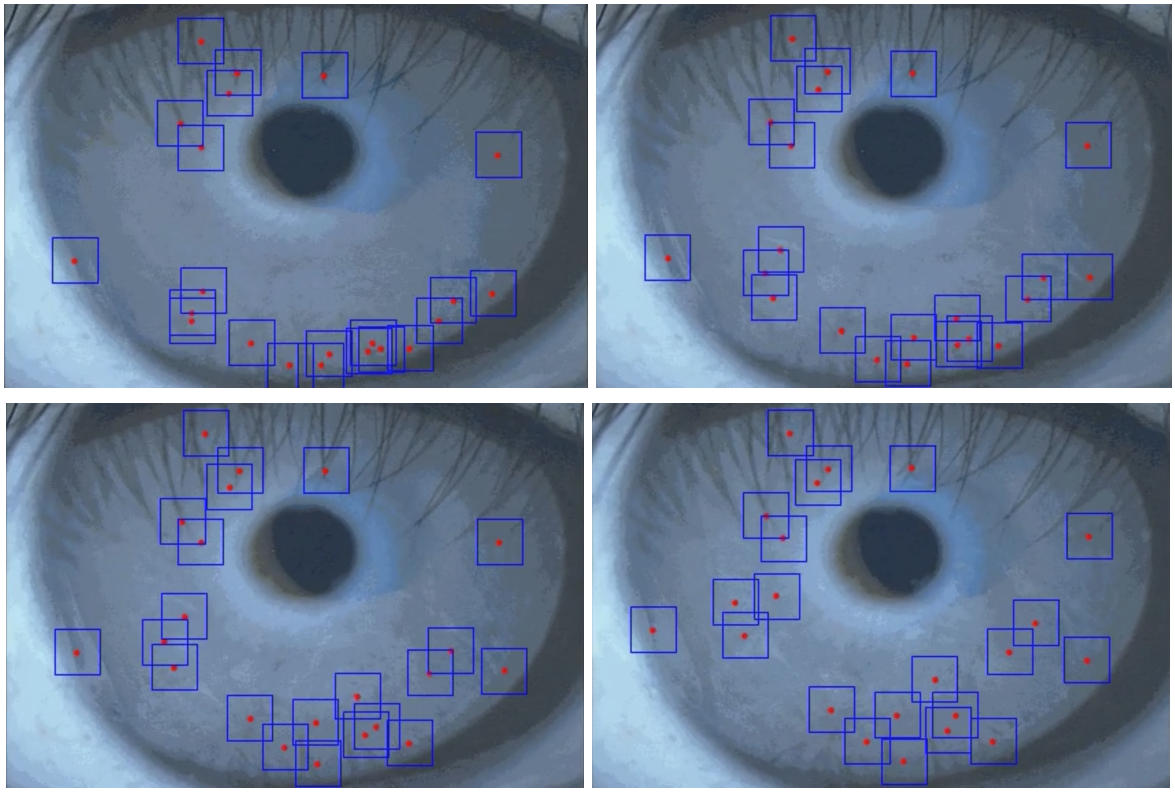

Figure 1. The above four pictures demonstrate the tracking of the lipid layer. The red dots represent the feature points tracked through the frames and the blue squares represent the patches associated with each feature point. The top left picture is right after a blink and as it can be seen all the feature points start out at the bottom of the iris. The top right is a few frames after and as it can be seen the feature points are moving up the iris. Similarly the chronological order is followed by the bottom left and the bottom right frames. The feature points move up the iris through the frames.

1. Abstract

This paper presents a robust way to track the lipid layer movement from a video of the eye. To the best of our knowledge this paper is the first of its kind in that it uses visual computing techniques to track the lipid layer. This paper will mainly focus on the methodology used for tracking. We start by detecting blinks in the video so that we can analyze each set of frames between blinks separately. We then align the frames in order to get an accurate average frame. This average frame is used to segment the iris with clustering based segmentation. The feature vectors used for clustering are a combination of the LUV color space value and the position of the pixel. Finally, random points are picked within the segmented iris region and tracked with Sum of Squared Differences (SSD) feature tracking. We show the significance of this procedure by analyzing the vertical displacement of the lipid layer through time and also the lipid layer thickness across the iris region.

2. Introduction

Analyzing the lipid layer has significance in diagnosing diseases related to the eye. Specifically the lipid layer movement can help understand the differences between dry eyes and healthy eyes [2, 4]. For example, the lipid layer behavior is on average different for Asian and Caucasian subjects [3]. The only previous work that has tried to track the motion of the lipid layer is done in [2]. However, their approach did not use visual computing techniques. Thus this paper contributes a first of its kind approach to analyzing the lipid layer. Figure 1 shows snapshots of the lipid layer being tracked over multiple frames. The rest of this paper will describe how the tracking demonstrated in Figure 1 is accomplished.

3. Methods

3.1. Blink Detection

The video of the subject eye is captured so that multiple eye blinks are included. After each eye blink the tear film is restructured. The goal is to track the tear film's motion in between blinks. Right after a blink, the tear film will rapidly restructure and move but it will slowly stagnate in motion and eventually settle in place until the next blink. It is thus important that the tear film motion is tracked separately for each blink.

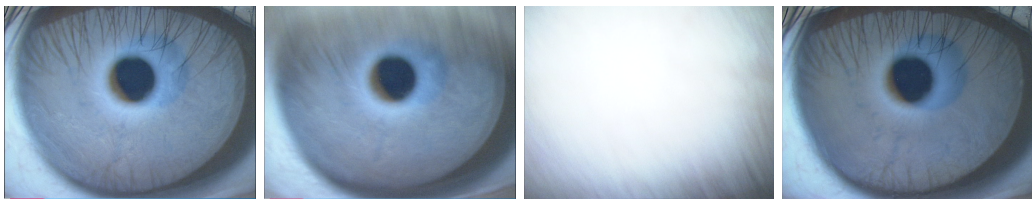

Figure 2. The above four pictures demonstrate what a blink appears as in the video. The first picture is right before a blink. The second picture is right when the blink is starting. The third picture is when the eyelid has completely occluded the eye. The fourth picture is right after the blink is complete.

Figure 2 illustrates what the frames of the video look like during a blink. The goal of blink detection is to find the frames where the eyelid is partially or completely occluding the eye. We then want to collect the frames in between blinks and analyze each set of frames separately.

We observe that the video of the eye looks relatively constant in nature aside from when the blinks appear. Thus we compute the average frame of the video. This is done by adding all the frames in the video together in the RGB colorspace and dividing by the total number of frames. Figure 3 illustrates what the average frame looks like.

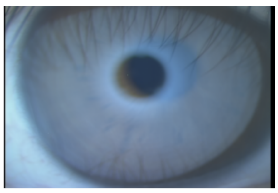

Figure 3. The average frame of the video.

Every frame is subtracted by the average frame and the frobenius norm is computed as a measure of distance from the average frame. Then the z-score of each frame's distance from the average frame is computed. Figure 4 shows the distance of each frame from the average frame and the thresholding used to determine blink frames.

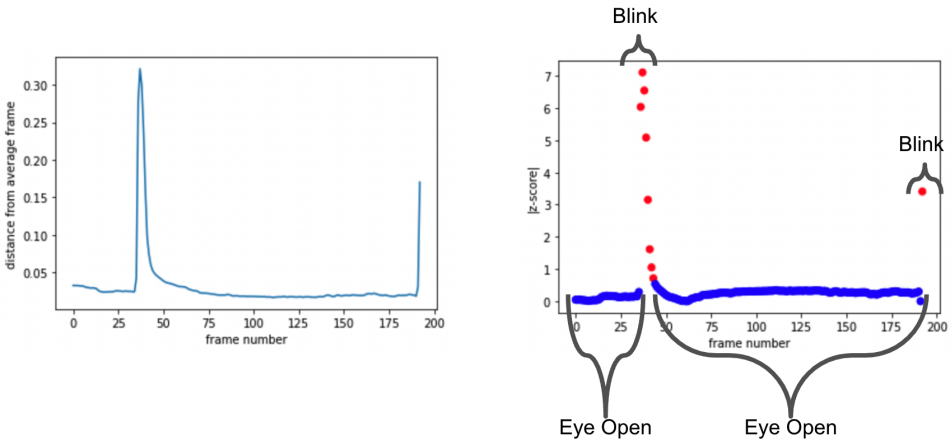

Figure 4. The plot on the left shows the distance of each frame from the mean frame. The plot on the right shows the absolute value of the z-score of each frame. The plot shows when the absolute value of the z-score is greater than 0.6 it will be considered a blink frame. If the frame's absolute value z-score falls below 0.6 it will be considered as the frame where the eye is open.

With blink detection each set of frames between blinks will be analyzed separately for tear film motion. In Figure 4 there are two separate sets of open eye frames that are separated by a blink.

3.2. Iris Segmentation

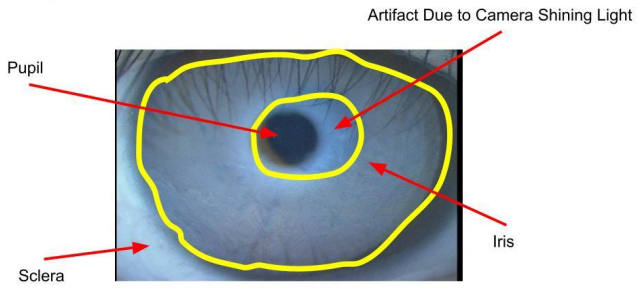

Figure 5. This picture labels the different parts of the eye that are of significance when trying to segment the eye.

When tracking the lipid layer, it is important to only do it over the iris region. Thus it is essential to discriminate all other pixels from the iris. This means we want a segmentation map where the iris is detected.

3.2.1. Single Frame Iris Clustering

We first try segmentation on a single frame. Specifically, the frame right after a blink is completed will be segmented for the iris. We use clustering, which is a form of unsupervised segmentation. The idea is to assign a cluster to every single pixel in the image. K-Means clustering is used for its efficiency and speed. Figure 6 illustrates the steps for segmenting the iris in the image.

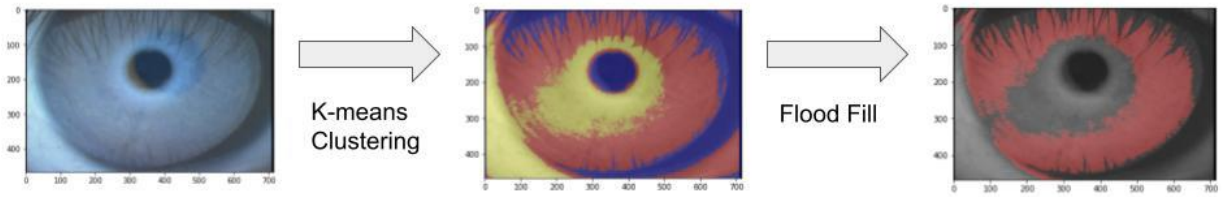

Figure 6. This graphic shows the pipeline for segmenting the iris. Each pixel feature vector is assigned a cluster with K-means. Then the cluster with the largest concentration of pixels near the center of the image is picked and flood filled to get the iris.

Several feature vectors for each pixel were considered. For the color value both the RGB and LUV colorspace were tested. In addition, the x,y coordinate values were concatenated to the colorspace feature as well to encode position of the pixel. These position pixels were multiplied by a gaussian kernel centered at the center of the image. The motivation behind multiplying by a gaussian kernel was to encode the natural circular shape of the eye into the pixel feature vectors.

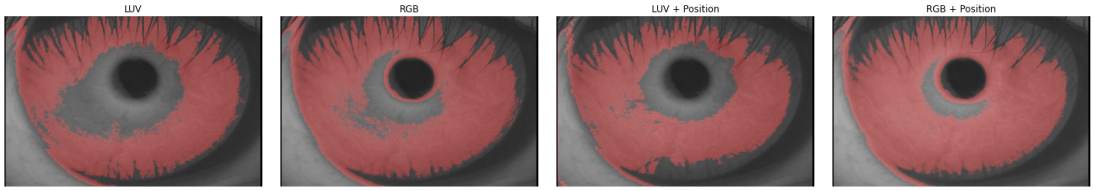

Figure 7. These pictures show the clustering segmentation results for different feature vector choices in the single frame setting. The title on top of each picture states the feature vector choice. We found that the "LUV + Position" feature vector was the best for single frame segmentation.

Figure 7 demonstrates the qualitative results of segmentation with clustering using different feature vector choices. We find qualitatively that using the LUV color space seems to do a better job for single frame clustering. In addition we find that encoding position into the feature vector results in a better segmentation of the iris. Ultimately, however we found that the segmentations for a single frame alone were too noisy and changed too much between frames.

3.2.2. Aligning Frames with Sum of Squared Differences

In order to address the noisy results of segmentation frame by frame we decided to start by aligning all the frames. When the subject's eye is recorded there will be some inherent movement in the eye. This will change the overall position of the iris between frames. Thus each frame will be aligned with its previous frame. The first valid frame will serve as template for every consecutive frame to match. Frame *n+1* will be aligned with frame *n*. Frame *n+1* will slide across frame *n*, and the alignment score will be computed

with Sum of Squared Differences (SSD). The borders of frame *n+1* will be cut out when computing the alignment. Thus only 90% of the frame *n+1* will be considered when aligning to frame *n*. Figure 8 illustrates the recursive process of aligning all the frames.

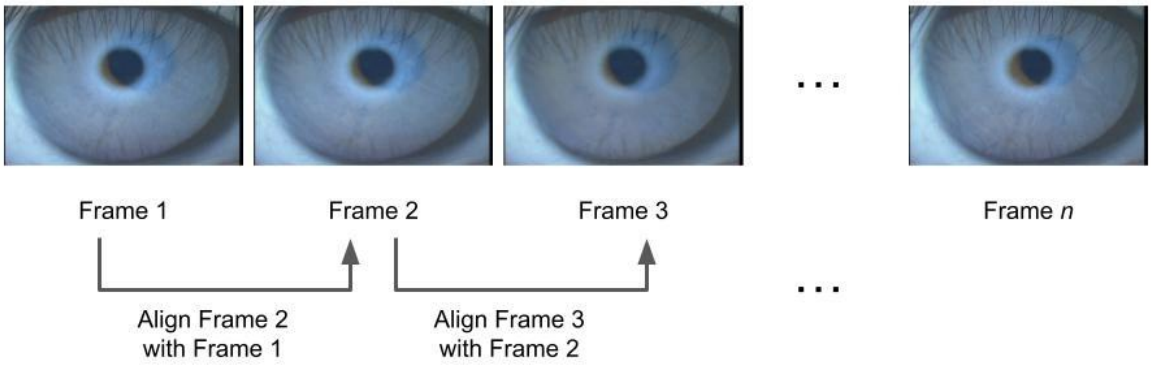

Figure 8. This graphic illustrates the process of aligning all the images together. The process starts with aligning frame 2 with frame 1 and every consecutive frame will be aligned with its previously (already aligned) frame. This recursive process will continue until frame n is aligned with frame n-1. Note when a frame is being aligned with its previous frame, the borders of the current frame will be cut out.

3.2.3. Aligned Mean Frame Iris Clustering

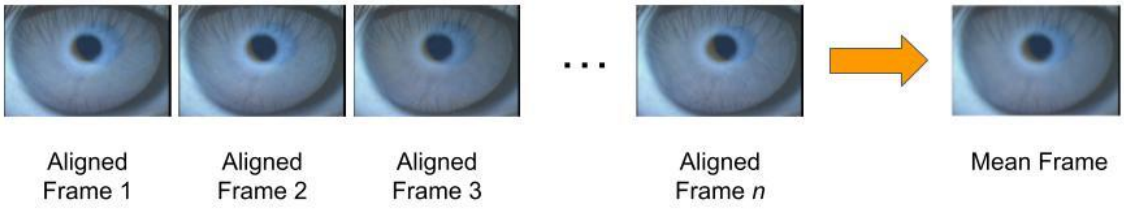

Figure 9. This graphic illustrates how the mean frame is computed. Once all the frames have been aligned they are averaged out in the RGB colorspace to yield the mean frame.

After all the frames of interest have been aligned the mean frame is computed as shown in Figure 9. Since all the frames have been aligned to this mean frame we can segment the iris in the mean frame and use the same segmentation map for all the aligned frames. Figure 10 shows the qualitative results of segmenting the iris in the mean frame for different choices of feature vectors. We find that the "*LUV + Position"* feature vector has the best qualitative results for segmentation. We can compare the results of segmentation on a single frame versus the average frame by looking at Figure 7 and Figure 10, respectively. We observe that the results for segmentation on the average frame is better across all the feature vector options.

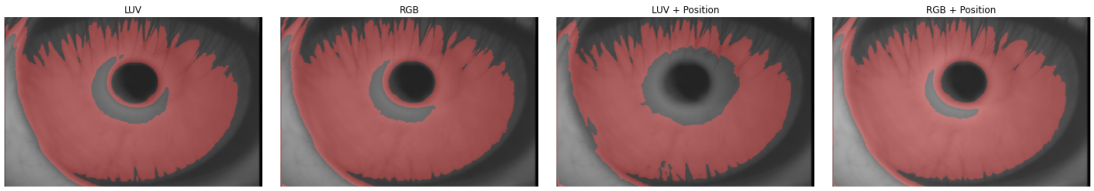

Figure 10. These pictures show the clustering segmentation results for different feature vector choices in the mean frame setting. The title on top of each picture states the feature vector choice. We found that the "LUV + Position" feature vector was the best.

3.3. Tracking Lipid Layer with SSD Feature Tracking

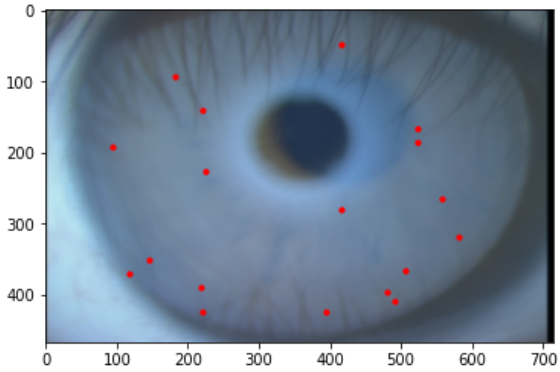

Figure 11. In order to track points on the tear film we randomly pick points on the segmented region of the iris. These feature points will be tracked across frames.

With the Iris region segmented, we can now track points on the tear film within the iris region. We take the starting frame and randomly pick feature points. Each feature point will have a 75 x 75 pixels patch associated with it where the feature point is at the center. These patches from the previous frame will be used to compute the SSD across the whole image for the next frame to find the most similar patch. Once the most similar patches are found for each feature point, the new patches are used for the next frame tracking. This process is repeated through all the frames. Note that we found randomly picking the feature points for the last frame and tracking backwards in time performed better.

The inspiration for tracking feature points comes from [1]. Their method tracks feature points on the retina but the same principles hold since they are also dealing with the noisy nature of eye movement. Their method uses Normalized Cross Correlation (NCC) instead of SSD. We found that SSD performs a lot better because it respects the intensity value of the pixels whereas NCC is invariant to local average intensity and contrast.

4. Clinical Results

4.1. Lipid Layer Thickness Heatmap

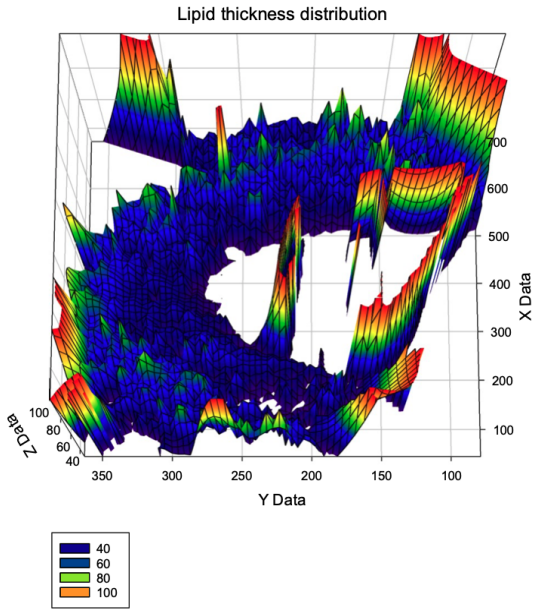

Figure 12. This graph shows the Lipid Layer thickness for one frame in time.

Our method is able to successfully segment out the iris region in each frame. Using the technique described in [5] we can compute the lipid layer thickness using the RGB values of the frame.

4.2. Exponential Decay Curve for the Vertical Displacement

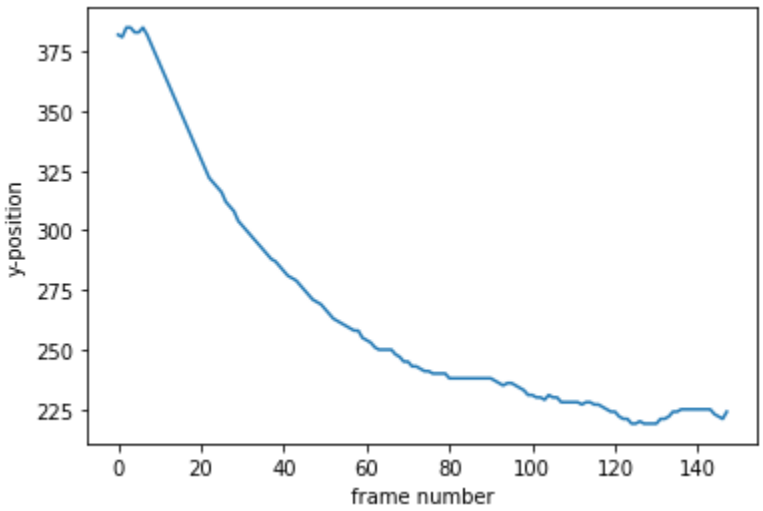

Figure 13. This graph shows the vertical displacement of the lipid layer is an exponential decay curve.

Our lipid layer tracking confirms the findings in [2]. They find that they can model the motion of the vertical displacement with an exponential decay curve. Figure 13 shows the vertical displacement of the lipid layer through the frames. As it can be seen our tracking method confirms that it is in fact an exponential decay curve.

5. Conclusion and Future Work

This work presents a first iteration to track lipid layer motion. Our work confirms previous findings and analysis of lipid layers. While we provide a robust method we intend to iterate on this method. In particular we want to improve the iris segmentation method by implementing contour finding. We also want to improve the tracking of the lipid layer in future iterations as well. We want to try to use a band-pass filter for the color space of the image so that only the lipid layer appears in the image without too much noise from the iris. In addition, we want to try to use optical flow to further improve the tracking of lipid layer.

Bibliography

1. Jay Shenoy, James Fong, Jeffrey Tan, Austin Roorda, & Ren Ng (2021). R-SLAM: Optimizing Eye Tracking from Rolling Shutter Video of the Retina.
2. Norihiko Yokoi, Hideaki Yamada, Yutaka Mizukusa, Anthony J. Bron, John M. Tiffany, Takahisa Kato, & Shigeru Kinoshita (2008). Rheology of Tear Film Lipid Layer Spread in Normal and Aqueous Tear–Deficient Dry Eyes.
3. Tatyana F. Svitova, & Meng C. Lin (2013). Racial Variations in Interfacial Behavior of Lipids Extracted from Worn Soft Contact Lenses.
4. James P McCulley, & Ward Shine (1997). A Compositional Based Model for the Tear Film Lipid Layer.
5. Hyeonha Hwang, Hee‑Jae Jeon, Kin Choong Yow,, Ho Sik Hwang, & EuiHeon Chung (2017). Image‑based quantitative analysis of tear flm lipid layer thickness for meibomian gland evaluation.