Segmentation of Crystalline Lens in Photorefraction Video

Mark Wendt and Shishir K. Shah[*]
Quantitative Imaging Laboratory
University of Houston
Dept. of Computer Science
Houston, TX, U.S.A.
shah@cs.uh.edu

ABSTRACT

With an increase in the percentage of people over the age of 65 years or older, there is a growing interest in finding ways to slow or reverse some of the effects of aging on the human body. One of these effects is the loss in the ability of the human eye to visually focus on near objects. Understanding the properties of the natural lens and the associated mechanisms of accommodation will greatly increase our knowledge to identify alternate solutions to reverse this phenomenon. Towards that end, a technique called photorefraction is currently being used in laboratory studies involving monkeys. To quantitate the lens properties, there is a need to accurately measure monkey lenses in images produced by this technique. In this paper, we present two probabilistic methods for segmenting the lens from photorefraction video sequences. Results of the developed methods are compared and evaluated against ground-truth segmentations. In addition, the results obtained are also compared to those obtained by a level set segmentation method.

Keywords

Segmentation, Photorefraction, Monkey Lens

1. INTRODUCTION

In 2003, 12% of the U.S. population was 65 years of age or older. That percentage is projected to increase to 20% by 2030 [6]. As the population ages, interest increases to find ways to slow or reverse some of the effects of aging on the human body. One of these effects is the loss of the ability to visually focus on a near object. Changing focus from far to near is called 'accommodation'. 'Presbyopia' is the age-related loss of the ability to accommodate. This process is complete by the time people reach their early fifties.

Research regarding treatments for presbyopia will benefit from an increased understanding of the mechanisms involved. Toward that end, a technique called photorefraction is currently being used in laboratory studies using monkeys. Monkeys have provided a useful model in the study of accommodation because of their ocular similarity to humans. Video sequences of the monkey eye are produced in photorefraction and calculations are made from frames of the video.

To acquire these sequences, each experiment is recorded to videotape at 30 Hz on a black and white CCD video camera. An opaque occluding device covers the lower half of the lens on the video camera (Figure 1A). An array of infrared light emitting diodes (IR-LED) is located on this occluding device. The system of video camera, occluding device, and LED array is placed in front the eye at a distance, d. Since the lower half of the camera lens is occluded, only the upper part of the light returning from the lens enters the camera. The originating location of that light on the lens is determined by the optics of the eye. If the eye is focused at distances greater than d, then the light entering the camera will originate from the upper portion of the lens (Figure 1B). If the eye is focused at a distance less than d, then the light entering the camera will be inverted such that it will come from the lower portion of the lens (Figure 1C). In images of the eye, the upper portion of the lens is brighter in the unaccommodated lens (Figure 2A) than in the accommodated lens (Figure 2B). This change in vertical brightness can be measured as a change in slope of the intensity profile (Figure 2C&D). Analysis of each video frame in a photore-

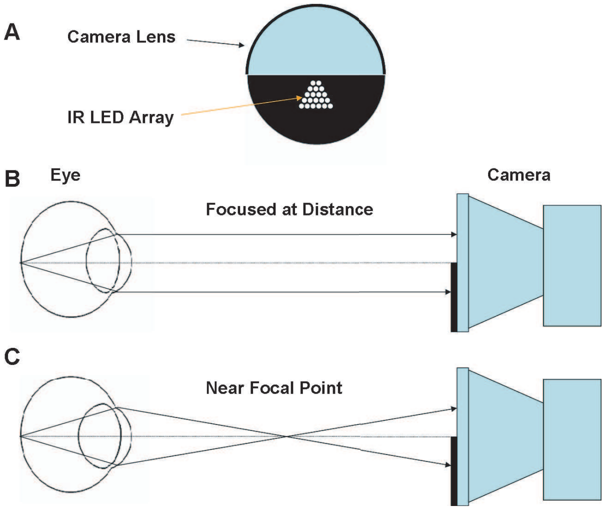

Figure 1: Diagram of a photorefraction experiment.

[*]Corresponding author

ICVGIP '10, December 12-15, 2010, Chennai, India
Copyright 2010 ACM 978-1-4503-0060-5/10/12 ...$10.00.

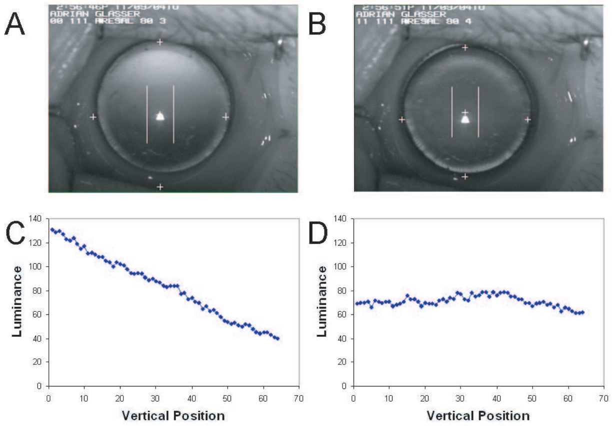

Figure 2: Images of a monkey lens in (A) the unaccommodated and (B) accommodated states produce pixel intensity profiles with (C) a negative slope and (D) a flatter slope, respectively.

fraction session returns a single slope value that is calculated from the intensity profile (Figure 3). This slope value has a strong inverse relationship with the actual refractive ability of the eye as measured in inverse meters (m^{-1}), also known as diopters (D).

In typical photorefraction studies with monkeys, the iris is surgically removed to provide a better view of the structures in the eye. Fully exposed lenses in video images of iridectomized monkey eyes provide a number of challenges to segmentation; occlusions from tears, eyelashes, and other structures are common; the defining features of the lens boundary vary from image to image and at different positions within the same image; and the lens illuminates differently at different angles. In addition, certain conditions can cause the appearance of false lens edges that strongly mimic true ones.

The main goal in these experiments is to record the accommodative state of the eye in each frame in order to make conclusions about changes in accommodation over time. Because dynamic changes in accommodation are being sought, it is necessary to make precise measurements with low noise. Typical experiments run at 30 frames per second for 48 seconds on the low end to more than an hour on the high end. The large volume of data coming from these experiments, combined with the need for consistent measurements, creates the need for these calculations to be performed automatically.

In this paper, we present an algorithm that is designed to segment lenses in individual photorefraction images. It is further extended such that the segmented boundary is refined by exploiting the temporal dimension of the video. The results of the developed algorithms are compared to results obtained using a Level Set method of segmentation. All three of these algorithms address the 'Lens Segmentation' module of the overall processing pipeline shown in Figure 3. The initial step of detecting the Purkinje image is related to detection of the lens center that forms the initialization for all segmentation algorithms discussed here. The lens center is detected by finding the projection of the LED array on the lens and is accomplished by template matching [14].

The rest of the paper is organized as follows: In Section 2, relevant previous work is discussed which lays the groundwork for the segmentation algorithms described here. Section 3 describes in detail the probabilistic edges (PE), and iterative probability (IP) methods as well as the level set method used for comparison. Methods of validating these algorithms are discussed in Section 4.1 and 4.2. Section 4.3 presents results from the individual methods as well as comparative results. Section 5 concludes with discussion of these methods and their significance.

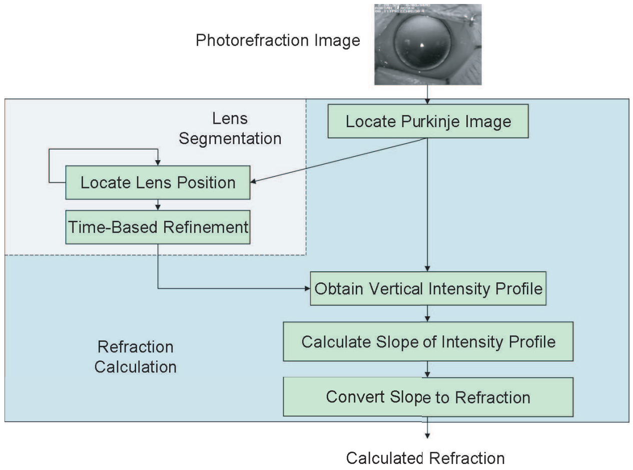

Figure 3: Diagram of photorefraction calculations.

2. PREVIOUS WORK

Many methods have been developed which are designed to follow or delineate the boundaries between regions based on image intensity or gradient. Active contours, or 'Snakes', as originally proposed by Kass, et al [8], make use of a parametric curve, $\mathbf{v}(s) = (x(s), y(s))$, which minimizes an energy functional:

$$E_{snake} = \int_0^1 E_{int}(\mathbf{v}(s)) + E_{ext}(\mathbf{v}(s)) + E_{con}(\mathbf{v}(s))\,\mathrm{d}s \quad (1)$$

The snake is designed such that the functional is minimized as the curve molds around the object of interest. In this functional, an external energy component is responsive to image features such as the gradient, an internal energy component defines the motion of the snake, and a constraint component incorporates an anchoring point or a repulsive force. As originally proposed, snakes have the disadvantages that they must be placed in close proximity to the object of interest and that they do not conform to concavities. Cohen and Cohen [1] addressed this by adding an inflating force to the active contour. Another adaptation to address these issues is to replace the external force with a gradient vector field [18]. In this method, a field of vectors point to the gradient and this field extends to all parts of the image, thus guiding the snake to the desired end point. Others have added an inertial force in a system that adds velocity and mass to the spline points [2]. The addition of mass and velocity to the spline points to other systems where particles have been used in an active contour like manner. In these systems, the particles behave like real world particles that have mass and follow Newtonian laws of motion. Szeliski et al. [15] used oriented particles which had not only a position, but an orientation in 3D space. Jalba et al. [7] represented their particles as charged particles in an electric field.

Level set formulations have been proposed for segmentation of curves by incorporating a means of following fronts which propagate with a velocity which is dependent on their curvature [13]. To do this, the object of interest is seen as the intersection of the plane (or volume) with a higher dimensional surface [11]. This method provides the advantage that it can deal with discontinuous regions and holes by representing them as branches of the higher dimensional surface. It also allows for initialization to occur far from the object of interest. A general problem with these techniques is the tendency to ‘leak’ when the magnitude of the stopping criterion, e.g. the gradient, decreases in certain regions. As an alternative, flux has been used to direct the level sets toward low contrast edges such as elongated blood vessels [17]. Instead of propagating level sets, flux has also been incorporated into graph cuts in combination with length/area [9].

The above methods are useful when a reliable edge type exists such as a gradient or a zero-crossing. However, many of the images in this study have competing edges with similar characteristics. For this, a method is desired which distinguishes between competing edges and assigns probabilities to each. Probabilistic methods have been applied to numerous problems in computer vision. Maximum a posteriori (MAP) was used for higher-level inference of 1-D structures to find roads in satellite images [5]. An analysis of this data indicated that MAP was effective only to the level that the images provided good visual cues [20]. Lower level segmentation was achieved by supervised learning of a large number of general image cues in regions centered at each point in training images [3]. Others have used a limited number of specifically chosen edge characteristics such as color and gradient to aid the classification of edge points [10, 12]. As opposed to edge points, Elder et al. [4] used supervised learning of contours where final boundaries were obtained by probabilistic methods such that a true probability could be assigned to the final contour.

Numerous papers report a multi-step approach using a variety of methods to segment or track objects in highly specific contexts. As one example, Yan et al. [19] described an approach for segmenting objects in medical images which could be described as a closed curve. In this approach, a region of the image which was centered on the object of interest was transformed into the polar domain. Then a pre-computing algorithm was used to locate the pixels on the boundary of the object. The image with selected pixels was transformed back into image coordinates and the pixels were fitted with a Fourier equation.

3. METHODS

This section presents the segmentation methods developed for detecting the lens boundary in all frames of the acquired photorefraction video. The probabilistic edge method (section 3.1) uses a multi-step approach reminiscent of Yan et al. [19] in that the image is pre-processed into a polar transform, edges are found on the polar image, and then a curve is fit to the edges. However, in this case, supervised learning of edge probabilities was used to find the edge. The results of the probabilistic edge method are further refined by an iterative probability method (section 3.2). Finally, these results are compared with a level set method (section 3.3).

3.1 Probabilistic Edge Method

As a first step, the image, $I(x, y)$, was transformed into a polar image, $I_p(\psi, r)$, using the following transform:

$$I_p(\psi, r) = I(a, b) \tag{2}$$

where

$$a = x_0 + \cos\psi * (r + R_0)$$

and

$$b = y_0 + \sin\psi * (r + R_0).$$

This is done to simplify edge calculations, which are expected to exhibit radial continuity due to the inherent lens shape. The point (x_0, y_0) is a selected point near the center of the image. The constant, R_0, is the distance from the image center at which the transform begins. The initial point in the polar transform (x_0, y_0) was chosen such that it was close to the center of the lens, (x_c, y_c). Assuming that the lens is circular and that (x_c, y_c) does lie within the circle, then the lens edge in the resulting polar transform will necessarily be a sinusoid with a single period, 2π. The equation for the sinusoid is given by:

$$r = r_0 + A * \sin(\psi + \phi). \tag{3}$$

The distance from the true center of the circle to the initial center is equal to the amplitude of the sinusoid:

$$A = \sqrt{(x_c - x_0)^2 + (y_c - y_0)^2} \tag{4}$$

and the the angular direction of (x_c, y_c) in relation to (x_0, y_0) corresponds to the phase, ϕ, in the sinusoid. The actual radius of the lens is given by r_0.

All subsequent operations were performed on the polar image. The set of all edge points, T, in the polar image were calculated using zero-crossings of first and second order derivatives:

$$T = T_1 \cup T_2 \tag{5}$$

where

$$T_1 = \{t_1,...,t_n\}, \frac{dI_{pt}}{dr} = 0$$

and

$$T_2 = \{t_1,...,t_n\}, \frac{dI_{pt}^2}{dr^2} = 0.$$

The subset of all edges which lie on the boundary of the lens are denoted as T^0. It was observed that first order lens edges, T_1^0, tended to appear as dark intensity minima in regions where the lens and the sclera came together, while second order edges, T_2^0, were high contrast gradients in regions where the retina could be seen between the lens and the sclera. Two first order texture values were also calculated from the polar image. Energy and entropy were calculated from histograms of pixel intensities in a 7x7 window centered at each pixel in I_p.

A set of characteristics, D, was collected at the positions of all edges, T, in each image. The set of characteristics can be enumerated as $D = \{d_1,..., d_n\}$. This set included (d_1) first derivative; (d_2) second derivative; (d_3) energy; (d_4) entropy; (d_5) the probability of an edge, $t \in T_2^0$, where $\frac{dI_p}{dr} < 0$; (d_6) distance from the previous edge, $t \in T_2$, where $\frac{dI_p}{dr} > 0$; (d_7) distance from the previous edge, $t \in T_2$, where $\frac{dI_p}{dr} < 0$; (d_8)

distance from the previous edge, $t \in T_1$; and (d_9) distance to the following edge, $t \in T_2$, where $\frac{dI_p}{dr} > 0$. Also an iterative probability factor, d_{10}, was set to a constant value for individual frames or calculated as described later for the Iterative Probability method.

Probability data was collected from thirty images in an image database. To account for the two types of lens edges, T_1^0 and T_2^0, points were manually selected separately for each. Two different sinusoids, S_1 and S_2 respectively, were then fit to each type of edge using least squares. Inclusion in T_1^0 or T_2^0 was determined by the distance to the corresponding set of calculated sinusoidal points, $S_1(\psi, r)$ or $S_2(\psi, r)$, in the polar image:

$$t_i \in T_1^0 \text{ if } |t_i - s_i| \leq 1,\ s_i \in S_1 \tag{6}$$

and

$$t_i \in T_2^0 \text{ if } |t_i - s_i| \leq 1,\ s_i \in S_2. \tag{7}$$

In thirty images, a total of 745,635 edge points were found such that $t_i \notin T^0$ while 6388 edge points were found such that $t_i \in T^0$.

During supervised learning, data was collected for each characteristic, $d_i \in D$ at each edge point, $t \in T$. The probabilities, $p(D|t \notin T^0)$ and $p(D\big|t \in T^0)$ were estimated by fitting a Gaussian to the data. As a result, two parameters, σ and μ were obtained at each edge point where $t \in T$. In addition, two parameters were obtained in the case where edge points were on the edge of the lens, $t \in T^0$. Once the probabilities were learned, the probability that any edge point, $t \in T$, would be on the edge of the lens could be calculated according to Bayes rule:

$$p(t \in T^0\big|D) = \frac{p(D\big|t \in T^0) * p(t \in T^0)}{p(D)} \tag{8}$$

For simplicity, the maximum a posteriori (MAP) method was used whereby the probability, $p(t \in T^0\big|D)$, was estimated by the posterior probability, $p(D\big|t \in T^0)$. The posterior probabilities were estimated with likelihood ratios for each edge trait, d_i, as follows:

$$p(t \in T^0\big|d_i) = \frac{1}{(1 + \frac{p(d_i|t \notin T^0)}{p(d_i|t \in T^0)})}. \tag{9}$$

In addition, two specific posterior probabilities were estimated in response to situations where true second order edges, $t \in T_2^0$, were preceded by a paired high probability first order edge, $t_{previous} \in T_1$. The first estimate modulates the probability of the distance from the previous edge by the probability of the previous edge:

$$p(t \in T_2^0\big|d_8) = \frac{1}{(1 + \frac{1}{p(d_8|t \in T_2^0) * p(t_{previous} \in T_1^0|D)})}. \tag{10}$$

Conversely, first order edges are multiplied by an inverse equation that reduces their probability if they are closely followed by a high probability second order edge, $t_{next} \in T_2$:

$$p(t \in T_1^0\big|d_9) = \frac{1}{p(d_9\big|t \in T_1^0) * p(t_{next} \in T_2^0\big|D)}. \tag{11}$$

Final posterior probabilities were combined differently for T_1 and T_2:

$$p(t \in T_1^0\big|D) = \prod p(t \in T_1^0\big|d_i) \tag{12}$$

$$p(t \in T_2^0\big|D) = \prod p(t \in T_2^0\big|d_j). \tag{13}$$

These probabilities are calculated for all edge points, $t \in T$, in I_p and stored as an image which is the same size as I_p.

The best fit to the probabilistic data is determined by an energy minimizing method. The total energy for each sinusoidal fit to the data is calculated by the following equation:

$$E_{total} = C_{dist} * E_{dist} + C_{conn} * E_{conn} + C_{prob} * E_{prob} \tag{14}$$

where

$$E_{dist} = \sum_{0}^{n=\psi_{max}} \big|r_p - t_n\big| * w\ , \tag{15}$$

$$E_{conn} = \sum_{0}^{n=\psi_{max}} \begin{cases} 1, (\big|r_p - t_n\big| * w) > 1 \\ o, (\big|r_p - t_n\big| * w) \leq 1 \end{cases}, \tag{16}$$

$$E_{prob} = \sum_{0}^{n=\psi_{max}} (w * p(t_n \in T^0\big|D))^{-1}\ , \tag{17}$$

$$w = \frac{\max(p(T_\psi\big|D))}{\max(p(T\big|D))} \tag{18}$$

where C_{dist}, C_{conn}, and C_{prob} are constants; r_p is the position of r that is predicted by the given sinusoid at the given position, ψ; and T_ψ is the maximum probability of all edges at any given value of ψ. The weight factor, w, is used to account for the low probabilities of certain regions of the probability image. Sinusoids are generated and the one which minimizes E_{total} is chosen as the best fit to the data. Rather than generate every possible sinusoid, initial ranges of parameters r_0, A, ψ, and ϕ were pre-selected using *a priori* knowledge of the system. The total energy, E_{total}, was calculated for three evenly spaced values of each parameter and then the parameters were narrowed in a binary search like manner.

3.2 Iterative Probability Method

The algorithm described so far segments the lens in images or frames. However, the size and position of the lens from frame to frame is strongly related. To take advantage of this, the iterative probability algorithm is used to characterize the data over the time domain.

Initial lens data is calculated using the probabilistic edge (PE) algorithm described previously. Individual frames along with the results of these calculations were stored in an array, M, of odd length l. A weighted mean and a weighted standard deviation are calculated for the lens data in the array. To weight these calculations, the lens results were multiplied by a Gaussian where $\sigma = l/4, \mu = l/2$, and x is the position of the data within the array. The PE algorithm is then performed on m frames beginning with the middle position in the array. In this case, the height, r, of the polar image, I_p, is set to $r_0 \pm 2 * sd$, where sd is the standard deviation which was previously calculated. In addition, an iterative probability factor, d_{10}, is set to a Gaussian centered at r_0 with $\sigma = sd$, thus increasing the probability of the previously obtained lens parameters. As before, the PE algorithm returns the lens parameters, (x_c, y_c) and r_0 which are then used to calculate a new mean and standard deviation. Additionally, the total probability, E_{total}, is also returned. The iterative probability (IP) algorithm minimizes E_{total} using gradient descent.

In addition to the information from the previous frame, the lead image can also use information from the previous IP calculation. The new IP factor, d_{10}, uses the average radii from the previous image and expands the sigma value based on the change in center position of the circle, according to:

$$\mu = \frac{r_{previous} + r_{IP}}{2} \tag{19}$$

and

$$\sigma = \sigma_{init} + \sqrt{dx_c^2 + dy_c^2}. \tag{20}$$

3.3 Level Set Method

For comparison, a level set method was applied to the same images and video segments as the proposed methods. The method used was the ShapeDetectionLevelSetImageFilter in the Insight Toolkit (ITK) [16] which is based on the paper by Malladi et al. [11]. Initially, the purkinje image was located as previously described. Then, the image was convolved with a GradientMagnitudeRecursiveGaussianImageFilter which acted as a Laplacian of Gaussian (LoG) filter that found the gradient magnitude of the Gaussian blurred image. Additionally, a sigmoidal filter was applied to the image intensity values using the equation:

$$I' = \frac{Max - Min}{1 + e^{\left(\frac{1-\beta}{\alpha}\right)}} + Min. \tag{21}$$

The user controlled the values of α and β in the sigmoidal filter as well as the value of σ in the LoG filter. Additionally, the user was able to start the level set boundary at a selected distance from the purkinje image.

The application of this filter typically stopped short of the actual lens edge at a position of high luminance. The actual lens edge is typically a dark boundary which can be described as a first order minimum in the image. For this reason, the boundary resulting of the level set method was used as the starting point of a second level set which operated on the original image to find an image minimum. Both applications of the level set method had components for propagation and curvature which were set by the user.

4. EVALUATION AND RESULTS

4.1 Data

An image database was created which had 126 photorefraction images. These were selected from videotapes of 29 different experimental sessions involving one of six different monkeys. Images from within a single experiment were selected to show a variety of accommodative states as well as different eye positions. Thirty of these images were used for supervised learning of the edge probabilities. Therefore, the remaining 96 images were used for ground truth comparisons with the different methods.

Comparisons of performance on video sequences were performed on a collection of 10 video sequences which ranged in length from 1300 to 1700 frames. Two of these sequences came from experiments in which rapid accommodative changes were stimulated while the remaining 8 sequences came from experiments with slower accommodative changes.

4.2 Experiments

To validate the developed algorithms in their ability to properly segment a lens in a still image, an expert established ground truth for the 96 test images in the image database. To do this, lens edge estimates were drawn on each image using a short program which draws a circle based on three selected points. Three acceptable estimates were averaged for each image returning an average radius, r_{GT}, and an average circle center (x_{GT}, y_{GT}).

Good segmentations of the lens do not exist as single values of radius and center, but rather as a range of these values. However, poor segmentations do not vary smoothly from good ones and tend to separate from the good ones when graphed. Therefore graphs were created which plotted positive segmentations against the percentage deviation from ground truth for both radius and circle center. An acceptable range, r_{limit} and d_{limit}, was defined for the radius and circle center, respectively .

Each algorithm was run on the same 96 test photorefraction images. The results were then evaluated such that if the distance of either the radius or the center point from ground truth exceeded the acceptable range, then the algorithm result was considered to be a false segmentation. Hence, each segmentation was assigned a score S according to:

$$S = \left\{ \begin{array}{ll} T, & r_{limit} \geq |r - r_{GT}| \text{ and} \\ & d_{limit} \geq \sqrt{(x - x_{GT})^2 + (y - y_{GT})^2} \\ & \\ F, & r_{limit} < |r - r_{GT}| \text{ or} \\ & d_{limit} < \sqrt{(x - x_{GT})^2 + (y - y_{GT})^2} \end{array} \right\}$$

Results are reported as percentages obtained from the number of true segmentations divided by the total number of images.

The algorithms were also run on 10 of the video sequences described earlier. In this case, the probabilistic edge algorithm was used both with and without the IP algorithm. Additionally, IP parameters were evaluated such that there were three different focus frames and 2 different window lengths (both defined on Figure 7) for a total of six combinations of parameters.

The radius from each frame was measured with a running average of 21 frames calculated from the same data. Then the differences of all of the radii from the running average were computed as a single root-mean-square (RMS) value for each video sequence. RMS values from all 10 video sequences were reported as a single average for each algorithm being tested.

In the Level Set method, the user controls the contributions of the various components of the level set equation and the associated preprocessing steps by setting the values of eight constants. No group of settings were found which worked best for all images tested. These settings were adjusted with varying results from image to image (Figure 4).

4.3 Results

The probability densities and associated training parameters for the proposed PE and IP algorithms were estimated from the 30 training images. Probabilities were obtained for each edge in the polar image based on the prior probabilities described in sections 3.1 and 3.2. Figure 5 shows an example polar image overlaid with edge probability estimates. Edges with higher probabilities are shown to be brighter (Figure 5C). Visual inspection of the probabilities overlaid on the original polar image (Figure 5B) shows that the higher probability edges correspond well with the actual lens edge. Additionally, some regions in the horizontal (ψ)

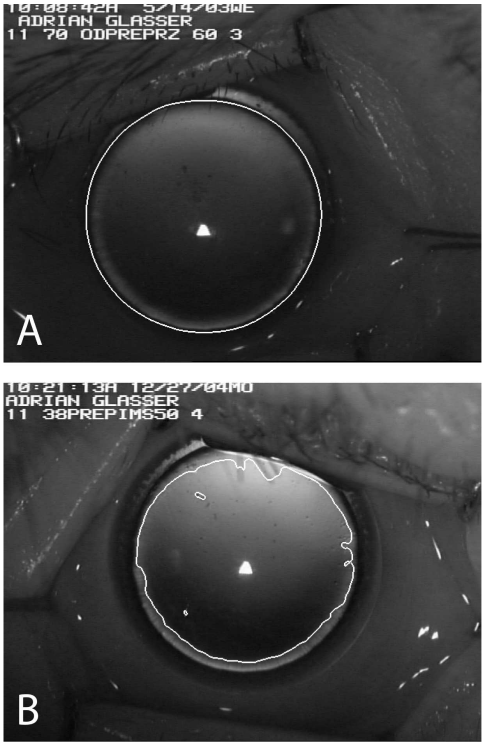

Figure 4: Two randomly chosen images of final level set positions on monkey lens image. Panel A shows a good fit to the monkey lens and panel B shows a boundary that stopped interior to the actual lens edge.

axis show higher probabilities than others indicating that various portions of a lens edge will contribute more than others to the edge calculation.

Quantitative evaluation was performed by comparing lens segmentation results to ground truth. Following manual segmentation of the 96 test images, each image was segmented using both algorithms, PE and Level Set. Two plots were created to find the range within which a result would be considered positive in relation to ground truth for the radius and for the center of the lens. In Figure 6A, the acceptable range from ground truth radius is held constant at a high value of 20% and increasing ranges from ground truth center position are tested. Conversely, the acceptable range from the ground truth center position is held at 20% and increasing ranges from ground truth radius are tested in Figure 6B. Based on these graphs, the acceptable limit was defined as being within 6% of the ground truth radius. Segmentations that deviated less than this amount from both the ground truth radius and the ground truth center were counted as positive matches for ground truth segmentation. Using these limits, it was found that the probabilistic algorithm correctly identified the lens in 89.6% of the images compared to 82.3% in the Level Set method.

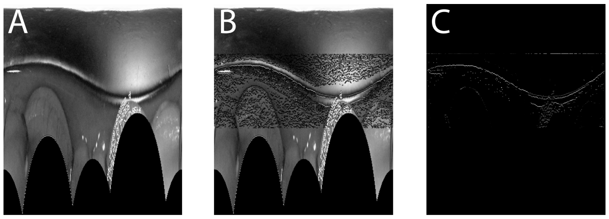

Figure 5: Edge probabilities from a sample polar image (A) are shown overlaid on the same image (B) and on a blank image (C).

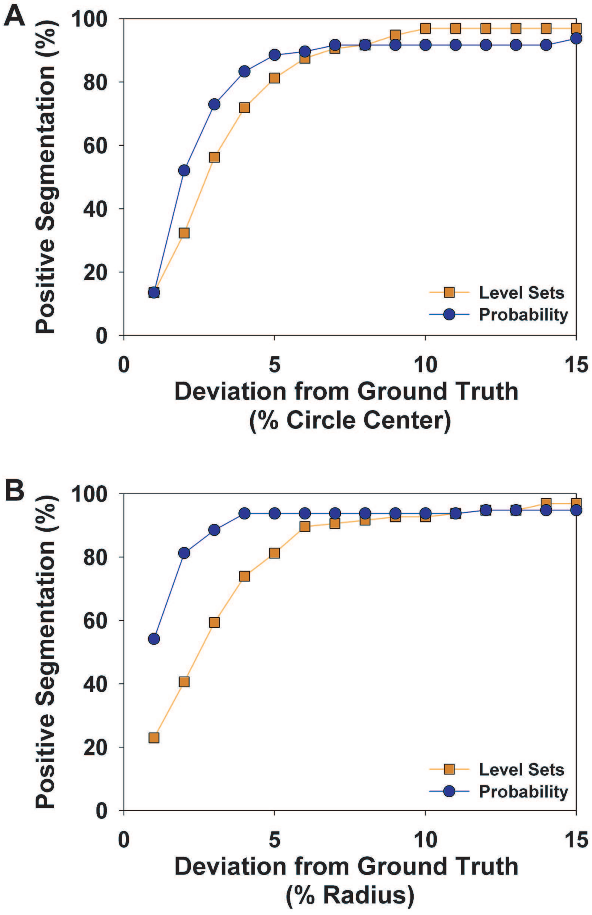

Figure 6: The percentage of images accepted as correct segmentations was plotted over the size of the range of deviation from ground truth for (A) variable distance to the lens center and (B) variable distance to the lens radius.

The iterative probability algorithm was analyzed with six combinations of parameters as described earlier. RMS values from 10 video sequences are shown in Figure 7A. There was essentially no difference in the variance whether the IP algorithm was run with one, two, or three focus frames. The larger window with 59 frames appeared to produce higher RMS values, but this was not a significant difference. However, the RMS values themselves had a higher standard deviation in video sequences of rapidly changing accommodation. Inspection of results from rapidly changing video sequences indicates that a large IP window size causes erroneous results in the vicinity of rapid accommodative changes. When data from these video sequences was removed (Figure 7B), there was a slight, non-significant trend toward lower RMS values with more focus frames.

The IP algorithm appears to aid the PE algorithm on disjoint video sequences where the lens position changes dramatically from one frame to the next. The RMS value from the Level Set method was large and variable at 1.9 ± 1.3 pixels, while those of the PE method (0.44 ± 0.12) and the PE method guided by the IP algorithm (0.43 ± 0.13) were lower, more consistent, and nearly identical to each other.

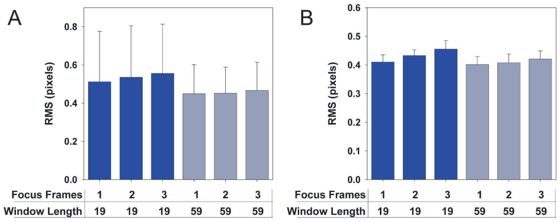

Figure 7: Two parameters were varied as the IP algorithm was used to test the variance of measured radius from either (A) all 10 video sequences or (B) 8 video sequences of eyes with slower accommodative responses.

Figure 8A shows results from a video sequence where the eye had moved during a time interval when the video camera had been turned off and on again. When the probabilistic algorithm was run on this sequence without guidance from the IP algorithm (red), it lost the lens location and wandered from that point on. The probabilistic algorithm was able to find the lens again when the IP algorithm was guiding the lead frame. The IP algorithm minimizes the sine fitting energy functional of the PE algorithm. Figure 8B shows how the final sine fitting energy increases as the probability of a good fit decrease. Figure 9 shows a comparison of RMS results from radius measurements of the level set method, the PE method, and the IP method used in combination with the PE method. The combination of IP parameters used for this comparison had an IP window of 19 frames and 3 focus frames. The level set algorithm parameters were adjusted at the start of each run.

5. DISCUSSION AND CONCLUSION

In comparison to the probabilistic edge method, the level set method had a lower percentage of correct segmentations on single frame ground truth images and a much higher variance as measured by RMS of radius measurements on video sequences. The level set method as used here applies the same criteria to the whole lens regardless of the edge conditions at any part of the lens. As such, it was vulnerable to varying conditions in different radial directions. As indicated by its reasonably good performance on single ground truth images, it segmented initial video frames correctly when the user was able to find the correct parameters. However, image conditions change during video sequences and there was no guarantee that the conditions in future frames would be sufficiently similar to those of the first such that the algorithm would continue to find the edge.

The probabilistic method could find multiple potential edges in any radial direction instead of just one. Each potential edge was assigned a probability and these probabilities were taken into account in the sine fitting functional. As a

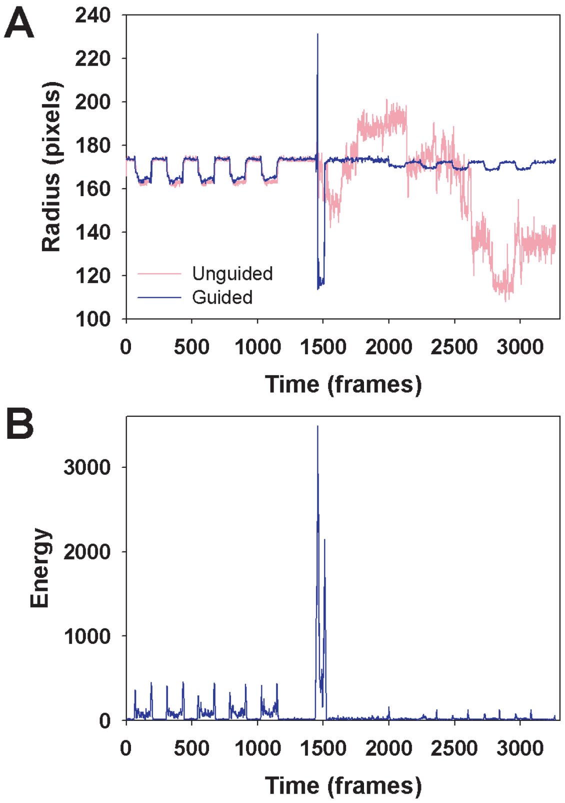

Figure 8: (A) The PE algorithm was used either alone (red) or in conjunction with the IP algorithm (blue) on a video segment in which the camera had been turned off in the vicinity of frame 1500. (B) The values of the sine fitting energy functional that is minimized in the IP algorithm are plotted over time.

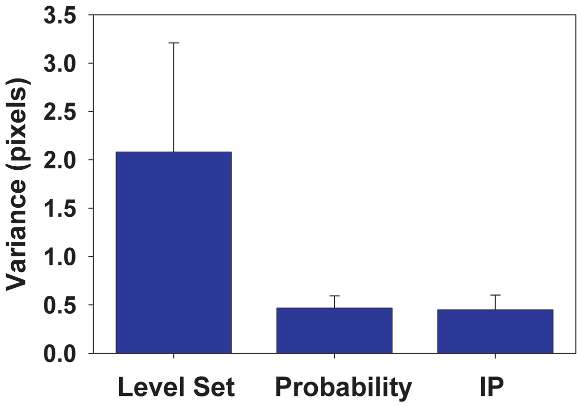

Figure 9: Shows the RMS variance in radius obtained on 10 video sequences.

result, this method was robust in the presence of random noise, although still vulnerable to competing structures in the image. A second advantage of the probabilistic algorithm was that probabilities could be learned for multiple edge types. In the current implementation, first order edges were one type of edge while second order edges were categorized as two types depending on the value of the first derivative. Second order edges with a first derivative below zero were never sought as a possible lens boundary, but rather they provided information which was used to calculate the probabilities of the other two types. Other edge types could be added if this method is to be extended or applied elsewhere. For example, the contact lens on the eye often had a detectable edge which could have been used to influence the probabilities of true edges. Boundaries can also include changes in other features such as texture.

Measurements of variance from the IP method showed no significant improvement over single, lead-edge measurements of the probabilistic method. The reasons for this are not clear and call for further investigation. However, the IP method did demonstrate the ability to steer the probability algorithm back to the lens when it was lost on a disjoint video segment. This is an important addition to the overall probabilistic method for reasons of both speed and accuracy. The probabilistic method is typically initialized with a full algorithm and then quickly shifted to a faster version which essentially means a smaller search window. This smaller search window is inherently faster because of the reduced calculations involved. It is also less vulnerable to other false segmentations because it is already focused on the region passed from the previous frame. The IP method now provides a mechanism to deal with situations where the lens is outside of the search window.

6. REFERENCES

[1] L. D. Cohen and I. Cohen. Finite-element methods for active contour models and balloons for 2-d and 3-d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):359–369, March 1998.

[2] B. Das and S. Banerjee. Inertial snake for contour detection in ultrasonography images. *IEEE Proceedings - Vision Image and Signal Processing*, 151(3):235–240, June 2004.

[3] P. Dollar, T. Zhuowen, and S. Belongie. Supervised learning of edges and object boundaries. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1964–1971. IEEE Computer Society, CVPR, 2006.

[4] J. H. Elder, A. Krupnik, and L. A. Johnston. Contour grouping with prior models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):661–674, 2003.

[5] D. Geman and B. Jedynak. An active testing model for tracking roads in satellite images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):1–14, 1996.

[6] W. He, M. Sengupta, V. A. Velkoff, K. A. DeBarros, and US Census Bureau. *65+ in the United States: 2005*. U.S. Government Printing Office, Washington, DC, 2005.

[7] A. C. Jalba, M. H. F. Wilkinson, and J. B. T. M. Roerdink. CPM: a deformable model for shape recovery and segmentation based on charged particles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1320–1335, 2004.

[8] M. Kass, A. Witkin, and D. Terzopoulos. Snakes - active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1987.

[9] V. Kolmogorov and Y. Boykov. What metrics can be approximated by geo-cuts, or global optimization of length/area and flux. In *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 564–571, Washington, DC, USA, 2005. IEEE Computer Society.

[10] S. Konishi, A. L. Yuille, J. M. Coughlan, and C. Z. Song. Statistical edge detection: learning and evaluating edge cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(1):57–74, 2003.

[11] R. Malladi, J. Sethian, and B.C.Vemuri. Shape modeling with front propagation - a level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158–175, 1995.

[12] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):530–549, 2004.

[13] S. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.

[14] M. Sonka, V. Hlavac, and R. Boyle. *Image processing, analysis, and machine vision*. Brooks/Cole Publishing Company, Pacific Grove, CA, 2 edition, 1999.

[15] R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling surfaces of arbitrary topology with dynamic particles. In *Computer Vision and Pattern Recognition, Proceedings IEEE Computer Society Conference on*, pages 69–74. IEEE Computer Society, CVPR, 2007.

[16] National Library of Medicine. `http://www.itk.org`.

[17] A. Vasilevskiy and K. Siddiqi. Flux maximizing geometric flows. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(12):1565–1578, 2002.

[18] C. Y. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, 7(3):1131–1147, November 1993.

[19] Z. Yan, B. J. Matuszewski, S. Lik-Kwan, and C. J. Moore. A novel medical image segmentation method using dynamic programming. In *Medical Information Visualisation - BioMedical Visualisation, International Conference on*, pages 69–74. MediVis 2007, 2007.

[20] A. L. Yuille and J. M. Coughlan. Fundamental limits of bayesian inference: order parameters and phase transitions for road tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(2):160–173, 2000.