Unpublishd manuscript, provides complete details of 2006 ETRA presentation.

Optical Eye Models for Gaze Tracking

Jeffrey B. Mulligan*
NASA Ames Research Center

Abstract

The traditional "bottom-up" approach to video gaze tracking consists of making measurements of images features, such as the position of the pupil centroid, corneal reflex, limbus, etc. These measurements are mapped to gaze angles using coefficients obtained from calibration data, obtained when a cooperative subject voluntarily fixates a series of known targets. This may be contrasted with a "top-down" approach in which the pose parameters of a model of the eye are adjusted in conjunction with a camera model to obtain a match to image data. One advantage of the model-based approach is provided by robustness to changes in geometry, as might occur due to slippage of a head-mount supporting a camera platform. A second advantage is that the pose estimates obtained are in units of degrees; traditional calibration serves only to determine the relation between the visual and optical axes, and provide a check for the model. While traditional grid calibration methods may not need to be applied, a set of views of the eye in a variety of poses is needed to determine the model parameters for an individual. This paper describes a simple eye model geometry, explores its behavior, and outlines a simple procedure for determining the model parameters from a small set of images.

Keywords: gaze tracking, optics, image formation, eye models

1 Introduction

This paper is concerned with the estimation of a human subject's direction of gaze from an image of the eye. A common approach is to assume a particular functional relationship between image feature measurements and gaze parameters, and then determine the parameters of the function from a set of calibration images in which the subject fixates known targets. We refer to this as the "bottom-up" approach, because we attempt to generate the gaze estimates from the raw measurement data, without using any knowledge of the geometry of the underlying eye-camera system. This approach can often attain satisfactory results, in large part because, for many geometries, the relationship between gaze angles and feature positions is linear.

The present work was motivated by two concerns. First, we would like to determine the limit of accuracy that can be obtained when a particular algorithm is applied to a particular imaging geometry. Second, we would like to have a gaze tracking method which requires minimal cooperation from the subject in order to obtain a calibration. We will address both of these goals by investigating what we call the "top-down" approach, in which the measurements are used to constrain the parameters of a geometric/optical model of the eye.

*e-mail: jmulligan@mail.arc.nasa.gov

Two recent applications of the top-down or model-based approach [Ohno et al. 2002; Beymer and Flickner 2002] have utilized measurements of the pupil location and shape, and the positions of one or more glints. (We use the term *glint* for the sake of brevity; it refers to the virtual image of the illuminator formed by reflection from the front surface of the cornea, and is often referred to as the first Purkinje image, or corneal reflex.) In both cases, these authors were estimating the gaze of a freely moving subject using a remote camera, with the possible eye poses forming a five dimensional manifold, (two rotational and three translational degrees of freedom). Both models were rotationally symmetric about the optical axis, in which case a torsion or twist about the optical axis produces no change in the shape or position of the pupil image, or glint.

The situation is significantly simpler for the case of a head-mounted camera; in this case, the pose manifold has three fewer dimensions, corresponding to the translational movements between the camera and the head. If the eye rotates about a fixed center, then the pose manifold may be associated with a pure rotation by placing the origin of the coordinate system at this center; it is known, however, that the center of rotation is not fixed, but follows a locus known as the *centrode* [Carpenter 1977]. In this case, for any choice of origin, a generic pose will be represented by both a rotation and an associated translation, with the composite transformations forming a two-dimensional manifold. While the case of a head-mounted camera offers some advantages due the dimensionality reduction of the pose space, it also generally implies a short camera distance, and therefore necessitates consideration of perspective effects.

In addition to modeling the pupil and glints, we also model the outer margin of the iris, or *limbus*. The limbus is an attractive feature for several reasons. First, unlike the pupil, it is viewed directly, without refraction by the cornea. Thus we can estimate parameters such as its size without regard to the estimation of corneal shape. Secondly, and again in contrast to the pupil, the limbus does not fluctuate in size, and so once we know its size we can make good pose estimates even when only one side of the limbus is visible, particularly when head movement is not a concern. Finally, the limbus is a strong, high-contrast feature, which (in unfavorable imaging situations) can be the only measureable feature. Interestingly, one of the early methods to obtain a translation-invariant measure of eye orientation [Rashbass and Westheimer 1960] involved a differential measurement of the limbus and glint, using an oscilloscope trace imaged on the limbus.

In remainder of the paper, we begin by presenting our geometric eye model, and its parameterization, along with some implementation details of the imaging simulation. We then use the simulation to generate synthetic data which we use to explore the performance of reduced models.

2 The Geometric Eye Model

We model the eye as a spherical ball with radius r_E, which rotates about its center (see figure 1). This sphere is cut by the plane of the iris, which is at a distance h_L from the eyeball center. The circular curve defined by the intersection of the sphere with this plane is the limbus. The parameters r_E and h_L determine the radius of the

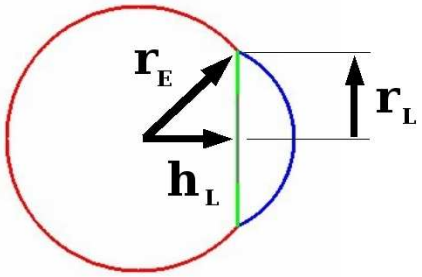

Figure 1: The eyeball is modelled as a sphere of radius r_E, which is cut by the plane of the iris a distance h_L from its center. The radius of the limbus, r_L is determined by these parameters.

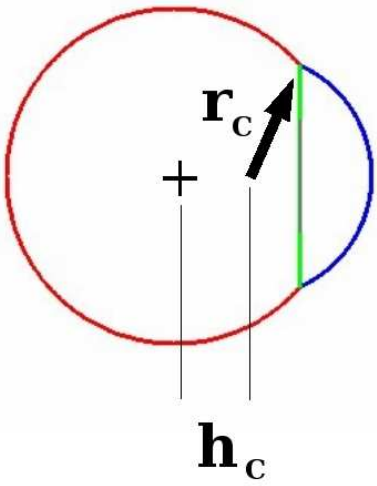

Figure 2: The cornea is modelled as a sphere of radius r_C, centered at a distance h_C from the center of the eyeball.

limbus:

$$r_L = \sqrt{r_E^2 - h_L^2} \tag{1}$$

We model the cornea as a spherical surface, bounded by the limbus (figure 2). (While the results presented here are all for a spherical cornea, most our computations are based on discrete samples from the corneal surface, and so can be performed for arbitrary corneal shapes such as the ellipsoid or conicoid.) Given the position of the limbus, there is only one free parameter; if we specify the position of the center of curvature of the cornea to be at a distance h_C from the eyeball center, then we can express the cornea radius of curvature as:

$$r_C = \sqrt{r_L^2 + (h_L - h_C)^2} \tag{2}$$

We model the pupil as a circle in the plane of the iris (figure 3). The primary pupil parameter is its radius (which generally varies from frame-to-frame); additionally, we include a pair of decentration parameters, x_P and y_P, which describe the position of the pupil

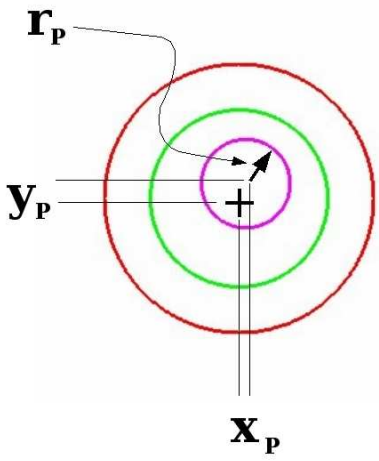

Figure 3: The pupil is modelled as a circle of radius r_P in the plane of the limbus. Decentration parameters x_P and y_P represent the position of the pupil center relative to the center of the limbus.

center relative to the center of the limbus. In our model, the pupil is constrained to be coplanar with the limbus. While this is probably not a bad approximation, some illustrations show the central portion of the iris pushed forward by the lens, and it is possible that an additional parameter may be necessary to account for this.

While we will in general be interested in varying the parameters to fit a particular subject, for the images rendered in figures 1-3 we used the values indicated in table 1. Also indicated are the corresponding values for the Gullstrand and Le Grand simplified eye models, as reported in [Smith and Atchison 1997]. Note that these models are not complete opto-mechanical models, but only describe the refracting surfaces and image formation, and do not include specification of the limbus, or the position of the eye's center of rotation. The overall eye length l_E, however, is specified, so that within our framework it is possible to specify a family of eyes for which $r_E + h_C = l_E - r_C$, each with a different limbus size governed by equation 1. The column in table 1 labelled "Cornsweet" gives values from Cornsweet and Crane's paper describing the dual Purkinje image eye-tracker [Cornsweet and Crane 1973]. In their paper, the eyeball radius r_E is not given directly, but can be inferred from other dimensions. These are not all mutually consistent; the first value (10.5) is obtained by subtracting the indicated distance from the corneal pole to the center of rotation from the stated overall eye length l_E, while the second value (10.2) is obtained from indicated values $l_E - r_C - h_C$. In addition, the stated values for r_C, h_C and r_L do not satisfy equations 1 and 2, which is easily understood when one notices that the eyeball in their figure is not drawn as a circle, but appears squashed in front: relative the center of rotation, r_E appears constant for the back of the eye but appears to decrease near the limbus. In the context of the present work, this is not a problem, because we are not concerned with retinal image formation, and so the overall eye length is irrelevant. The appearance of the anterior structures of such an eye will be equivalent to our model eye having r_E consistent with r_C and h_C.

2.1 Corneal sampling

The corneal surface is represented by a two-dimensional array of 3-D points. We desire that this array cover the cornea completely and sample it uniformly. This is would trivial if the cornea were

param.	default	Gullstrand	Le Grand	Cornsweet
r_C	7.25	7.8	8	7.8
r_E	10			10.5 (10.2)
h_C	4.5			6
r_P	3.3			
r_L	6.59			6.25

Table 1: Default eye model parameters used in the present simulation, along with comparable values from various schematic eyes.

a flat rectangular patch, but as it is a spherical cap, we have a bit more work to do. While this problem may be viewed as a mundane technicality, we present it in some detail because it affects the performance of our algorithms for modelling refraction and reflection, to be described in the next section.

We begin by generating an array of samples on a circular disk of unit radius. We imagine the unit circle with an inscribed square grid pattern, and "inflating" the grid pattern until it fills the circular region. We present two alternative schemes to achieve this, with slightly different properties. In each case, the result is to map the edges of the square to the circular arcs having the same endpoints.

The first method consists of radially expanding the pattern by a factor which depends on the angle at each point. Let N_C represent the linear dimension of the sample array, and index the array using variables i and j which each range from 0 to $N_C - 1$. We define u_{ij} and v_{ij} to be the coordinates of the undistorted square array:

$$u_{ij} = \sqrt{2}\left(\frac{j}{N_C - 1} - \frac{1}{2}\right), \tag{3}$$

$$v_{ij} = \sqrt{2}\left(\frac{i}{N_C - 1} - \frac{1}{2}\right). \tag{4}$$

First we transform to polar coordinates:

$$r_{ij} = \sqrt{u_{ij}^2 + v_{ij}^2}, \tag{5}$$

$$\phi_{ij} = \tan^{-1}(v_{ij}, u_{ij}). \tag{6}$$

Now we wish to increase r_{ij}, by the factor by which the circle radius (1) exceeds the distance from the center to the edge of the square in the direction ϕ_{ij}. For $-\pi/4 \le \phi_{ij} \le \pi/4$ (corresponding to the right side of the square), this factor is $\sqrt{2}\cos\phi_{ij}$, while for the $\pi/4 \le \phi_{ij} \le 3\pi/4$ (the top of the square), it is $\sqrt{2}\cos(\phi_{ij} - \pi/2)$. When all four quadrants are considered, we obtain

$$x_{ij} = r_{ij}\ \cos\phi_{ij}\ \mathrm{f}(\phi_{ij}), \tag{7}$$

$$y_{ij} = r_{ij}\ \sin\phi_{ij}\ \mathrm{f}(\phi_{ij}), \tag{8}$$

$$\mathrm{f}(\phi) = \sqrt{2}\ \min\left(|\cos\phi|, \left|\cos\left(\phi - \frac{\pi}{2}\right)\right|\right) \tag{9}$$

The resulting array of disk samples is shown in figure 4.

While the mesh shown in figure 4 achieves our goal of attaining approximately uniform sampling of the disk, we note that the orientation of the gridlines changes discontinuously as they cross the diagonals. This may be undesireable in cases where we want to use partial derivatives with respect to the array dimensions, as in a gradient descent search. Therefore we now present an alternative scheme which achieves continuity at the expense of reduced sampling uniformity. As in the first scheme, the first row of sample points samples the circular arc bounded by the two upper vertices of

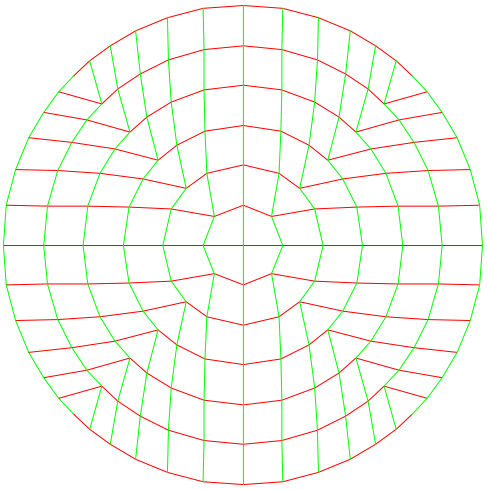

Figure 4: Sampling grid on the unit disk.

the square. But, unlike the first scheme, now we make the samples on this arc uniformly spaced:

$$x_{0j} = \cos\left(\frac{3\pi}{4} - \frac{j\pi}{2(N_C - 1)}\right), \tag{10}$$

$$y_{0j} = \sin\left(\frac{3\pi}{4} - \frac{j\pi}{2(N_C - 1)}\right). \tag{11}$$

Similarly, the left-most column of samples are spaced uniformly on the left-hand arc of the circle:

$$x_{i0} = \cos\left(\frac{3\pi}{4} + \frac{i\pi}{2(N_C - 1)}\right), \tag{12}$$

$$y_{i0} = \sin\left(\frac{3\pi}{4} + \frac{i\pi}{2(N_C - 1)}\right). \tag{13}$$

Arbitrary rows of sample points are generated as linear combinations of the top row and the horizontal diameter of the circle. We define a set of uniform samples on this diameter:

$$s_j = -1 + \frac{2j}{N_C - 1}. \tag{14}$$

Note that the samples on the diameter will only belong to the array in the case where N_C is odd. An arbitrary sample point (x_{ij}, y_{ij}) will be expressed as a mixture of the diameter and either the top row (for $i \le N_C/2$) or the bottom row (for $i > N_C/2$). We first consider the case $i \le N_C/2$:

$$x_{ij} = \alpha_i x_{0j} + \beta_i s_j \tag{15}$$

$$y_{ij} = \alpha_i y_{0j} \tag{16}$$

Now we must obtain the weights α_i and β_i. To satisfy equations 12 and 13, the following system of equations must be satisfied:

$$\begin{pmatrix} x_{i0} \\ y_{i0} \end{pmatrix} = \begin{vmatrix} x_{00} & s_0 \\ y_{00} & 0 \end{vmatrix} \begin{pmatrix} \alpha_i \\ \beta_i \end{pmatrix}. \tag{17}$$

The weights α_i and β_i are computed by inverting the matrix, and applying the inverse to the left-hand points given by equations 12 and 13 for $0 \le i \le N_C/2$. The rows below the diameter are obtained from upper rows by inverting the y coordinate. The resulting sampling of the unit disk is shown in figure 5.

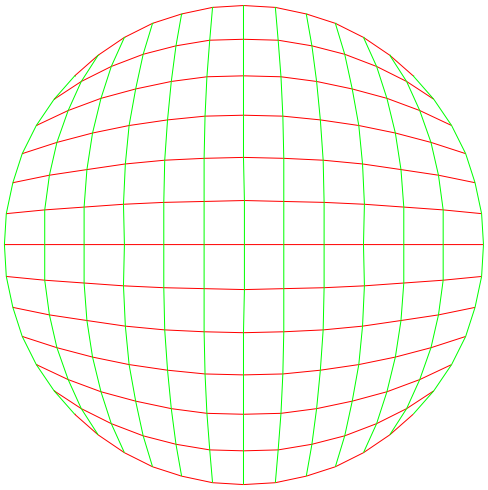

Figure 5: An alternate sampling of the unit disk. The orientation of the local coordinate system does not have the discontinuites seen at the diagonals in figure 4, but achieves this by sacrificing uniformity of sample spacing at the "corners."

We have presented above two methods for obtaining an approximately uniform sampling of a circular disk by a two-dimensional array of points. Now we need to map the points on the unit disk to the spherical cap representing the cornea. At this point we could simply scale up all the x and y coordinates by $\mathrm{r_L}$, and compute the corresponding z such that the resulting 3-vector has a length of $\mathrm{r_C}$. While this would result in apparently uniform sampling in a frontal view, the edges of the cornea would be undersampled in the angular domain. To rectify this, we apply a radial transformation to our disk samples. First we transform to polar coordinates:

$$r_{ij} = \sqrt{x_{ij}^2 + y_{ij}^2}, \tag{18}$$

$$\phi_{ij} = \tan^{-1}(y_{ij}, x_{ij}). \tag{19}$$

We then compute new coordinates in which we treat r_{ij} as a desired slant angle, and scale up by $\mathrm{r_L}$:

$$x_{ij} = \mathrm{r_L} \frac{\sin(\theta_\mathrm{L} r_{ij})}{\sin\theta_\mathrm{L}} \cos\phi_{ij}, \tag{20}$$

$$y_{ij} = \mathrm{r_L} \frac{\sin(\theta_\mathrm{L} r_{ij})}{\sin\theta_\mathrm{L}} \sin\phi_{ij}, \tag{21}$$

$$\theta_\mathrm{L} = \tan^{-1}(\frac{\mathrm{r_L}}{\mathrm{r_C}}). \tag{22}$$

We compute z_{ij} so that the final point lies on the corneal sphere:

$$z_{ij} = -(\mathrm{h_C} + \sqrt{\mathrm{r_C^2} - x_{ij}^2 - y_{ij}^2}). \tag{23}$$

Note that we have made the z values negative, so the unrotated model points in the negative z direction. The resulting meshes obtained when $\mathrm{N_C} = 13$ are shown in figures 7 and 6.

3 Rendering the Model

We simulate image formation using a simple pinhole camera model. The eyeball center is placed at the origin of the coordinate system (x right, y up, and z away). We adopt a left-handed coordinate system, so that when we view the model in the positive z direction

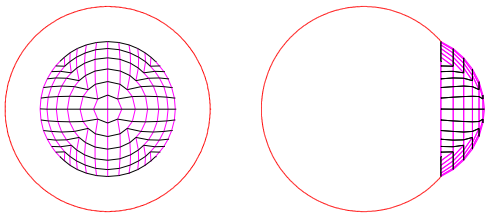

Figure 6: The corneal surface is sampled by a two-dimensional array of points. The rectangular sampling array attempts to sample the corneal surface uniformly. For the purposes of illustration, the mesh is shown here for $\mathrm{N_C} = 13$, but for the simulations reported here a value of 512 was typically used.

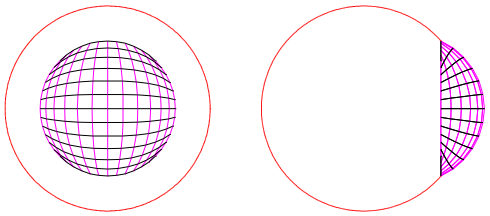

Figure 7: An alternate mesh for sampling the corneal surface. Compared to the mesh depicted in figure 6, the mesh depicted here samples more uniformly in the "corners," but the local gridlines change orientation discontinuously along the diagonals.

with the positive y axis pointing up, the positive x axis points to the right. The camera is located on the negative z axis at distance d, pointing towards the origin. For any point $\mathbf{p} = (x, y, z)$, we compute normalized image coordinates (u, v) as follows:

$$u = \frac{1}{2} + \frac{fx}{2(z+\mathrm{d})r_s}, \tag{24}$$

and

$$v = \frac{1}{2} + \frac{fy}{2(z+\mathrm{d})r_s} \tag{25}$$

where f represents the focal length and r_s is the "radius" of the sensor chip. In addition, we calculate images using orthographic projection, $u = x$, $v = y$. Orthographic projection represents the limit as the distance from the eye to the camera and light source increases; unless otherwise stated, calculations reported for perspective projection were made using a camera positioned at (0,0,-75), looking in the positive z direction at an eye located at the origin. The coordinates represent distance in millimeters.

To render the limbus, we tabulate an array of 3-D points corresponding to the limbus, and project as described above. For visualization purposes, we also render the outline of the eyeball, which is slightly more complicated because the outline is not a fixed locus which moves with the eyeball. For orthographic projection, we simply draw a circle of radius $\mathrm{r_E}$ centered at the projection of the eyeball center. For perspective projection, we first calculate the radius of the circle of tangency between the eyeball sphere and the viewing cone with apex at the camera center, and then generate a list of points on this circle as trigonometric combinations of a pair of unit vectors normal to the viewing axis. We use a similar procedure to render the occluding edge of the cornea, when appropriate.

To render the image of the pupil, we begin by tabulating an array of points on the inner margin of the iris. For each iris point, we must find the corresponding point in the cornea where the image will appear. We find this point by exploiting Fermat's principle of

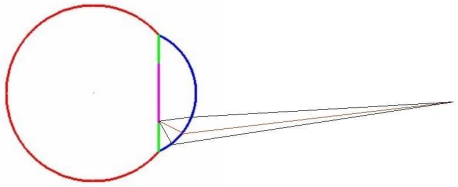

Figure 8: For each pupil point, the apparent position in the cornea is determined by finding the corneal point for which the total optical distance from the pupil to the camera is minimal.

least action, which states that the path taken by a ray of light minimizes the optical distance considered over all possible paths. For a given point in the plane of the iris, we compute the optical distance between the point and the camera center for a set of samples on the corneal surface, and then choose the sample for which the distance is minimal. (The position of the glint produced by an illuminator can be computed in the same way: the locus of points in three dimensions which form the boundary of the illuminator assumes the role of the pupil locus.) Note that in computing the optical distances, the physical distance from a pupil point to each corneal point must be multiplied by the index of refraction (1.33). This method requires dense sampling of the corneal surface to obtain accurate results, although in principle we should be able to obtain "sub-pixel" estimates through interpolation. For the initial frame, we compute the distance over the entire cornea, and scan the array of distances to find the global minimum. For subsequent points which traverse a continuous locus (such as the pupil margin), we can reduce the amount of computation needed by only calculating the optical distances for a small sub-array of the corneal samples. In our algorithm, the size of the subwindow is a tuneable parameter: expressed as a number of samples, it must be increased as the density of corneal samples is increased, but can be decreased as position change between successive samples in the iris plane decreases (such as by increasing the number of pupil samples). The subwindow is centered on the corneal sample corresponding to the solution for the previous pupil sample; if the minimum value falls on the edge of the subwindow, it is repositioned up to 5 times to ensure that the global minimum is found. It is for this reason that we went to so much trouble to ensure uniform sampling of the corneal surface; an early implementation used polar coordinates to parameterize the corneal surface, producing an unusually high sampling density in the vicinity of the corneal pole, which broke our search algorithm.

The circularly symmetric model we have described above has an obvious axis, namely the axis of symmetry which is the line line joining the eyeball center with the center of curvature of the cornea. In our model, this axis (which we shall refer to as the *corneal axis*) coincides with the optical axis and the pupillary axis (terms which shall be defined as needed). This axis does not generally correspond to the line-of-sight or direction of gaze, however. The sampling of visual space by the photoreceptor array is extremely nonuniform, and there is a specialized region called the *fovea* which has the highest sampling density of receptors. When a person "looks at" a target, they move their eye so as to cause the image of the target to fall on the fovea. The line of sight defined this way does not generally coincide with the optical axis but deviates by an angle, often represented by the symbol κ. Determination of angle κ requires either a cooperating subject reporting when they are fixating directly "on" a reference target, or a stimulus so powerful that there can be no doubt that any subject must fixate it. For the remainder of this paper, we will ignore this complication, and discuss the corneal axis as if it were the line-of-sight.

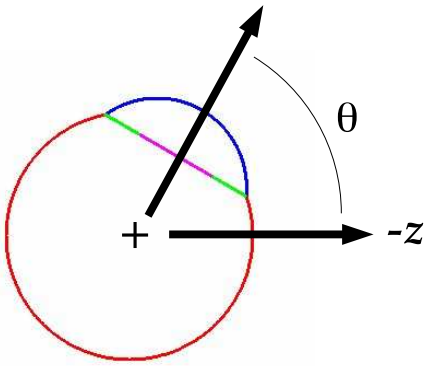

Figure 9: Slant angle θ is subtended by the model axis and the negative z axis.

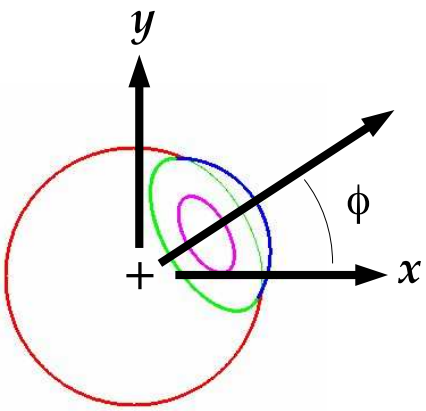

Figure 10: Tilt angle ϕ is subtended by the projection of the model axis to the xy plane and the positive x axis, when viewed in the positive z direction.

We parameterize the orientation of the model by two angles: the slant θ describes the angle the corneal axis makes with the negative z axis (figure 9, while the tilt ϕ describes the angle the projection of the axis into the xy plane makes with the positive x axis, when viewed in the positive z direction (figure 10). Figure 11 shows the model rendered for a variety of gaze directions. At the center, we see the model pointing directly toward the camera. In this view, and only this view, the pupil and limbus appear as concentric circles. For the case of orthographic projection (as in figure 11), the shape of the pupil and limbus depend only on the slant, while their orientations in the image depend only on tilt. (For perspective projection, these relations remain approximately true when the camera is pointed at the eye.) Note that, for any of the other views corresponding to a non-zero slant angle, the limbus and the pupil appear as ellipses whose minor axes (possibly extended) passes through the projection of the eyeball center; the various views are positioned in the figure so that these lines all meet in the center. In figure 11, the outer ring of eyes all have a slant of 60 $^\circ$ and tilts which span the range from 0 $^\circ$ to 360 $^\circ$.

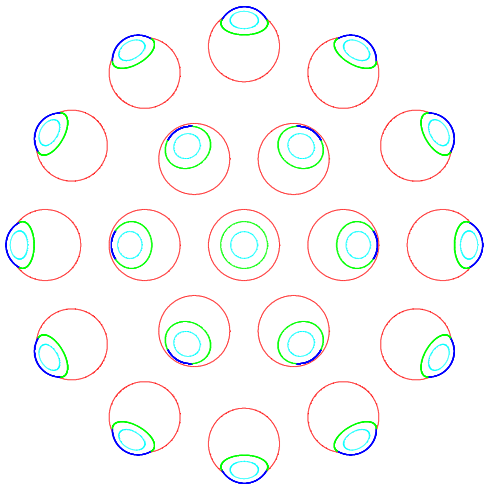

Figure 11: Eye model rendered using orthographic projection for slants 0 ° (center), 30 ° , and 60 ° .

4 Empirical relations

In the preceding sections, we have described the geometric model of the eye, and our method for determining the positions of various features. In this section we explore how the feature positions and shapes covary as the eye rotates. For the purpose of gaze estimation, this may be our final goal; we may have no interest in the radius of curvature of the cornea, but we do want to know how much motion of the pupil and glint relative to the limbus should be expected as the eye rotates.

In our model, the limbus is modelled as a circle; thus, it's projection in the image will always be an ellipse, whether rendered in orthographic or perspective projection. The image of the pupil, however, is not in general perfectly elliptical after refraction by the cornea. For moderate gaze angles, however, it will be very nearly elliptical, and we will summarize its shape by fitting an ellipse to the set of points we obtain after mapping the pupil. The results reported here were obtained by fitting general conic of the form

$$ax^2 + bxy + cy^2 + dx + ey + f = 0. \tag{26}$$

A fit minimizing the algebraic error with respect to N points is obtained by constructing an Nx6 matrix of terms made from the point coordinates, which when multiplied by the vector of coefficients results in a vector of N zeros:

$$\begin{pmatrix} 0_1 \\ \vdots \\ 0_N \end{pmatrix} = \begin{vmatrix} x_1^2 & x_1y_1 & y_1^2 & x_1 & y_1 & 1 \\ \vdots & & & & & \\ x_N^2 & x_Ny_N & y_N^2 & x_N & y_N & 1 \end{vmatrix} \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \end{pmatrix} \tag{27}$$

We compute the singular value decomposition of the matrix of data terms, using the routine svdcmp from the Numerical Recipes library [Press et al. 1992]. The desired coefficients are obtained from the eigenvector with the smallest eigenvalue. This procedure could be improved by minimizing the geometric error instead of the algebraic error, and by insuring that the points to be fit are uniformly spaced in the image, rather than on the underlying structures. An ellipse-specific fitting algorithm [Pilu et al. 1996] is reported to give better results for noisy or incomplete data; however, we assume that, in the present case (where the curves are close to being perfectly elliptical and are densely sampled with little noise), incorporation of one or more of these niceties would not appreciably affect the results.

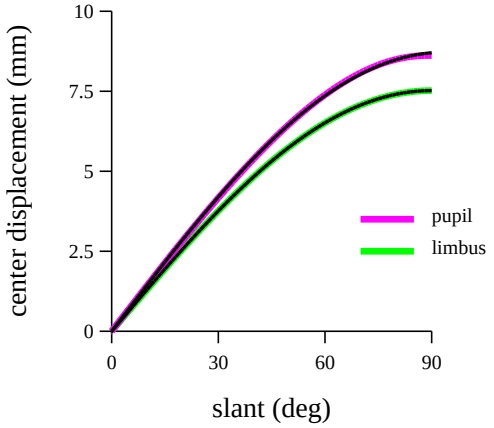

Figure 12: Position of the center of the pupil (upper curve) and limbus (lower curve) as a function of slant under orthographic projection, expressed in millimeters. The curve for the limbus is exactly described by a sinusoid; the curve for the pupil is more complicated due to the effect of refraction by the cornea, but is well-approximated by a stretched sinusoid (see text).

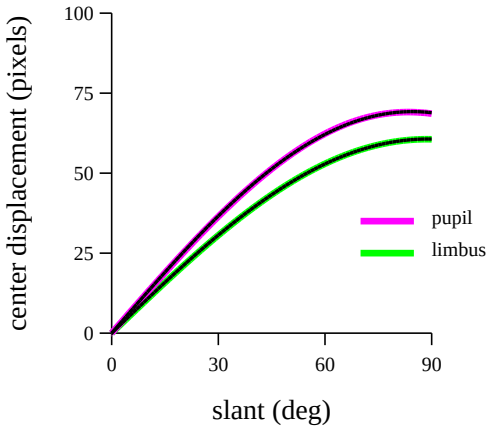

Figure 13: Similar to figure 12, but for perspective projection. Vertical units are now in sensor pixels.

Figure 12 shows the computed position of the center of the pupil and limbus under orthographic projection, as a function of slant, sampled in 1 degree increments from 0 to 90. For the limbus, the relation is expressed exactly by:

$$\mathrm{f}(\theta) = \mathrm{h_L} \sin(\theta). \tag{28}$$

For the pupil, the data plotted are derived from the ellipse fits to the computed pupil images. It can be seen in figure 12 that, at a first glance the pupil appear approximately sinusoidal in form, like the limbus, but with slightly higher amplitude. We fit the pupil data with a stretched version of the limbus function:

$$\mathrm{f}(\theta) = \alpha \mathrm{h_L} \sin(\beta\theta). \tag{29}$$

In this equation, α represents the vertical stretching factor, while β represents the horizontal factor. In figure 12, the data are overlaid with a descriptive fit obtained by application of equation 29, with parameters $\alpha = 1.16$, $\beta = 0.96$. These values were obtained

using the STEPIT optimization procedure [Chandler 1969]; the optimization sought to minimize the total squared error e_{fit} between the observed data $\mathbf{d}(\theta)$ (shown above in figure 12) at a set of slants $\{\theta_i\}$ and the fit:

$$e_{fit} = \sum (d(\theta_i) - \alpha h_L \sin(\beta \theta_i))^2. \quad (30)$$

Figure 13 shows similar results for perspective projection by a camera located 75 mm from the eyeball center.

The results shown in figures 12 and 13 were obtained using the default model parameters (see table 1). These simulations were repeated for different values of r_C, from 7 to 8.5 mm in steps of 0.1 mm. Data sets similar to those shown in figures 12 and 13 were tabulated, sampling slant in 5 degree steps. The data were again fit with the descriptive model (equation 29); the fit parameters are shown in figures 14 and 15.

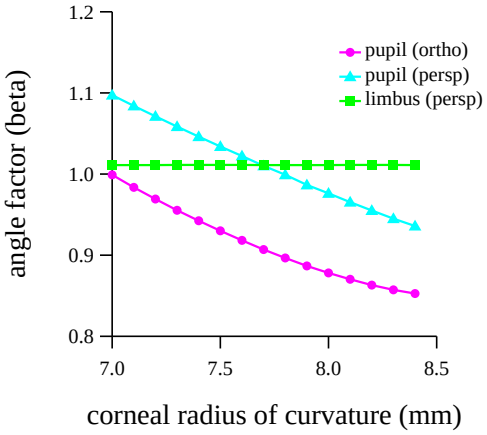

Figure 14: .

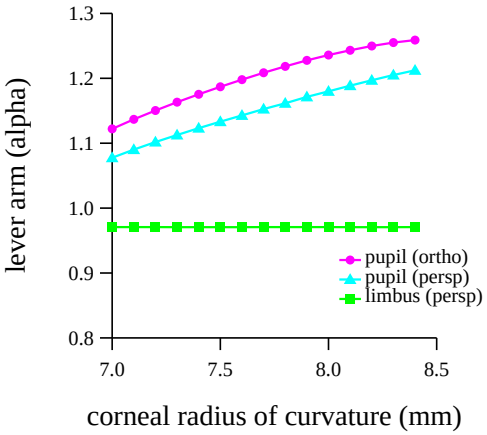

Figure 15: .

Next we consider the shape of the pupil and limbus. Because the image loci are either exactly elliptical, or well-fit by an ellipse, we describe the shape using two parameters: the orientation (of the major axis) and the aspect ratio (minor axis length divided by major axis length). We select this definition of aspect ratio because the resulting dependence on slant is particularly simple for the unrefracted limbus under orthographic projection: it is simply $\cos\theta$. Figures 16 and 17 show empirically tabulated results for orthographic and perspective projection, respectively. Again, we introduce a descriptive model to fit the observed data by stretching the slant axis:

$$f(\theta) = \cos(\lambda\theta). \quad (31)$$

Figure 18 shows the values of λ obtained for various values of r_C.

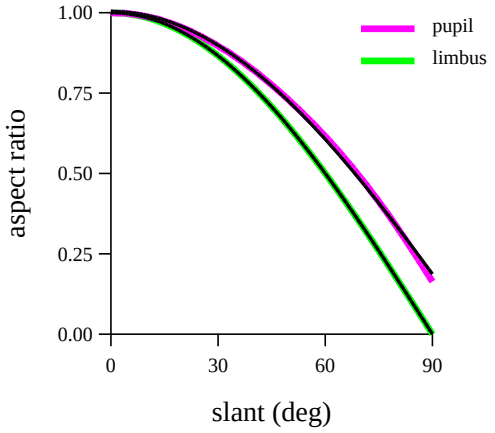

Figure 16: Length of the minor axis of the best-fitting ellipse, expressed as a fraction of the major axis length, for the limbus and pupil viewed with orthographic projection.

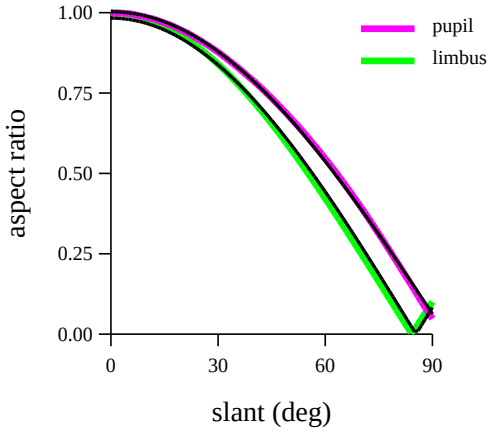

Figure 17: Same as figure 16, but for perspective projection with a camera located 75 mm from the eyeball center.

5 Parameter estimation

In the previous section, we explored how the observable features would be expected to vary with changes in pose and structural parameters. In practice, we need to solve the inverse problem: given a set of observations, we wish to recover estimates of the structural parameters (fixed for the entire set) and pose parameters (independent estimates for each image). Beymer and Flickerner estimated the structural parameters from a set of 18 stereo pairs using a nonlinear optimization routine[Beymer and Flickner 2002]. Obviously, we can improve the accuracy of our estimates by including more images, but each added image brings with it additional unknown pose parameters, and corresponding increase in the computational cost of performing the optimization.

In some cases, we may be interested in data sets for which the features cannot be reliably located by automatic procedures. One such data set, collected outdoors with shadows cast by the sun, was the

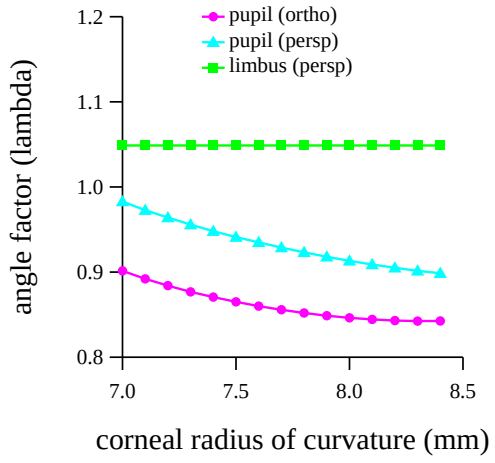

Figure 18: Horizontal stretch parameters for model fits to data as shown in figures 16 and 17.

motivation for the present work. When the features cannot be located automatically (i.e., the program has not been written), it may still be possible for a human to visually locate at least some of the features. To make this process efficient, we do not want the operator to label the pupil and limbus separately and independent, rather we wish to provide him or her with a set of knobs corresponding to the pose parameters (for a head-mounted camera, slant θ, tilt ϕ and pupil radius r_P), which move all the observable features together. To do this for the first image requires arbitrarily assigning default values to the intrinsic structural parameters; if they are wrong, a good fit to the first image can often be obtained through compensating adjustments to the pose parameters, but eventually an inconsistency will arise. We would like to find an efficient method for determination of the intrinsic parameters from a small set of images.

We consider the case of a head-mounted camera, and assume a fixed relation between the camera and the eyeball center. One of the simplest parameters to obtain is the limbus radius r_L, provided we have a frame in which the limbus is mostly visible, and not too slanted. We fit an ellipse to the visible limbus points, and take half the length of the major axis. This value will be in pixels. Up until now, we have expressed all quantities in their physical units (mm), but for the time being we will find it convenient to work in pixels, understanding that they are related to physical units by a (possibly unknown) scale factor which depends on the sensor resolution, the lens focal length, and the camera distance.

We can determine the image location of the projection of the eye center from two images, simply by finding an elliptical feature (pupil or limbus) in each images, and then finding the intersection of the two minor axes. This point is defined as long as the eye slant is not zero (ellipse orientation defined) and the two ellipses have different tilts (minor axes not parallel). For this purpose, it is clearly beneficial to select images with as large a slant as possible, to reduce uncertainty in the direction of the minor axis. Also, it is beneficial to choose a pair for which the two ellipses have more-or-less orthogonal orientations, to reduce uncertainty in the location of the intersection point. While ellipses from two images determine a unique center point, when more images are available we can obtain a least squares solution; by representing points and lines using homogenous coordinates (see, for example [Hartley and Zisserman 2003]), the problem of finding the best intersection of a group of lines can be solved in the same manner as the dual problem of fitting a line to a set of points. We construct a matrix of the line coordinates, and seek a point (x_0, y_0) for which

$$\begin{pmatrix} 0_1 \\ \vdots \\ 0_N \end{pmatrix} = \begin{vmatrix} l_{11} & l_{12} & l_{13} \\ \vdots & & \\ l_{N1} & l_{N2} & l_{N3} \end{vmatrix} \begin{pmatrix} x_0 \\ y_0 \\ 1 \end{pmatrix}. \qquad (32)$$

As in our solution for the best fitting ellipse parameters (equation 27), the desired result is using the SVD to obtain the eigenvector of the matrix with the smallest eigenvalue.

Once the center has been determined, we are in a position to determine the distance of the plane of the limbus h_L from the center of rotation. For an image a moderately large eye slant, we determine the apparent aspect ratio of the limbus, which we will call a. Assuming orthographic projection, it follows that the slant θ is simply $\cos^{-1} a$. For perspective projection, if we know the imaging geometry we can construct a set of predictions (as in figure 17), determine the parameter λ, and obtain the slant θ as $(\cos^{-1} a)/\lambda$. With an estimate of the slant, we are now in a position to estimate h_L: for orthographic projection, the displacement of the ellipse from the center is given by $h_L \sin\theta$, as in figure 12. So for a measured displacement d, h_L can be computed as $d/\sin\theta$.

We used the position and aspect ration limbus to determine h_L because they are unaffected by the curvature of the cornea. If the corneal radius were known we could equally well have used the pupil, after making appropriate corrections. But if we determine h_L as described above, then we can now determine the radius of curvature of the cornea from an image in which the limbus and pupil are both visible. From the limbus, we can determine the slant θ as described above. Figures 14 and 15 tell us the parameters necessary to predict the variation of pupil position with slant, as in figures 12 and 13. We choose the corneal radius r_C so that the displacement predicted for slant θ matches the observed displacement. This might be done using an iterative search procedure, or directly by fitting lines or low-order polynomials to the curves in figures 14 and 15.

Note that at this point, we have not used the apparent aspect ratio of the pupil, but we have determined all of the intrinsic structural parameters of the model. Comparison of the predicted and observed pupil aspect ratios provides a consistency check.

The simple procedure outlined above is intended to be more an example of the general approach than a definitive method. The alert reader will have noticed that in the preceding section we made no mention of the glint. The position of the glint provides independent evidence concerning the shape of the cornea. For a head-mounted camera system, incorporation of measurements of glint position offer a consistency check, enabling detection of slippage of the head mount. For a remote system, in which the eyeball center cannot be assumed to be fixed, measurement of the position of one or more glints becomes a necessity.

In the absence of pupil decentration, the model predicts that ellipses fit to the pupil and limbus will have the same orientation. Occasionally, we observe images for which this does not seem to be true. One possibility is that our assumption of a circular pupil is false. This could be tested by inspection of images with slants near zero. Alternatively, mis-aligned ellipses might arise from a de-centered pupil combined with an eye rotation in a different direction.

6 Conclusion

We have described a simple model for predicting the appearance of the anterior structures of the eye. By exploring the behavior of the model, we are able to develop simple parametric approximations describing the variation of observable image parameters with gaze

angle. When relative motion between the head and the camera is eliminated (as with a head-mounted camera), the model parameters can be determined from as few as two images. Incorporation of the model into a manual eye labelling tool increases operator efficiency, and enables gaze estimates for images in which many of the features not seen.

References

BEYMER, D., AND FLICKNER, M. 2002. Eye gaze gracking using an active stereo head. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, IEEE, 451–458.

CARPENTER, R. H. S. 1977. *Movements of the eyes*. Pion Limited.

CHANDLER, J. D. 1969. Subroutine STEPIT: Finds local minima of a smooth function of several parameters. *Behavioral Science 14*, 81–82.

CORNSWEET, T. N., AND CRANE, H. D. 1973. Accurate two-dimensional eye tracker using first and fourth Purkinje images. *J. Opt. Soc. Am. 63*, 8, 921–928.

CRANE, H. D., AND STEELE, C. M. 1969. Accurate three-dimensional eyetracker. *Applied Optics 17*, 691–82.

FRY, G. A., AND HILL, W. W. 1962. The center of rotation of the eye. *Am. J. Optom. 39*, 581–595.

HARTLEY, R., AND ZISSERMAN, A. 2003. *Multiple view geometry in computer vision*, 2nd ed. Cambridge University Press.

OHNO, T., MUKAWA, N., AND YOSHIKAWA, A. 2002. Freegaze: a gaze tracking system for everyday gaze interaction. In *ACM Eye Tracking Research and Applications*, ACM, 125–132.

PILU, M., FITZGIBBON, A. W., AND FISHER, R. B. 1996. Ellipse-specific direct least-square fitting. In *Proc. ICIP*, IEEE, 001–002.

PRESS, W. H., TEUKOLSKY, S. A., VETTERLING, W. T., AND FLANNERY, B. P. 1992. *Numerical Recipes in C : the art of scientific computing*, 2nd ed. Cambridge University Press.

RASHBASS, C., AND WESTHEIMER, G. 1960. Recording rotational eye movements independently of lateral displacements. *J. Opt. Soc. Am. 50*, 512.

SMITH, G., AND ATCHISON, D. A. 1997. *The eye and visual optical instruments*. Cambridge University Press.