2.11, note the chosen spacing for the control points of the two models. The ILM displays relatively quick changes in topology due to the presence of blood vessels just under the surface. A spacing of $a = b = 1$ for the ILM allows flexibility and more finely grained deformations than the $a = 7, b = 1$ spacing selected for the RPE. Referring to Fig. 2.2, this makes intuitive sense, as the RPE's slope is gradually changing; the larger spacing will most likely improve the segmentation. This is an example of the intrinsic smoothness of using FFD to evolve a deformable model [27]. Taken from a different perspective, the model $S_{\mathrm{R}}$ has less difficulty moving through a patch of noise when there are more points, i.e. more information to contribute to each individual deformation and avoid incomplete boundaries.

With the means to minimize the objective function using FFD explained, the iterative process of minimization can now be explained. The total topology and data energy terms are described in terms of their arguments:

$$E_{\text{topology}}(\Phi^t(z)) = \iint \Phi^t(\mathbf{z}_{\text{ILM}})\,dz + \iint \Phi^t(\mathbf{z}_{\text{RPE}})\,dz \tag{2.18}$$

$$E_{\text{data}}(\Phi^G_{\text{ILM}}(z), \Phi^G_{\text{RPE}}(z)) = \iint \Phi^G_{\text{ILM}}(\mathbf{z}_{\text{ILM}})\,dz + \iint \Phi^G_{\text{RPE}}(\mathbf{z}_{\text{RPE}}))\,dz \tag{2.19}$$

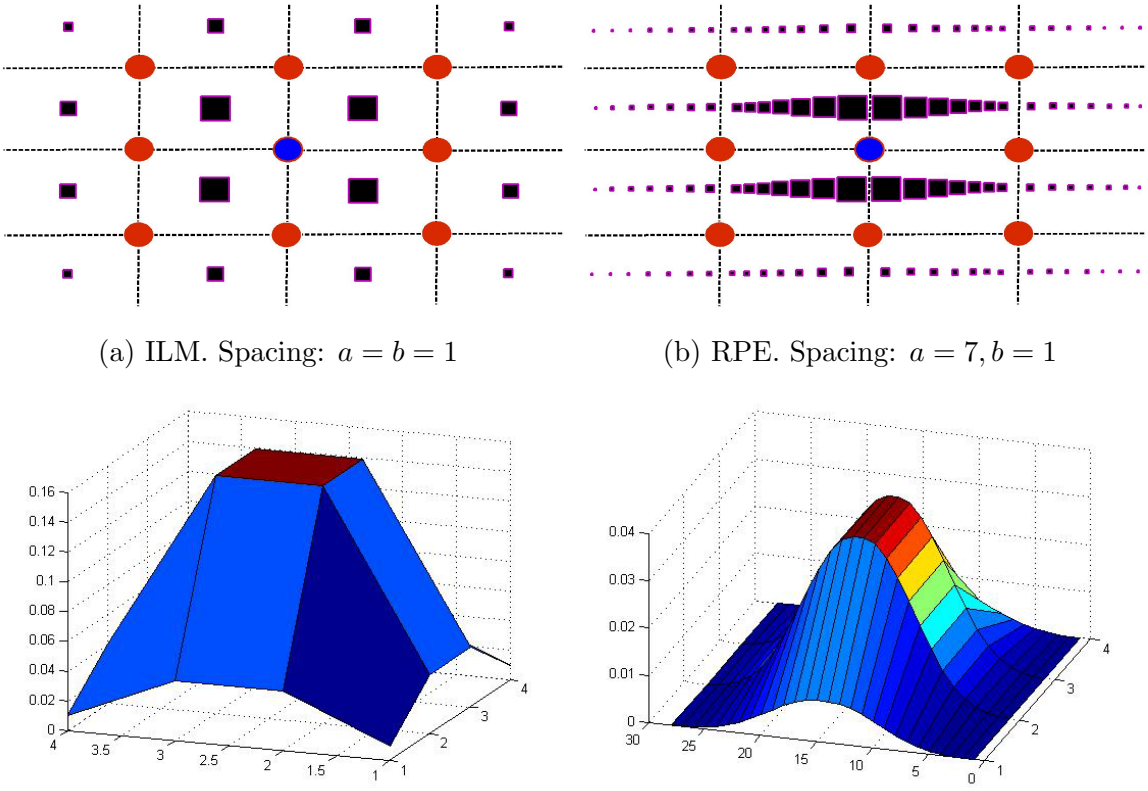
A single iteration of model deformations, given $\hat{\mathbf{z}}$ as new energy values on a given map for a given model, results in an energy reduction of

$$\frac{\partial E_{\text{total}}}{\partial z} = \frac{\partial E_{data}}{\partial z} + \frac{\partial E_{topology}}{\partial z} \tag{2.20}$$

$$\frac{\partial E_{\text{data}}}{\partial z} = \iint \Phi^G_{\text{ILM}}(\mathbf{z}_{\text{ILM}} - \hat{\mathbf{z}}_{\text{ILM}})\,dz + \iint \Phi^G_{\text{RPE}}(\mathbf{z}_{\text{RPE}} - \hat{\mathbf{z}}_{\text{RPE}})\,dz \tag{2.21}$$

$$\frac{\partial E_{\text{topology}}}{\partial z} = \iint \Phi^t(\mathbf{z}_{\text{ILM}} - \hat{\mathbf{z}}_{\text{ILM}})\,dz + \iint \Phi^t(\mathbf{z}_{\text{RPE}} - \hat{\mathbf{z}}_{\text{RPE}})\,dz \tag{2.22}$$

for each energy term and together the total energy of the models. This process is repeated until model energy falls below a predetermined $\epsilon$. $\Phi^t(z)$ will naturally push the models past outlying noise to within an intuitive range of each other given the $\beta$ distance buffer, at which point the topology term effectively drops out of the equation. The data-driven term provides the precise information past this point for an accurate segmentation.




(a) ILM. Spacing: $a = b = 1$

(b) RPE. Spacing: $a = 7, b = 1$

(c)

(d)

Figure 2.11.: Maps of the weights used to impart deformations $d_S$ to control points of the grid $L$ by surrounding voxels $\mathbf{z}$. For either, the deformation given to the blue control point at center is a weighting of the deformation values of the surrounding black squares (voxels) proportional to their size. Voxel values are normalized to faithfully map vertex distances to appropriate control-point-deformation distances. (a) For the model evolving to the ILM, each deformation is more flexible with only 16 voxels influencing each control point. (b) Spacing for the model evolving to the RPE, $a = 7, b = 1$, allows for more voxels, 112 in total, to contribute each control point's deformation. This adds smoothness to the model at the expense of flexibility. The 3D representations of these pixel grids are included in (c) and (d) for clarity. In these latter two figures, the intersections represent voxels, and the height of each intersection is the proportion of influence of that voxel to a control point's movement. All proportions sum to 1.

# 3 RESULTS AND CONCLUSIONS

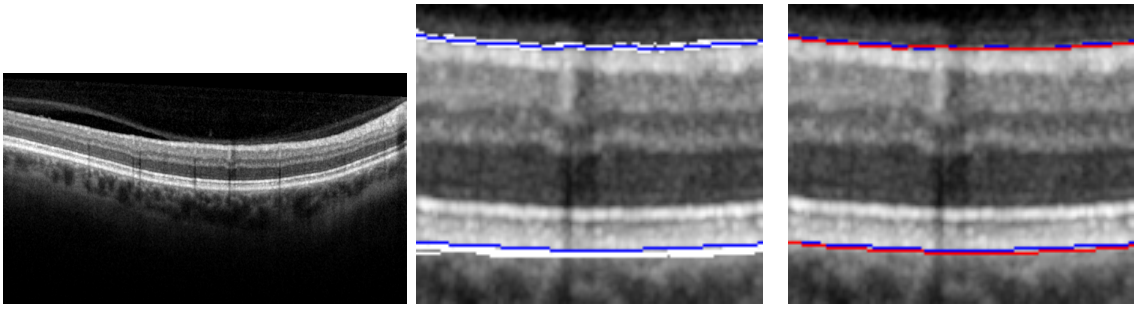
Ten healthy study participants were given bilateral macular SD-OCT scans using the Heidelberg Spectralis SD-OCT tomographer, resulting in 20 full image stacks of $768 \cdot 496 \cdot 61$ voxels. Spectralis segmentation results for the ILM and the RPE were extracted from native Heidelberg Eye Explorer (HEYEX) .vol software files using the ImageJ plugin Open Heyex Raw [28]. The segmentation results for the authors' method was obtained using an Intel i7 2.8GHz laptop, and the average segmentation time for a given image stack was 7 seconds. When including HCF, average segmentation time was 3 minutes per image stack. For clarity's sake, segmentation data for a layer is the determined integer-valued voxel height of the layer from the top of the stack for each column of voxels in the processed image stack. It should also be noted that though the optic nerve head (ONH) is present at least partially in all macular images taken by the Spectralis, the area up to and including 1/2 the optic nerve radius was removed from statistical consideration for this study. Thickness measurements around the optic nerve head are not used in the clinical testing for which this method is designed, and the methods used here are unsuitable for such thickness determination near the ONH.

Two qualified graders were asked to manually segment the RPE and the ILM for 30 images chosen randomly from the set of 20 macular image stacks. Segmentation was performed in triplicate for each of the 30 images to allow for intra-grader variability. To demonstrate segmentation efficacy, the error of the proposed method was compared with that of native Heidelberg Spectralis software, the current state of the art for retinal SD-OCT methods, for three different datasets: the means of the first grader's delineations, those of the second grader's delineations, and those of both graders (Table 3.1). The first three rows of the data in Table 3.1 summarize the main points of the results. For the ILM, the Spectralis software's mean error is slightly better with

Table 3.1.: Mean errors and corresponding standard deviations for the proposed method with and without HCF and the Heidelberg Spectralis software when compared to both graders or each grader separately. Superior retinal thickness calculation is achieved by the deformable models. All values are in $\mu$m, and for the given application $3.8717\mu\text{m} = 1$ voxel.

| | | **Error ($\mu$m)** | | | | | |
|---|---|---|---|---|---|---|---|
| | | ILM | | RPE | | Thickness | |
| **Grader** | **Method** | Mean | Dev. | Mean | Dev. | Mean | Dev. |
| **Both** | Deform. | 3.33 | 2.64 | 4.23 | 2.77 | 6.61 | 3.90 |
| | HCF | 3.25 | 2.38 | 4.22 | 2.47 | 6.87 | 3.69 |
| | Heid. | 2.74 | 2.36 | 6.71 | 2.26 | 8.57 | 3.63 |
| **Gr. 1** | Deform. | 2.90 | 2.58 | 3.59 | 2.63 | 5.30 | 3.71 |
| | HCF | 2.82 | 2.28 | 3.58 | 2.36 | 5.47 | 3.57 |
| | Heid. | 2.48 | 2.28 | 5.94 | 2.32 | 6.98 | 3.72 |
| **Gr. 2** | Deform. | 4.07 | 2.94 | 5.04 | 3.20 | 8.23 | 4.36 |
| | HCF | 3.98 | 2.71 | 5.01 | 2.92 | 8.51 | 4.14 |
| | Heid. | 3.38 | 2.75 | 7.49 | 2.78 | 10.23 | 4.10 |

a mean error $2.74\mu$m versus the authors' program at $3.33\mu$m. The RPE results show much less error for the authors' program versus the Spectralis at 4.23 and 6.71 mean errors respectively. As would be expected, this results in an overall better fit of the deformable model method over the Spectralis when considering the retinal thickness calculation $S_{\text{R}} - S_{\text{I}}$. Final results show a mean retinal thickness error of $6.61\mu$m versus $8.57\mu$m for the Spectralis. The addition of HCF does not significantly change these findings. Values are comparable to the deformable-model-only approach for all cells of the table. This is an expected result; the B-scans used for these results did not suffer from poor initialization of $\Phi^G_{\text{ILM}}(z)$ and $\Phi^G_{\text{RPE}}(z)$. These results demonstrate



(a) (b) (c)

Figure 3.1.: Results from a graded SD-OCT image. (a) Original image. (b) For a close-up of a section of (a), graders' six delineations for the ILM and RPE are shown in white. Red for the deformable-model method and blue for the native Spectralis algorithm are shown where not in agreement. Nowhere does our method disagree with all six delineations at once. (b) The same close-up, but only exhibiting the deformable models' results in red, with Spectralis results shown in blue where not identical. It can be seen that the proposed method more consistently estimates the RPE to be deeper than the Spectralis.

clearly the effectiveness of both methods in determining retinal thickness, the value of interest.

Although manually segmented cube-like sections of the image stacks would be ideal for comparison, or perhaps, in addition to standard images, cross-sections of the stack would have been more informative, neither time nor feasibility of accurate segmentation allowed these other manual segmentation applications (Fig. 3.3(c)). Perpendicular images prove too grainy for accurate segmentation, and cube-like sections of a stack blind a grader from proper segmentation due to a lack of information. Given the image data was collected using a Spectralis, it is natural to use the Spectralis segmentation results as benchmarks by which to compare our results with the graders' ground truth. A clinical tool as accurate and swift as the current accepted state of the art is useful in accomplishing the clinical goals of this project. Demonstration of the validity of the results is shown in Fig. 3.1.

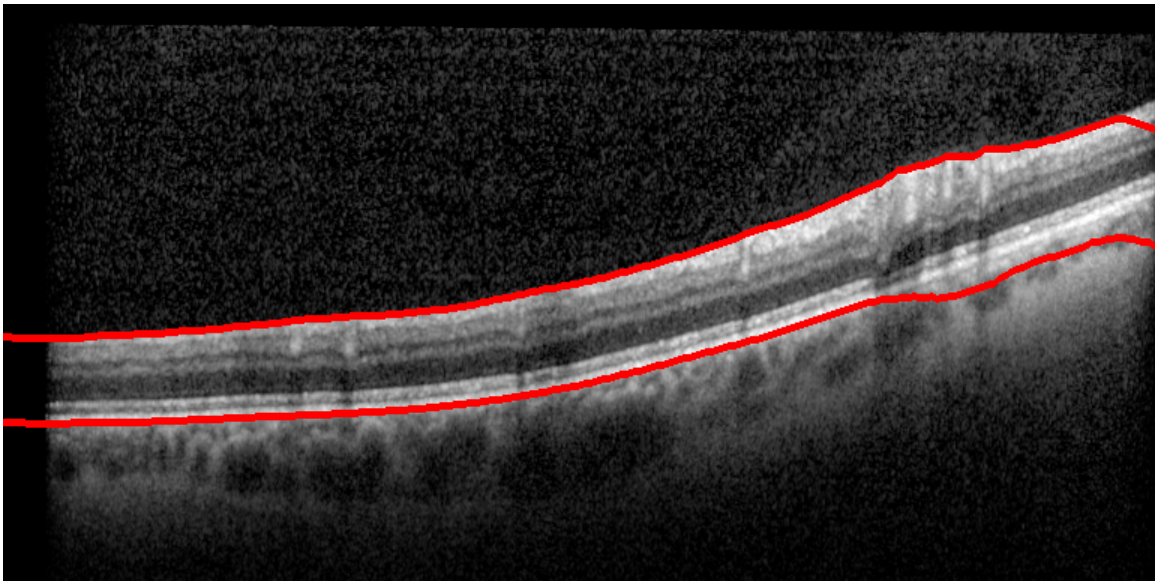
Table 3.2.: Mean errors and corresponding standard deviations for the proposed method with and without HCF from the Spectralis segmentation values for the entire dataset of 46 total retinal volumes used in the study. Values are included for the ILM, the RPE and for the determined retinal thickness. The number of data points successfully segmented is also included. Superior evaluation of layer location as well as overall retinal thickness is achieved by including HCF in the segmentation process. All values are in voxels.

**Error (voxels)**

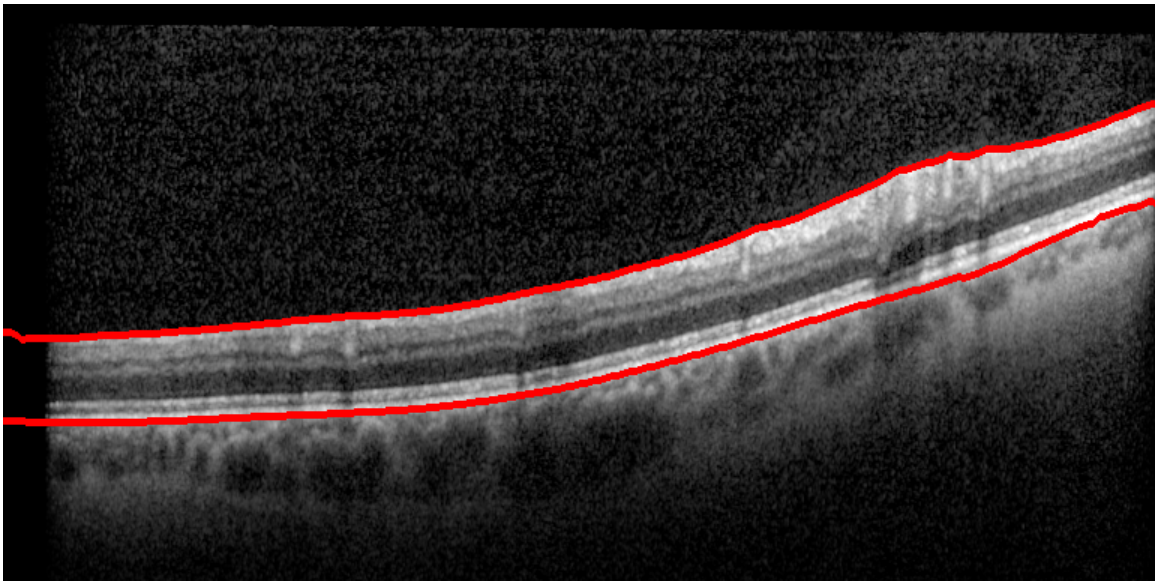
| | ILM | | RPE | | Thickness | | Data Points |
|---|---|---|---|---|---|---|---|
| **Alg.** | Mean | Dev. | Mean | Dev. | Mean | Dev. | |
| Def. Only | .6901 | .7061 | 1.0369 | 1.1603 | .9613 | 1.2730 | 1,838,530 |
| With HCF | .6189 | .7840 | .8336 | .8352 | .8523 | 1.0393 | 2,155,008 |

Finally, we can compare the overall segmentation results directly to the Spectralis results. This shows that, in the data our graders were unable to grade due to time constraints (the vast majority of our data), our results do not stray too far from acceptable. For the 46 total 61-image volumes segmented, including HCF improved the algorithm's performance in ILM, RPE, and total retinal thickness segmentation; see Table 3.2 and Fig. 3.2. Also shown in the table is the number of data points successfully segmented, showing that an adaptation was made for the second algorithm to be able to segment all areas despite the cropping necessary before due to "blank" A-scans. This change gives on average 17% more segmentation information per volume.

This thesis presents a novel approach to the segmentation of the inner and outer layers of SD-OCT image stacks. Previous studies using deformable models for this purpose had not exploited the information present between individual B-scans, and the presented dual-model framework provides accurate segmentation for a 3D image stack in a time acceptable even for clinical applications. An eventual goal of the

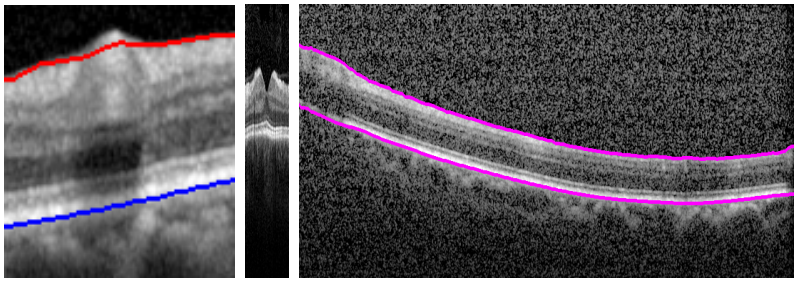


(a)



(b)

Figure 3.2.: Comparison of segmentations using the initial deformable-model algorithm with the results after inclusion of the proposed HCF method (right). The HCF method efficiently adjusts some errors in the binary volumes $G_{\mathrm{ILM}}$ and $G_{\mathrm{RPE}}$ to ensure selected areas of contrast agree with the information present between images.



(a) (b) (c)

Figure 3.3.: (a) A tendency of the deformable model for the ILM is to underapproximate sharp peaks in the layer. (b) Example cross-section taken along the y-direction of an image stack. It was determined that this type of scan could not be manually graded due to its low resolution. (c) Model results from a highly degraded scan from a glaucoma patient show promise for the method's eventual purpose as a clinical tool to distinguish between glaucoma and healthy patients.

proposed clinical study of which this algorithm is a key part is longitudinal evaluation of RNFL thickness changes in a larger subset of healthy and glaucoma patients; the program has covered some ground towards this effort as evidenced by our current findings. Results on highly degraded scans are promising as can be seen in Fig. 3.3(c) and also in the additional validation of our results with Heidelberg's segmentation on both healthy and glaucoma patient scans. Despite a thinning (less reflective) retinal nerve fiber layer in glaucoma patients, successful segmentation is possible given the current methods.

Despite considerable success in segmentation, the algorithm can still be improved. The errors corrected by the novel application of HCF are not perfect, and instead of the four-step algorithm for determining the binary volumes $G_{\text{ILM}}$ and $G_{\text{RPE}}$ from contast, a more content-aware algorithm for following a layer's contrast could be developed. For example, during the growth phase (step 4), the use of the contrast information more than one A-scan ahead (and perhaps from neighboring images) could better direct the creation of the final binary images.

One important improvement to the algorithm would be to successfully implement a multilayer segmentation approach. The proposed method has been designed for many-surface classification and can be easily extended to segment the remaining retinal layers. Such multilayer segmentation has immediate applicability to other biomedical images that involve crossing or non-crossing layers, where a variation of the class vector $\mathbf{x}_s$ may be used. For example, an object (instead of thin boundaries) may not exist in every column of an image volume. Without a constant number of objects to segment for each column, the vector may be allowed a fixed length $d$. Values may be binary in the case of single object, or integer-valued if the number of objects is greater. The clique potentials defined here would then be adjusted to smooth the identified locations of the boundaries of the object(s) in the image, specifically, the locations where the sequential values in $\mathbf{x}_s$ change.

Attempts were made to segment each of the layers in the OCT using the HCF approach; however, rarely does enough information exist for each layer to be confidently segmented. One of the most difficult tasks facing a segmentation algorithm is performing well under ambiguities, but performing reasonably when information is not present is even more difficult. I am confident that successful segmentation can be achieved using the approach presented here, provided significant functionality in the event of a lack of information for a layer is added to the program. A final improvement to the algorithm would add more exact inference by allowing fractional voxel locations for the deformable models and for the "contrast" locations.

REFERENCES

# REFERENCES


[1] Eye Diseases Prevalence Research Group et al. Prevalence of open-angle glaucoma among adults in the united states. *Archives of Ophthalmology*, 122(4):532, 2004.

[2] Harry A Quigley and Aimee T Broman. The number of people with glaucoma worldwide in 2010 and 2020. *British Journal of Ophthalmology*, 90(3):262–267, 2006.

[3] Mauro T Leite, Harsha L Rao, Linda M Zangwill, Robert N Weinreb, and Felipe A Medeiros. Comparison of the diagnostic accuracies of the spectralis, cirrus, and rtvue optical coherence tomography devices in glaucoma. *Ophthalmology*, 118(7):1334–1339, 2011.

[4] Sanjay Asrani, Jullia A Rosdahl, and R Rand Allingham. Novel software strategy for glaucoma diagnosis: asymmetry analysis of retinal thickness. *Archives of Ophthalmology*, 129(9):1205–1211, 2011.

[5] D Cabrera DeBuc. A review of algorithms for segmentation of retinal image data using optical coherence tomography. *Image Segmentation*, pages 15–54, 2011.

[6] Pratul P Srinivasan, Stephanie J Heflin, Joseph A Izatt, Vadim Y Arshavsky, and Sina Farsiu. Automatic segmentation of up to ten layer boundaries in sd-oct images of the mouse retina with and without missing layers due to pathology. *Biomedical Optics Express*, 5(2):348–365, 2014.

[7] Min Chen, Andrew Lang, Elias Sotirchos, Howard S Ying, Peter A Calabresi, Jerry L Prince, and Aaron Carass. Deformable registration of macular oct using a-mode scan similarity. In *Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on*, pages 476–479. IEEE, 2013.

[8] Stephanie J Chiu, Xiao T Li, Peter Nicholas, Cynthia A Toth, Joseph A Izatt, and Sina Farsiu. Automatic segmentation of seven retinal layers in sdoct images congruent with expert manual segmentation. *Optics Express*, 18(18):19413–19428, 2010.

[9] Mona Kathryn Garvin, Michael D Abràmoff, Xiaodong Wu, Stephen R Russell, Trudy L Burns, and Milan Sonka. Automated 3-d intraretinal layer segmentation of macular spectral-domain optical coherence tomography images. *Medical Imaging, IEEE Transactions on*, 28(9):1436–1447, 2009.

[10] Kang Li, Xiaodong Wu, DZ Chen, and M Sonka. Optimal surface segmentation in volumetric images-a graph-theoretic approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(1):119–134, Jan 2006.

[11] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[12] Mohammed J Zaki and Wagner Meira, Jr. *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press, May 2014.

[13] Delia Cabrera Fernández, Harry M Salinas, and Carmen A Puliafito. Automated detection of retinal layer structures on optical coherence tomography images. *Optics Express*, 13(25):10200–10216, 2005.

[14] Mircea Mujat, Raymond Chan, Barry Cense, B Park, Chulmin Joo, Taner Akkin, Teresa Chen, and Johannes de Boer. Retinal nerve fiber layer thickness map determined from optical coherence tomography images. *Optics Express*, 13(23):9480–9491, 2005.

[15] Akshaya Mishra, Alexander Wong, Kostadinka Bizheva, and David A Clausi. Intra-retinal layer segmentation in optical coherence tomography images. *Optics Express*, 17(26):23719–23728, 2009.

[16] Azadeh Yazdanpanah, Ghassan Hamarneh, Benjamin Smith, and Marinko Sarunic. Intra-retinal layer segmentation in optical coherence tomography using an active contour approach. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2009*, pages 649–656. Springer, 2009.

[17] Paul B Chou and Christopher M Brown. The theory and practice of bayesian image labeling. *International Journal of Computer Vision*, 4(3):185–210, 1990.

[18] PB Chou, PR Cooper, MJ Swain, CM Brown, and LE Wixson. Probabilistic network inference for cooperative high and low level vision. In R Chellappa and A Jain, editors, *Markov Random Fields: Theory and Applications*. Academic Press, 1993.

[19] Y Tsaig and A Averbuch. Automatic segmentation of moving objects in video sequences: a region labeling approach. *Circuits and Systems for Video Technology, IEEE Transactions on*, 12(7):597–612, Jul 2002.

[20] N Paragios and G Tziritas. Detection and location of moving objects using deterministic relaxation algorithms. In *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, volume 1, pages 201–205. IEEE, 1996.

[21] Yefeng Zheng, Huiping Li, and David Doermann. Machine printed text and handwriting identification in noisy document images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(3):337–353, 2004.

[22] Shiloh L Dockstader and A Murat Tekalp. Multiple camera tracking of interacting and occluded human motion. *Proceedings of the IEEE*, 89(10):1441–1455, 2001.

[23] Qi Yang, Charles A Reisman, Zhenguo Wang, Yasufumi Fukuma, Masanori Hangai, Nagahisa Yoshimura, Atsuo Tomidokoro, Makoto Araie, Ali S Raza, Donald C Hood, et al. Automated layer segmentation of macular oct images using dual-scale gradient information. *Optics Express*, 18(20):21293–21307, 2010.

[24] MATLAB Cubic Spline Interpolation. *Version 2012b*. The MathWorks Inc., Natick, Massachusetts, http://www.mathworks.com/help/curvefit/cubic-spline-interpolation.html, 2015. Accessed: 2015-01-29.

[25] MATLAB. *Version 2012b*. The MathWorks Inc., Natick, Massachusetts, 2015.

[26] Thomas W Sederberg and Scott R Parry. Free-form deformation of solid geometric models. In *ACM Siggraph Computer Graphics*, volume 20, pages 151–160. ACM, 1986.

[27] Xiaolei Huang, Dimitris Metaxas, and Ting Chen. Metamorphs: deformable shape and texture models. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 1, pages I–496. IEEE, 2004.

[28] EJ Knott, KG Sheets, Y Zhou, WC Gordon, and NG Bazan. Spatial correlation of mouse photoreceptor-rpe thickness between sd-oct and histology. *Experimental Eye Research*, 92(2):155–160, 2011.