Gaze Estimation Using Image Segmentation, Feature Extraction, & Neural Networks

by Katie Morgan
Human gaze estimation is a rapidly growing field of research with many useful applications. It is especially useful for human-computer interaction (HCI) applications, such as video conferencing and eye-typing.

Many eye and gaze tracking methods exist, but not all are desirable for the average computer user, either because they are intrusive, expensive, or require frequent recalibration.
One method [10] uses an infrared camera and an infrared light to obtain bright and dark pupil images which highlight the pupil and cause an infrared glint on the eye. By fitting an ellipse to the pupil and using the glint coordinates, user gaze is accurately approximated.

Requires the use of expensive and specialized equipment.
Another method [3] uses a simple camera to capture grayscale images of the user. A small eye image is segmented from this large image, and is then input to a neural network, which returns the gaze position coordinates.

Although straightforward, it was thought that the neural networks in this method would be very large, since it would need to represent images containing 600 pixel values.
The method researched combines the approaches of the infrared method and the image neural networking method:

1. A small eye image is segmented from the larger image of the user.
2. Pupil is identified and an ellipse is fit around it.
3. The area of the sclera (white of the eye) to the left and right of the pupil is calculated.
4. Parameters from the fitted ellipse and the ratio of the left and right sclera areas are saved in feature vectors used to train 2 neural networks to approximate screen gaze coordinates.

By using feature vectors instead of images to train the neural networks, it was hypothesized that the networks would be smaller and more efficient.
Equipment and Setup

- The system setup was designed to run in an open office environment.
- Users sat about one meter away from the camera in front of the left monitor, after which the camera was manually adjusted once to put the subject’s right eye near the center of the resulting images.
Using thresholding and known information about average pupil features, extract the small image of the user’s right eye from the larger image.
Feature Extraction

Pupil Ellipse Fitting

Identify the pupil in the eye image and fit an ellipse around it
Histogram normalization is performed on the eye image to enhance the brightest and darkest values.

This image is thresholded so that only those points with the lowest 3% of pixel values remain.

The resulting clusters of points are analyzed based on size, and the cluster with the largest area is assumed to be the pupil.

The image indices corresponding to the pupil cluster points are input to an ellipse fitting method to determine the parameters of the ellipse which best fits the pupil.

Angle of ellipse orientation, x & y coordinates of the ellipse center, and the ratio of it’s major to minor axis are the 4 ellipse parameters used to train the networks.
Feature Extraction

Sclera Detection & Area Calculation

Detect the sclera to the left and right of the pupil and calculate the ratio of their areas.
Feature Extraction

Sclera Detection & Area Calculation

- Starting from the center of the pupil and moving left, compare the difference in value between neighboring pixels
  - If two pixels are found whose difference is greater than the difference threshold, then the edge between the iris and sclera has been found
  - If no edge is detected, the difference threshold decreases and the process is repeated until an edge is found

- The maximum and minimum values in a neighborhood around the first sclera pixel determine the range of pixel values of the sclera

- Using these values, a recursive algorithm searches a set window for neighboring pixels within value range, and increases the sclera area count until all appropriate pixels are found

- The same procedure repeated to find the right sclera area

- Once both counts are obtained, their ratio is calculated and used as the 5th element in the feature vector
Four black, square points were produced in the center of the four screen quadrants.

One by one, in random order, each of these points was made much larger than the remaining three.

As a point was enlarged, the program allowed a one second delay time for the user to focus on the point.

The interface captured three different images to avoid having an image of the subject blinking.

The images and the position of the corresponding gaze point were appended to the end of a data file.
Training Data Collection

- Segmentation was then used to attempt to find the subject’s right eye.
- If an eye candidate was not found, then the program segmented the second image, and then the third.
- If no eye candidates were found, the test data was rejected.
- Once an eye candidate was found, the segmented image was saved to the same file as the data and full image.
- This procedure was repeated for all 4 gaze points.
- The operator then manually checked the segmented images of the saved data.
- Any segmented image not of an eye, then this image, along with the full image and data corresponding to it, were deleted.
- By doing this, the neural networks would only be trained with relevant data.
Timing Results

- Timing results from the average runtimes of 77 images (combined) and 50 successful images (segmentation and feature extraction).
- Improving segmentation time will improve the combined speed.

<table>
<thead>
<tr>
<th></th>
<th>Image segmentation</th>
<th>Feature Extraction</th>
<th>Combined</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. Time (s)</td>
<td>0.3190</td>
<td><strong>0.0300</strong></td>
<td>0.3596</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>IR</th>
<th>Image NN</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Segmentation (fps)</td>
<td>--</td>
<td>--</td>
<td>3.13</td>
</tr>
<tr>
<td>Feature Extraction (fps)</td>
<td>--</td>
<td>--</td>
<td><strong>33.33</strong></td>
</tr>
<tr>
<td>Combined Method (fps)</td>
<td>20</td>
<td>20</td>
<td>2.78</td>
</tr>
</tbody>
</table>
The method was tested on twelve test subjects of varying age, gender, and race (Asian and Caucasian). The method performed fairly well for many test subjects.
Feature Extraction Results

- Once the image was correctly segmented, features were accurately extracted in most cases. Glare, squints, and eyelashes were the usual causes of incorrect extraction.

- A correlation between gaze position and ellipse orientation can be seen – although there is some difference for each test subject, the main ellipse shape and orientation remains similar for each of the four gaze points.
The feature extraction procedure also performs well for some users wearing glasses, although the results are dependent on the type of glasses worn, which affects the position of the glare on the lens.
Incorrect Results

Hair Detected

Eye Corner Detected

One Sided Sclera

Eyebrow Fit with Ellipse
At this time, there is not sufficient training data for the appropriate training of the neural networks – there was simply not enough time before this presentation to collect the hundreds of necessary feature vectors required.

- The 50 feature vectors obtained in the data collection were from tests of many different people.
- In order to effectively train the neural networks for this many subjects, many more feature vectors per person would be required.
- Until the feature extraction and segmentation procedure can be improved and sped up, the neural network data would also be inconsistent and incorrect.
Future Work

Training the Neural Networks

- Eventually, there will be two neural networks, each trained with the same large set of input feature vectors and target gaze coordinates.

- One network will approximate the vertical coordinate, and the other the horizontal, since there is a larger difference between horizontal movement than vertical movement.

- Many other methods use General Regression Neural Networks because of their faster training times.

- However, they often use varying numbers of layers, a capability not available in Matlab, so the use of Radial Basis Neural Networks will also be examined during future study, since the two networks are very similar.
Future Work

There is much future work to be done to explore and improve this method.

- Testing the method with different head angles and experimenting with the eye segmentation portion of the method.
- Using edge enhancement to improve sclera detection.
- Improving the accuracy of the system to divide the screen into 8 regions.
- Implementing the method with a web-camera instead of the Sony DVR camera would also be a desirable future study in order to decrease the cost of this method even further.


