

- Problem1

- *Explanation:*

- 1. Filtered all the frames with an HSV filter. Used this mask to determine the contours of the ball. Found the center by averaging the points.
 - 2. Noted all the trajectory points of the center of a circle. Used the least approach to find the coefficient of the parabola using the following equation.

$$X^T X B = X^T Y \quad Nc$$

$$B = (X^T X)^{-1} (X^T Y)$$

- Noted the initial point of the trajectory and added 300 to it. Used “B Matrix” by least square to find the coefficient of the parabola equation. Put these values in the formula to find the roots of the equation.

- *Problems and solution:* Tracking HSV was difficult. Handled it by using the HSV tracker provided by OpenCV “Trackbars”

- Problem2 Covariance Matrix

- *Explanation:*

- 1. Read the CSV file and differentiated the x, y, and z data. Found means of each x_data, y_data, and z_data. Found the normalized x_data, y_data, and z_data by subtracting the mean from the original data. Used these normalized data to find covariance along xx, yy, zz, xy, yz, xz directions. Used the formula below to find the covariance matrix.

$$\Sigma = \begin{bmatrix} \sigma_{x_R}^2 & \sigma_{x_R} \sigma_{x_G} & \sigma_{x_R} \sigma_{x_B} \\ \sigma_{x_R} \sigma_{x_G} & \sigma_{x_G}^2 & \sigma_{x_G} \sigma_{x_B} \\ \sigma_{x_R} \sigma_{x_B} & \sigma_{x_G} \sigma_{x_B} & \sigma_{x_B}^2 \end{bmatrix}$$

- Found the eigenvectors of the covariance matrix. Then found the two dominating directions of data spread. These directions give an axis along which most of the data is spread. Then computed normal to these directions by the cross product of these two vectors.

- Problem2_LS

- *Explanation:*

- Read the CSV file and differentiated the x, y, and z data. Found means of each x_data, y_data, and z_data. Combined x_data, y_data, and vector of ones to get the matrix X. Y was constructed from z_data. Used the

equation below to get the LS solution of the data.

$$X^T X B = X^T Y \quad Nc$$

$$B = (X^T X)^{-1} (X^T Y)$$

- *Problems faced and solution:* Unable to plot the data due to some corrupt dependencies of pyQT. Tried reinstalling the OenCV did not work
- Problem2_TLS
 - *Explanation:*
 - Read the CSV file and differentiated the x, y, and z data. Found the normalized x_data, y_data, and z_data by subtracting the mean from the original data. Used these normalized data to find the “U matrix” by combining normalized x_data, normalized y_data, and normalized z_data. Found the eigen values and eigen vectors of $U^T \cdot U$. The B matrix was found by eigenvectors corresponding to the least eigenvalue.
- Problem2_Ransac
 - *Explanation:* Read the CSV file and differentiated the x, y, and z data, then following are the parameters used

```
p = 0.95
t = 13
num_outlier = 6
max_inlier_count = 0
e = num_outlier/x_data.size
min_num_data_points = 3
N = int(math.log(1-p)/math.log(1-1/3))
```

Calculated the maximum number of iterations. For each iteration, sampled 3 random points. Calculated plane fitting those three points using LS approach. Then calculated error assigned with this plane

Probelms faced : getting random number and finding optimal solution