MouseEyeTracker is Python software for acquisition and analysis of video data of the mouse eye.

Dependencies

- Python 2 or 3 (image acquisition from camera tested with PC running Python 2.7; analysis of saved data tested with PC running Python 2.7 and Mac running Python 3.4)
- Numpy, Scipy
- PyQt4
- PyQtGraph
- OpenCV3
- h5py (if saving acquired video data or opening saved data files)
- Pymba (if acquiring data with AVT camera)
- PyDAQmx and nidaq.py (if using NIDAQ board for digital input/output)

Hardware

Currently we use an Allied Vision GigE GC660 IR camera (659W x 494H pixels, max 119 frames/s at full resolution, grayscale) with a Navitar Zoom-6000 lens (adapter mount 1-6010 and body tube 1-6265). Four IR LEDs (Thorlabs, LED851L, 8060-2) spaced equally around the lens (parallel to the video image) are used to determine the center of the corneal surface closest to the camera. Additional IR LEDs (Mightex SLS-0208-A) are used for illumination. A single-axis manipulator (Thorlabs, MT1/M) moves the camera parallel to the image axis for calibrating mm/pixel. A National Instruments board (USB-6009) is used to trigger data acquisition and/or broadcast timing of saved frames.

Starting MouseEyeTracker

From the command line:

```
python MouseEyeTracker.py
```

From a Python console:

```
import MouseEyeTracker
MouseEyeTracker.start()
```


Summary of keyboard shortcuts

Key	Action
> or <	Navigate video one frame at a time; add ctrl key to move several frames at a time
arrow keys	Context dependent: move ROI, or move (left/right) or flip (up/down) saccade marker
+ or -	Resize ROI
n	Set pupil tracking data for current frame to NaN
ctrl n	Toggle 'Set Data to NaN' on or off
escape	Toggle 'Stop Tracking' on or off
space bar	Set pupil area/position plots to full time range
f	Find saccades

delete	Delete currently selected saccade marker (use right click to select the saccade)
ctrl delete	Delete all saccade markers

Acquiring data

To acquire images from a camera, select 'Use Camera' from the camera menu and press the Start Video button. Use Camera Menu > Settings to change the buffer size (number of frames in camera memory), spatial binning of pixels, exposure (fraction of frame interval), or frame rate.

To save images, check the Save Video Data checkbox. Alternatively, saving can be triggered by a digital input to a NIDAQ board after selecting Camera Menu > NIDAQ IO > Use Save Trigger (NIDAQ Input P0.0). To broadcast the timing of saved frames to other devices, select Camera Menu > NIDAQ IO > Signal Saved Frames (NIDAQ Output P1.0).

Video data is saved to a hdf5 file. Each frame is a dataset named by the frame number. The file has attributes 'frameRate', 'numFrames', and 'mmPerPixel'. Frames have the attribute 'acquisitionTime', which is a timestamp in seconds from the camera's internal clock.

Analyzing data (pupil tracking and saccade detection)

To analyze data from an hdf5 file saved by MouseEyeTracker or a video file, select Open > Frame Data from the file menu. Press the Start Video button to play the video. Press the '>' and '<' keys to navigate the video one frame at a time, or use 'control' plus '>' or '<' to navigate several frames at a time.

If the pupil is detected (see 'Tracking the pupil'), the pupil area, horizontal position, and vertical position plots will continuously update. Press the 'n' key or 'control+n' to set these values to NaN for the current frame or all subsequently viewed frames, respectfully. To move the video to a particular frame, drag the vertical red line in one of the data plots or use the frame number box below these plots (type in in frame number or press the arrows). Pressing the spacebar resets the x range to the full video duration.

Frames can be analyzed more quickly by selecting Analysis Menu > Analyze All Frames to analyze all frames from the current frame to the end of the video without updating the display (until the final frame).

Once pupil position is determined for several or all frames, find horizontal saccades using Analysis Menu > Saccades > Find (or press the 'f' key). Double-click near the horizontal pupil position trace to manually mark a saccade. Right click near any saccade marker (blue triangle) to select that saccade, and use the arrow keys to move the event marker in time or the delete key to remove that event. Press control-delete to clear all marked saccades.

Use File Menu > Save > Data to save the pupil area/position data, reflection center, frame times (if available), and saccade times to an hdf5, npz, or mat file. This data can be reloaded with File > Open > > Tracking Data. Use File Menu > Save > Movie or Annotated Movie to save a movie or an annotated movie showing pupil and reflection markers.

Setting a region of interest (ROI)

Using an ROI increases performance and reduces file size. Pressing the Set ROI button resets the ROI to the full image size. Drag the bottom-right corner of the ROI box to resize the ROI. Click within the ROI box and drag the mouse to re-position the ROI. Alternatively, use the + and – keys the resize the ROI and the arrow keys to change its position. Pressing the Set ROI button again (or any other button) causes the ROI to fill the image window.

Tracking the pupil

MouseEyeTracker has three methods for tracking the pupil: 'Starburst', 'Line', and 'Gradients' (see Track Menu). Starburst is generally most useful. Using this method, press the Find Pupil button and double click on the image near the center of the pupil. The clicked point serves as the pupil center 'seed' that MouseEyeTracker uses to find the pupil edges, which are fit to an ellipse. The center of the ellipse is the pupil center and serves as the seed for the next frame. The pupil center seed can be reset at any time (including while the Start Video button is depressed) by double clicking on the image.

The pixel intensity along lines at different angles from the calculated pupil center ('radial profiles') are shown in the top plot. Drag the horizontal red line in this plot up or down to change the pupil edge threshold. To automatically adjust the pupil edge threshold for overall changes in image intensity, select Track Menu > Adaptive Threshold.

The middle plot shows the running sum of the radial profiles up to the 'minimum number of pixels threshold'. This threshold determines the minimum consecutive number of pixels that must be above the pupil edge threshold. Drag the horizontal red line in this plot to change the minimum number of pixels threshold.

The bottom plot shows the distance of the pupil edge points from the center of the ellipse. Points above or below the horizontal red lines in this plot are excluded for fitting the ellipse. Dragging these lines adjusts their position relative to the edge distance mean and standard deviation.

Use Track Menu > Circularity to adjust the circularity threshold. Circularity is the ratio between the lengths of the major and minor axes of the pupil fit ellipse. Pupil fits with circularity below the circularity threshold are rejected (data is set to NaN).

The 'Line' method of for pupil tracking searches for edges along a horizontal line crossing the pupil. This can be useful for detecting horizontal saccades when the pupil is very large (pupil area and vertical position are not determined with this method). Press the Find Pupil button and position/resize the ROI rectangle (as described under 'Setting a region of interest (ROI)') such that it crosses one side of the pupil (vertical columns of pixels are averaged). Set Track Menu > Line Origin to left or right depending on the side of the pupil at which the pupil ROI is placed. Adjust the edge threshold as described for the Starburst method of pupil tracking.

Find the function findPupilWithGradients in MouseEyeTracker.py to learn more about this pupil tracking method. It is slower and less accurate than the Starburst method, but can be useful for some images with low resolution and or poor contrast. If this method is selected, press the Find Pupil button and move/resize the pupil ROI to cover the region of possible pupil positions. To speed up tracking, use Track Menu > Gradient Downsample to reduce the fraction of pixels used for tracking.

Image masking during pupil tracking

For pupil detection, specific regions of the image (LED reflections, for example) can be masked. Press the Set Mask button and double-click on the image near the center of a region to mask. A small ROI box should appear; adjust the size and position of this box as described under 'Setting a region of interest (ROI)'. Double-click on the image again to set additional mask regions. Press the 'delete' key to delete a mask region. Pixel intensities within masked regions are set to 0 for the purposes of pupil detection when the Use Masks checkbox is checked.

Tracking corneal reflections

Pupil position is typically calculated relative to the location of a corneal reflection to better isolate movements caused by eye rotation. When the location of the corneal reflection is the center of the corneal curvature closest to the camera, this point can be used to estimate angular rotation (see 'Estimating rotational angle of the mouse eye').

Press the Find Reflection button and double click the image near the center of a reflection to cause a small ROI box to appear. Adjust the size and position of this box as described under 'Setting a region

of interest (ROI)'. Press the Find Reflection button again and the reflection center is shown at the centroid of pixels within the reflection ROI that are above the reflection threshold. The reflection threshold is set by selecting Tools menu > Reflection > Set Threshold.

The reflection can be a single spot or a ring of four small spots resulting from LEDs placed around the camera. In the latter case, select Tools menu > Reflection > Set Type > Ring and double click of the four spots. The reflection center is then calculated as the centroid of the centers of the four spots.

Calculating mm per pixel

Calculating the mm per pixel conversion of the video image is required to estimate the rotational angle of the eye and/or display pupil area in units of mm^2 rather than $pixels^2$. To determine this value while in acquiring camera images, find track a corneal reflection as described above and select Tools menu > mm/pixel > Measure. A window will appear to provide directions. Move the camera along the image plane 0.5 mm, then press 'Ok' on the pop-up window. The camera can then be moved back to any location. The mm/pixel value can also be set manually using Tools menu > mm/pixel > Set.

Estimating rotational angle of the mouse eye

Video-based eye tracking involves locating the center of the pupil on an image and converting this position to angular rotation of the eye. One approach to the image-to-angle calibration problem is to have the subject fixate to known visual angles on a screen. A second approach, more practical in mice, is to move the camera instead (i.e. rotate the camera around the corneal surface while the pupil is still; Stahl et al. 2000, Zoccolan et al. 2010). There are several disadvantages to this method: (1) it requires an apparatus for complicated movements of the camera; for some rigs this might necessitate suboptimal placement of the camera and use of mirrors; (2) the calibration process it time consuming and must be performed at the beginning of every experiment; this reduces the time available for the actual experiment; (3) the eye must be still during the calibration process; (4) it assumes that the center of rotation of the corneal surface and the eyeball are equivalent; this is known to be erroneous.

Another approach to eye tracking in mice is described by Sakatani et al. 2004. The intersection of the corneal reflections of 4 IR LEDs placed around the camera lens indicates the center of the corneal curvature closest to the camera. On the video image, the position of the pupil center relative to the center of corneal curvature is related to the center of rotation of the eyeball as follows (Fig. 5 in Sakatani et al. 2004):

$$\frac{x_0 - x_p}{x_c - x_p} = \frac{R_{pupil}}{R_{pupil} - Offset_{eye-cornea}}$$

or

$$x_0 = (x_c - x_p)\frac{R_{pupil}}{R_{pupil} - Offset_{eye-cornea}} + x_p$$

where,

x_p = image coordinate of pupil center
x_c = image coordinate of corneal reflection intersect
x_0 = image coordinate of center of eye
R_{pupil} = rotational radius of pupil
$Offset_{eye-cornea}$ = offset between the rotational centers of the eye and cornea

The rotation angle of the pupil is:

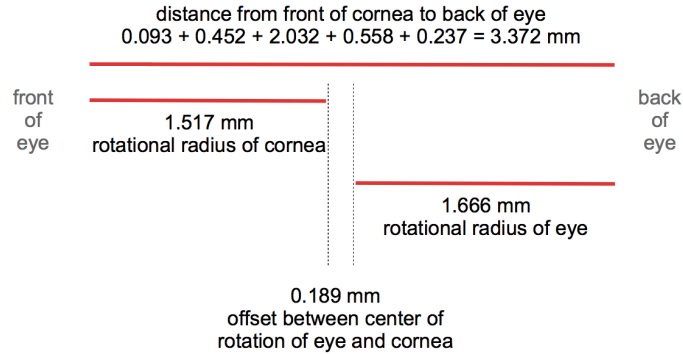$$\text{pupil rotation} = \arcsin\left(\frac{X_p - X_0}{R_{pupil}}\right)$$

thus,

$$\text{pupil rotation} = \arcsin\left(\frac{(X_c - X_p)\dfrac{R_{pupil}}{R_{pupil} - \text{Offset}_{eye\text{-}cornea}}}{R_{pupil}}\right)$$

Using this equation and the values of R_{pupil} and $\text{Offset}_{eye\text{-}cornea}$ discussed below, one can determine the vertical and horizontal rotational angles of the eye from the image coordinates of the pupil center and corneal reflection intersection. Note that by tracking and comparing the movement of both the corneal reflection intersection and the pupil each frame, rotations of the pupil are isolated from translational movement of the eyes (with movement of the head, for instance; DiScenna et al. 1995, Stahl et al. 2000).

One potential disadvantage of this method is that is uses values of R_{pupil} and $\text{Offset}_{eye\text{-}cornea}$ derived from the literature (Remtulla and Hallett 1985) rather than measured for each mouse. However the variation in these parameters across mice of different age and sex is small (less than 10% after the mice are 6 weeks old; Puk et al. 2006, Tkatchenko et al. 2010).

From Table 1 in Remtulla and Hallett 1985 (note this table is erroneously labeled 'rat' rather than 'mouse') we can derive the following:

distance from front of cornea to back of eye
0.093 + 0.452 + 2.032 + 0.558 + 0.237 = 3.372 mm

front
of
eye

1.517 mm
rotational radius of cornea

back
of
eye

1.666 mm
rotational radius of eye

0.189 mm
offset between center of
rotation of eye and cornea

Similarly, the offset between the center of rotation of the eye and pupil is:

$\text{Offset}_{eye\text{-}lens}$ = |2.032+0.558+0.237-1.248-1.666| = 0.087 mm
$\text{Offset}_{eye\text{-}lens}$ is negative because the center of rotation of the lens is offset towards the back of the eye relative to the rotational center of the eye.

Sakatani et al. 2000 round $\text{Offset}_{eye\text{-}cornea}$ to 0.2 mm and $\text{Offset}_{eye\text{-}lens}$ to 0.1 mm.

Correcting for pupil size (Sakatani et al. 2000),

$$R_{pupil} = \sqrt{R_{lens}^2 - P_{radius}^2} - \text{Offset}_{eye\text{-}lens}$$

where,

R_{lens} = rotational radius of the lens = 1.25 mm (Remtulla and Hallett 1985)
P_{radius} = pupil radius (dilation; as opposed to the rotation radius, R_{pupil})

References

DiScenna et al. (1995). Evaluation of a video tracking device for measurement of horizontal and vertical eye rotations during locomotion. J Neurosci Methods.

Puk et al. (2006). Variations of eye size parameters among different strains of mice. Mamm Genome.

Remtulla and Hallett (1985). A schematic eye for the mouse, and comparisons with the rat. Vision Res.

Sakatani et al. (2004). PC-based high-speed video-occulography for measuring rapid eye movments in mice. Neurosci Res.

Stahl et al. (2000). A comparison of video and magnetic search coil recordings of mouse eye movements. J Neurosci Methods.

Tkatchenko et al. (2010). Analysis of postnatal eye development in the mouse with high-resolution small animal magnetic resonance imaging. Invest Ophthalmol Vis Sci

Zoccolan et al. (2010). A self-calibrating, camera-based eye tracker for the recording of rodent eye movements. Front Neurosci.