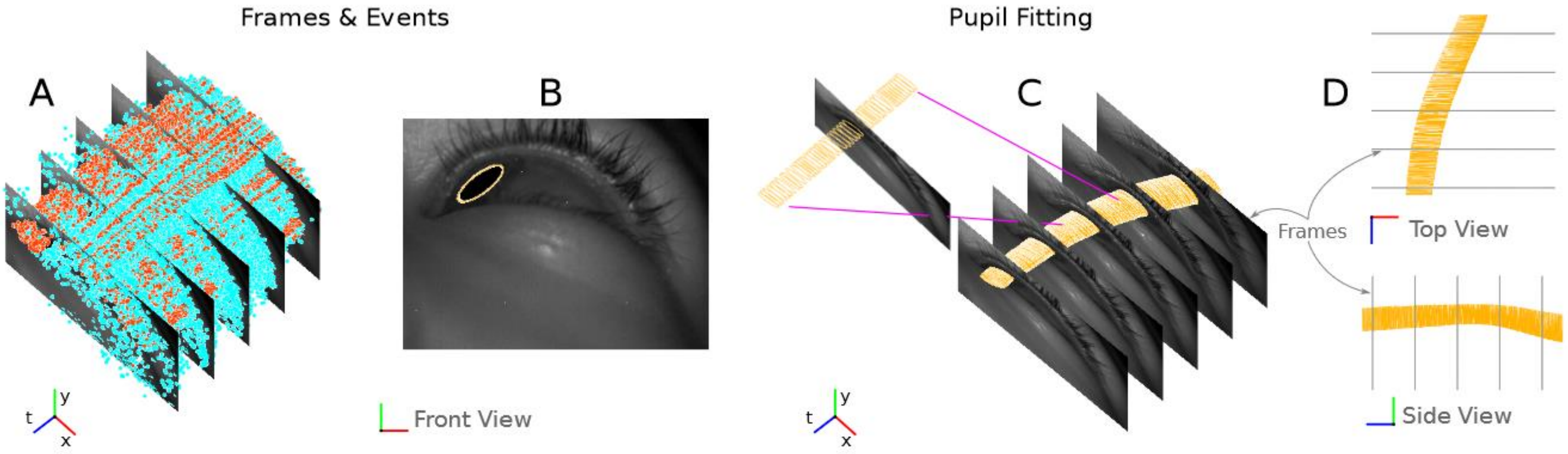
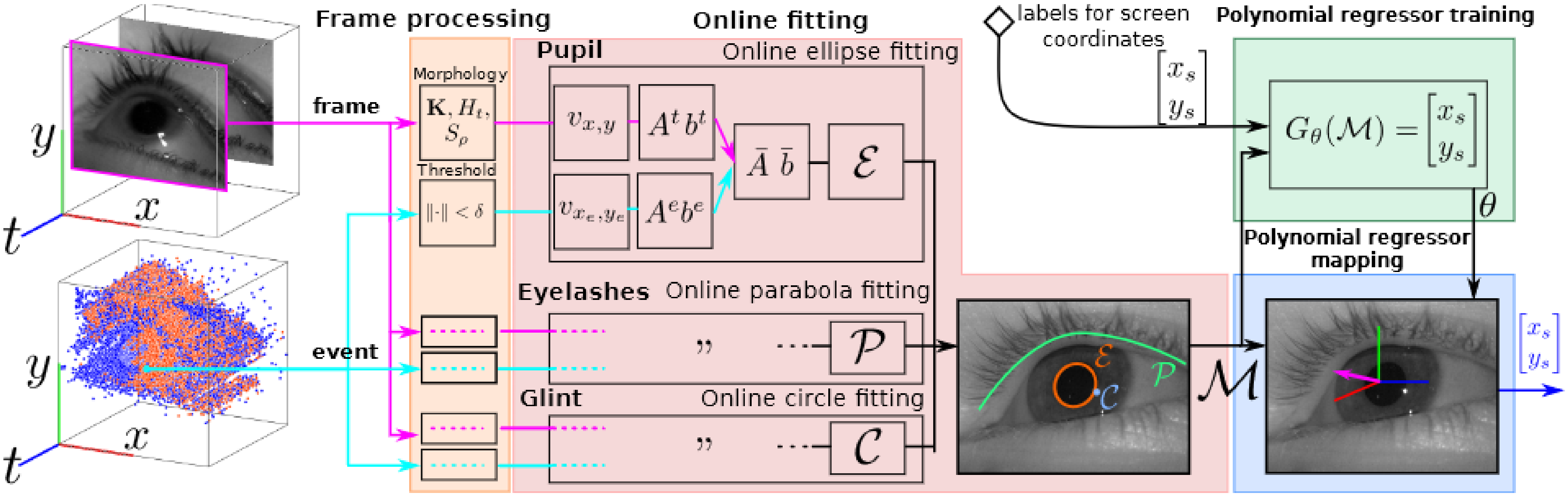
Event-Based Near-Eye Gaze Tracking



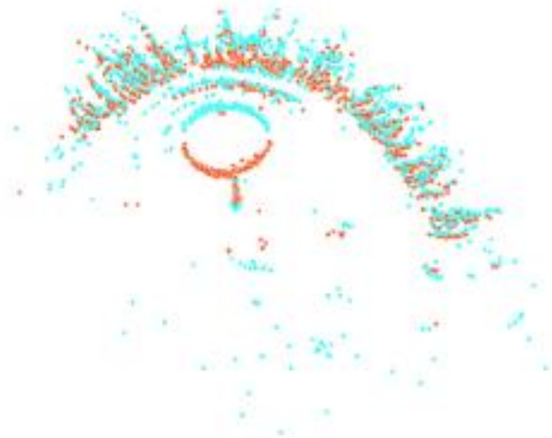
System flow



Frame processing
Online fitting
labels for screen coordinates
Polynomial regressor training
Pupil
Online ellipse fitting
Morphology
$\mathbf{K}, H_t, S_\rho$
Threshold
$\|\cdot\| < \delta$
frame
event
$v_{x,y}$
$A^t b^t$
v_{x_e,y_e}
$A^e b^e$
$\bar{A}\ \bar{b}$
$\mathcal{E}$
$G_\theta(\mathcal{M}) = \begin{bmatrix} x_s \\ y_s \end{bmatrix}$
θ
Polynomial regressor mapping
Eyelashes
Online parabola fitting
$\mathcal{P}$
Glint
Online circle fitting
$\mathcal{C}$
$\mathcal{M}$
$\begin{bmatrix} x_s \\ y_s \end{bmatrix}$

Eye Modle

- M = {E , P, C }
- E representing pupil ellipse, R^5
- P representing eyelid parabola, R^3
- C representing glint circle, R^2

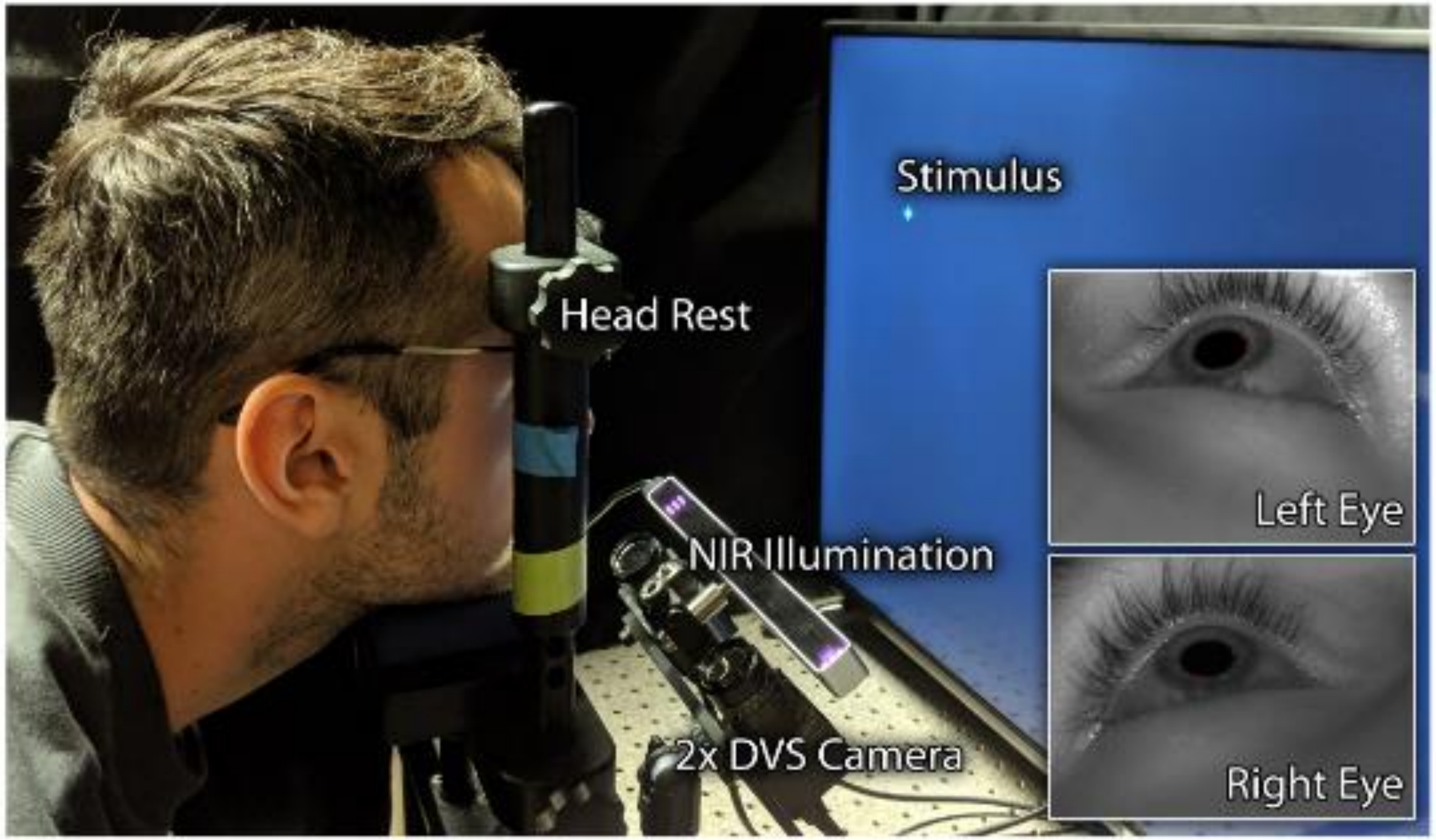


Mapping to a point of gaze

- Using two 2nd order polynomial functions to regress the output (xs, ys) from the pupil center (xc, yc)

$$G_{\theta}(x_c, y_c) = \begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} G_{\theta^1}|_x(x_c, y_c) \\ G_{\theta^2}|_y(x_c, y_c) \end{pmatrix}$$

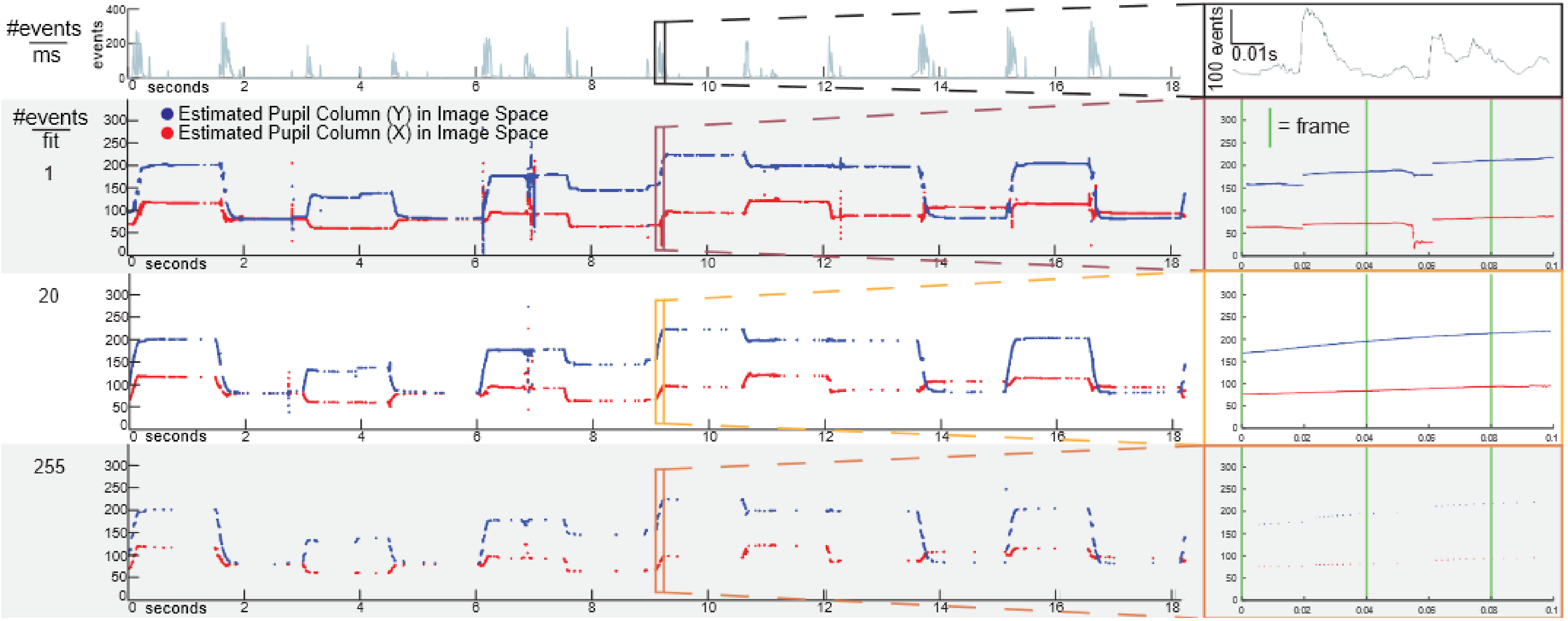
Dataset



Stimulus
Head Rest
Left Eye
NIR Illumination
2x DVS Camera
Right Eye

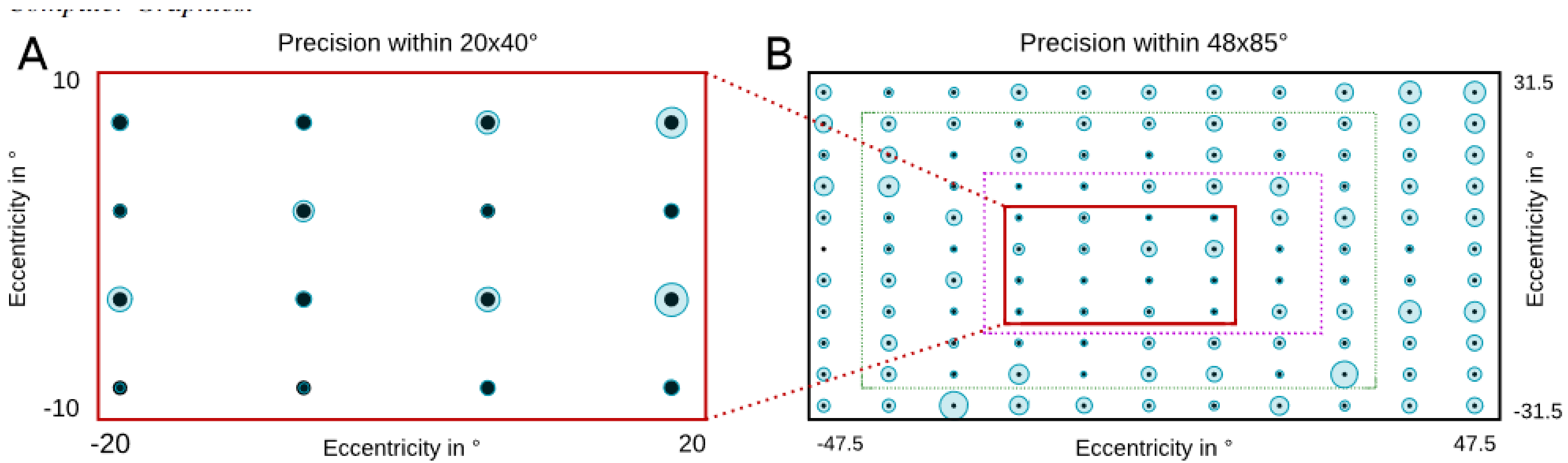
Result

- an inherent tradeoff between the smoothness, sparsity, and update rate.



Result

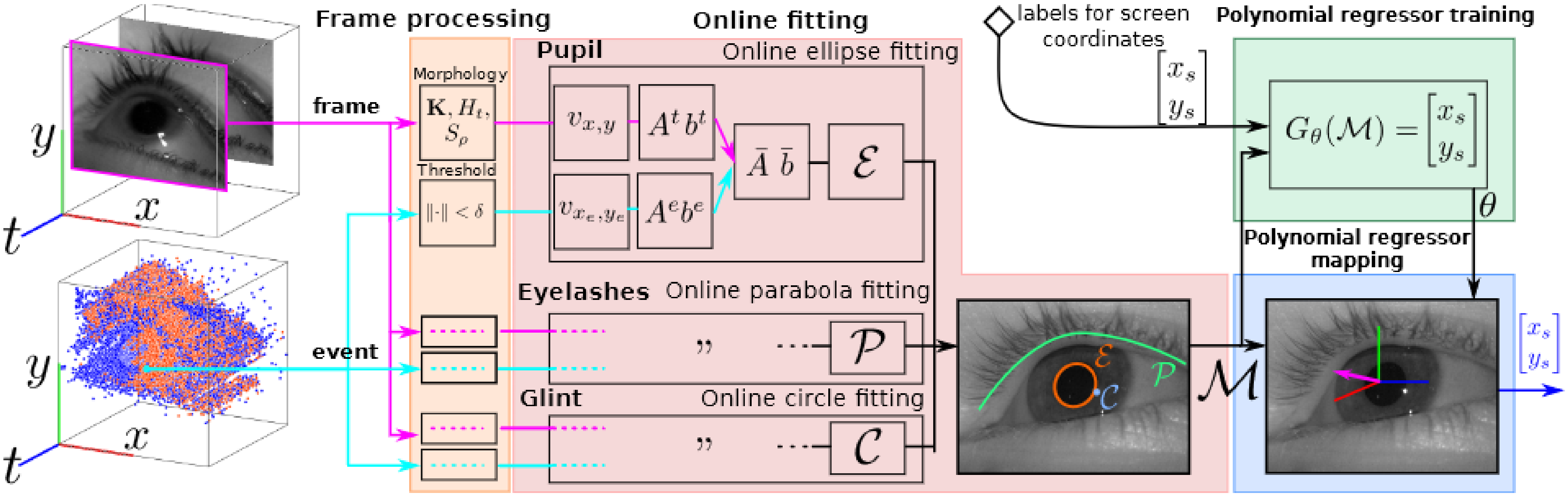
- Gaze Mapping accuracy and precision worsen in the edges of the field of view



Possible Improvements

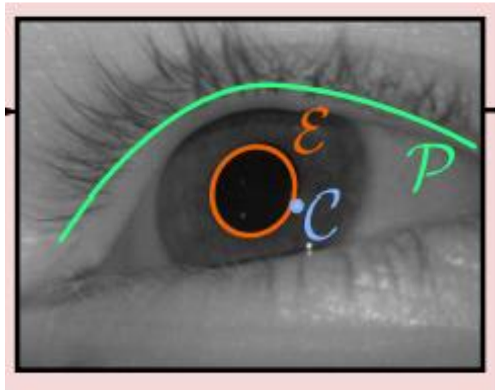
- Noise reduction and refresh rate enhancement
- Edge accuracy improvement
- Large distance camera
- Non fixed head position
- Single picture with two eyes
- New algorithms
- New evaluation methods

System flow



Frame processing
Online fitting
labels for screen coordinates
Polynomial regressor training
Pupil
Online ellipse fitting
Morphology
frame
$\mathbf{K}, H_t, S_\rho$
Threshold
$\|\cdot\| < \delta$
$v_{x,y}$
$A^t b^t$
v_{x_e,y_e}
$A^e b^e$
$\bar{A}\ \bar{b}$
$\mathcal{E}$
$G_\theta(\mathcal{M}) = \begin{bmatrix} x_s \\ y_s \end{bmatrix}$
θ
Polynomial regressor mapping
Eyelashes
Online parabola fitting
event
$\mathcal{P}$
Glint
Online circle fitting
$\mathcal{C}$
$\mathcal{M}$
$\begin{bmatrix} x_s \\ y_s \end{bmatrix}$

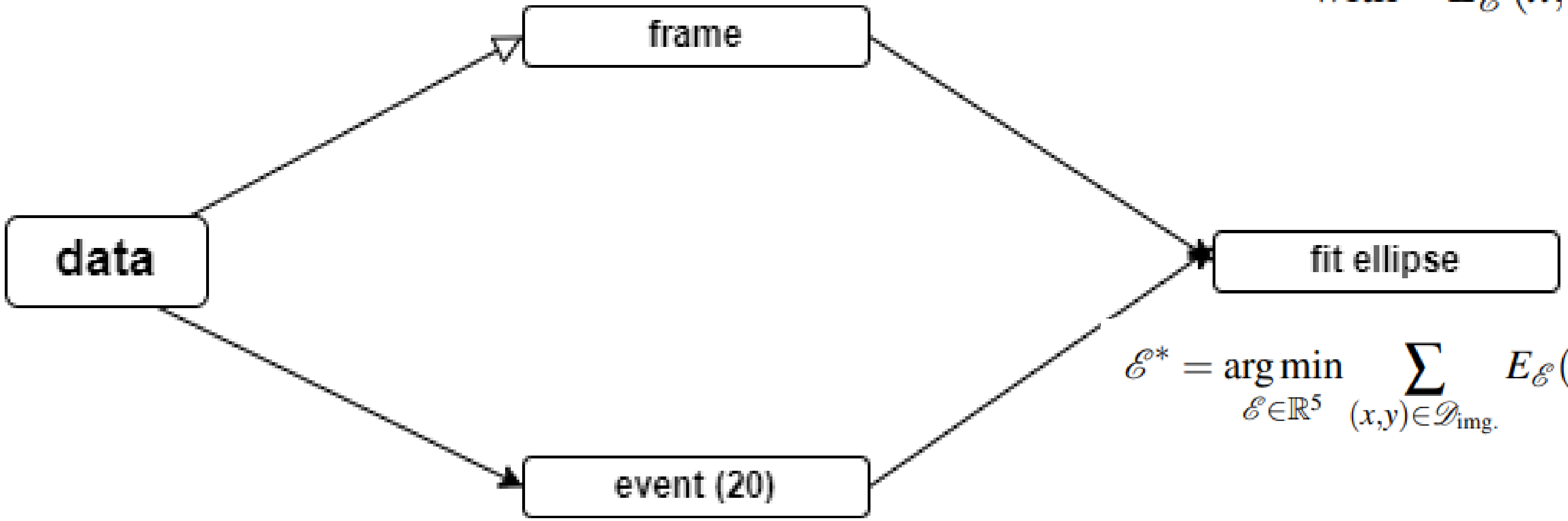
Eye Model-pupil ellipse



$$E_{\mathscr{E}}(x,y) = 0$$

$$\text{with} \quad E_{\mathscr{E}}(x,y) = ax^2 + hxy + by^2 + gx + fy + d$$

$$\mathscr{D}_{\text{img.}} = \left\{(x,y) \mid \mathbf{K}\big(\mathrm{H}_\theta(I) \circ S_\sigma\big)(x,y) = 1\right\},$$



$$\mathscr{E}^* = \underset{\mathscr{E} \in \mathbb{R}^5}{\arg\min} \sum_{(x,y) \in \mathscr{D}_{\text{img.}}} E_{\mathscr{E}}(x,y)^2$$

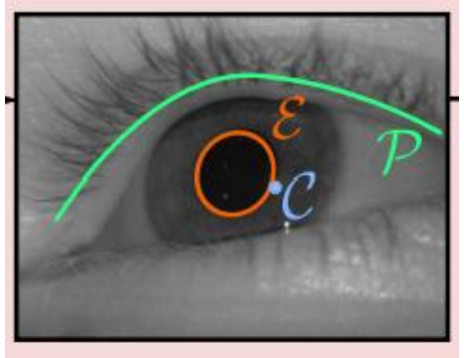
$$\mathscr{D}_{\text{evt.}} = \left\{(x,y) \mid \|P_{\mathscr{E}}\big((x,y)\big) - (x,y)\|_2 < \delta\right\},$$

for all $(x,y) \in \mathscr{D} = \mathscr{D}_{\text{evt.}} \cup \mathscr{D}_{\text{img.}}$ **do**

$$
\begin{aligned}
v_{x,y}^{\intercal} &\leftarrow (x^2, xy, y^2, x, y) \\
A &\leftarrow \textstyle\sum_{(x,y) \in \mathscr{D}} v_{x,y} v_{x,y}^{\intercal} \\
b &\leftarrow \textstyle\sum_{(x,y) \in \mathscr{D}} v_{x,y}
\end{aligned}
$$

$$
\begin{aligned}
\bar{A} &\leftarrow \gamma \bar{A} + (1-\gamma) A \\
\bar{b} &\leftarrow \gamma \bar{b} + (1-\gamma) b \\
\bar{A}_{\text{inv.}} &\leftarrow \bar{A}^{-1} \\
\mathscr{E} &\leftarrow \bar{A}_{\text{inv.}} \bar{b}
\end{aligned}
$$

Eye Model- eyelid parabola



$$E_{\mathscr{P}}(x,y) = 0$$

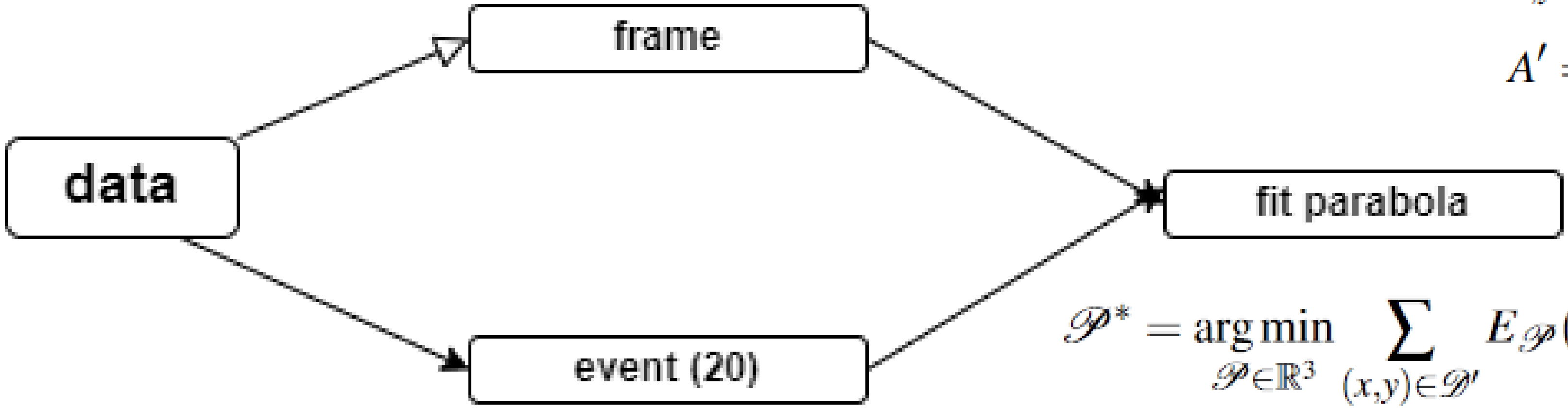
$$\text{with} \quad E_{\mathscr{P}}(x,y) = a'\,y^2 + g'\,y + d' - x$$

$$\mathscr{D}'_{\text{img.}} = \Big\{(x,y) \mid (x,y) \in \text{HarrisCorner} \circ \text{clip}(I, t_1, t_2), \\ \text{and } \|(x,y) - (x_e, y_e)\|^2 < \rho', \\ \text{and } y < \frac{\text{rows}}{2}\Big\}$$

$$v'^{1}_{x,y} = (y^2, y, 1) \quad v'^{2}_{x,y} = x\, v'^{1\,\intercal}_{x,y} = (x y^2, x y, x)$$

$$A' = \sum_{(x,y)\in\mathscr{D}_{\text{img.}}} v'^{1}_{x,y}\, v'^{1}{}_{x,y}{}^{\intercal}, \quad b' = \sum_{(x,y)\in\mathscr{D}_{\text{img.}}} v'^{2}_{x,y}$$

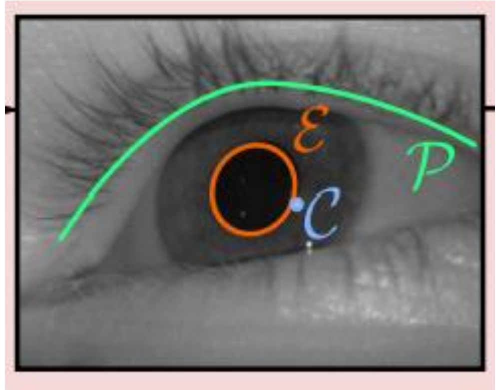
$$\mathscr{P}^* = A'^{-1}\, b'$$



$$\mathscr{P}^* = \underset{\mathscr{P}\in\mathbb{R}^3}{\arg\min} \sum_{(x,y)\in\mathscr{D}'} E_{\mathscr{P}}(x,y)^2$$

$$\mathscr{D}'_{\text{evt.}} = \Big\{(x,y) \mid |E_{\mathscr{P}}((x,y))| < \delta\Big\}$$

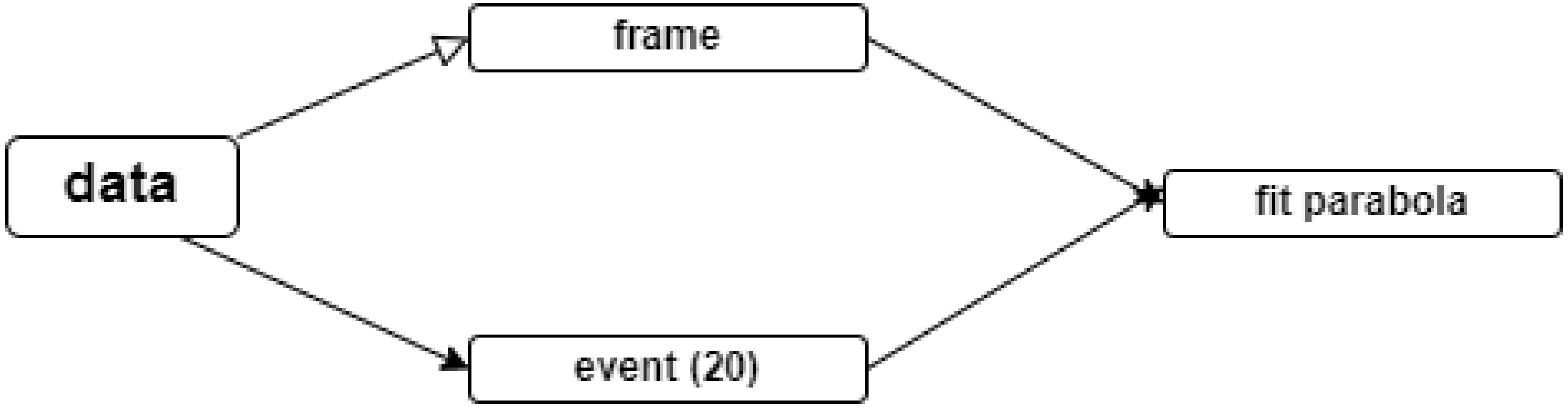
Eye Model- glint circle



$$E_{\mathscr{C}}(x,y) = 0$$

$$\text{with} \quad E_{\mathscr{C}}(x,y) = x^2 - 2xc_x - 2yc_y + (c_x^2 + c_y^2 - r^2)$$

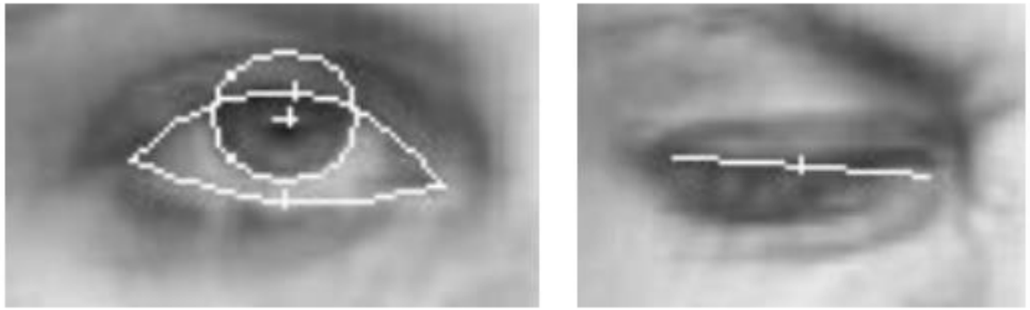
$$\mathscr{D}''_{\text{img.}} = \Big\{(x,y) \mid (x,y) \in H_{t_3}(I(x,y)) \text{ and } \|(x,y) - (x_e,y_e)\|^2 < \rho''\Big\}$$



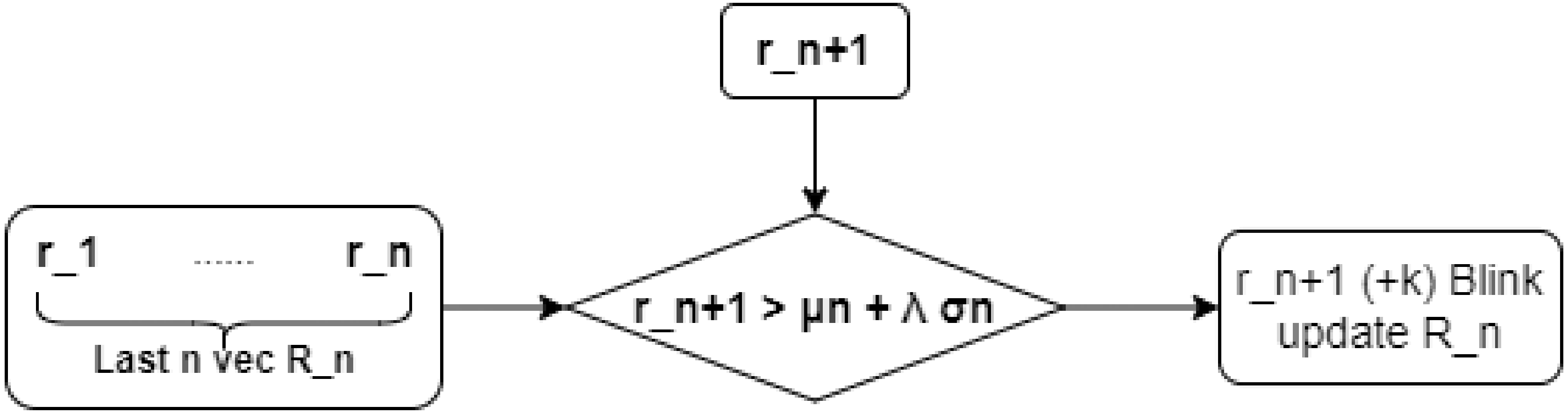
$$\mathscr{D}''_{\text{evt.}} = \{(x,y) \mid |E_{\mathscr{C}}(x,y)| < \delta\}$$

Similar to the ellipse

Blink Detector



Detect changes in *eccentricity* of pupil ellipse on frames



The Regressor – eye model to screen

$$G_{\theta^i}|_{x/y}(x_e, y_e) = \alpha_i x_e^2 + \gamma_i x_e y_e + \beta_i y_e^2 + \varepsilon_i x_e + \zeta_i y_e + \eta_i$$

$$\arg\min_{\theta^1} \|G_{\theta^1}|_x(x_e, y_e) - x_s\|^2$$

$$\arg\min_{\theta^2} \|G_{\theta^2}|_y(x_e, y_e) - y_s\|^2$$

$$\boxed{\text{eye center}} \longrightarrow \boxed{\text{screen pixel}}$$

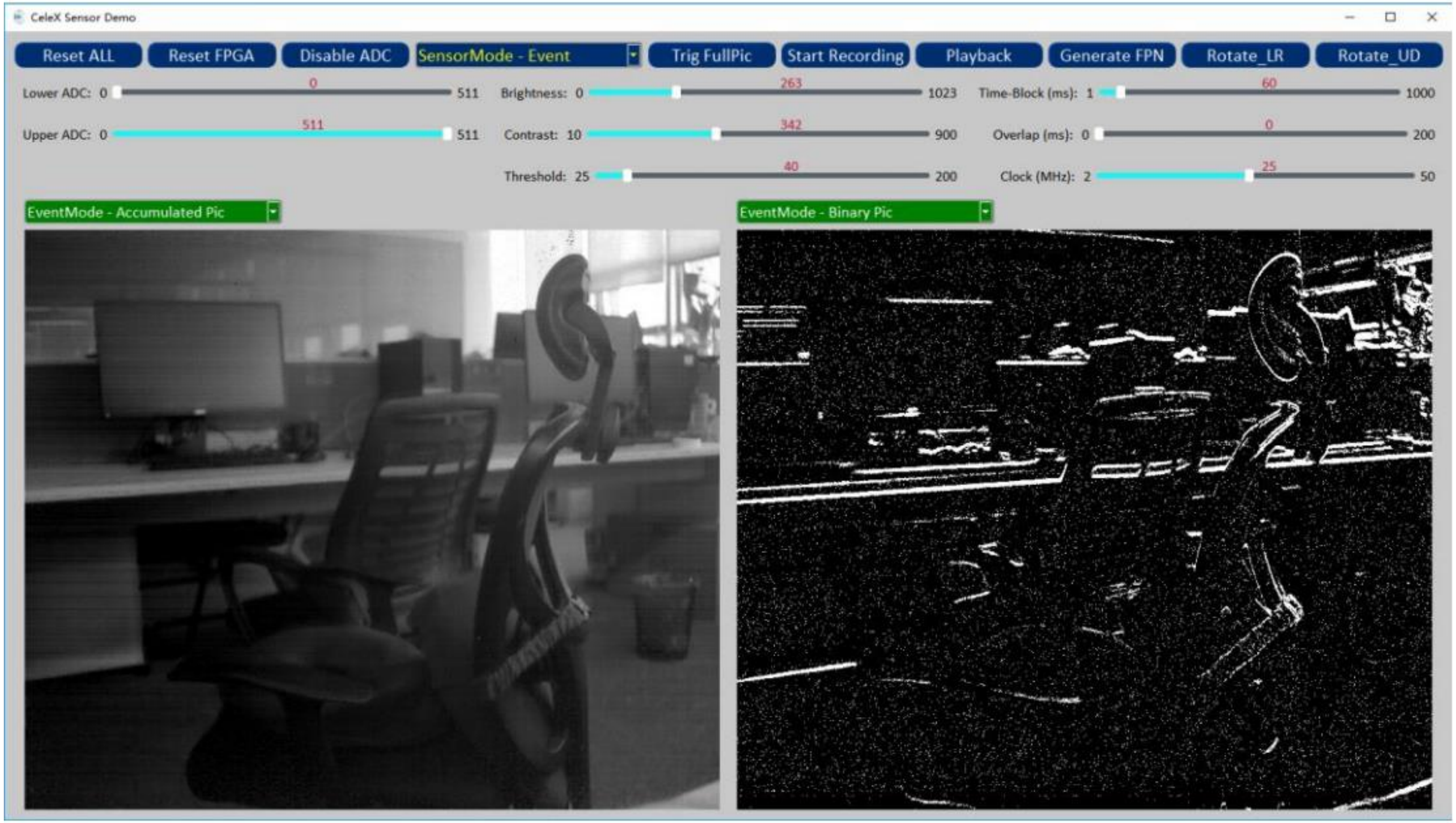
$$x_e = \frac{2bg - hf}{h^2 - 4ab}, \text{ and } y_e = \frac{2af - hg}{h^2 - 4ab}$$

$$\mathscr{E} = (a, h, b, g, f, d)$$

$$G_\theta(\mathscr{E}) = \begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} G_{\theta^1}|_x(\mathscr{E}) \\ G_{\theta^2}|_y(\mathscr{E}) \end{pmatrix}$$

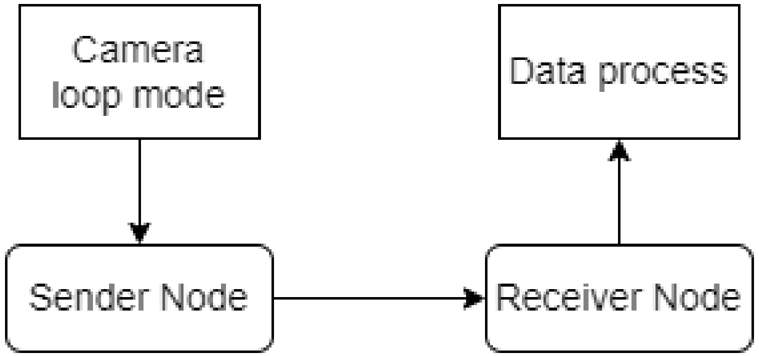
CelePixel

CelePixel Event Camera

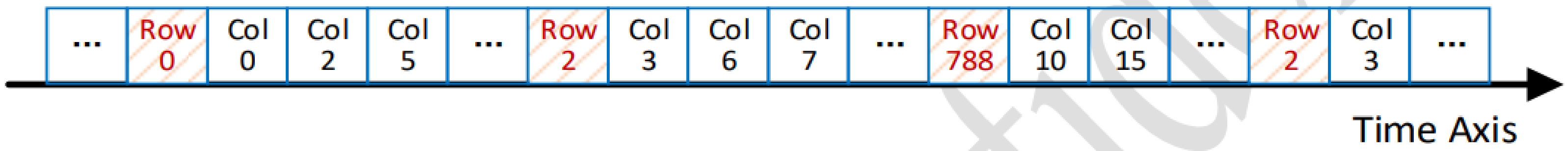


CelePixel Data Collection under ROS

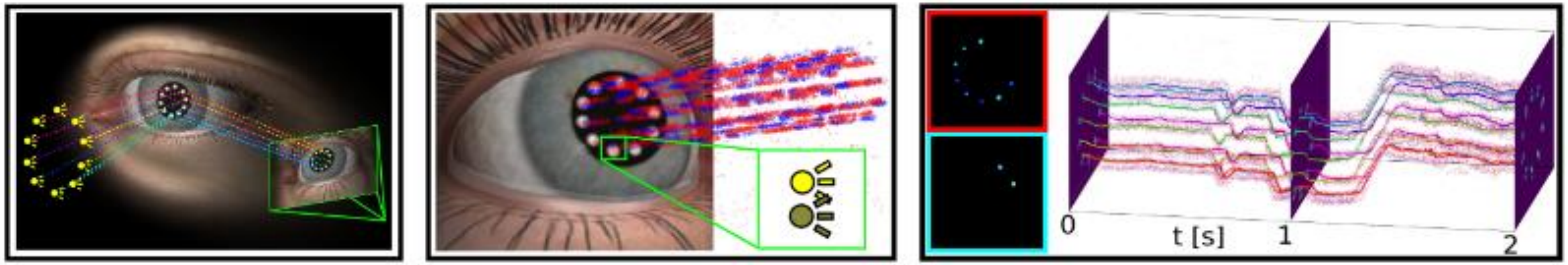
Full-Picture ⟷ Event Off-Pixel Timestamp

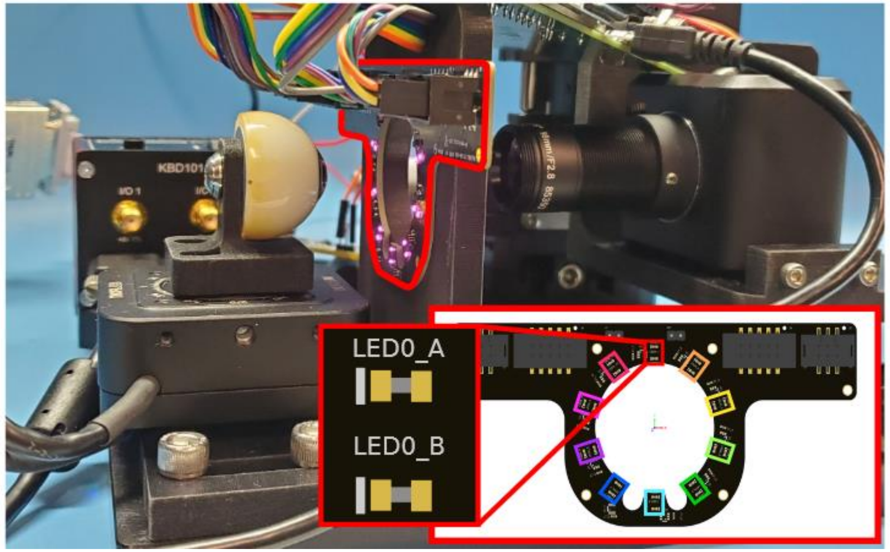


Sensor 的工作模式	SDK 输出的图像数据类型
Full-Picture Mode	Full Pic Buffer/Mat
Event Off-Pixel Timestamp Mode	Event Binary Pic Buffer/Mat Event Denoised Pic Buffer/Mat Event Count Pic Buffer/Mat Event Vector<row, col, off-pixel timestamp>

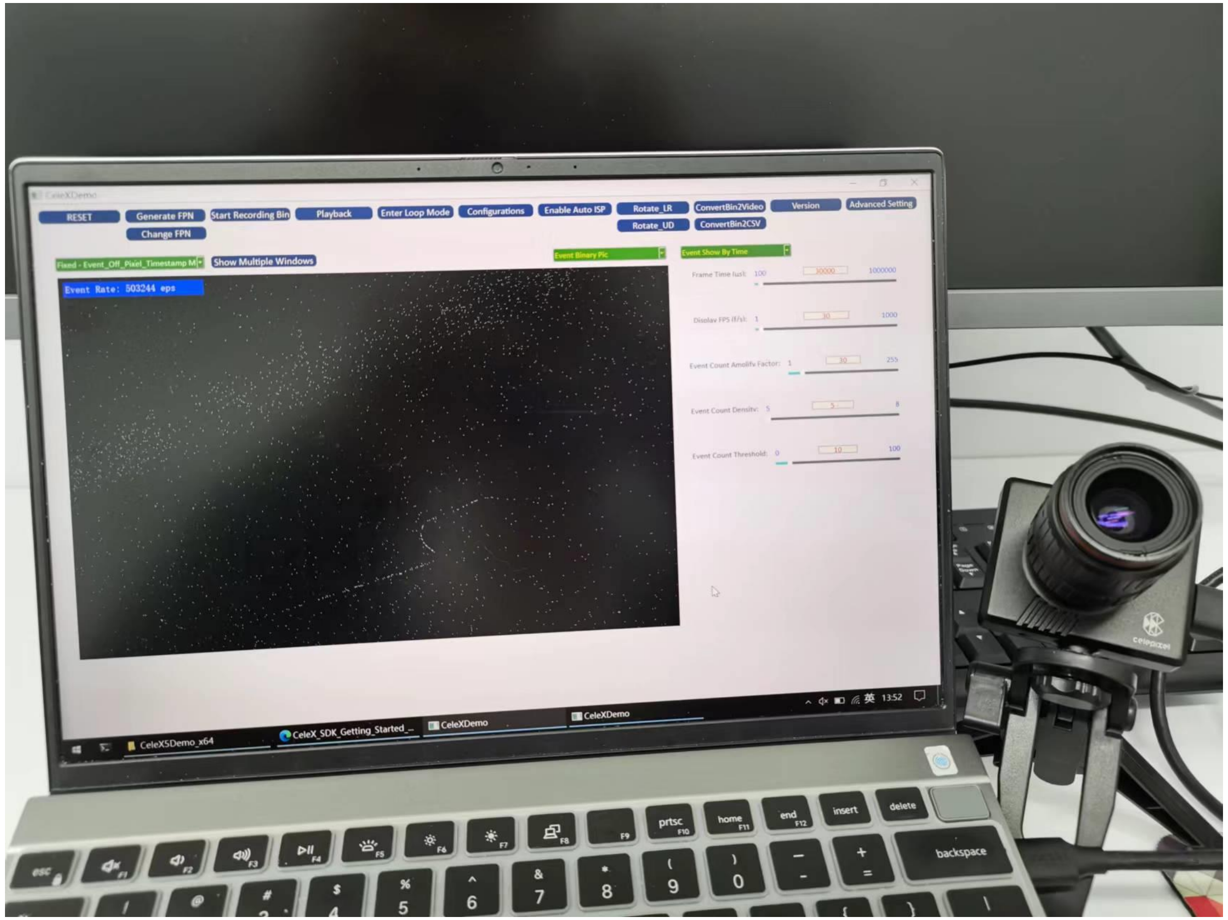


Event-Based Kilohertz Eye Tracking using Coded Differential Lighting





Data record


RESET
Generate FPN
Change FPN
Start Recording Bin
Playback
Enter Loop Mode
Configurations
Enable Auto ISP
Rotate_LR
Rotate_UD
ConvertBin2Video
ConvertBin2CSV
Version
Advanced Setting
Show Multiple Windows
CeleXDemo

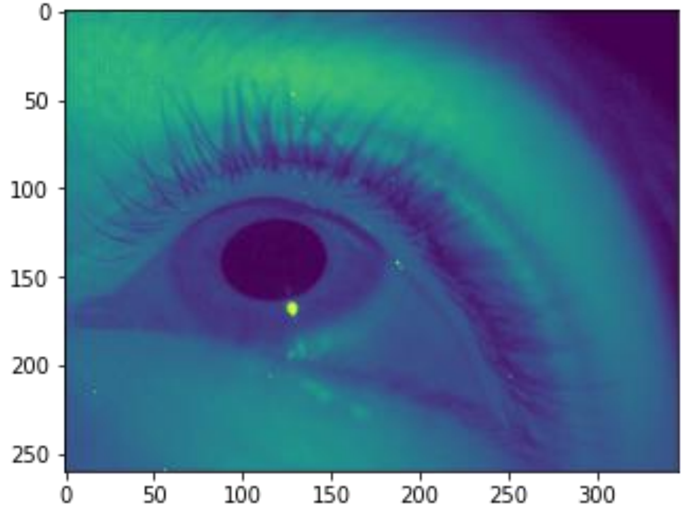

Event Binary Pic

Ellipse pupil fit

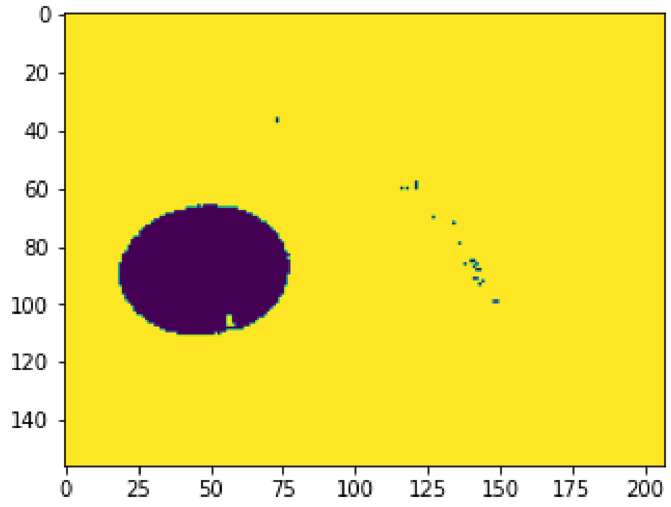
$$E_{\mathscr{E}}(x,y)=0$$

with $$E_{\mathscr{E}}(x,y)=ax^2+hxy+by^2+gx+fy+d$$

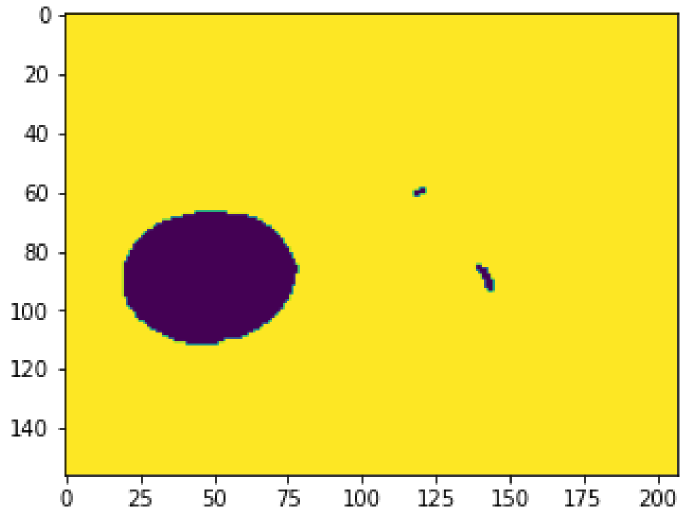
$$\mathscr{D}_{\text{img.}} \leftarrow \{(x,y) \mid \mathbf{K}(\mathrm{H}_\theta(I^t)\circ S_\rho)(x,y)=1\}$$



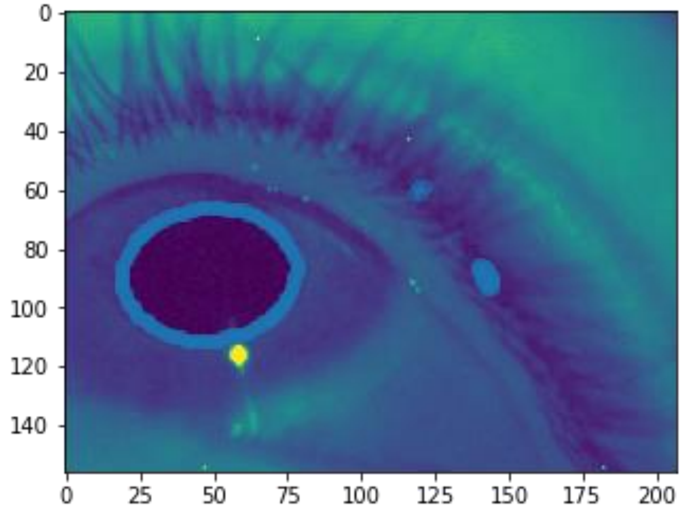
frame



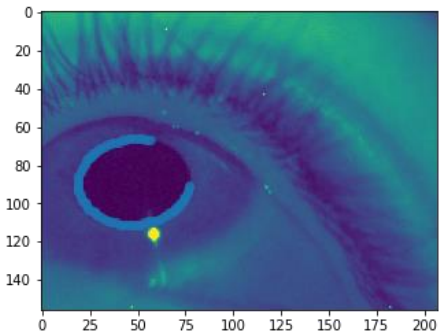
threshold



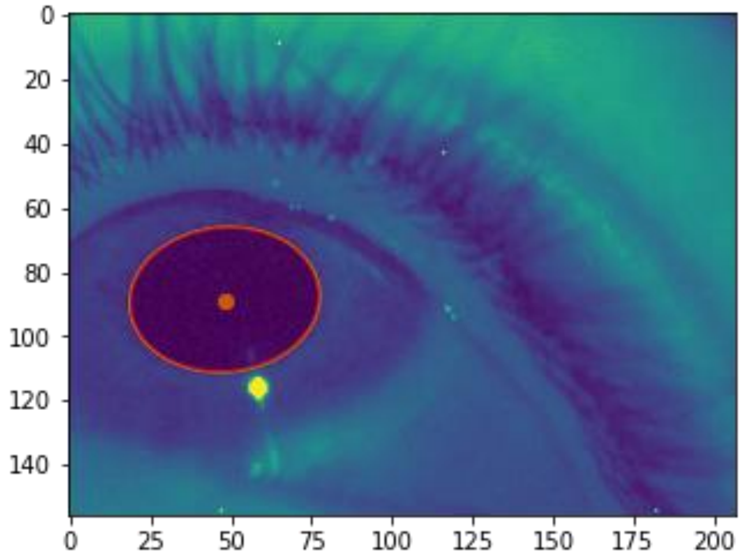
morphological opening



edge detect



filter by distant



Fit result

Red: opencv
Yellow: paper

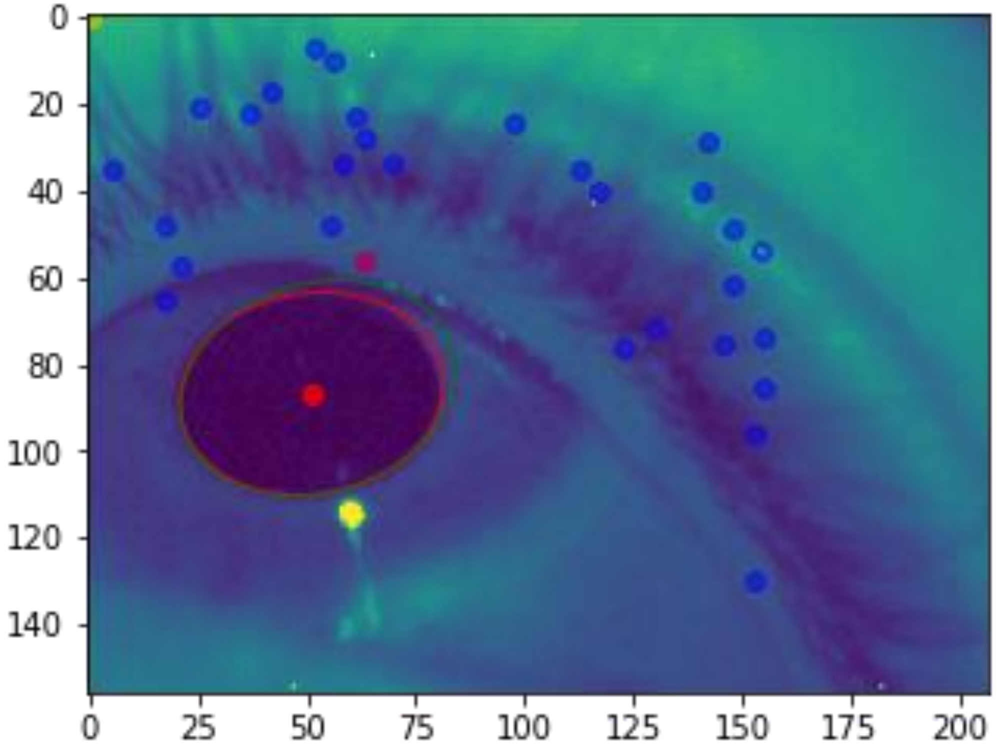
Coincident

Ellipse pupil fit

$$E_{\mathscr{E}}(x,y) = 0$$
$$\text{with} \quad E_{\mathscr{E}}(x,y) = ax^2 + hxy + by^2 + gx + fy + d$$

$$\mathscr{D}_{\text{evt.}} = \left\{(x,y) \mid \|P_{\mathscr{E}}((x,y)) - (x,y)\|_2 < \delta\right\},$$

In implement
Distantance (evt, center) < avg(a, b) +- tol



Event fit

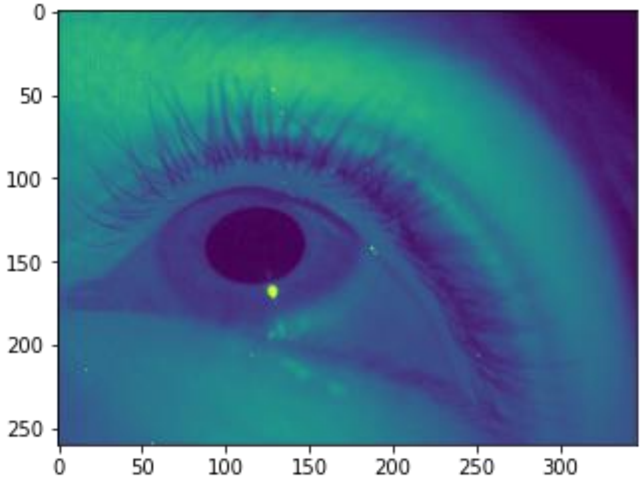
Blue Point: events
Red Point: filtered events

Red ellipse: fitted by frame
Green ellipse: fitted with new events

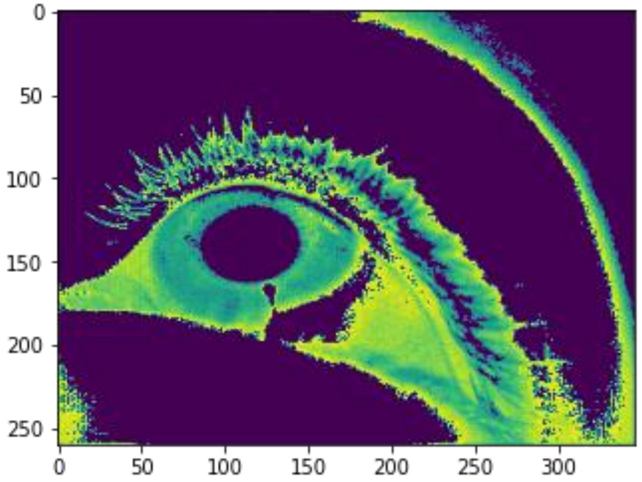
Parabola eyelid fit

$$E_{\mathscr{P}}(x,y) = 0$$
$$\text{with} \quad E_{\mathscr{P}}(x,y) = a'\,y^2 + g'\,y + d' - x$$

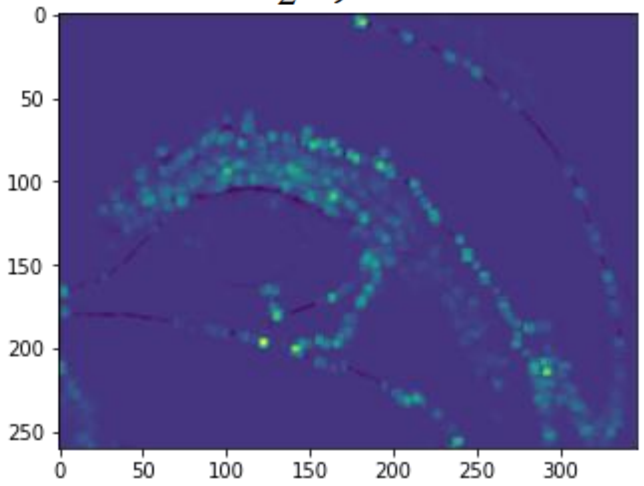
$$\mathscr{D}'_{\text{img.}} = \Big\{ (x,y) \mid (x,y) \in \text{HarrisCorner} \circ \text{clip}(I, t_1, t_2), \; \text{and } \|(x,y) - (x_e, y_e)\|^2 < \rho', \; \text{and } y < \frac{\text{rows}}{2} \Big\}$$



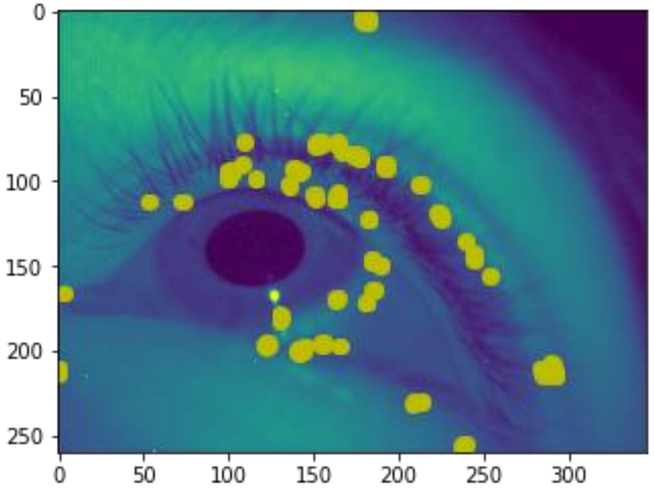
frame



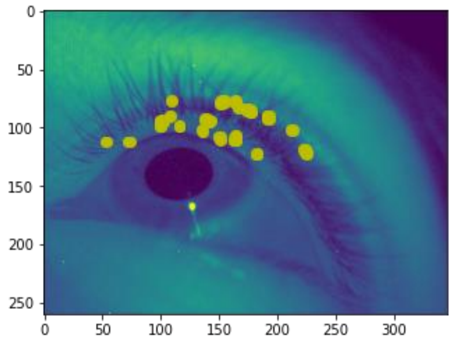
Clip



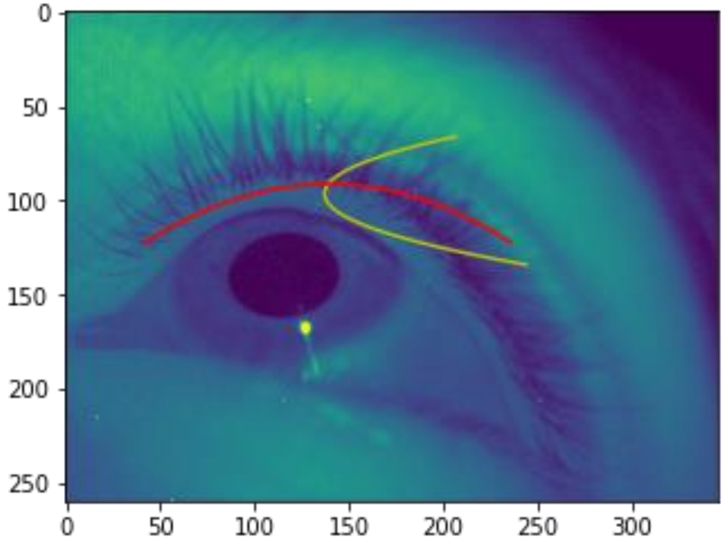
Harris Corner Detect



Corner points



Cut below



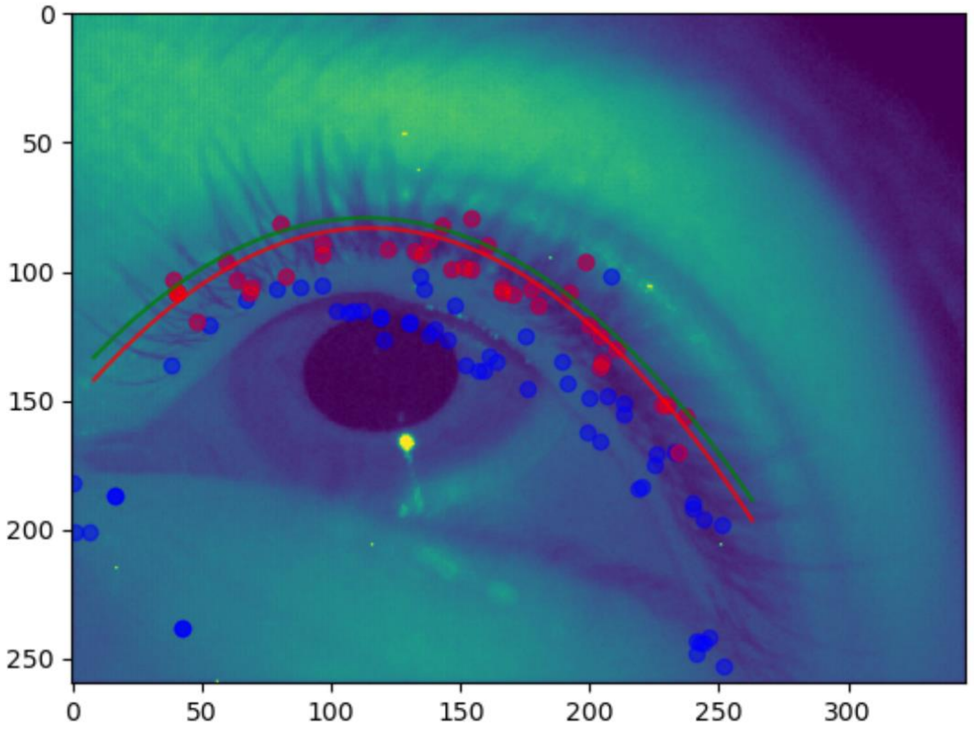
Fit result

Yellow: x=y^2
Red: y=x^2

Parabola eyelid fit

$$E_{\mathscr{P}}(x,y) = 0$$
$$\text{with} \quad E_{\mathscr{P}}(x,y) = a'y^2 + g'y + d' - x$$

$$\mathscr{D}'_{\text{evt.}} = \left\{(x,y) \mid |E_{\mathscr{P}}((x,y))| < \delta\right\}$$



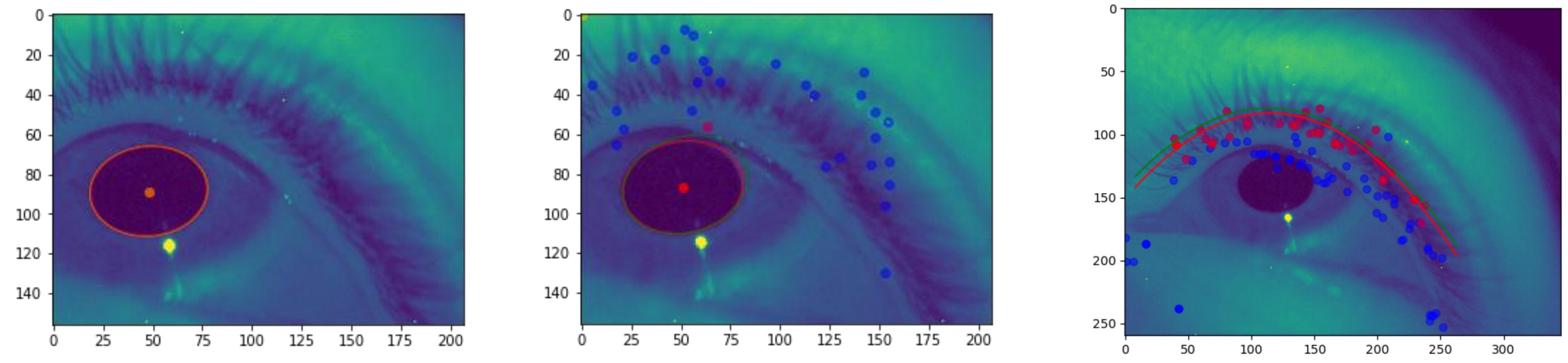
Event fit

Blue Point: events
Red Point: filtered events

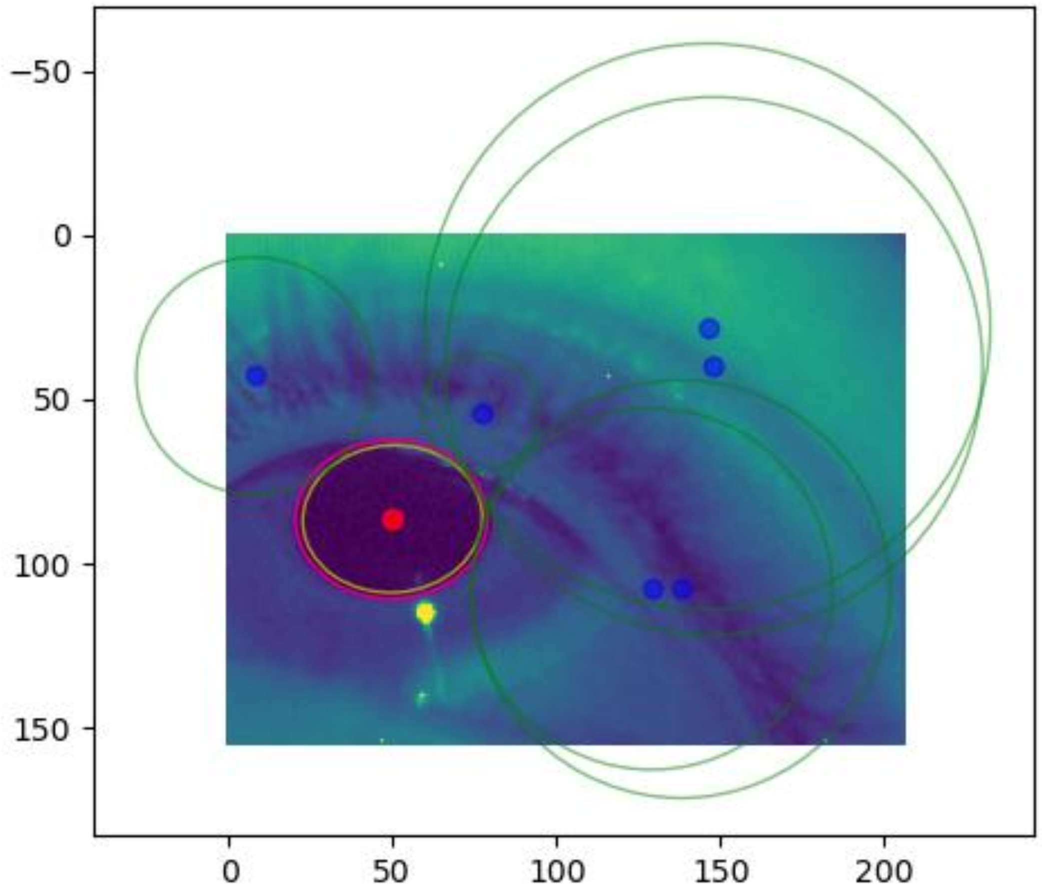
Red ellipse: fitted by frame
Green ellipse: fitted with new events

Last week

- Ellipse pupil fit and Parabola eyelid fit from flames and events



Distance to ellipse



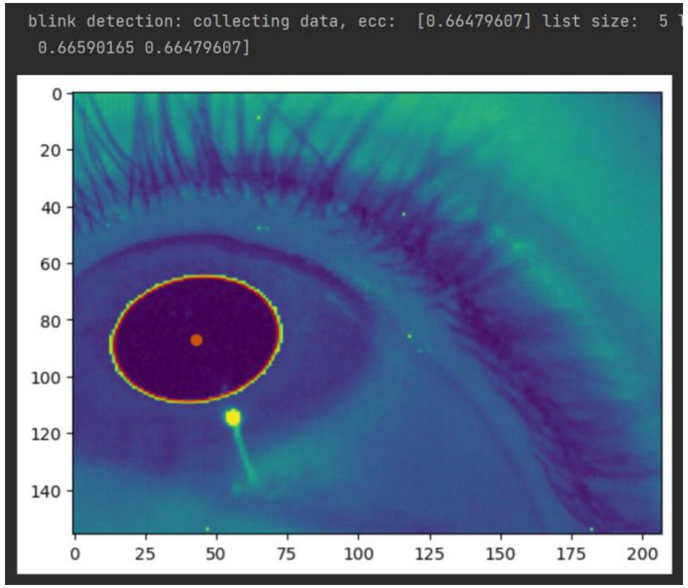
Each **event** with calculated **distant circle** to **pupil ellipse**

Method 1:
① dy/dx(ellipse) * line slope = -1
② Ellipse equation

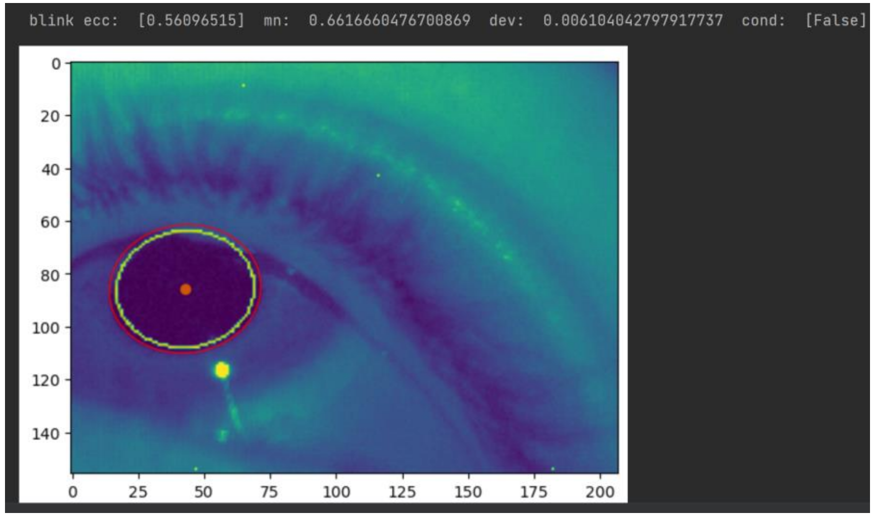
Method 2
Newton method
Transform to Polar forms

f = (x - a*cos(t))**2 + (y - b*sin(t))**2
t_n = t_n-1-df/ddf
Tol = 1e-2, max_itr = 3

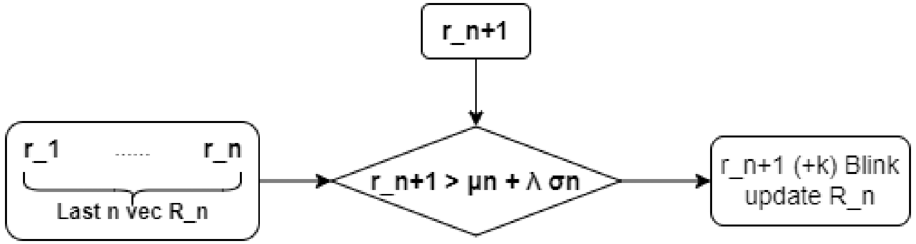
Blink detection



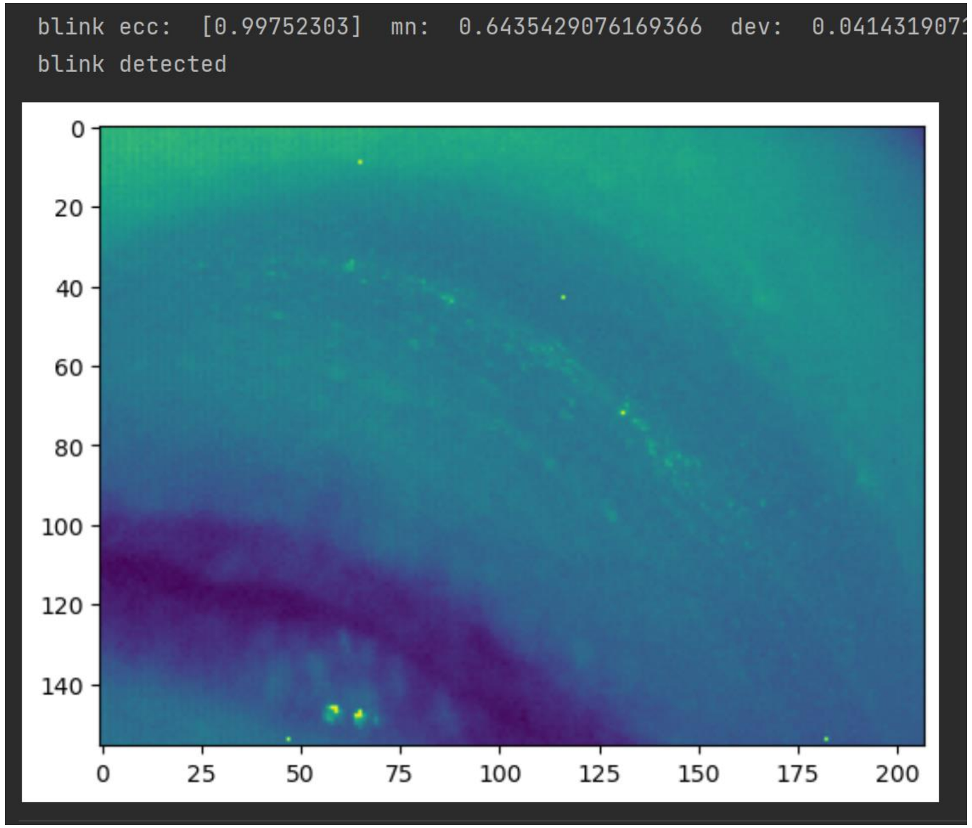
Collect data for first 5 flames



Not blink



Condition = mean+3*var, list length = 5



Blinked, will ignore following 3 flames

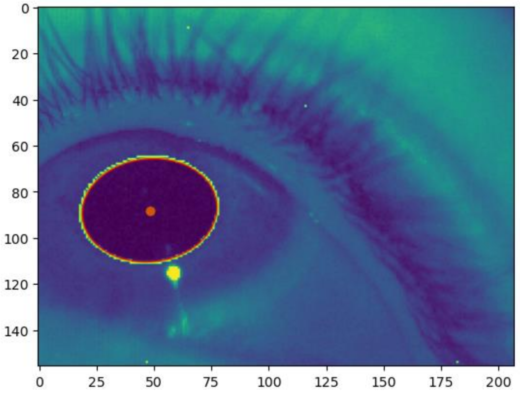
The Regressor – eye model to screen

$$G_\theta(\mathscr{E}) = \begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} G_{\theta^1}|_x(\mathscr{E}) \\ G_{\theta^2}|_y(\mathscr{E}) \end{pmatrix}$$

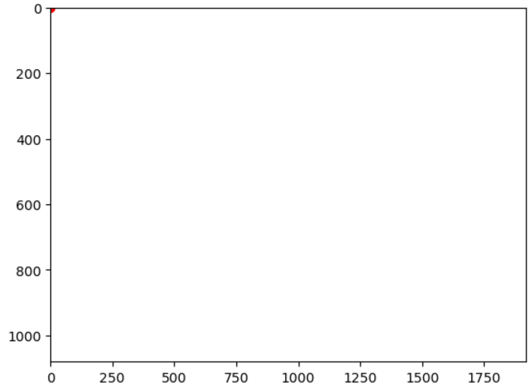
$$\arg\min_{\theta^1} \|G_{\theta^1}|_x(x_e, y_e) - x_s\|^2$$

$$\arg\min_{\theta^2} \|G_{\theta^2}|_y(x_e, y_e) - y_s\|^2$$

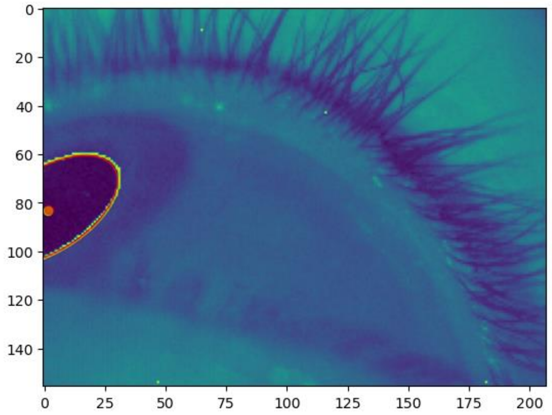
$$G_{\theta^i}|_{x/y}(x_e, y_e) = \alpha_i x_e^2 + \gamma_i x_e y_e + \beta_i y_e^2 + \varepsilon_i x_e + \zeta_i y_e + \eta_i$$



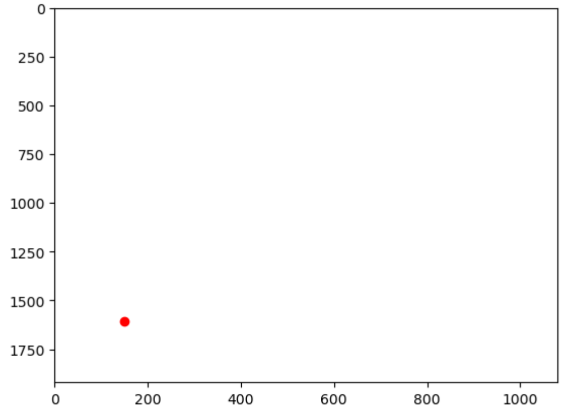
48.12, 88.09



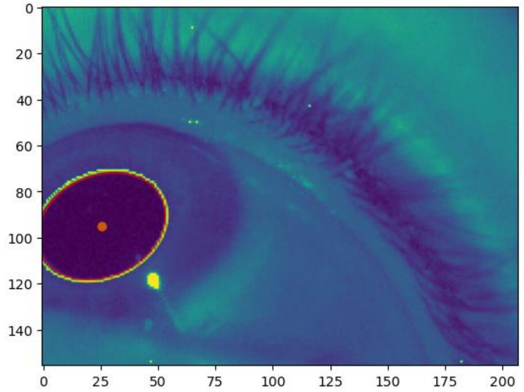
0 , 0



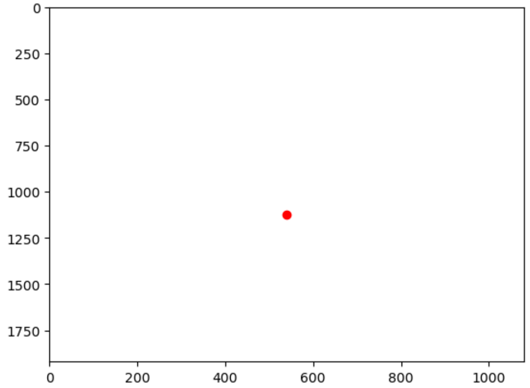
1.28, 83.10



150, 1608



25.34, 95.20

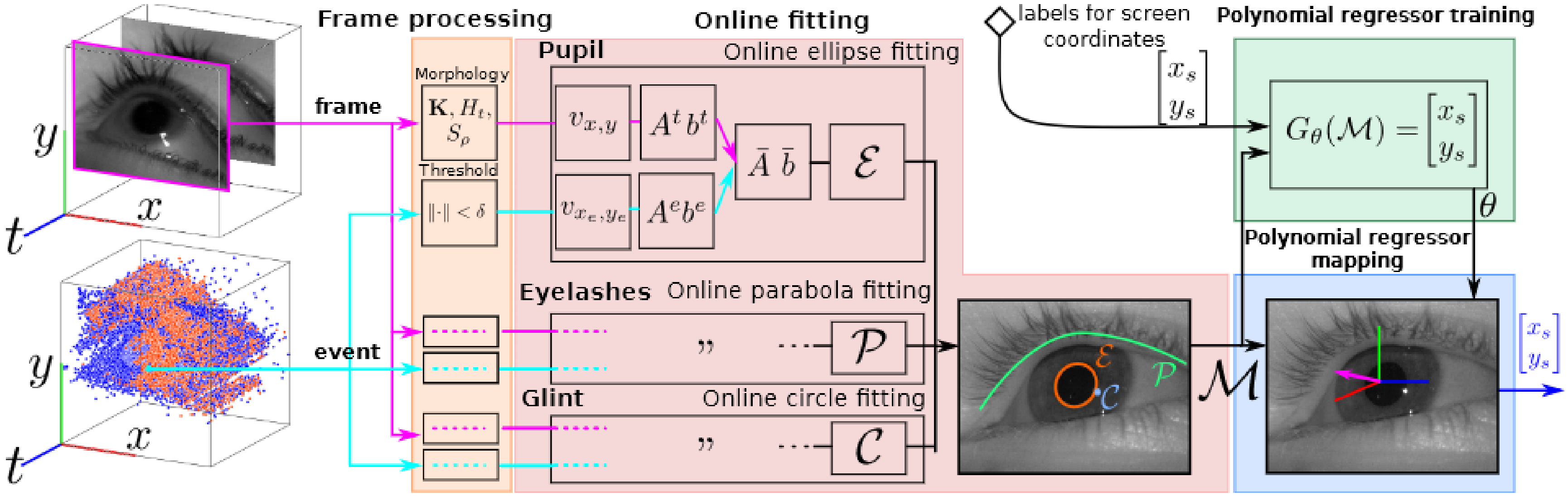


540, 1122

Next week

- Dynamic demo
- Personal data process and test
- More event and remote eye tracking papers

Pipeline



Finished
- Parabola eyelash model fit (frame/event)
- Circle glint model fit (frame/event)
- Blink detection (frame)

Bug fixing
- Ellipse pupil model fit (frame/event)
- Screen mapping polynomial regressor

Under processing
- Accuracy and Precision Examination

Dynamic demo

Problems:

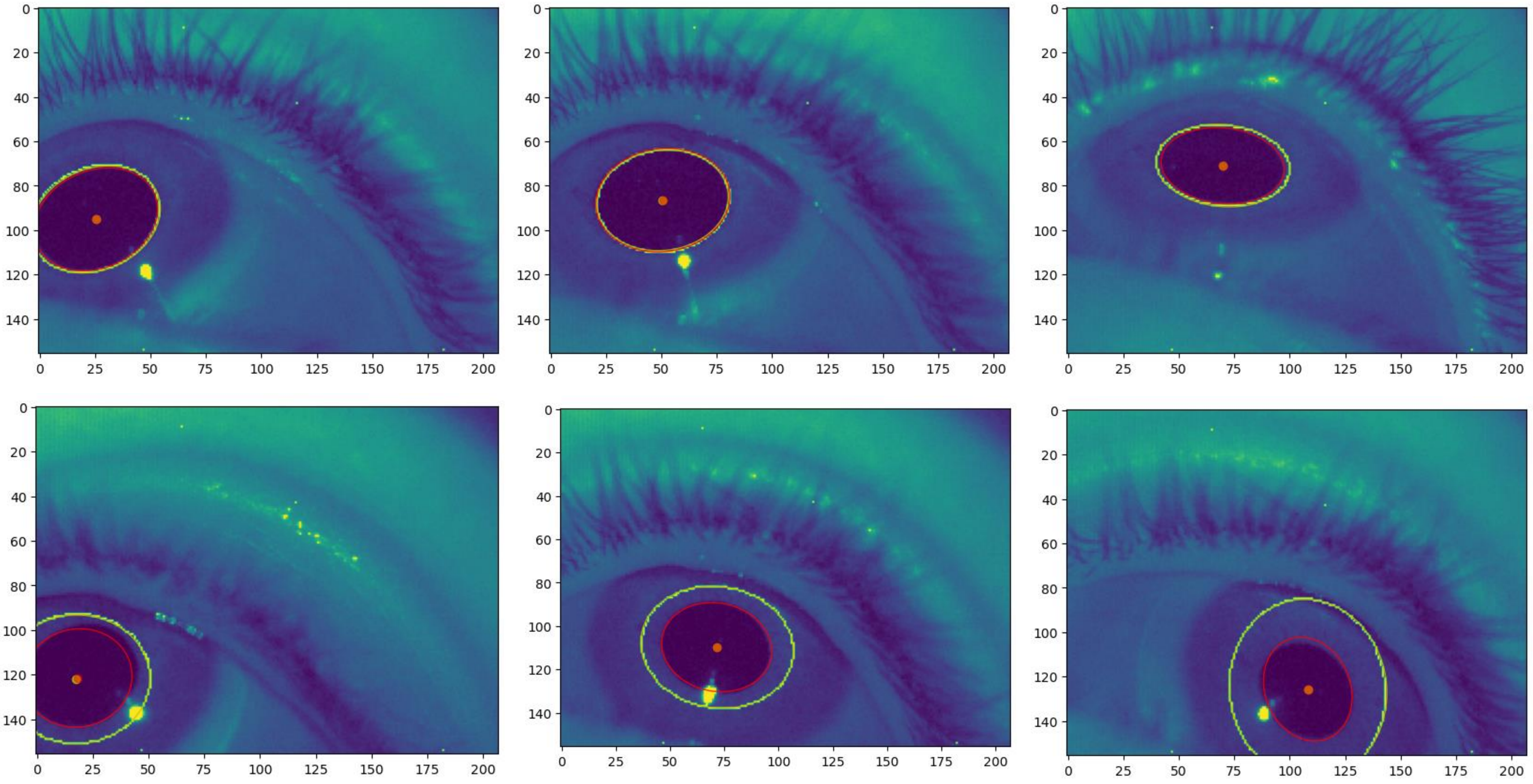
- Between frames, events fit is not reliable
  - With parameter filter distance=2, update event num = 50 (20 in paper)
  - Decrease distance to get better ellipse, but not sensitive to eye move
  - Increase update event num to get better ellipse, but react more slowly
- when near the blink, massive chaos events may fit ellipse badly m
- When blink, can’t fit a valid ellipse, then blink detector is unusable

Ellipse pupil model fit

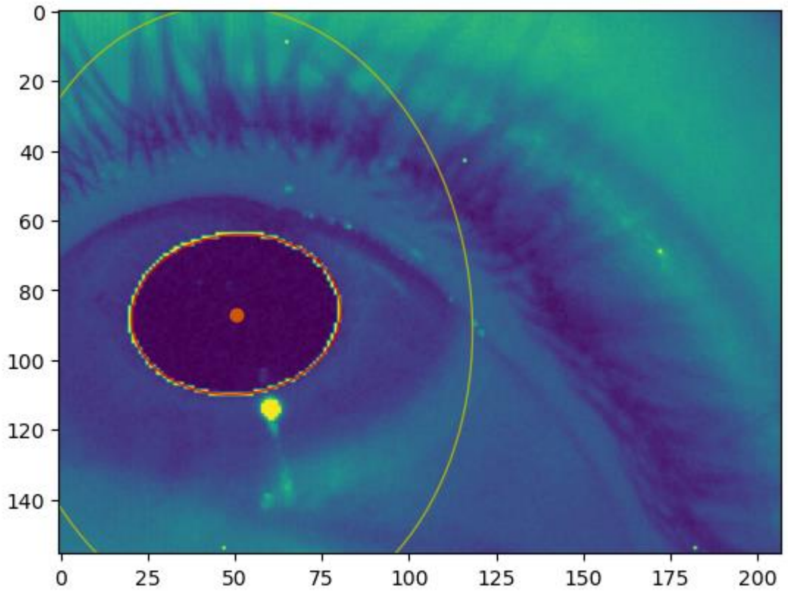
Red circle: cv2 ellipse
Yellow circle: paper method

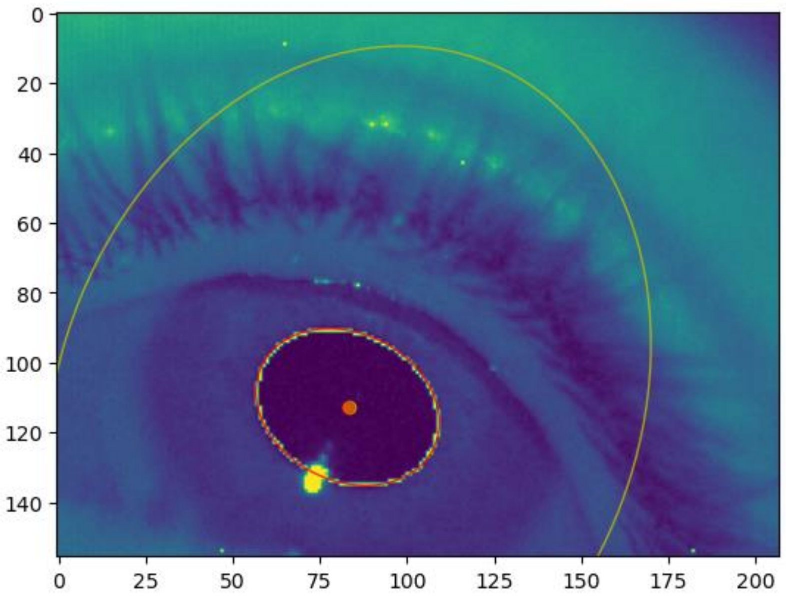
Good fit

Bad fit



Ellipse pupil model fit





Process

- Above img: direct fit result of paper method. Compared with cv2
- Processed the result of paper method to fit the cv2 result
- a = a/3, b = b/3
- But not suitable for some other frames.

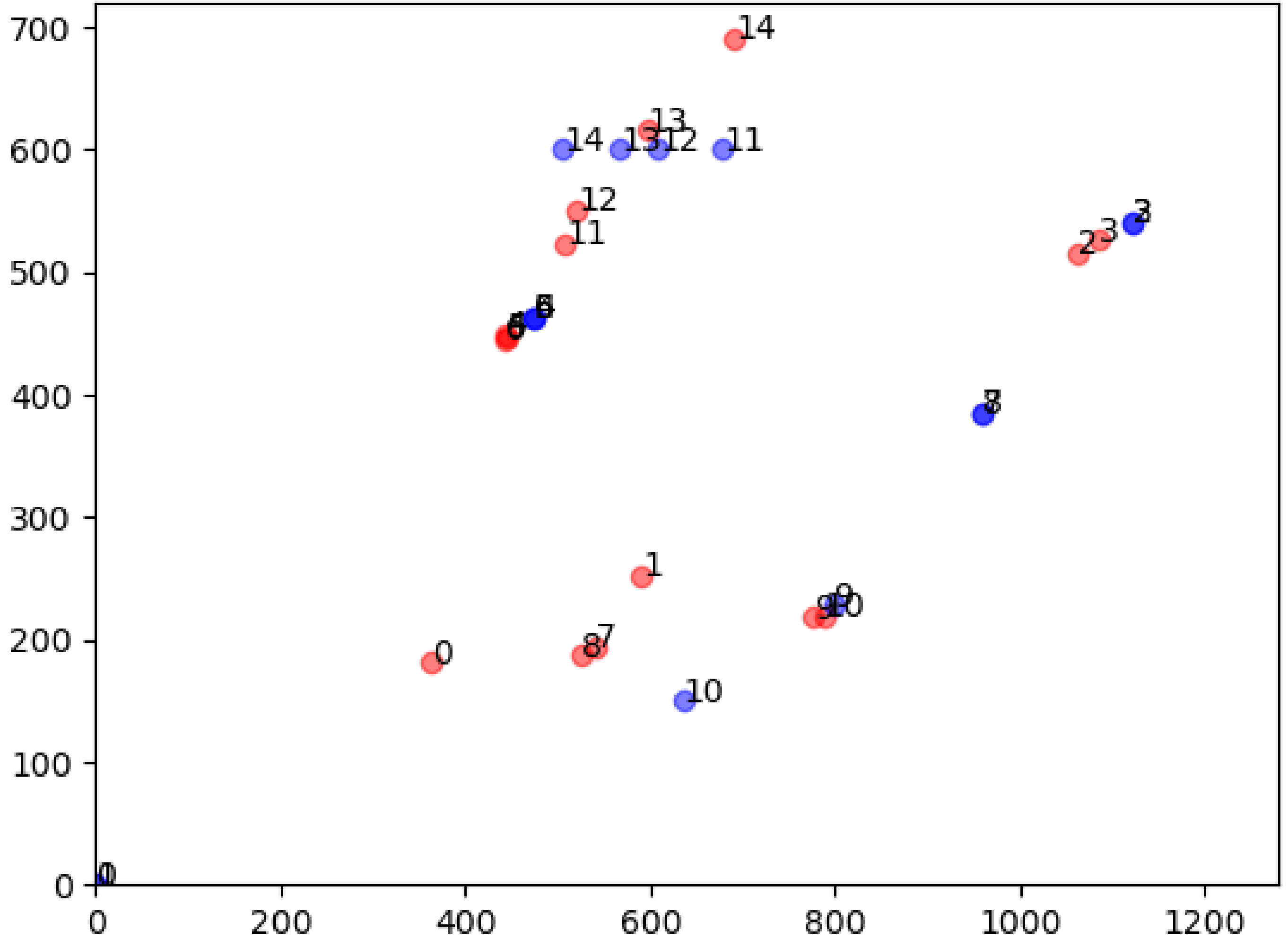
Screen map

$$G_\theta(\mathscr{E}) = \begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} G_{\theta^1}|_x(\mathscr{E}) \\ G_{\theta^2}|_y(\mathscr{E}) \end{pmatrix}$$

$$\arg\min_{\theta^1} \|G_{\theta^1}|_x(x_e, y_e) - x_s\|^2$$

$$\arg\min_{\theta^2} \|G_{\theta^2}|_y(x_e, y_e) - y_s\|^2$$

$$G_{\theta^i}|_{x/y}(x_e, y_e) = \alpha_i x_e^2 + \gamma_i x_e y_e + \beta_i y_e^2 + \varepsilon_i x_e + \zeta_i y_e + \eta_i$$



Red points: map

Blue points: tag

Used selected 30 frames to train,
And tested on the same frames.

Process:
For x_s
A = Mat of stack [above func of x_e]
b = vec of some x_s
Solve for p=[a,b,c,d,e,f]
p = np.linalg.lstsq(A, b)[0]

Dynamic Demo

- Improvement:
  - Remove events on parabola before update the ellipse: not that useful
  - Limited the a, b, center of ellipse in event update (+- 0.1)
  - Blink detection: by parabola difference of “c” between last frame and current events: $y = a(x+b)^2+c$

Parabola eyelid fit

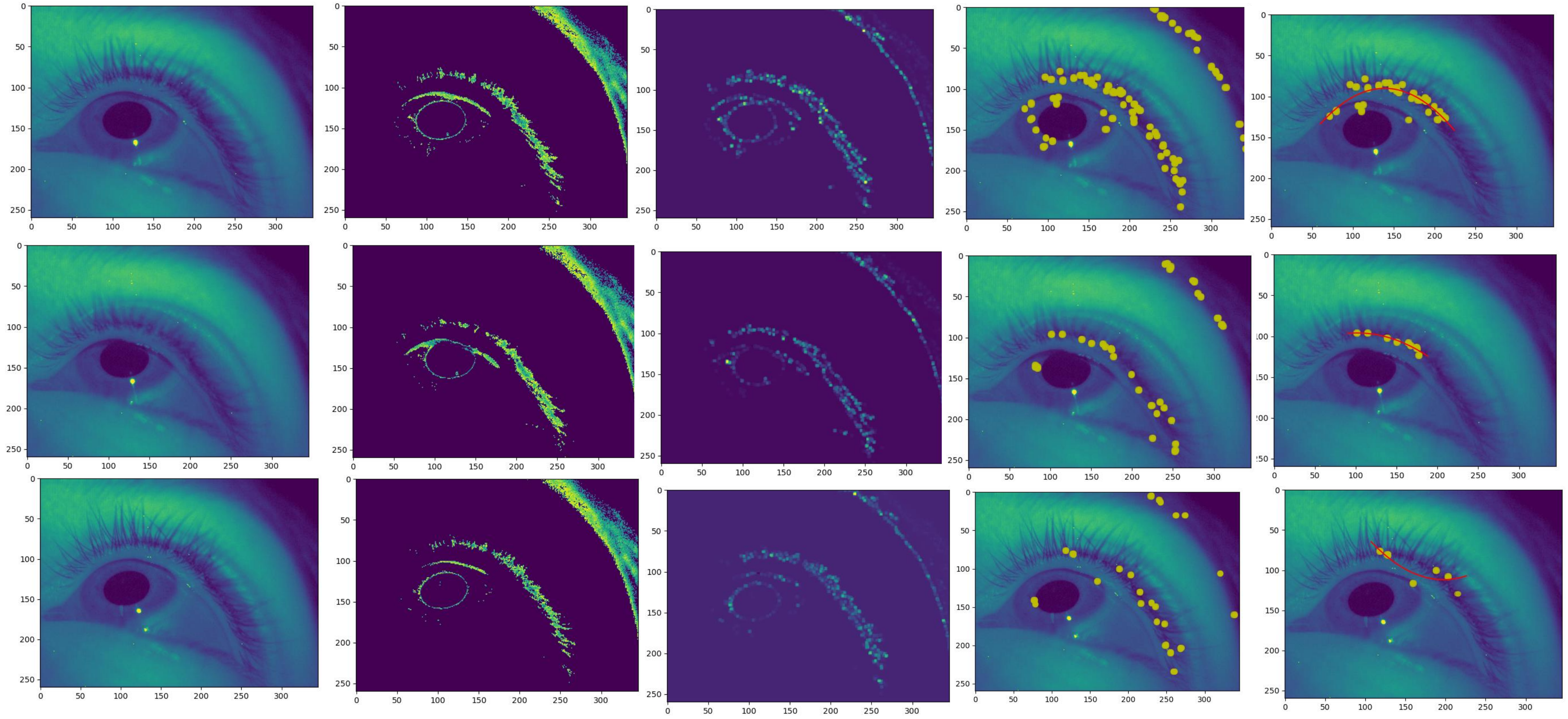
$$E_{\mathscr{P}}(x,y) = 0$$

$$\text{with} \quad E_{\mathscr{P}}(x,y) = a'\,y^2 + g'\,y + d' - x$$

$$\mathscr{D}'_{\text{img.}} = \Big\{ (x,y) \mid (x,y) \in \text{HarrisCorner} \circ \text{clip}(I, t_1, t_2), \\ \text{and } \|(x,y) - (x_e, y_e)\|^2 < \rho', \\ \text{and } y < \frac{\text{rows}}{2} \Big\}$$

Events fit better than frame

Bad frame fit examples:

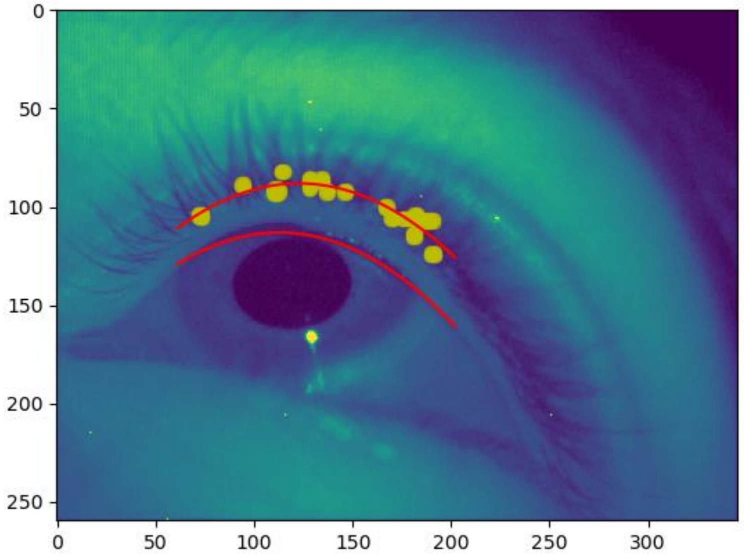


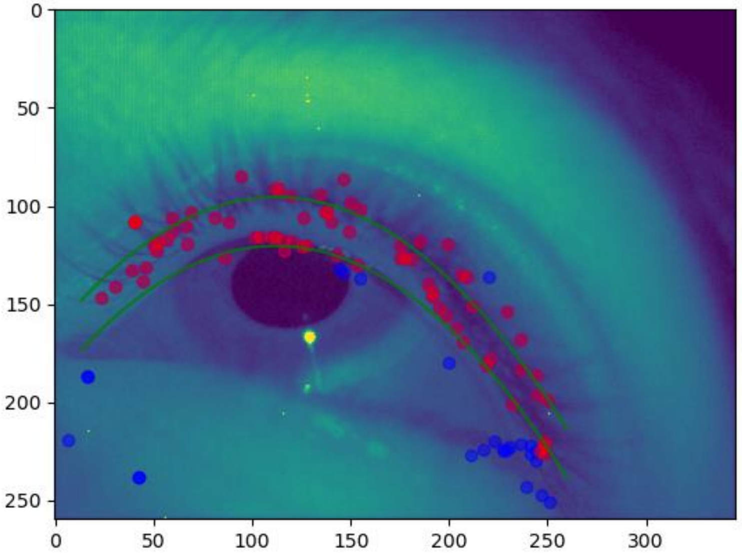
Consider fit the upper and lower curves of eyelash respectively, distinguish by distance to ellipse

Parabola eyelid fit

Update:

1. Add second lower eyelid curve, share parameters with first curve, and moving down and left (for the left eye) (Second eyelid helps filter events when fit pupil ellipse by events)
2. When fit eyelid parabola, remove corner points that too near or too far from the ellipse
3. Make the parabola curve must open downside





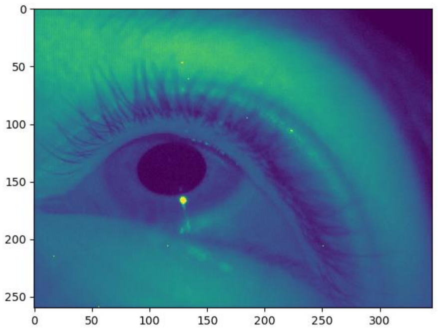
Tone Mapping

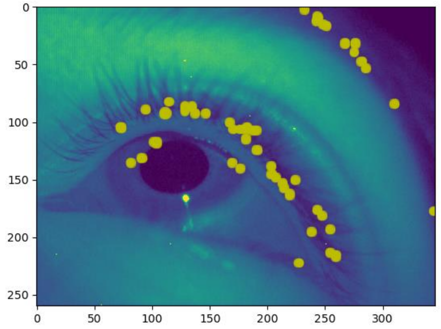
frame

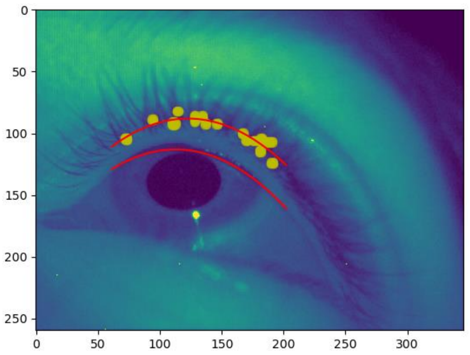
Corner points

Filter and fit

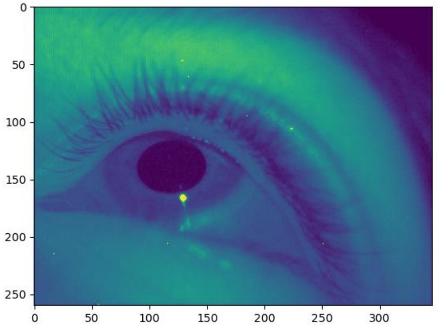
Normal

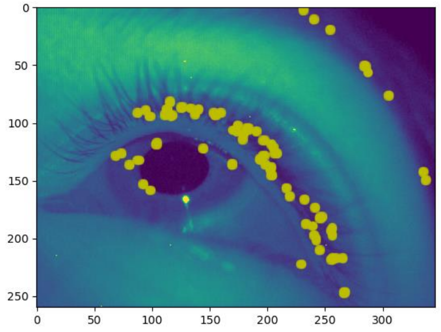


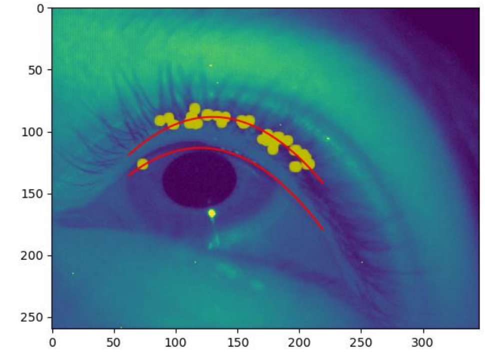




Tone Mapping







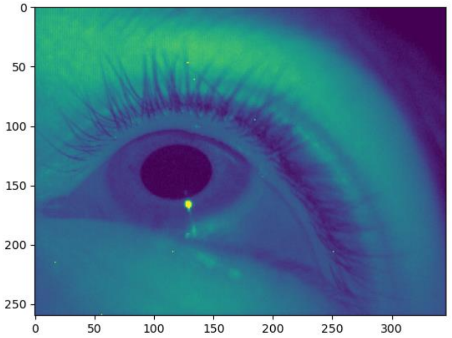
Tone Mapping

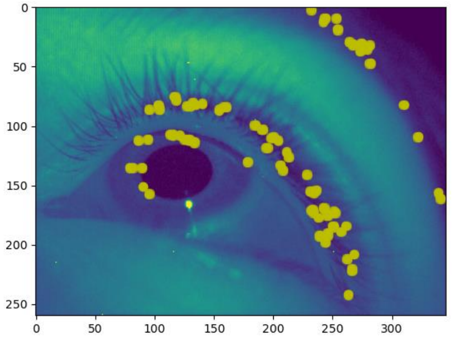
frame

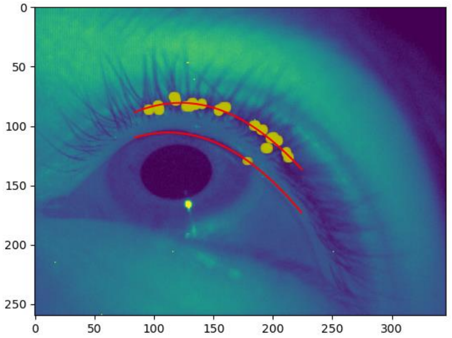
Corner points

Filter and fit

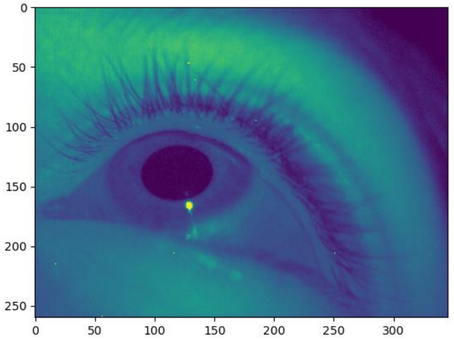
Normal

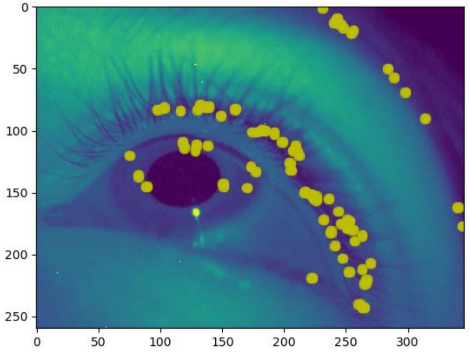


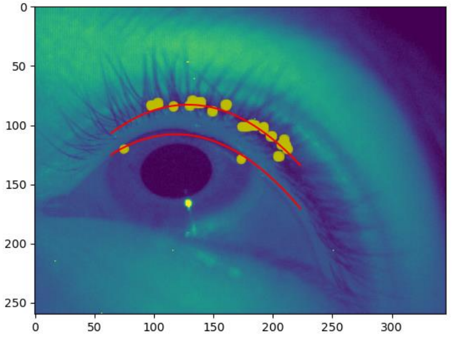




Tone Mapping





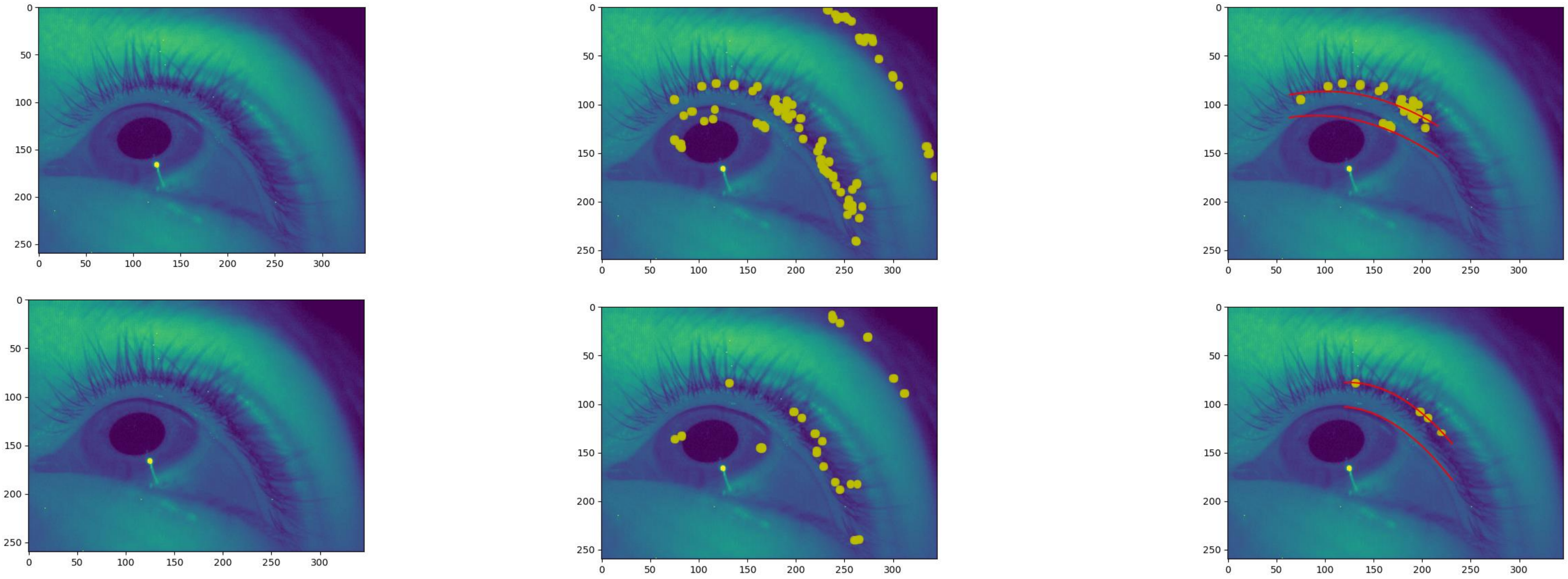


Tone Mapping

frame | Corner points | Filter and fit

Normal

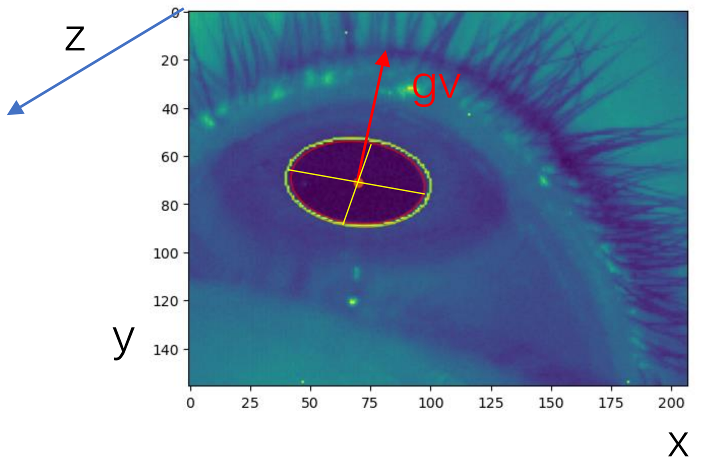
Tone Mapping

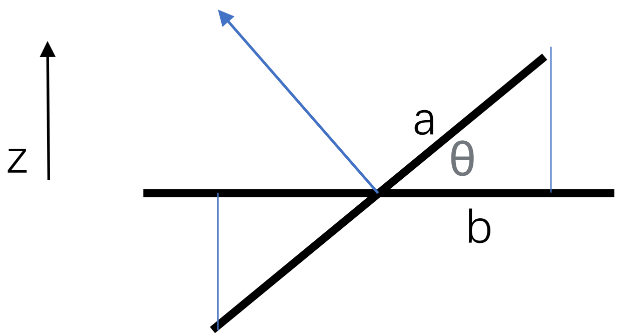


Screen Map

Original Paper: $G_\theta(\mathscr{E}) = \begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} G_{\theta^1}|_x(\mathscr{E}) \\ G_{\theta^2}|_y(\mathscr{E}) \end{pmatrix}$ $\quad \arg\min_{\theta^1} \|G_{\theta^1}|_x(x_e, y_e) - x_s\|^2$, $\arg\min_{\theta^2} \|G_{\theta^2}|_y(x_e, y_e) - y_s\|^2$ $\quad G_{\theta^i}|_{x/y}(x_e, y_e) = \alpha_i x_e^2 + \gamma_i x_e y_e + \beta_i y_e^2 + \varepsilon_i x_e + \zeta_i y_e + \eta_i$

Gaze Vec:

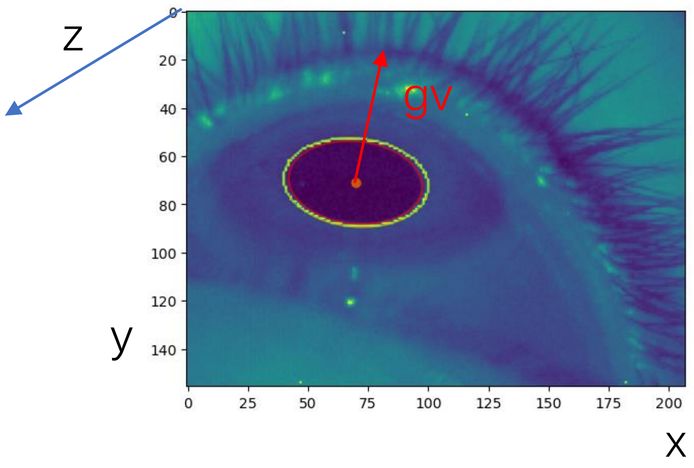




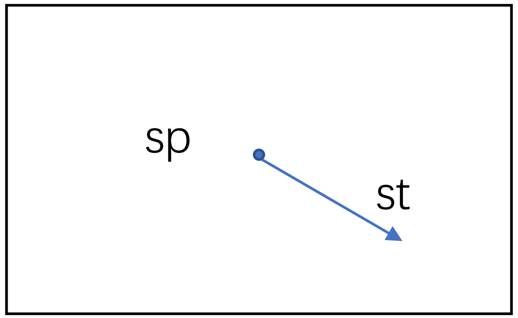
cos θ= b/a

Screen Map

Gaze Vec:



Screen Plain:



sp = (x0,y0,z0)
st = (x1,y1,z1)
Transformation Matrix = [2*3]

intersection points (xi,yi,zi) = (gv+st) * |sp|
pixel tag (xs, ys) = TransMat * (xi, yi, zi)

Last Meeting

Previous work:

- Run without intermediate frames

Problems:

- Map Gaze to Screen
- Kalman Filter

Map Gaze to Screen

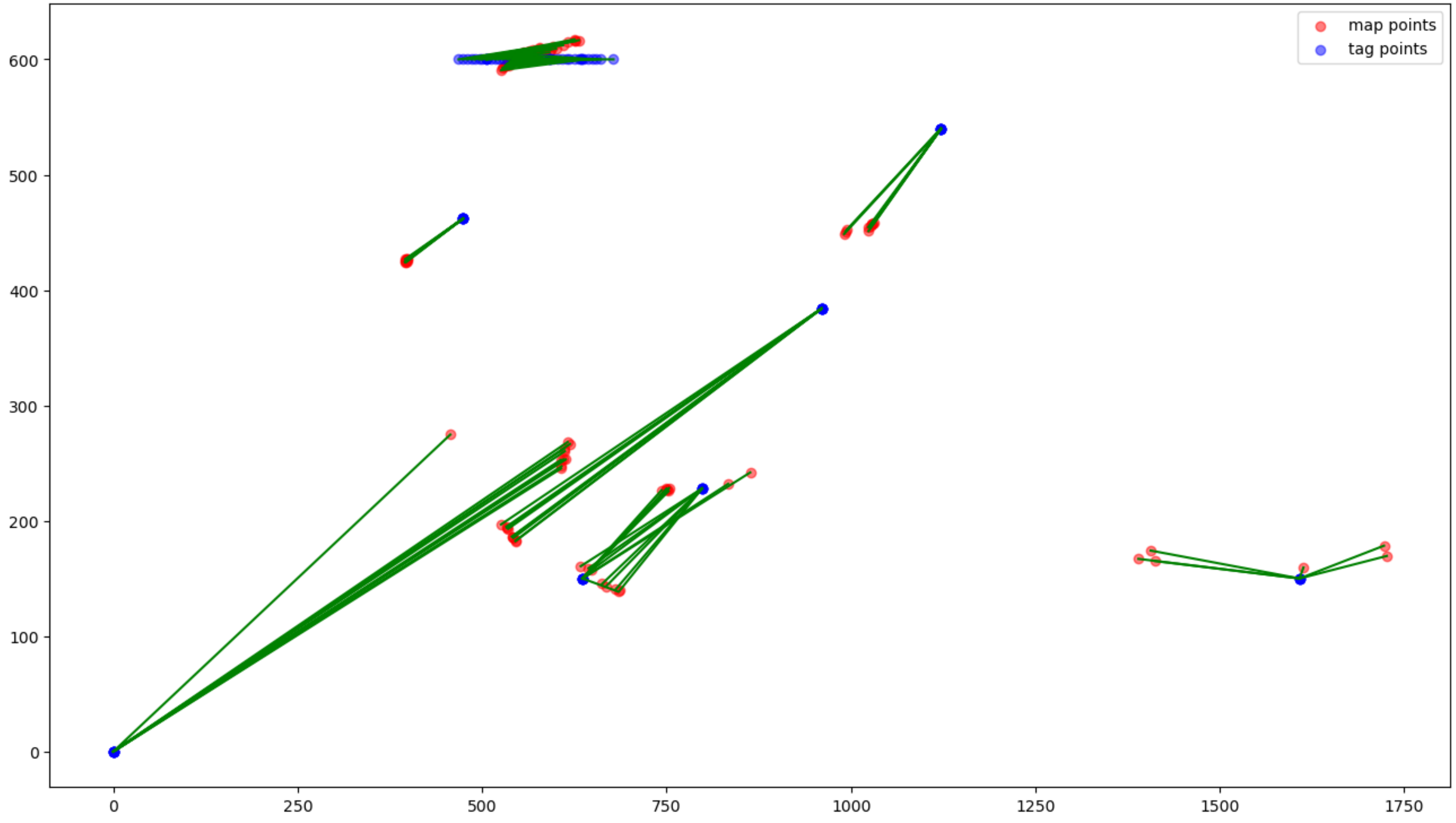
$$G_\theta(\mathscr{E}) = \begin{pmatrix} x_s \\ y_s \end{pmatrix} = \begin{pmatrix} G_{\theta^1}|_x(\mathscr{E}) \\ G_{\theta^2}|_y(\mathscr{E}) \end{pmatrix}$$

$$\arg\min_{\theta^1} \|G_{\theta^1}|_x(x_e, y_e) - x_s\|^2$$

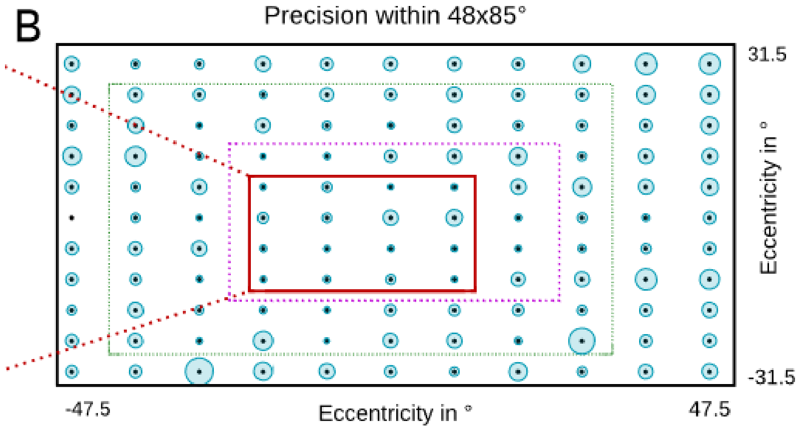
$$\arg\min_{\theta^2} \|G_{\theta^2}|_y(x_e, y_e) - y_s\|^2$$

$$G_{\theta^i}|_{x/y}(x_e, y_e) = \alpha_i x_e^2 + \gamma_i x_e y_e + \beta_i y_e^2 + \varepsilon_i x_e + \zeta_i y_e + \eta_i$$

- Paper Method



p = [a, b, c, d, e, f]
A = [x^2, xy, y^2, x, y , 1]
B = [x_s, y_s]
Ap=B
np.linalg.lstsq(A,b), train set: 200



Kalman Filter

$x_k = A * x_{k-1} + B * u_k + w_{k-1}$ -------(1)

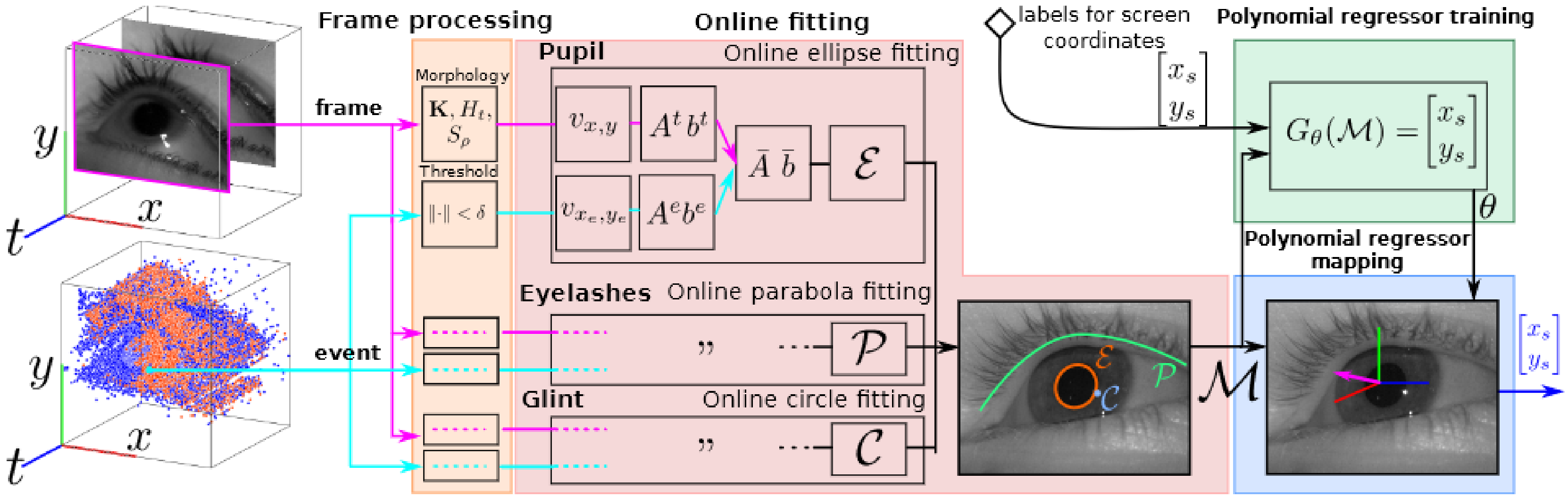
$z_k = H * x_k + v_k$ --------------------(2)

Correct and Predict the ellipse for each fit:

$$x_k = \begin{bmatrix} x_e & y_e & a & b & \theta \\ dx_e & dy_e & da & db & d\theta \end{bmatrix}$$

$$x_{ek} = x_{ek-1} + \Delta t \, dx_{ek-1}$$

Summary



- Add corner point filter step on frame eyelid fit
- Add second eyelid to help denoise event pupil fit
- Change blink detection to eyelid movement
- Change screen map to 3D coordinate system map

Kalman Filter

Correct and predict the ellipse for each fit:

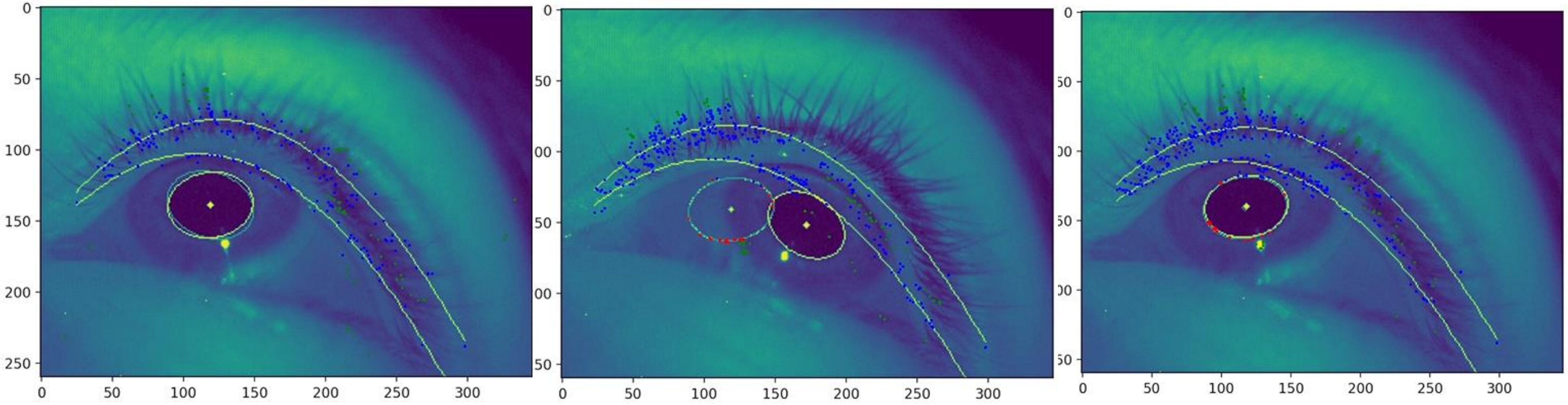
$$state: x_k = [x_e \quad y_e \quad a \quad b \quad \theta \quad dx_e \quad dy_e \quad da \quad db \quad d\theta]$$
$$measurement: z_k = [x_e \quad y_e \quad a \quad b \quad \theta]$$

$$x_k = A * x_{k-1} + B * u_k + w_{k-1} \text{-------(1)}$$

$$z_k = H * x_k + v_k \text{ -------------------(2)}$$

Strategy: discard new fitted pupil if it’s much different from the last prediction

Yellow Ellipse: Current Fit
Green Ellipse: Kalman Prediction



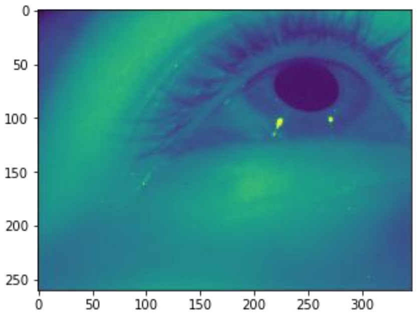
3 adjacent frames (the middle frame is replaced on purpose)

Problem on parameters setting

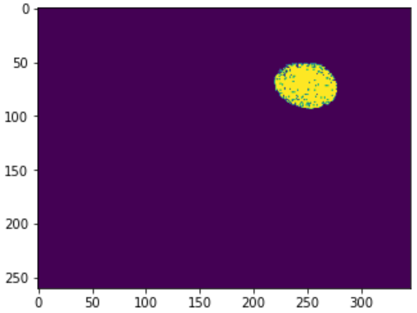
- Same group parameters can not run on all the data, need to set self-adaption, trying tone mapping, percentage parameter, ect.

On another eye side:

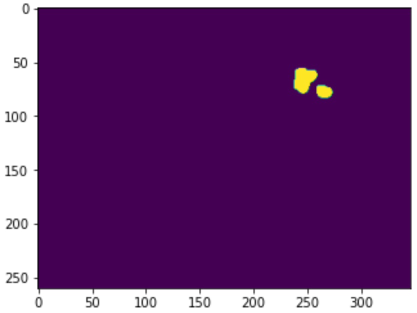
Frame pupil ellipse bad fit:



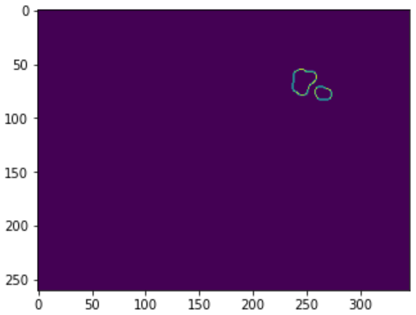
Frame



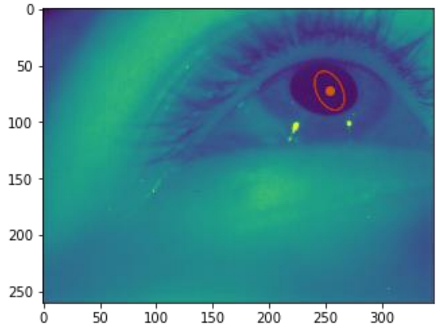
threshold



morphological opening



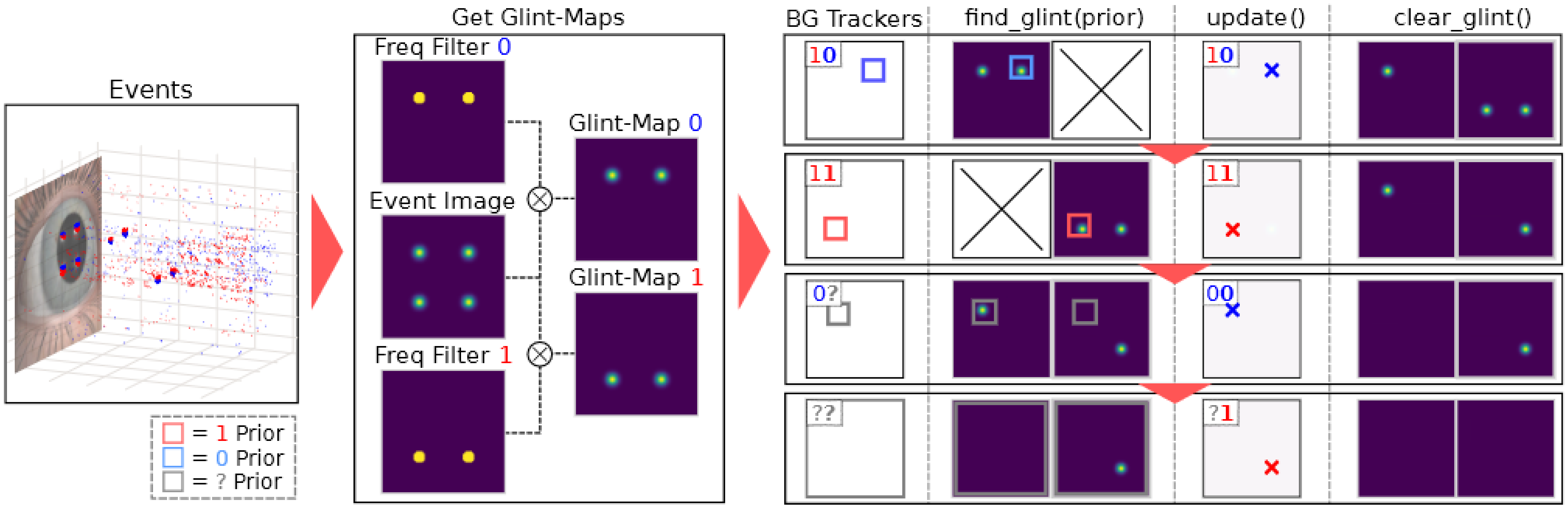
edge detect



Fit result

Next Week

- Start reproduction of Tracking Corneal Glints using Coded Differential Lighting
- Data collection



This week

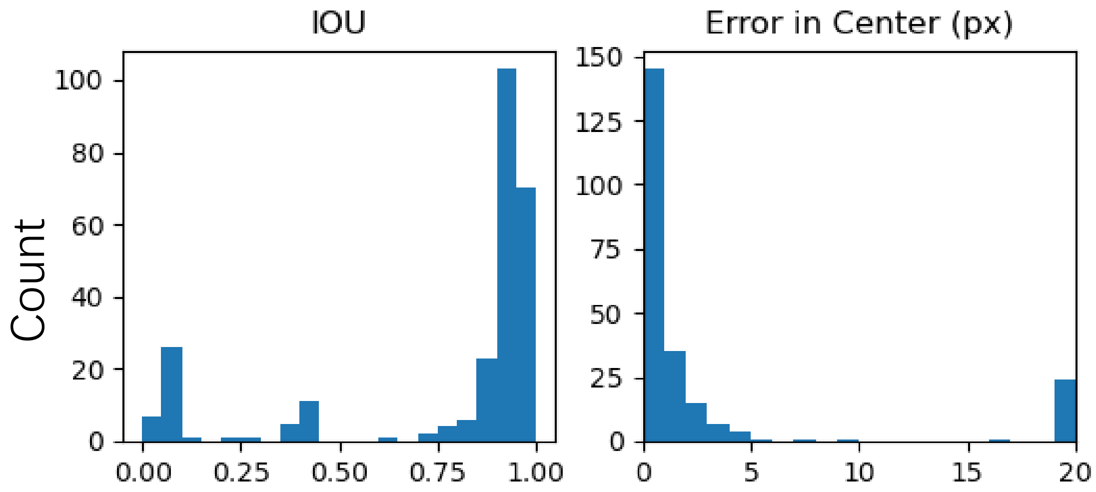
- Kalman Filter On eyelid, trying new strategy
- Pupil Tracking evaluation
- Gaze vectors analysis

Kalman Filter

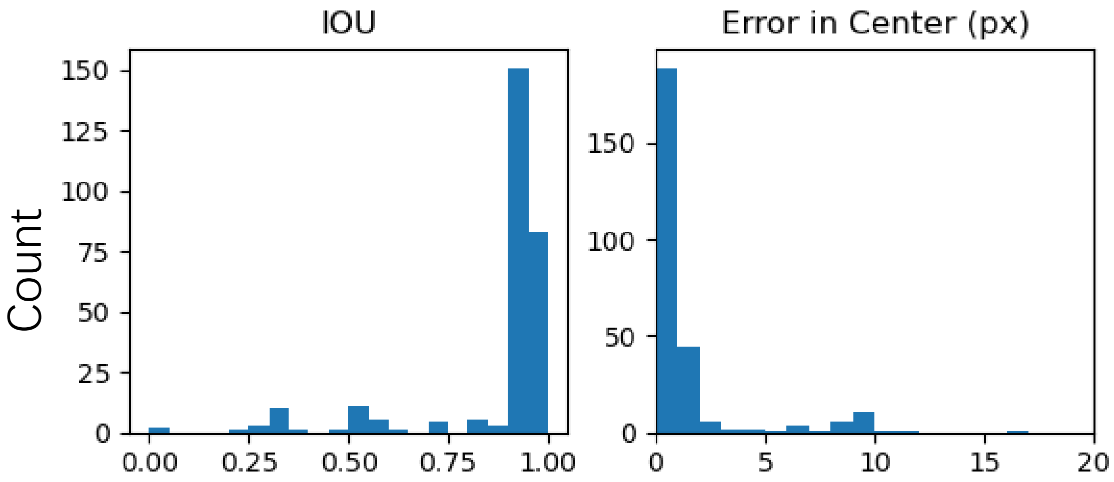
- Changed strategy: compare the prediction and current fitted ellipse, and set the less changed from last output as current output.
- Got worse result.

Pupil Tracking evaluation

Standard: The intersection over union (IOU) and The error in center estimation
Compare: Frame estimation & Event estimation (just before the frame)



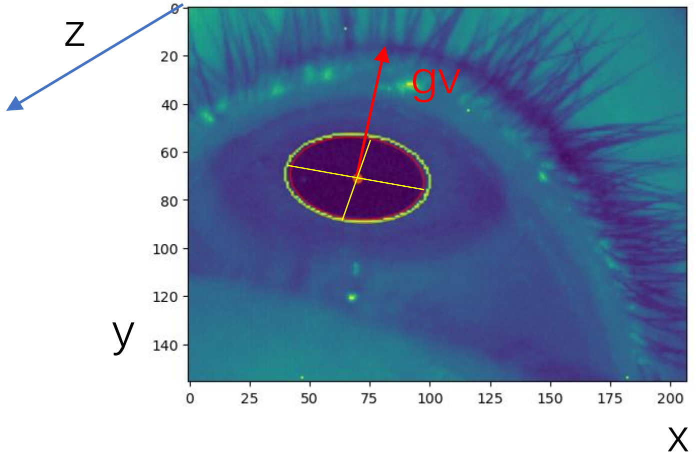
Paper Method

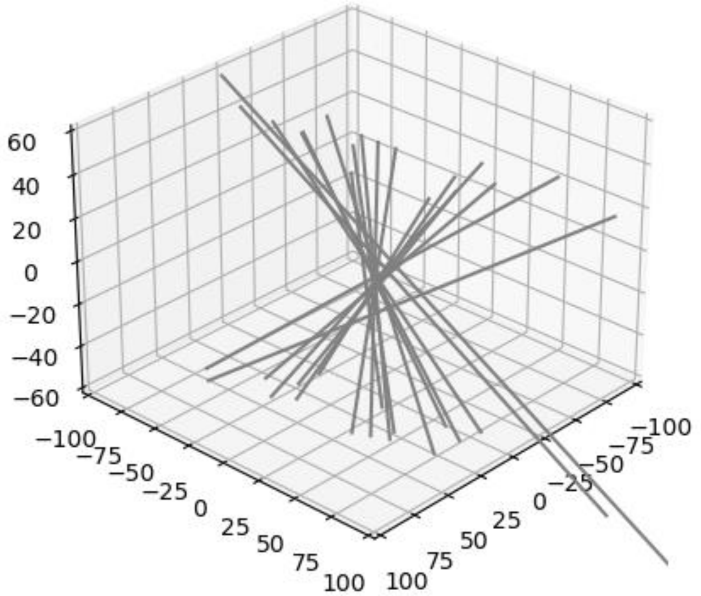


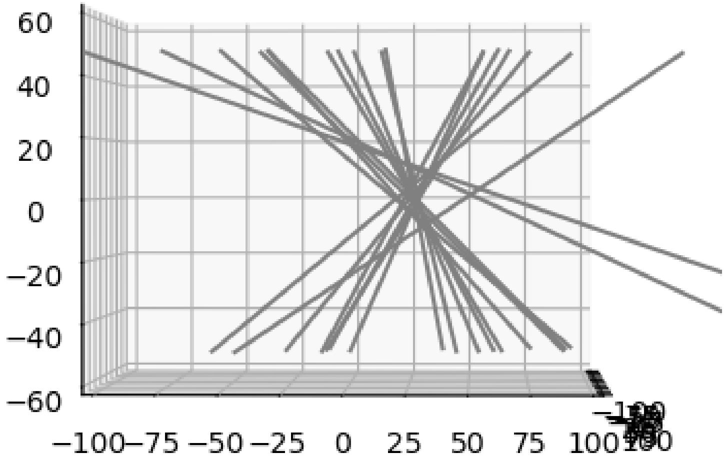
Our Method (without Kalman filter)

Gaze vectors analysis

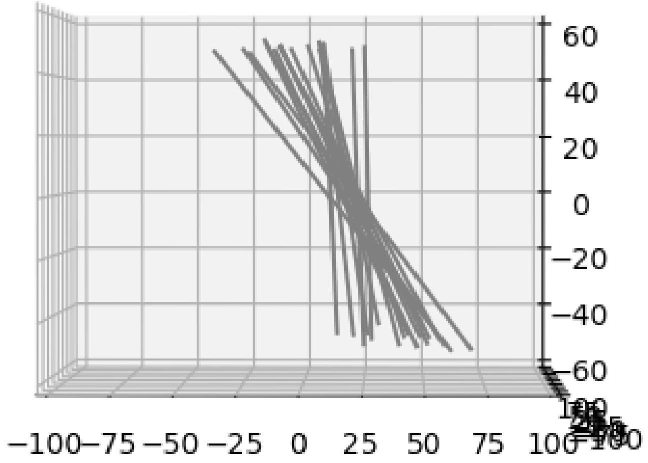
The x-y plane (z=0) is set as eye frame.
The lines are based on unit gaze vectors and pupil centers.





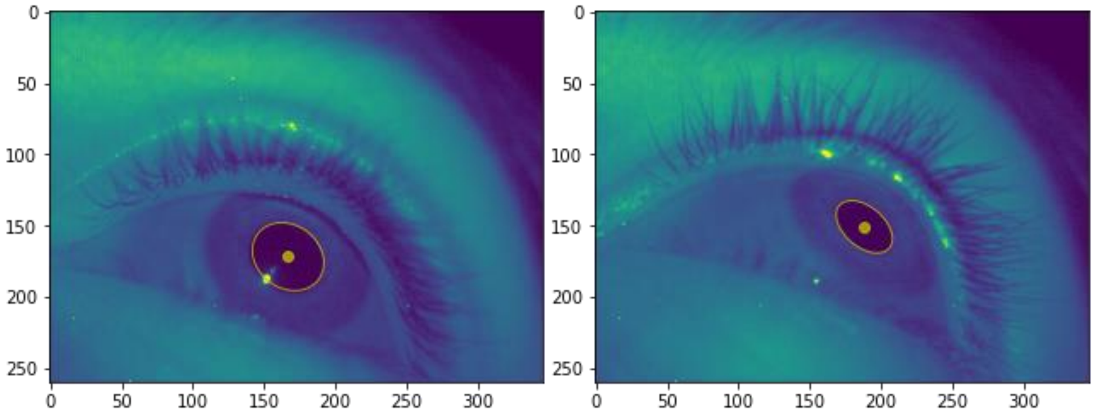


x-axis view



y-axis view

Gaze vectors analysis



Pupil size may change (not by angle)

Next week

- Finish the screen map evaluation
- Analysis gaze vectors and build advanced eye model