

Segmentation of Three-dimensional Retinal Image Data

Alfred R. Fuller, Robert J. Zawadzki, Stacey Choi, David F. Wiley, John S. Werner, and Bernd Hamann

Abstract—We have combined methods from volume visualization and data analysis to support better diagnosis and treatment of human retinal diseases. Many diseases can be identified by abnormalities in the thicknesses of various retinal layers captured using optical coherence tomography (OCT). We used a support vector machine (SVM) to perform semi-automatic segmentation of retinal layers for subsequent analysis including a comparison of layer thicknesses to known healthy parameters. We have extended and generalized an older SVM approach to support better performance in a clinical setting through performance enhancements and graceful handling of inherent noise in OCT data by considering statistical characteristics at multiple levels of resolution. The addition of the multi-resolution hierarchy extends the SVM to have “global awareness.” A feature, such as a retinal layer, can therefore be modeled within the SVM as a combination of statistical characteristics across all levels; thus capturing high- and low-frequency information. We have compared our semi-automatically generated segmentations to manually segmented layers for verification purposes. Our main goals were to provide a tool that could (i) be used in a clinical setting; (ii) operate on noisy OCT data; and (iii) isolate individual or multiple retinal layers in both healthy and disease cases that contain structural deformities.

Index Terms—support vector machine, segmentation, image analysis, retinal, optical coherence tomography, volume visualization, image processing.

1 INTRODUCTION

Advancements in medical imaging are facilitating the extraction of accurate information from volumetric data making three-dimensional (3D) imaging an increasingly useful tool for clinical diagnosis and medical research. This development makes possible non-invasive examination and analysis of diseases by providing clinicians insight into the morphology of disease within the body and how it changes over time and through treatment. Common non-invasive imaging modalities are magnetic resonance imaging (MRI) and computed tomography (CT). Our efforts focus on the analysis and visualization of volumetric OCT retinal data. OCT, described in [8], is an acquisition system based on back-scattering of coherent light producing a stack of images similar to MRI and CT. A light beam is directed into a patient’s eye where reflected light is merged with a reference beam eliciting an interference pattern that is used to gauge reflectance at various depths along the beam path. Quickly sweeping the beam across the retinal surface, in a structured pattern, produces the image stack.

The ophthalmology field historically identified diseases by examining *fundus images* (captured using an ophthalmoscope showing the retina, macula, and optic disc) and more recently by 2D thickness maps of retinal layers. OCT has drastically improved the type of information available to vision scientists allowing for a more intuitive view as well as analysis of retinal layer information. Recently, 3D OCT imaging has gained popularity by giving practitioners more information for their evaluations due to advancements in OCT technology. As a result, we have built software that turns what is an otherwise qualitative evaluation into a quantitative form.

An automatic approach that segments, classifies, and analyzes retinal layers from 3D OCT would be ideal. However, the morphology of retinal layers depends on the patient and the disease in question,

which has caused problems for existing automatic retinal layer extraction methods [19]. To address this problem we have developed a semi-automatic segmentation system in which the morphology of retinal structures can be discovered and refined by a clinician. The clinician interactively specifies the location of a retinal layer on a few select slices of the volume. This selection is then extrapolated throughout the entire volume using a SVM classifier in order to create a segmentation. Once segmented, we provide visualizations and measurements of the resulting segmentation to aid in disease diagnosis. The main visualization interface is an interactive 3D volume rendering of the segmented portions of the volume. We also provide more familiar visualizations such as a thickness map [1], currently a common diagnosis tool, and a 2D summed-intensity projection of the data resembling a fundus image (feature included for completeness, but we show no images of it in this paper). Additionally, the user can compute volume and thickness from layer segmentations, which have proven useful in retinal disease diagnosis.

Speckle noise is a normally distributed noise component introduced by the data acquisition process. Our SVM approach, based on the original work [24], considers a voxel’s mean value and variance across multiple resolutions in order to gracefully handle this noise and to give the SVM a global perspective over feature shapes. Additionally, this SVM is more tolerant of misclassification by the user, variation between patients and diseases, and adapts well to the data variation constituting retinal layer morphology.

Our main goals were to provide a tool that could (i) be used in a clinical setting; (ii) operate on noisy OCT data; and (iii) isolate individual or multiple retinal layers. Our main contributions to achieve these goals are (i) integration of a hierarchical data representation into SVM computations to counter noise and to better find retinal layers and (ii) several speedups for improving SVM performance allowing its practical use within a clinical setting.

1.1 Related Work

We evaluated a number of methods in order to find one that met our needs for OCT retinal data. One commonly used volumetric data segmentation method is a visualization technique that uses a one-dimensional transfer function to map scalar intensity to color and opacity [12]. This approach effectively segments regions based on scalar intensity so that areas of interest are shown opaquely while other regions are rendered transparently. Problems arise when different biological features share similar scalar values, manifesting as opaque regions that should not be, and is exacerbated when noise is present. Some improved methods address these problems [9, 10, 11, 12]. However, the user interface for these types of methods is typically too cum-

-
- Fuller and Hamann are with Institute for Data Analysis and Visualization (IDAV) and Department of Computer Science, University of California, Davis, E-mail: arfuller@ucdavis.edu and hamann@cs.ucdavis.edu.
 - Zawadzki, Choi, and Werner are with Vision Science and Advanced Retinal Imaging Laboratory, Department of Ophthalmology and Vision Science, University of California, Davis, E-mail: {rjzawadzki, sschoi, js Werner}@ucdavis.edu.
 - Wiley and Hamann are with Stratovan Corporation, E-mail: {wiley, hamann}@stratovan.com.

Manuscript received 31 March 2007; accepted 1 August 2007; posted online 27 October 2007.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

bersome for clinicians to use in everyday practice since they need intimate knowledge of the intensity-to-color mapping in order to properly understand the user interface. Our work is based on the work described in [22], which describes how artificial intelligence algorithms can be used to construct N -dimensional transfer functions through an intuitive user interface.

Machine learning algorithms such as artificial neural networks have been used in medical imaging research somewhat successfully [5, 7]. However, SVMs [3, 15, 4] have yielded more reliable results for feature detection [2, 21, 16]. The method discussed in [23] compares the use of neural networks and SVMs when constructing an N -dimensional transfer function (mapping). In our case, the ability of SVMs to handle error, both in the form of speckle noise and user misclassification, makes them attractive.

Typical characteristics used to train machine learning algorithms are scalar value, gradient magnitude, spatial location, orientation, and neighborhood information. However, these characteristics can cause a SVM to be sensitive to noise or objects that are structurally deformed, resulting in poor segmentations. We have found that careful selection of the characteristics defining the SVM input vector is typically better than adding as many as possible. Too many characteristics dilute the input vector by slowing down SVM computations and this fact often leads to unwanted segmentation results. Additionally, SVMs are typically fed local data characteristics establishing a dependence on the base resolution that ignores global feature trends that are apparent when looking at the data macroscopically. The method described in [17] uses wavelets combined with a SVM classifier to identify texture properties of image data. This concept is useful since the method identifies patterns with distinct texture characteristic on a more global scale. The method described in [20] employs a multi-resolution SVM kernel to account for macroscopic features and differs from the method discussed in this paper in that we instead compute a multi-resolution representation of the data.

Good examples of the current state of the art of 3D OCT visualization and analysis are described in [25, 13]. To our knowledge, no existing system for 3D OCT retinal visualization and analysis is as complete and accurate as that presented in this paper.

1.2 OCT Retinal Data

Volumetric OCT data are captured by directing a light beam at a patient’s retina in a gridded fashion. Back-scattered interference patterns are captured through a complex feedback system, described in [26], to produce β -scans similar to slices obtained via MRI, CT, or ultrasound. However, neighboring slices are not registered to one another as they are in MR imaging, due to naturally occurring unconscious eye movements. During the scanning process, the clinician monitors a realtime display of the β -scans in order to eliminate low-frequency movements. High-frequency vibrations are almost always present. Typically, the clinician collects 80 to 200 β -scans having dimensions ranging from 500×250 to 1000×500 pixels in size (corresponding to a region of about $8mm \times 8mm \times 300\mu m$ in size).

Figure 1 shows a single β -scan and also a cross section of several β -scans showing the (mis-)alignment along the stack axis. In a pre-processing step, slices are registered using standard registration techniques. Our clinicians use ImageJ [18], which computes rigid body transformation (translation and rotation) to minimize the difference between neighboring β -scans. Figure 2 shows volume data sets of three different retinal diseases compared to a healthy human retina. Often, the disease type is obvious from a single β -scan. However, a time-series volumetric data set can show disease progress that is not apparent from a single scan. In addition, a clinician can extract volume information of fluid, in the case of retinal detachment (fluid collects beneath the photoreceptor layer), in order to gauge severity of a disease before surgery and improvement after surgery.

For a normal healthy retina, the retinal surface and layers are well-defined but obfuscated by noise inherent in OCT data. This is exacerbated by the fact that patient’s move their eye while being scanned resulting in “spikes” in β -scans (as indicated in the left-side of the top image of Figure 1). The clinician cannot always capture blemish-

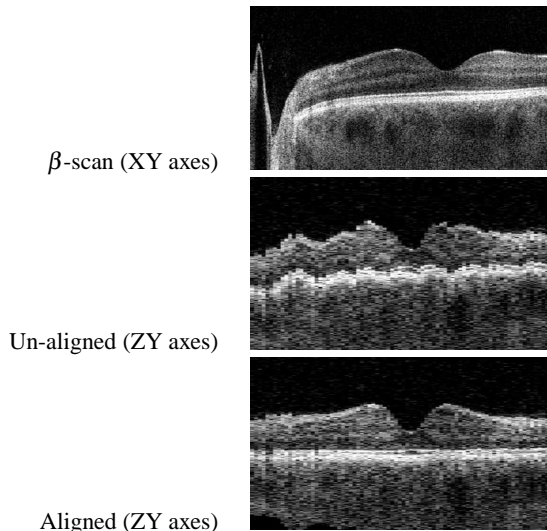


Fig. 1. Top image shows one OCT β -scan. Middle image shows 80 unaligned β -scans along the stack axis. Bottom image shows the scans after alignment.

free volumes since the patient must not move their eye for about ten seconds, which is not always possible due to natural involuntary eye movements. In addition, we needed to build a system that could segment and analyze diseased retina. In diseased cases, the retinal layers are not well-defined and can i) be missing entirely, ii) vary in thickness across the retina, iii) be very thin, iv) be very thick, v) be very bumpy, and vi) be erratic in morphology; all of which are handled effectively by a SVM due to its flexibility.

2 SVM SEGMENTATION METHOD

The two main choices when implementing a SVM are the kernel and input vector used to classify the feature space. We use a radial basis function kernel since we assume it can represent our feature space well and also that speckle noise is normally distributed across the data. Additionally, it allows non-linear separation of the space.

2.1 Feature Space

The data characteristics we include in our input vector are scalar intensity, gradient, spatial location, mean of the neighbors, and variance. The most obvious inclusion for a voxel $\mathbf{i} = \{i, j, k\}$ is scalar intensity $f_{\mathbf{i}}$ and allows for efficient segmentation of regions having relatively low standard deviation (noiseless data) and in our case, even though we have substantial noise, this characteristic still proves useful. The spatial location $\mathbf{p}_{\mathbf{i}} = \{x_{\mathbf{i}}, y_{\mathbf{i}}, z_{\mathbf{i}}\}$ of a voxel (considered with the other characteristics) allows the differentiation between features having similar data-distribution characteristics but residing in different locations.

The method described in [22] suggests that six neighbor intensity values $f_{i\pm 1, j\pm 1, k\pm 1}$ should be considered at \mathbf{i} to counter noise. They suggest that if a voxel value has been perturbed by noise, inclusion of its neighbors will help determine the “actual value.” We found, in our case, that this can lead to disconnected components due to those regions having the appropriate neighbors, but not necessarily the correct local data distribution. Our method instead uses as a parameter the mean of the six neighbors, leading to improved results. We also include the variance (instead of the standard deviation) to include a data distribution characteristic and additionally include gradient magnitude to identify tissue boundaries.

In summary, for a voxel \mathbf{i} at world-space location $\mathbf{p}_{\mathbf{i}}$ with scalar value $f_{\mathbf{i}}$ having neighbors $N = \{i \pm 1, j \pm 1, k \pm 1\}$ the data characteristics we use for our input vector are:

$$f_{\mathbf{i}}, \quad (1)$$

$$x_{\mathbf{i}}, \quad (2)$$

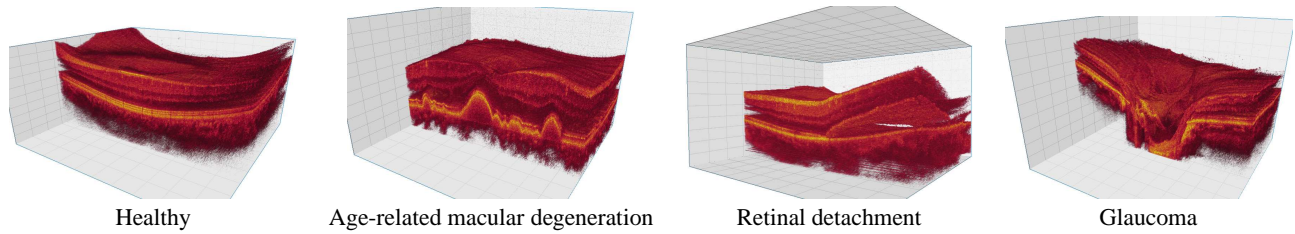


Fig. 2. Comparison of different retinal diseases. The front portion of each volume has been clipped in order to reveal internal structures. *Drusen*, extracellular deposits on the photoreceptor layer, are indicative of age-related macular degeneration. The retinal detachment data were captured after surgery. All images are of the foveal region (the dip in the middle) except for the glaucoma case, which is of the optic nerve head.

$$y_{\mathbf{i}}, \quad (3)$$

$$z_{\mathbf{i}}, \quad (4)$$

$$\bar{f}_{\mathbf{i}} = \frac{1}{|\mathcal{N}|} \sum_{\mathbf{n} \in \mathcal{N}} f_{\mathbf{n}}, \quad (5)$$

$$\sigma_{\mathbf{i}}^2 = \frac{1}{|\mathcal{N}|} \sum_{\mathbf{n} \in \mathcal{N}} [f_{\mathbf{n}} - \bar{f}_{\mathbf{i}}]^2, \quad (6)$$

$$\nabla \hat{f}_{\mathbf{i}} = \frac{1}{|\mathcal{N}|+1} \left\{ \nabla f_{\mathbf{i}} + \sum_{\mathbf{n} \in \mathcal{N}} \nabla f_{\mathbf{n}} \right\}. \quad (7)$$

Thus, we include for each voxel \mathbf{i} the scalar value $f_{\mathbf{i}}$, location $\mathbf{p}_{\mathbf{i}}$, mean value $\bar{f}_{\mathbf{i}}$ of the neighbors around \mathbf{i} , variance $\sigma_{\mathbf{i}}^2$ around \mathbf{i} , and the gradient $\nabla \hat{f}_{\mathbf{i}}$ at \mathbf{i} using a local difference operator.

2.2 Multi-resolution Hierarchical SVM

Even in the presence of noise, a person can seemingly automatically classify features by the distribution of data intensities in both local neighborhoods and globally. When confronted with subtle changes in these distributions, a person is able to perceive material (tissue) boundaries that are not necessarily well-defined by scalar value, but are instead identified by the transition of one noisy distribution to another. To expand the operation of our SVM beyond the local scope of data characteristics, we construct a hierarchy of representations of the volume and sample the mean value, gradient magnitude, and variance at each level of this hierarchy in order to capture the distribution across all levels and accumulate a weighted average of these samples prior to inserting them into the SVM.

Our algorithm constructs a mipmap-like hierarchy having the full resolution data as level l_0 and successive levels $l_i, i > 0$, having half the resolution (in each dimension) of the previous level l_{i-1} , see Figure 3. The user specifies “feature” and “background” voxels on l_0 and then specified voxels are mapped to coarser levels $l_i, i > 0$ in order to determine the intensity value, mean value, gradient, and variance at that level. Since each lower-resolution level voxel covers a larger region, we weigh each successive level’s result to reasonably diminish its influence on the final value. Since each successive level in our hierarchy is an eighth the size as the previous level, each level l receives a weight $w_l = 1/8^l$. We allow the user to manipulate this value to increase or further diminish the influence of lower resolution levels. Additionally, we normalize the weights so that $\sum_{\{l\}} w_l = 1$ to maintain the integrity of the values. Figure 4 shows the effect of different hierarchy-level weights on SVM segmentations. To include the multi-resolution hierarchy into our SVM computations, we applied weighted averaging each to scalar intensity, gradient, mean of the neighbors, and variance characteristics. Thus, when obtaining characteristics for each voxel (either to train or classify), we access the hierarchy at all levels to produce a weighted average for each of those characteristics. We did not include spatial location in this process since this characteristic was not affected by multiple levels of detail. Additionally, we found that the maximum number of levels was less than ten in most cases since one of the primary axes would vanish by this point.

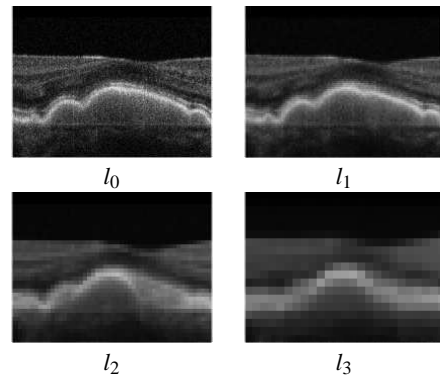


Fig. 3. We construct a mipmap-like hierarchy in order to compute varying levels of data distributions to sample as input to our SVM algorithm. Top-left image shows a slice through a level having a resolution of $475 \times 150 \times 48$. Remaining images show levels having a resolution half the one before it. (The change in the “bump” is due to the influence of neighboring slices in front and behind the shown slice.)

2.3 Feature Specification

A user provides SVM training data through an intuitive interface in order to create the “segmentation function.” Our system allows a user to quickly classify features by using a small number of specification (training) points. We perform this specification on axis-aligned 2D slices of the volume similar to [22]. The user can slice a plane through the volume to see a 2D intersection image and can “paint” on that plane to mark points as “feature” or “background” as indicated by the green and red marks, respectively, in Figure 4. This approach can be learned quickly and has proven to be user-friendly within a clinical setting. The user is required only to draw through regions of interest and to indicate regions not of interest. Unfortunately, as the training data set grows through painting, so does the complexity of the SVM. This additional complexity leads to drastically longer segmentation times.

2.4 Speed Improvements

SVMs have a large computational cost that hinders their application to an entire volume with real-time response behavior. We have implemented several techniques that reduce computational cost for both SVM training and classification while having minimal effect on the resulting segmentation quality. These methods are intended to minimize clinician time needed to iterate between specifying training data and viewing results. The methods include (i) training-data reduction; (ii) evaluation on a single slice; (iii) checkerboard SVM sampling; (iv) SVM multi-threading; and (v) clipping planes; each is discussed in detail in the following sections.

Training-data reduction We found that users typically specified many training data points resulting in duplication of the data characteristics fed into the SVM since our painting interface lays down a “block” of specification points whenever the user marks a region. Thus, using every voxel marked by this block introduces some redundancies resulting in substantial SVM computation penalties while

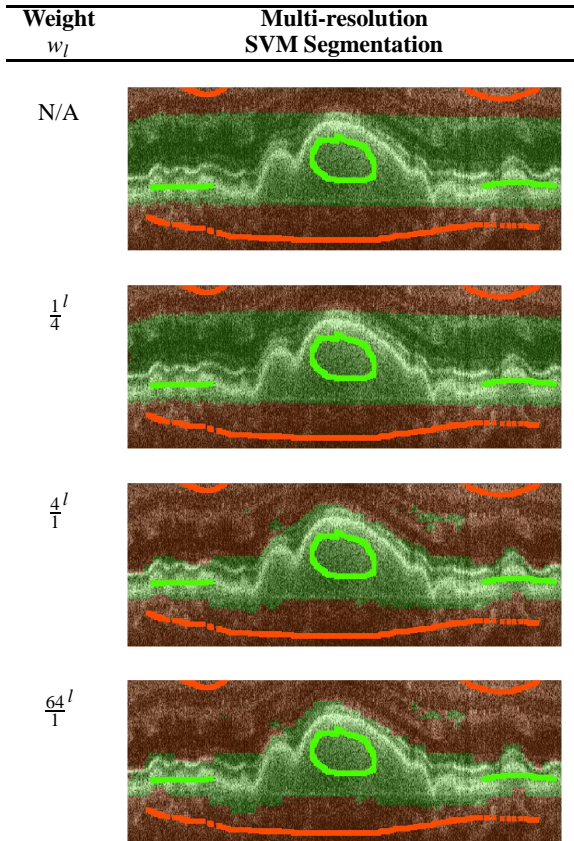


Fig. 4. Influence of multi-resolution hierarchy on SVM segmentation. Top-image uses standard local data characteristics while subsequent segmentations accumulate data characteristics across the hierarchy using the weight shown. Each segmentation used the same training data points as indicated by the green (feature) and red (background) marks. As indicated in the bottom image, the presence of an overlying grid becomes noticeable when lower-resolution levels are weighted more. In this case, the entire segmentation region should be connected, but in general, this is not the case, for example, when segmenting fluid pockets caused by retinal detachment.

not necessarily producing better results. A representative subset $\mathbf{R} = \{\mathbf{r}_0, \mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_R\}$ from the entire training set $\mathbf{T} = \{\mathbf{t}_0, \mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_T\}$ is sufficient to achieve nearly the same results while requiring much less computation time. To obtain the subset $\mathbf{R} \subset \mathbf{T}$, we implemented a discrete form of *best-candidate sampling* where a random point $\mathbf{r}_0 = \mathbf{t}_{rand(|\mathbf{T}|)}$ is first added to \mathbf{R} . Then, we repeatedly add to \mathbf{R} the point from \mathbf{T} that is farthest (in Cartesian distance) from all points currently in \mathbf{R} . This process continues until (i) the number of points $|\mathbf{R}|$ reaches a user specified maximum threshold or (ii) the next point to be added is closer to another point in \mathbf{R} than a user-specified minimum distance. Both of these thresholds mitigate the growth of training data by reducing the set of points used in a fashion that covers all training regions well. We found in practice that having no maximum threshold while setting a minimum distance of two voxels between training data points in \mathbf{R} has little to no effect on the resulting segmentation while speeding up the SVM computation by nearly 60%.

Evaluation on a single slice Our software allows a user to test the SVM segmentation on individual 2D slices to evaluate the resulting segmentation for that slice. If needed, the user can modify the painted regions or browse additional slices and subsequently apply the SVM or further mark those slices. The application of the SVM to a single slice requires only a few seconds (even for larger volumes of $1000 \times 500 \times 200$ voxels). Once satisfied with the segmentation on individual slices, the user can apply the SVM to the whole volume.

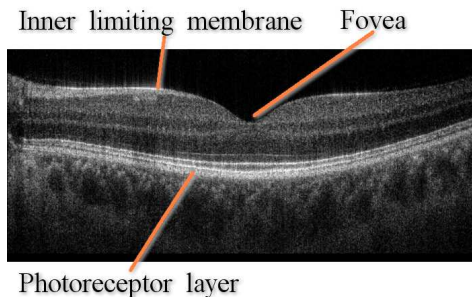


Fig. 5. Regions isolated by clinicians are the photoreceptor layer (PRL) and the *full retinal thickness* indicated by the boundary at the bottom of the PRL and ending at the top of the inner limiting membrane.

Checkerboard SVM classification After the SVM has been trained, we apply the assumption that retinal features are much larger than an individual voxel by using a checkerboard scheme that first classifies every other voxel by the SVM as either a feature or background. Each unclassified voxel is then determined by finding the majority classification of its six neighbors. If the classification is tied, we then apply the SVM to that voxel to decide. We found the checkerboard scheme consistently reduces SVM classification time by roughly 45% and consequently leads to smoother object boundaries and less “orphaned” voxels (sparse individual misclassified voxels).

SVM multi-threading Once trained, an SVM inherently classifies each voxel independently of others. We take advantage of this by incorporating multi-threaded SVM classification in order to take advantage of popular multi-processor computers (the primary computer used to run our software has four processor cores). We have found this to do the expected by speeding up the SVM classification step according to the number of physical processors.

Clipping planes Clipping planes can be specified to restrict the SVM application to a sub-volume of the data. Our software allows a user to specify axis-aligned clipping planes in order to specify a sub-region (bounded by the clipping planes). This approach reduces the (i) number of voxels to be processed, (ii) specification time by the user, and (iii) training data size while subsequently reducing the complexity of the SVM resulting in both faster training and classification. For retinal data, clipping planes are useful for isolating broad regions around relatively flat retinal layers of interest. Thus, while a simple concept, this method is quite useful for our application.

3 LAYER THICKNESS ANALYSIS

Once a retinal layer has been isolated, measurements are computed in order to track patient progress over time or to diagnose diseases such as glaucoma, which can be identified by retinal-layer volumes outside normal parameters. Additionally, the fluid volume beneath the photoreceptor layer caused by a retinal detachment can be measured before and after surgery to insure that the surgery was a success and additionally to monitor disease advancement.

Typically, a clinician isolates the *full retinal thickness* as the region between the bottom of the photoreceptor layer and the top surface of the retina in order to examine thickness maps, see Figure 5. Thickness maps provide insight into gradually changing thickness and other abnormalities. The clinician also isolates the photoreceptor layer itself in order to reveal thickness abnormalities. More details on thickness map analysis of retinal layers can be found in [14, 6, 1].

4 RESULTS

Our software is currently being used by vision scientists at the University of California, Davis Medical Center. We present work by two clinicians to examine our software’s practicality in a clinical setting as well as the quality of segmentation results. We have applied our

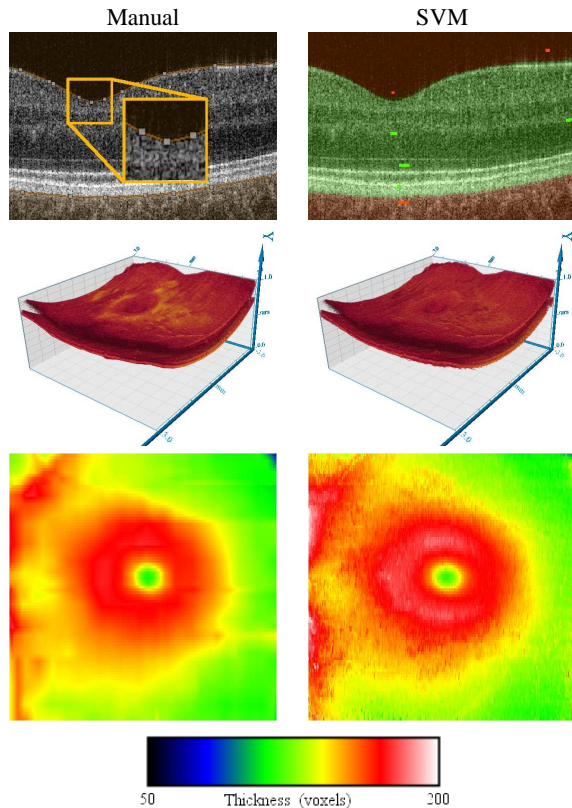


Fig. 6. Extracting the “healthy” thickness map from *full retinal thickness* segmentations. Left-column indicates manual segmentation while right-column is the SVM. Top-images show the segmentation specification on a slice. Middle-images show the segmentation applied to the whole volume. Bottom-images show the extracted thickness maps. For thickness, one voxel unit is approximately equal to $1\mu\text{m}$.

methods to data from patients having age-related macular degeneration and retinal detachment and compare these to a healthy retina, see Figure 2. We examined how well our method segments healthy retinal layers, diseased retinal layers, and pockets of fluid by comparing our semi-automatically generated segmentations to manual generated segmentations.

We included a manual segmentation tool within our software allowing a clinician to isolate retinal layers by defining top- and bottom-bounding polylines on key frames throughout the volume. The specified polylines are linearly interpolated between key frames in order to segment every slice. We use this segmentation as a “gold standard” to gauge the quality of our SVM-based segmentations. We asked one clinician to define three gold standards for our test data and found that this process required 30 minutes on the healthy data and up to 50 minutes on the diseased data to specify. Figure 6 shows a manually- and SVM-generated segmentation as well as the process from (i) slice segmentation to (ii) volume segmentation to (iii) full retinal thickness map of the healthy data.

Data acquisition In practice, the clinician spends approximately 30 to 45 minutes with a patient in order to capture the 3D OCT data. The patient must fill out medical forms, is educated about equipment being used, their pupils may be dilated, and is then positioned into a chin rest for the scanning. The clinician then goes through a systematic process of instructing where to fixate enabling the capture of several scans of various regions on their retina. The clinician evaluates realtime 2D scan images in order to guarantee data quality. Once scanning is finished, the clinician proceeds with data processing and visualization.

There is a series of steps involved in order to prepare the data for our

Improvement	# Training Points	SVM Training Time (sec)	SVM Class. Time (min)
(laptop) 2GHz Intel Core 2 Duo + 2GB main memory			
None	2902	8.92	29.74
CH only	2902	8.73	14.74
CH+MR	2902	8.75	15.94
CH+RED	736	1.02	6.36
CH+MR+RED	732	1.00	6.59
(lab machine) Dual 3GHz Intel Xeon + 3GB main memory			
None	2902	5.25	7.54
CH only	2902	5.05	3.76
CH+MR	2902	5.13	4.00
CH+RED	736	0.53	1.60
CH+MR+RED	732	0.52	1.64

Table 1. Improvement influence on *full retinal thickness* segmentation performance as applied to the healthy data. Improvements tested were checkerboard (CH), multi-resolution hierarchy (MR), and best candidate reduction (RED). (Multi-threading was used in all tests.) All tests began with the same training data. Tests were performed on a laptop having a 2GHz Intel Core 2 Duo processor (two cores total) and 2GB of main memory and also on a desktop computer used by the clinicians having dual 3GHz Intel Xeon processors (four cores total) with 3GB of main memory. The multi-resolution hierarchy used the weights $w_l = 1/8^l$. The best candidate training data reduction included a minimum distance of two voxels between training data.

software. The majority of time is spent exporting 2D slices from the acquisition software and subsequently registering them to one another. It takes approximately 15 minutes for a clinician to process all of the collected patient data in this fashion.

Protocol We asked the two participating clinicians (identified as C1 and C2) to isolate the photoreceptor layer (PRL) and the full retinal thickness (encompassing the space from the retinal surface up to and including the PRL) using the painting interface on all three data sets. At first, we allowed them to spend as much time as they thought necessary to obtain adequate results and found that they spent around 20 to 30 minutes per feature segmentation. Then, we asked them to repeat each feature segmentation in under ten minutes. Their results are explained in detail over the following sections.

4.1 SVM Performance

We measured the running time for the checkerboard, multi-resolution hierarchy, and best-candidate training-data reduction improvements. We performed the tests on the healthy data using the “unlimited time” SVM training data from clinician C1. We used two computers having different amounts of main memory and processor capabilities, see Table 1. Overall, we found that the checkerboard speedup slightly improved segmentation quality while significantly speeding up the entire process. Thus, we left this option on for these tests.

Overall, we determined that the multi-resolution hierarchy had a negligible effect on SVM training while having a SVM classification penalty of only 3% to 8% in all cases. We had expected the multi-resolution hierarchy to impact the SVM classification running time significantly more, but it seems that the SVM computations are far more complex than the additional data accesses needed to implement the hierarchy. The checkerboard improved performance by about 50% in all cases, which is expected since it applies the SVM classification to about half of the data. The best-candidate training data reduction (having a minimum distance of two voxels between training points) improved SVM training performance by about 90% and further reduced SVM classification running time by about 60% while not significantly effecting the resulting segmentations. Quality measurements are detailed in the next section.

Table 2 shows the running time results for all SVM segmentations (as performed on the lab computer). The full thickness segmentations required on average less than two minutes. The SVM classification

time did not seem to directly correlate with the training data size. Additionally, we found that clinicians tended to paint aggressively up front when under a time constraint in order to avoid more iterations later and, when under no time limit, the clinicians tended to paint sparsely and iterate more often. This led to the ten-minute training data being comparable or even larger than the unlimited-time training data.

Time Constraint	# Training Points	# Reduced Points	Total SVM Time (min)
Healthy - Full Thickness			
No Limit - C1	3348	838	1.90
No Limit - C2	5725	1264	1.53
10 mins - C1	4469	1046	1.49
10 mins - C2	16113	3631	3.48
Healthy - PRL			
No Limit - C1	8636	2049	2.81
No Limit - C2	20569	4752	6.22
10 mins - C1	13232	3029	3.18
10 mins - C2	19135	4212	4.42
Macular Degeneration - Full Thickness			
No Limit - C1	2831	676	0.78
No Limit - C2	3772	952	0.62
10 mins - C1	2236	556	0.36
10 mins - C2	3936	939	0.55
Macular Degeneration - PRL			
No Limit - C1	7429	1753	1.43
No Limit - C2	6913	1635	1.30
10 mins - C1	7925	2013	1.53
10 mins - C2	14277	3213	1.40
Retinal Detachment - Full Thickness			
No Limit - C1	4832	1138	1.80
No Limit - C2	4381	1027	1.66
10 mins - C1	4325	997	1.26
10 mins - C2	4574	1008	1.46
Retinal Detachment - Fluid			
No Limit - C1	5058	1267	1.22
No Limit - C2	4623	1147	0.93
10 mins - C1	9598	2171	2.34
10 mins - C2	2985	742	0.89

Table 2. Measured running times for *full retinal thickness* segmentations using multi-threading, clipping planes, checkerboard, multi-resolution hierarchy, and best-candidate training-data reduction. C1 and C2 indicate each clinician. Tests were performed on our “lab computer,” a dual processor 3GHz Intel Xeon with 3GB of main memory (four processor cores total). Multi-resolution hierarchy weights were set to $w_i = 1/8^i$ and the best-candidate training-data reduction had a minimum distance of two voxels between training points. Measured running time without our improvements ranged between from 30 min to 2 hours in some cases.

4.2 SVM Segmentation Quality

We derived a thickness map from each segmentation by considering the distance (thickness) between the top and bottom surfaces of the segmentation. To estimate the quality of our SVM method, we compared thickness maps extracted from the gold-standard manually generated segmentation to that obtained by our clinician’s SVM segmentation. We computed the difference, at each point, between the thickness maps as our error metric. We computed the mean and standard deviation of these differences and additionally found a useful metric to be the value that 68% of the differences fell below, see Table 3. Furthermore, we investigated the effect our checkerboard, multi-resolution hierarchy, and best-candidate training data reduction schemes had on segmentation quality, see Table 4. We found in each of these cases that the resulting SVM segmentation better matched the gold standard.

Prior to our addition of the multi-resolution hierarchy, our clini-

Time Constraint	68% are less than	Mean	STD
Healthy - Full Thickness			
No Limit - C1	3.233	3.875	10.288
No Limit - C2	7.074	6.115	5.329
10 mins - C1	4.318	4.293	7.285
10 mins - C2	5.188	4.563	4.496
Healthy - PRL			
No Limit - C1	3.829	5.048	8.119
No Limit - C2	3.746	4.983	8.063
10 mins - C1	3.813	4.868	7.009
10 mins - C2	6.811	8.712	11.434
Macular Degeneration - Full Thickness			
No Limit - C1	4.322	3.331	2.617
No Limit - C2	3.13	2.604	2.076
10 mins - C1	3.063	2.748	4.227
10 mins - C2	2.443	2.036	1.684
Macular Degeneration - PRL			
No Limit - C1	6.049	5.061	4.428
No Limit - C2	5.502	4.614	3.739
10 mins - C1	4.93	4.431	4.575
10 mins - C2	5.695	4.751	4.114
Retinal Detachment - Full Thickness			
No Limit - C1	3.986	3.451	3.427
No Limit - C2	4.512	3.693	3.018
10 mins - C1	5.111	4.263	4.038
10 mins - C2	5.594	4.842	5.051
Retinal Detachment - Fluid			
No Limit - C1	7.34	4.895	6.438
No Limit - C2	7.21	4.865	6.16
10 mins - C1	7.78	4.901	5.694
10 mins - C2	9.63	5.9	6.675

Table 3. Comparison of SVM segmentations to gold-standard manual segmentations. C1 and C2 indicate each clinician. The values in the table indicate the thickness difference in voxel units between the SVM and manual segmentations. The 68% metric states that 68% of the thickness differences are less than the value specified in the table. Layer thicknesses typically range from 30 to 100 voxel units depending upon the disease.

icians had difficulty isolating the PRL in many patient cases due to the PRL being thin (< 30 voxels) and the local data characteristics used in standard SVM computations were not able to capture the “thin” morphology of the PRL. With the addition of the multi-resolution hierarchy, we are now able to isolate the PRL accurately. This is due to the ability of the hierarchy to examine lower-resolution levels (low-frequency information) in order to find feature characteristics that define the thin PRL. Figure 7 compares the manual segmentation with the multi-resolution SVM segmentation of the PRL for the healthy data. The multi-resolution SVM segmentation captures the foveal region well while additionally revealing retinal blood vessels.

Our SVM method performed well on both healthy and disease data. The ability of the SVM to isolate arbitrary features, such as the pocket of fluid beneath the retina in the case of retinal detachment, furthers the application of our method. Figure 8 shows the results of our fluid segmentations for the retinal detachment data. We noticed that all of the SVM segmentations underestimated the volume slightly. This is apparent in the histogram of thickness-map differences also shown in Figure 8. Additionally, we were especially pleased with the results in disease cases having retinal layer deformation. This was primarily evident with the drusen on the PRL of the age-related macular degeneration data. We obtained reasonable reproducibility among clinicians as well as tolerable error, see Figure 9.

Improvement	68% are less than	Average	STD
None	5.598	5.335	6.043
CH only	5.451	5.143	5.689
CH+MR	5.433	5.021	5.339
CH+RED	4.386	3.898	3.793
CH+MR+RED	4.1	3.806	4.393

Table 4. Improvement influence on *full retinal thickness* segmentation quality as applied to the healthy data set. The values in the table indicate the thickness difference in voxel units between the SVM segmentation computed using the improvement and the manual segmentation. Improvements tested were checkerboard (CH), multi-resolution hierarchy (MR), and best candidate reduction (RED). All tests began with the same training data. The multi-resolution hierarchy used the weights $w_l = 1/8^l$. The best-candidate training data reduction included a minimum distance of two voxels between training points. Improvements tended to increase segmentation quality.

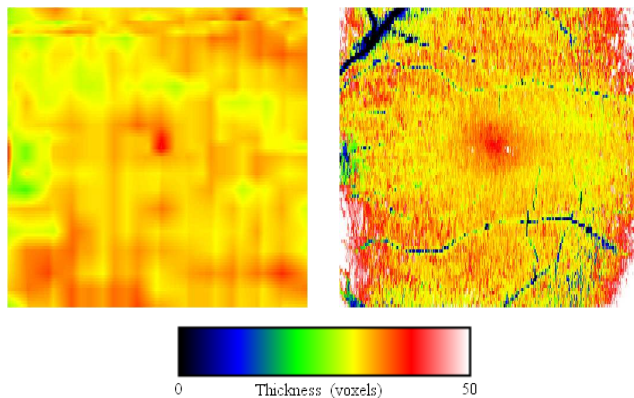


Fig. 7. Comparison between manual and multi-resolution SVM segmentation of the PRL of the healthy data. Left-image shows gold standard manual segmentation. Right-image shows the multi-resolution SVM segmentation. Note how the SVM segmentation reveals blood vessels and better indicates the location of the fovea (red area in the middle). The difference in how the manual and SVM segmentations handle blood vessels manifested as voxel difference error in our thickness maps.

5 CONCLUSION AND FUTURE RESEARCH

We used a combination of volume visualization and data analysis techniques to better diagnose and subsequently treat retinal diseases. We have found that applying volume visualization techniques to 3D retina image data collected in a clinical setting has achieved success by revealing subtle features that standard diagnosis procedures miss as well as providing accurate quantitative measurements of retinal structures.

This tool is currently being used in a clinical environment and is continually providing insight into challenging retinal visualization and analysis problems. We found that handling noise is a difficult task when training and using any type of machine learning algorithm for segmentation. We plan to investigate image filtering techniques to reduce speckle noise prior to SVM processing. We also plan to create training data that can be applied to multiple volumes eliminating the need to retrain the SVM for new patients. Our method shows reproducibility among different clinicians and yields good accuracy. The primary differentiating factor between clinicians is their interpretation as to where a feature begins and ends, which is better controlled through a clinical protocol. Our method currently produces desirable results in about ten minutes. This is a significant improvement over past machine learning applications to volumetric segmentations.

The most prominent drawback from our method is that the SVM mis-classifies some voxels resulting in scattered noise. This is noticeable in the thickness maps shown in Figure 9. However, the clinicians are aware of this issue and are willing to cope with it (by weighting

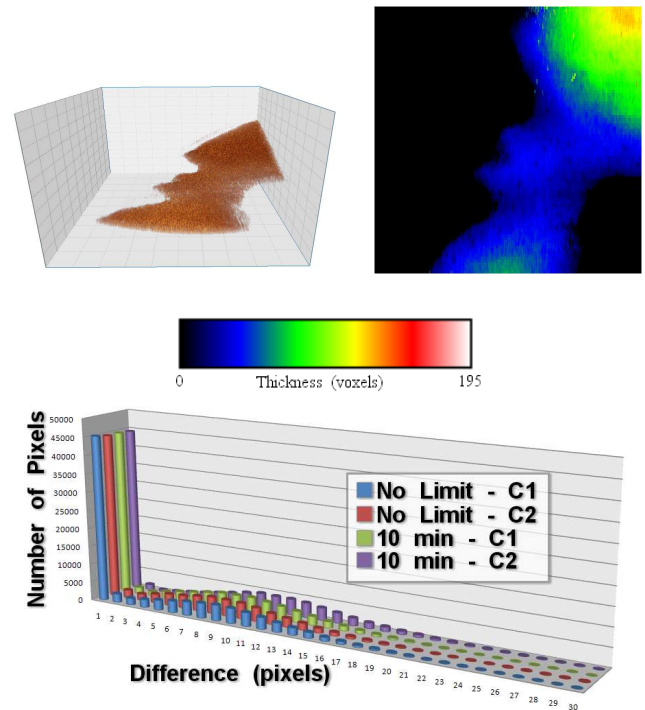


Fig. 8. Isolation of fluid beneath retina from retinal detachment data. Top-left image shows SVM segmentation from clinician C2 having a ten minute time constraint. Top-right image shows associated thickness map. Bottom-image shows the histogram of thickness map differences between the manual and SVM segmentations for all fluid SVM segmentations. SVM segmentation consistently slightly underestimated the fluid volume as compared to the manually specified segmentation. We noticed that this was because the manually specified polylines were placed on the layer boundary as opposed to just inside of it, which is how the SVM identifies the region.

the hierarchical levels differently) due to the enormous time savings involved with using the SVM over other methods.

As a result of this work, vision scientists are using this software to provide quantitative information useful in treatment planning that is not otherwise available. It is certain to contribute to research and eventually may facilitate clinical diagnosis and monitoring.

ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation under contracts ACI9624034 (CAREER Award), NEI grant 014743, through the Large Scientific and Software Data Set Visualization (LSSDSV) program under contract ACI 9982251, and a large Information Technology Research (ITR) grant; and the National Institutes of Health under contract P20 MH60975-06A2, funded by the National Institute of Mental Health and the National Science Foundation.

Alfred Fuller has been supported by a Student Employee Graduate Research Fellowship (SEGRF) from Lawrence Livermore National Laboratory.

We thank the members of the Visualization and Computer Graphics Research Group at the Institute for Data Analysis and Visualization (IDAV) at the University of California, Davis.

REFERENCES

- [1] S. Asrani, R. Zeimer, M. Goldberg, and S. Zou. Application of rapid scanning retinal thickness analysis in retinal diseases. *Ophthalmology*, 104(7):1145–1151, 1997.
- [2] V. Blanz, B. Schlkopf, H. Blthoff, C. Burges, V. Vapnik, and T. Vetter. Comparison of view-based object recognition algorithms using realistic 3d models. In *Proc. Int'l Conf. Artificial Neural Networks*, pages 251–256, 1996.

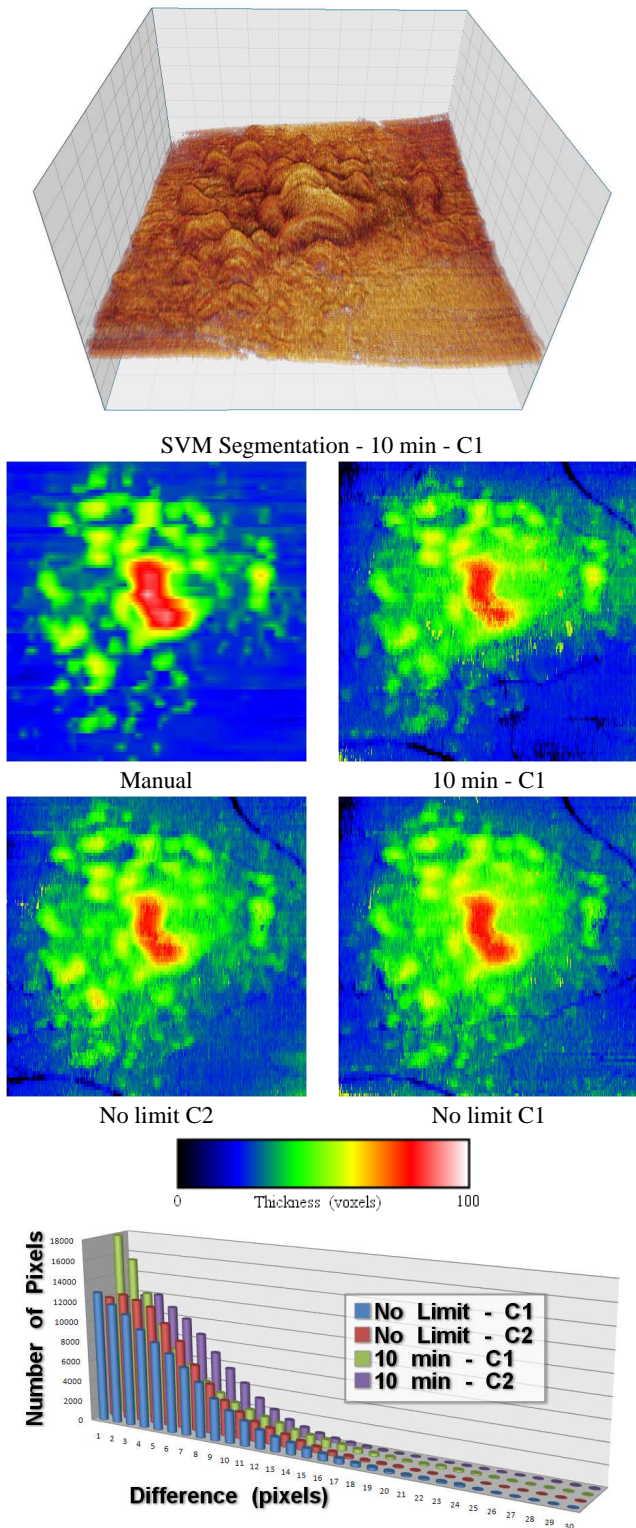


Fig. 9. Isolation of PRL and drusen (bumps on the PRL) for the age-related macular degeneration data. Top-image shows the 10 min SVM segmentation from clinician C1. Middle-images show the thickness maps obtained for these data. Bottom-image shows the histogram of thickness map differences between the SVM and manual segmentations.

[3] B. E. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. Fifth Ann. Workshop Computational Learning Theory*, pages 144–152, 1992.

[4] C. Cortes and V. Vapnik. Support vector network. *Machine Learning*, 20(3):273–297, 1995.

[5] E. Gelenbe, Y. Feng, K. Ranga, and R. Krishnan. Neural networks for volumetric mr imaging of the brain. In *Proc. Int'l Workshop Neural Networks for Identification, Control, Robotics, and Signal/Image Processing*, pages 194–202, 1996.

[6] V. Guedes and et al. Optical coherence tomography measurement of macular and nerve fiber layer thickness in normal and glaucomatous human eyes. *Ophthalmology*, 110(1):177–189, 2003.

[7] L. O. Hall, A. M. Bensaid, L. P. Clarke, R. P. Velthuizen, M. S. Silbiger, and J. C. Bezdek. A comparison of neural network and fuzzy clustering techniques in segmenting magnetic resonance images of the brain. *IEEE Trans. Neural Networks*, 3(5):672–682, 1992.

[8] D. Huang, E. Swanson, C. Lin, J. Schuman, W. Stinson, W. Chang, M. Flotte, K. Gregory, and C. Puliafito. Optical coherence tomography. *Science*, 254:1178–1181, 1991.

[9] R. Huang and K. L. Ma. Rgvis: Region growing based techniques for volume visualization. In *Proc. Pacific Graphics '03 Conf.*, pages 355–363, 2003.

[10] G. Kindlmann and J. W. Durkin. Semi-automatic generation of transfer functions for direct volume rendering. In *Proc. '98 IEEE Symp. Volume Visualization*, pages 79–86, 1998.

[11] J. Kniss, G. Kindlmann, and C. Hansen. Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *Proc. IEEE Visualization '01 Conf.*, pages 255–262, 2001.

[12] M. Levoy. Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(3):29–37, 1988.

[13] S. Makita, Y. Hong, M. Yamanari, T. Yatagai, and Y. Yasuno. Optical coherence angiography. *Opt. Express*, 14(17):7821–7840, 2006.

[14] 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. *Optics Express*, 13(23):9480–9491, 2005.

[15] K. R. Muller, S. Mika, and G. Ratsch. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, 2001.

[16] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: An application to face detection. In *Proc. '97 Conf. Computer Vision and Pattern Recognition*, pages 130–137, 1997.

[17] K. M. Rajpoot and N. M. Rajpoot. Wavelets and support vector machines for texture classification. In *Multitopic Conference, 2004. Proceedings of INMIC 2004. 8th International*, pages 328–333, 2004.

[18] W. S. Rasband. Imagej. U. S. National Institutes of Health, Bethesda, Maryland, <http://rsb.info.nih.gov/ij/>, 1997-2006.

[19] S. R. Sadda, Z. Wu, A. C. Walsh, L. Richine, J. Dougall, R. Cortez, and L. D. LaBree. Errors in retinal thickness measurements obtained by optical coherence tomography. *Ophthalmology*, 113(2):285–293, 2006.

[20] X. Shao and V. Cherkassky. Multi-resolution support vector machine. *Neural Networks, 1999. IJCNN apos;99. International Joint Conference on*, 2:1065–1070, 1999.

[21] J. Thorsten. Text categorization with support vector machines: Learning with many relevant features. *Proc. 10th European Conf. Machine Learning*, pages 137–142, 1998.

[22] F. Y. Tzeng, E. B. Lum, and K. L. Ma. A novel interface for higher-dimensional classification of volume data. In *Proceedings of the IEEE Visualization 2003 Conference*, pages 505–512, 2003.

[23] F. Y. Tzeng, E. B. Lum, and K. L. Ma. An intelligent system approach to higher-dimensional classification of volume data. *IEEE Transactions on Visualization and Computer Graphics*, 11(3):273–284, 2005.

[24] V. Vapnik. *Estimation of Dependences Based on Empirical Data*. English translation: Springer Verlag, New York, 1982.

[25] M. Wojtkowski, V. Srinivasan, J. G. Fujimoto, T. Ko, J. S. Schuman, A. Kowalczyk, and J. Duker. Three-dimensional retinal imaging with high-speed ultrahigh-resolution optical coherence tomography. *Ophthalmology*, 112(10):1734–1746, 2005.

[26] R. J. Zawadzki, A. R. Fuller, M. Zhao, D. F. Wiley, S. Choi, B. A. Bower, B. Hamann, J. A. Izatt, and J. S. Werner. 3d oct imaging in clinical settings: Toward quantitative measurements of retinal structures. *Photonics West – Electronic Imaging 2006*, 6138:8–18, 2006.