Locus Determination

REN HANFEI
任 含菲

CS3218 Final Project

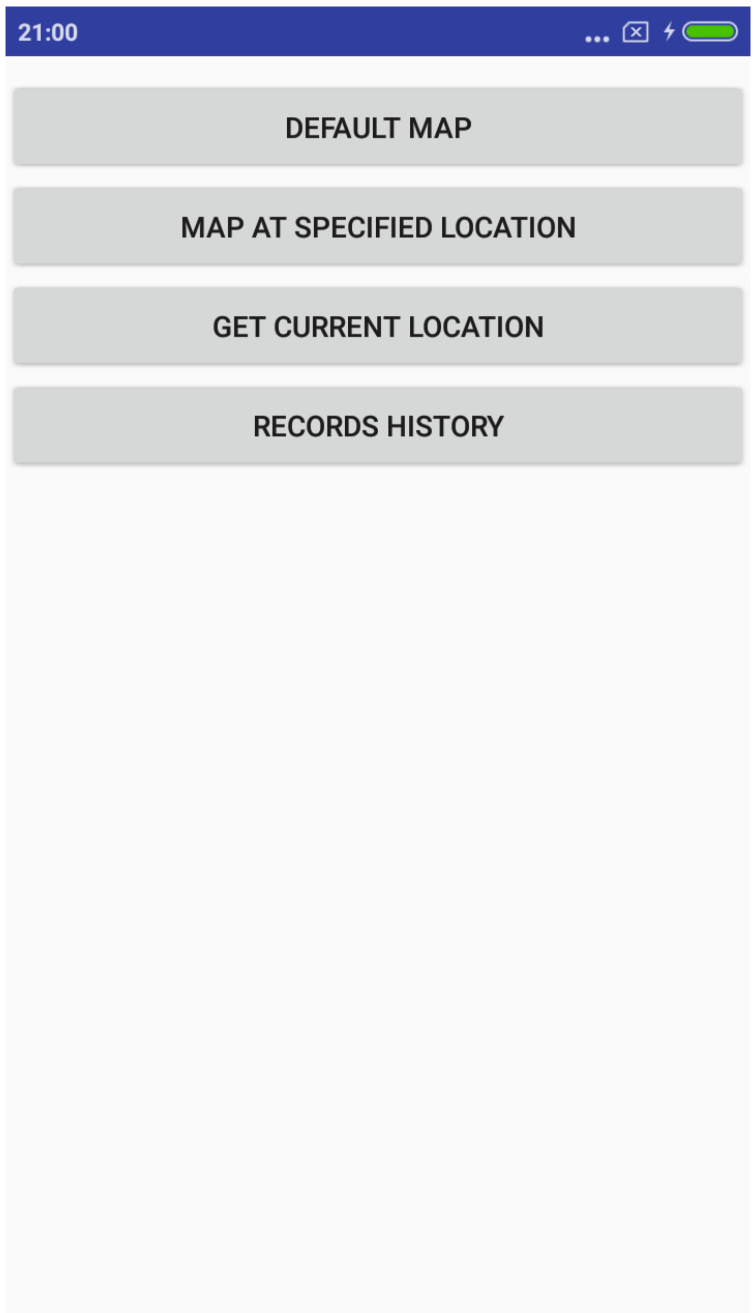
GPS Error Determination

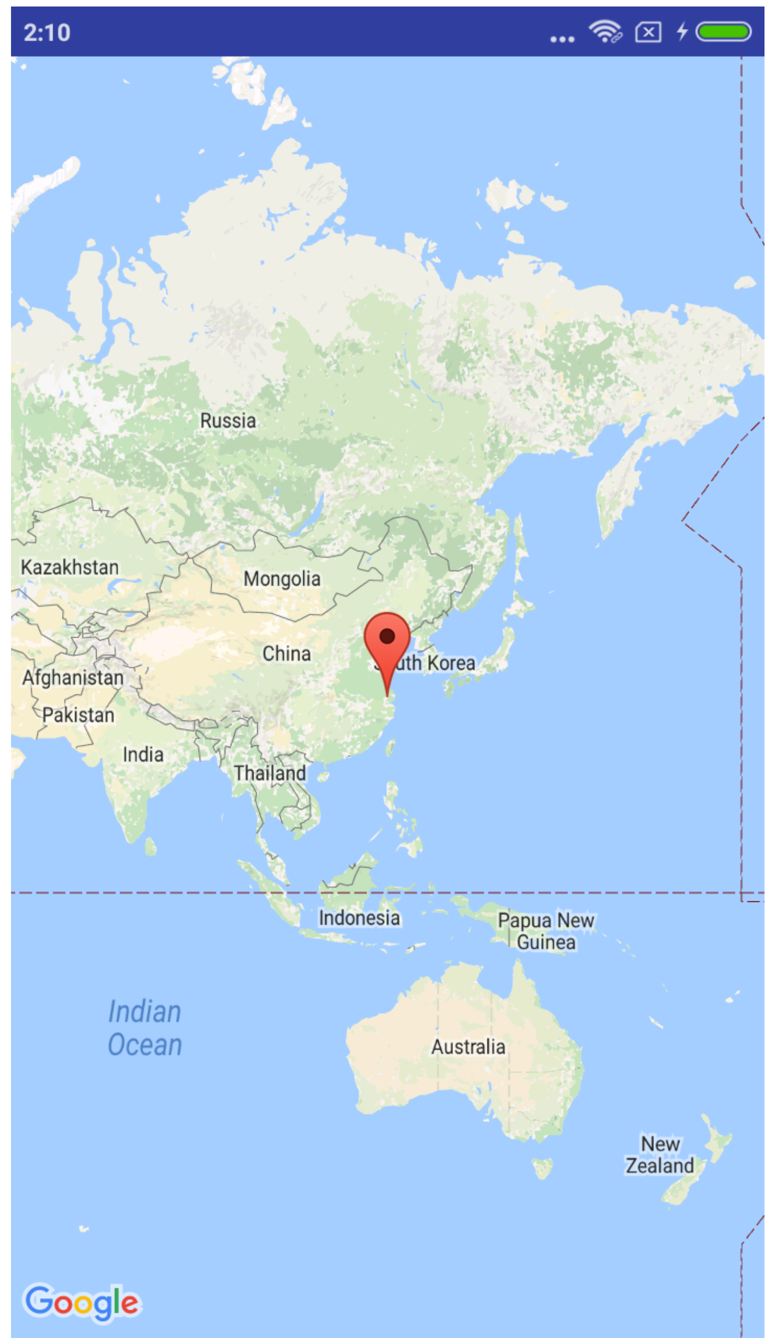
- Database Structure
- User Interface
- Experimental Process
- Calculation
- Result

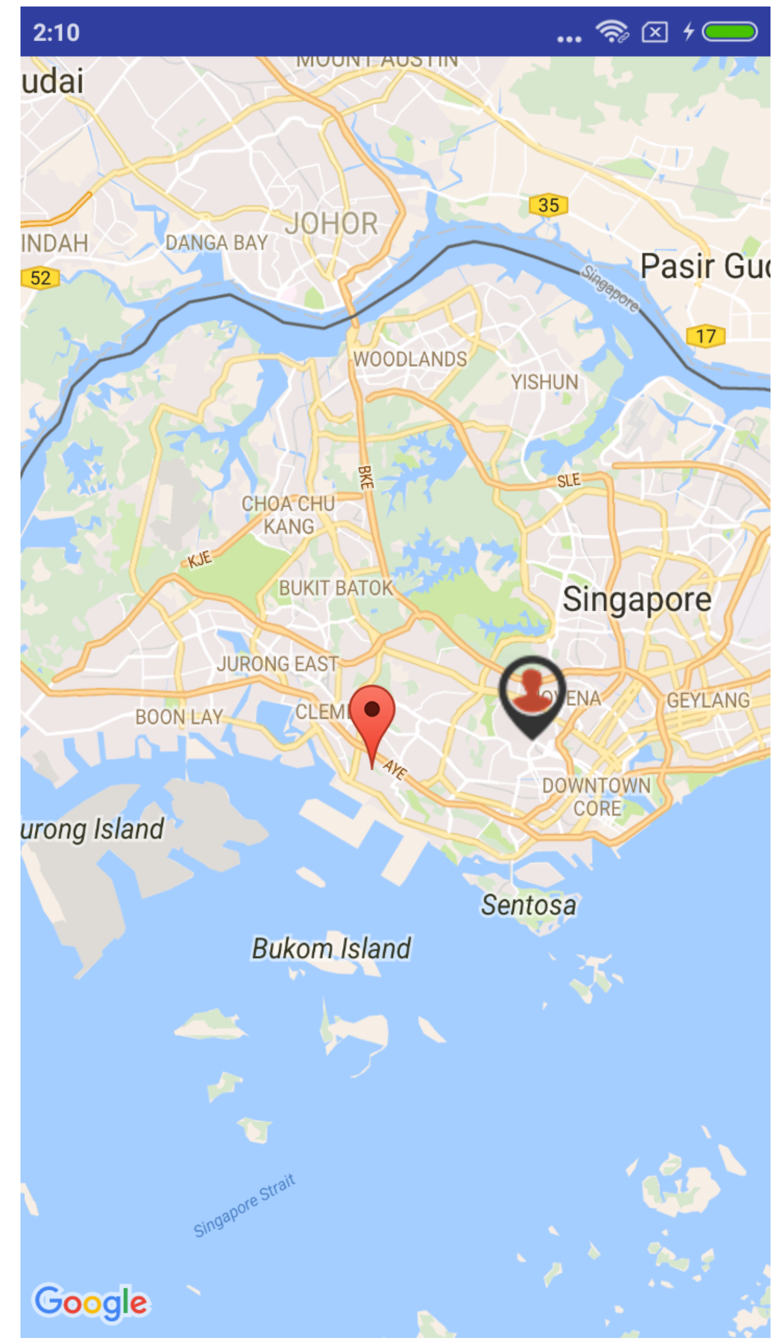
Database Structure

Column	Type	About
id	long	auto increment
time	text	YY-MM-DD HH:mm:ss
position	text	east / south / west / north
latitude	text	ensure precision
longitude	text	
address	text	

User Interface

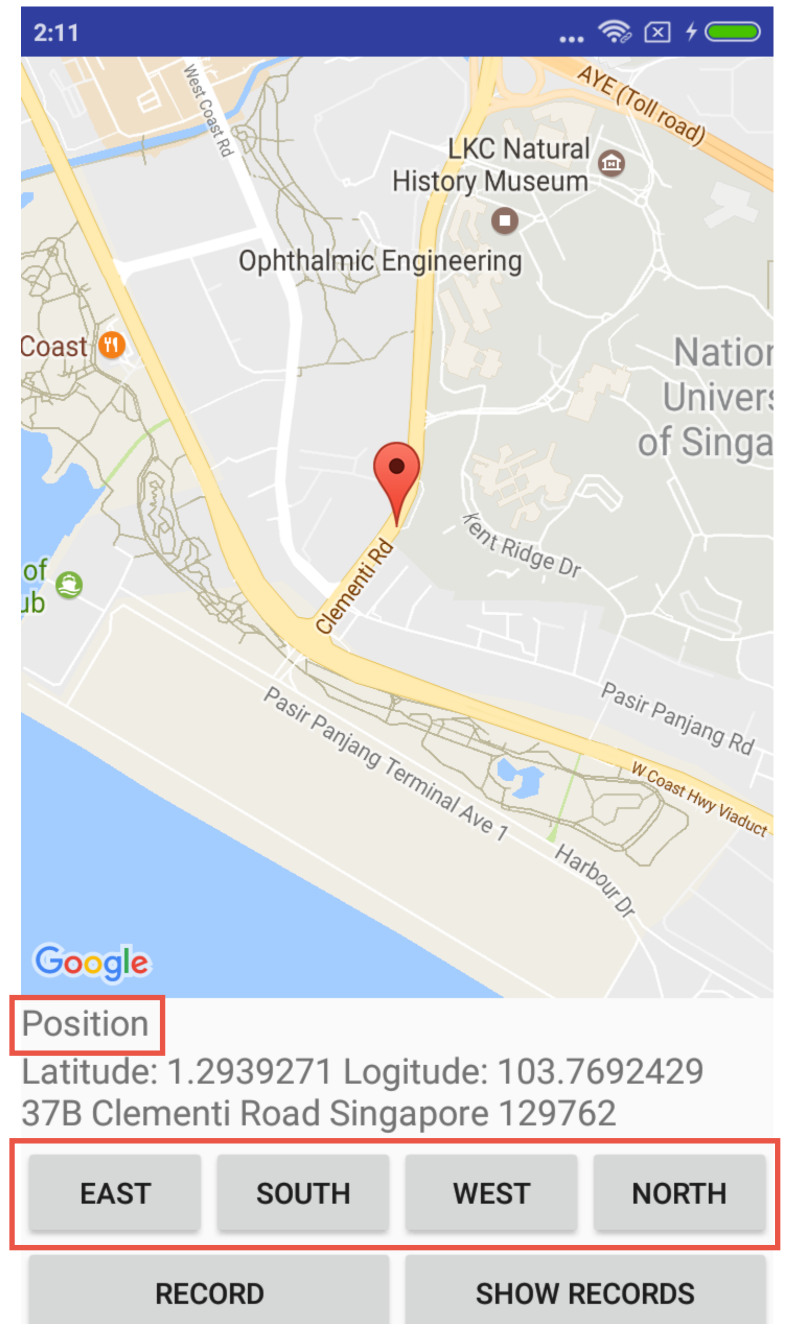


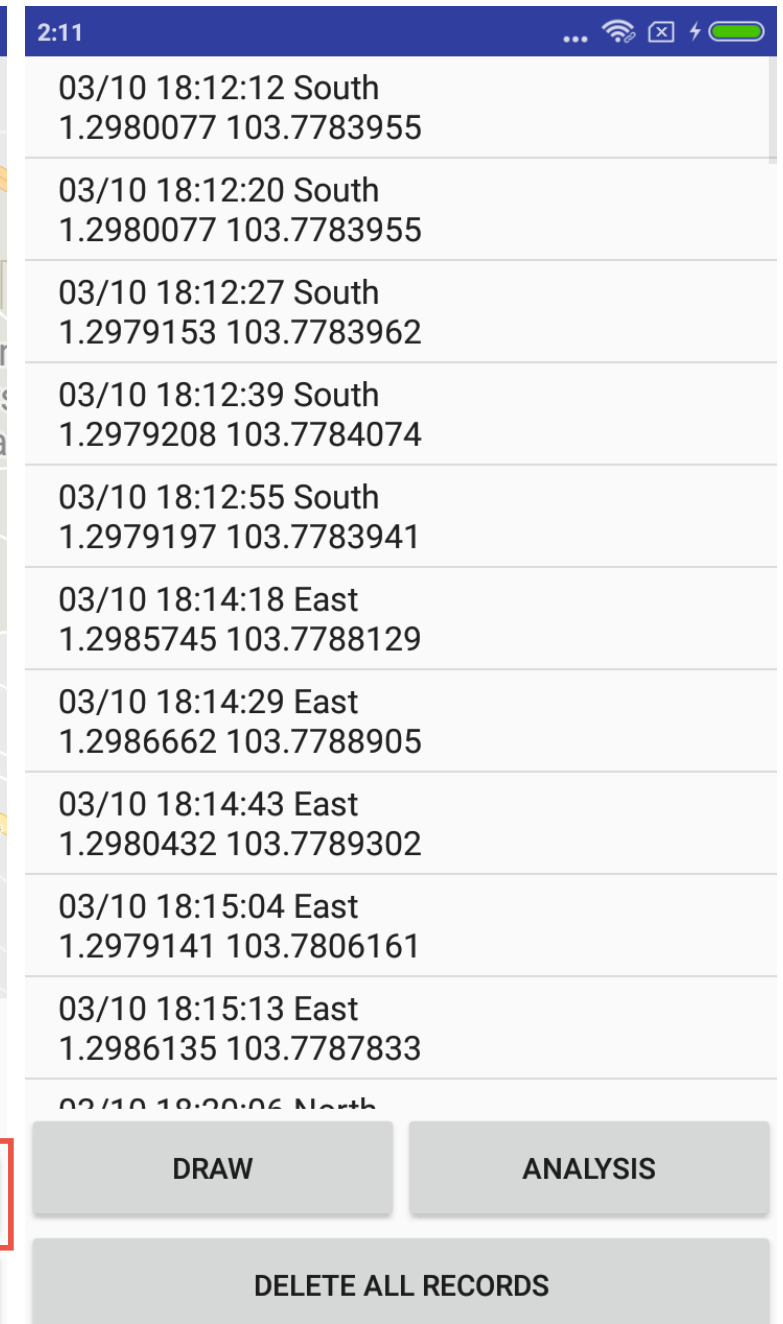




Record Interface

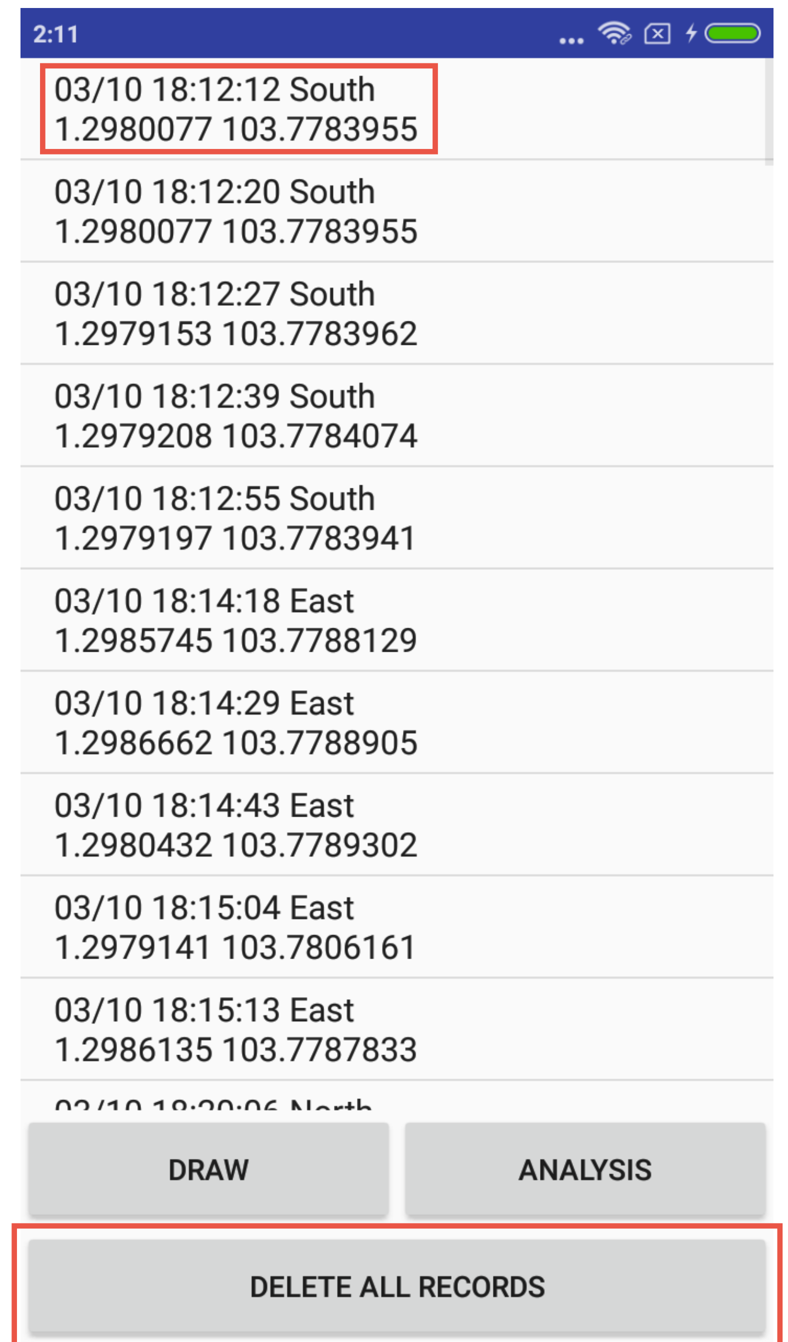
- Click east / south / west / north to change the record position.
- Clicked position is shown above coordinates.
- Coordinates recorded with different position will be drawn in different color.
- “Show Records” as the right.

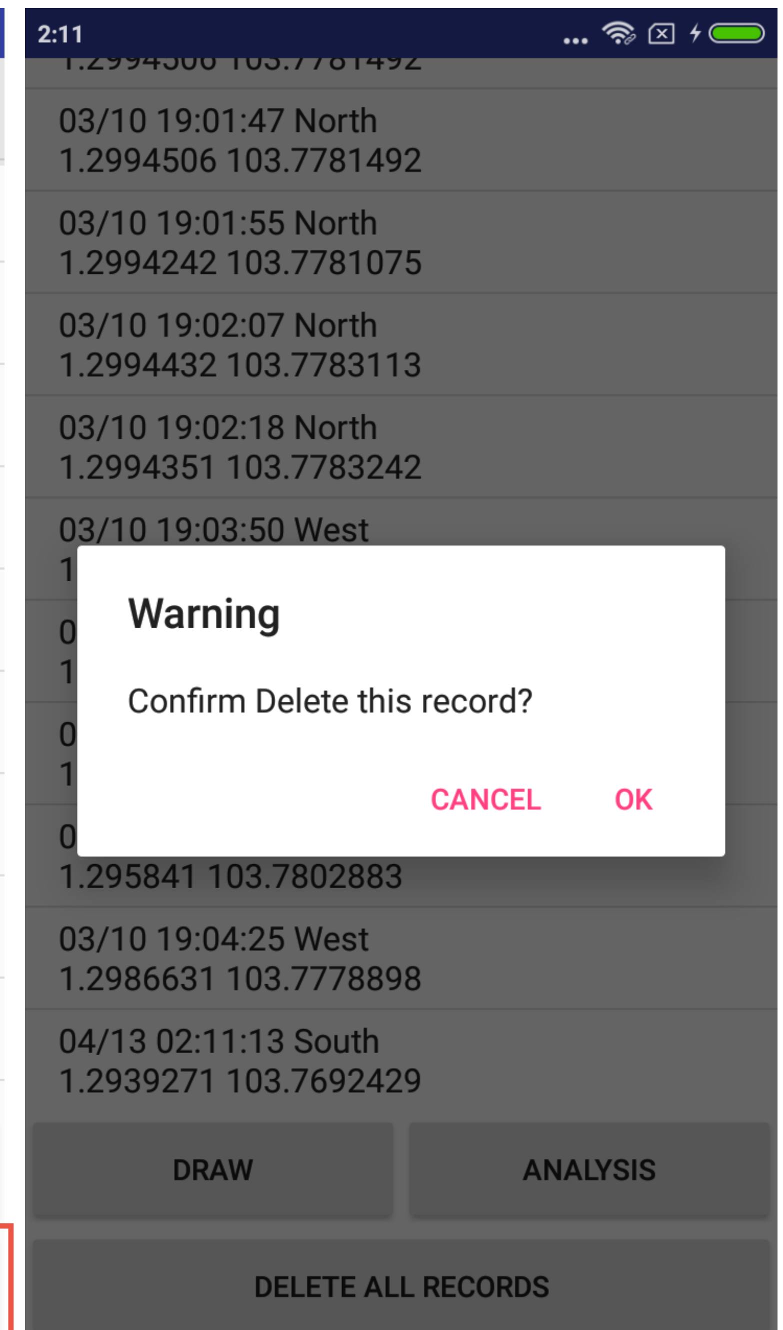




History Interface

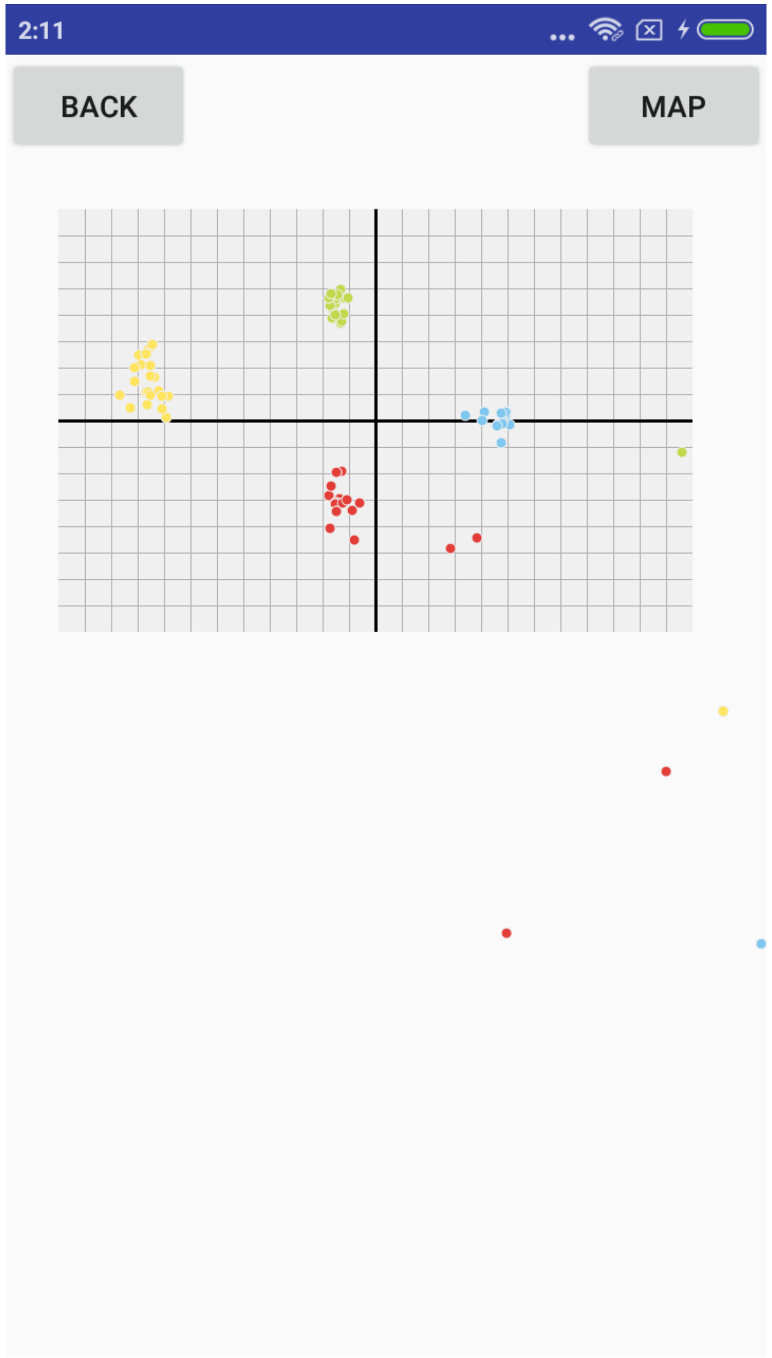
- Click on certain item then warning dialog shows up (to confirm deletion). Click “ok” to delete or “cancel” to cancel deletion.
- Also works for “delete all records”.
- In prevent of my finger slip.

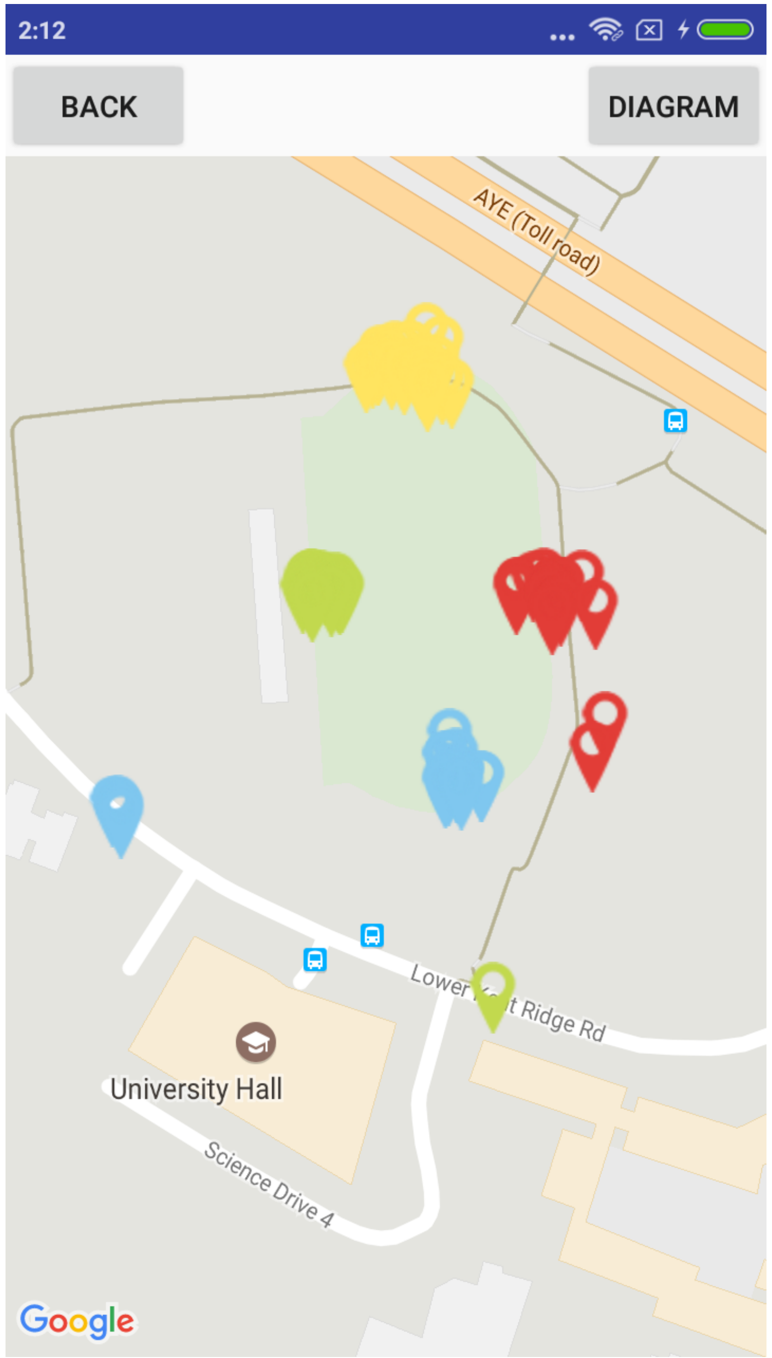




Plot Interface

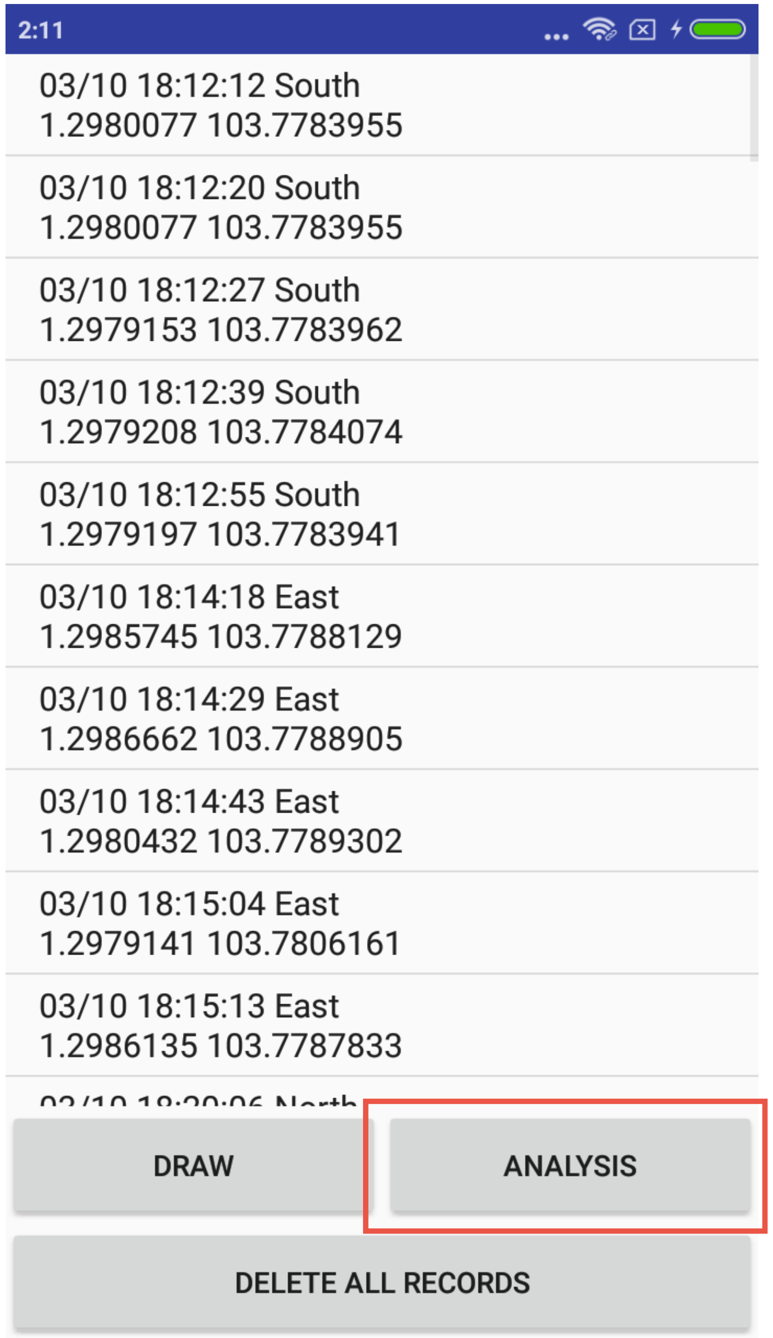
- Click “draw” in records interface to plot coordinates.
- Click “map” to draw coordinates on map (to have a direct view.)

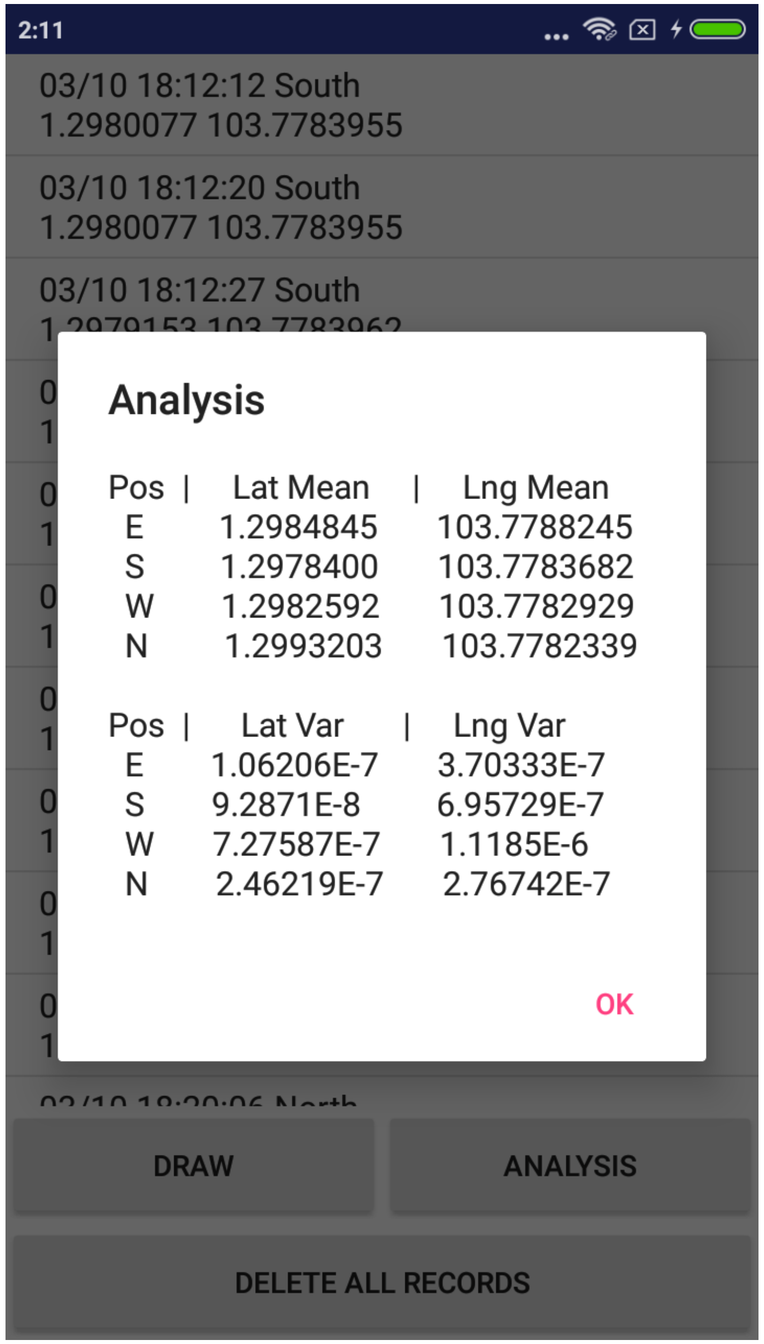




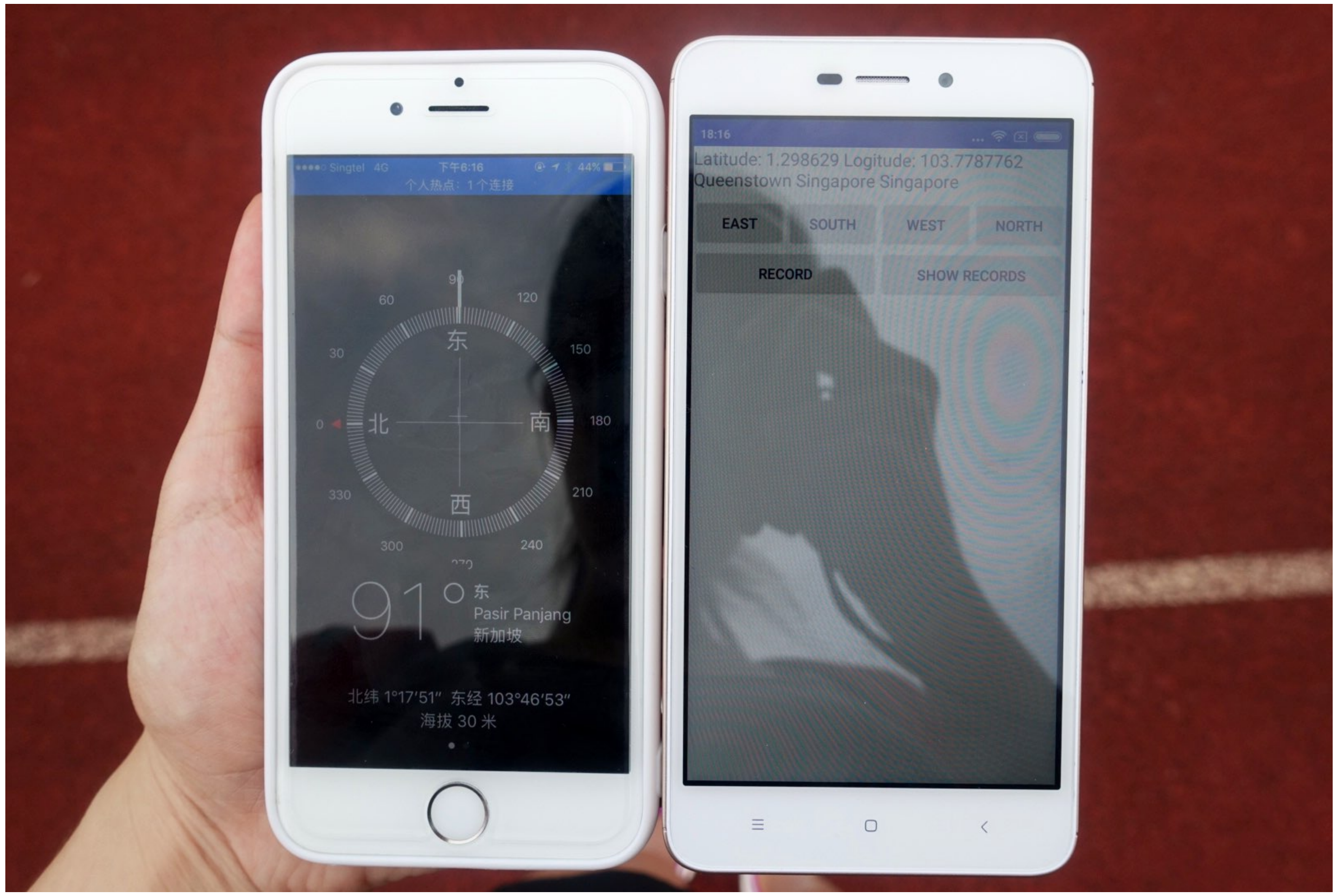
Analysis Interface

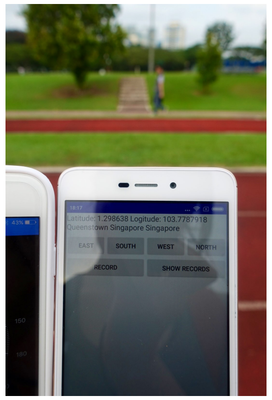
- Click “Analysis” to see analysis (in dialog).
- Mean and variance values are shown respectively in position.

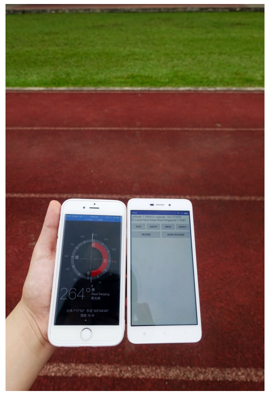


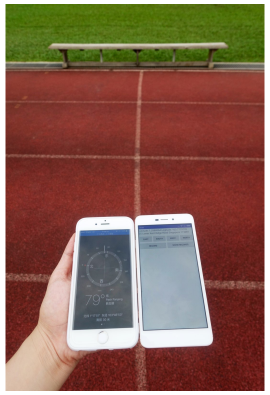


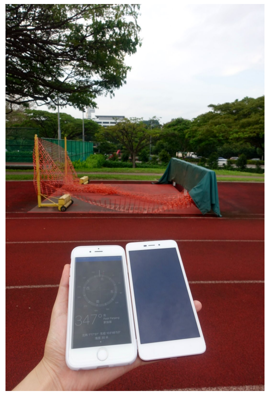
Experimental Process







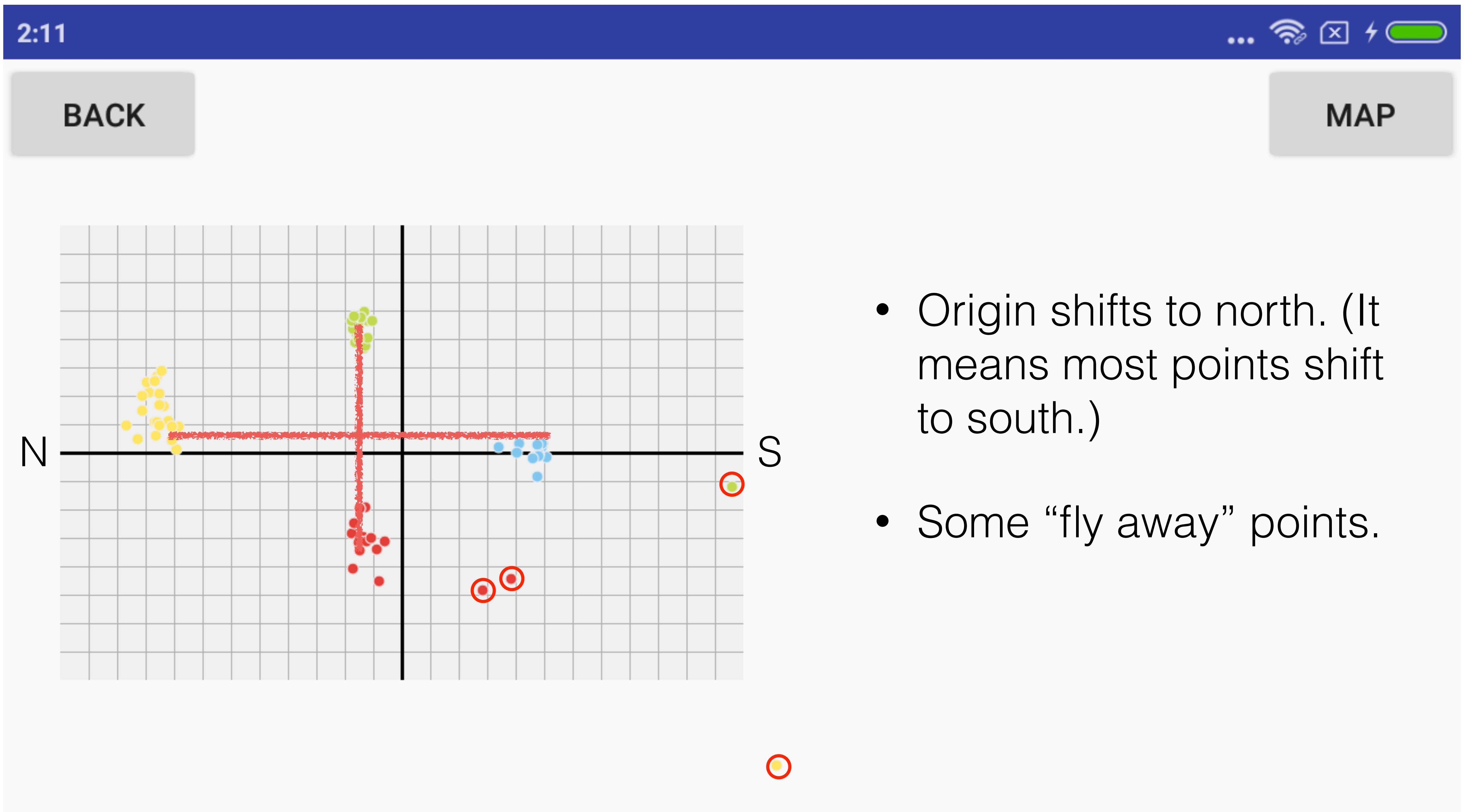




Calculation

- origin = {mean(lat), mean(lng)};
- kilometerPerLat = 111;
  kilometerPerLng = 111 * cos(origin[0] * PI / 180);
- latDiff = (origin[0] - geo[0]) * kilometerPerLat;
  lngDiff = (origin[1] - geo[1]) * kilometerPerLng;
- rect[0] = (int) (origin[0] + latDiff * 1000 * pixelPerMeter);
  rect[1] = (int) (origin[1] + lngDiff * 1000 * pixelPerMeter);

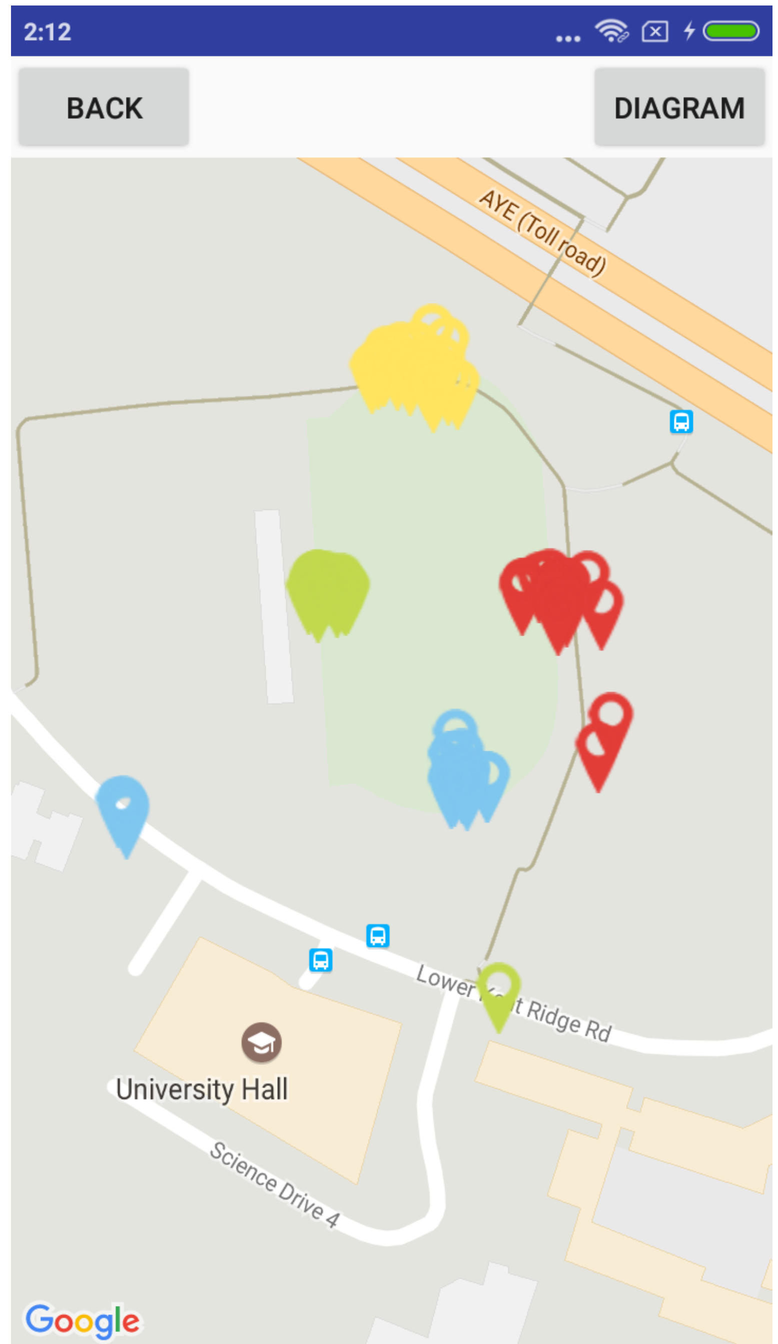
Result — Diagram

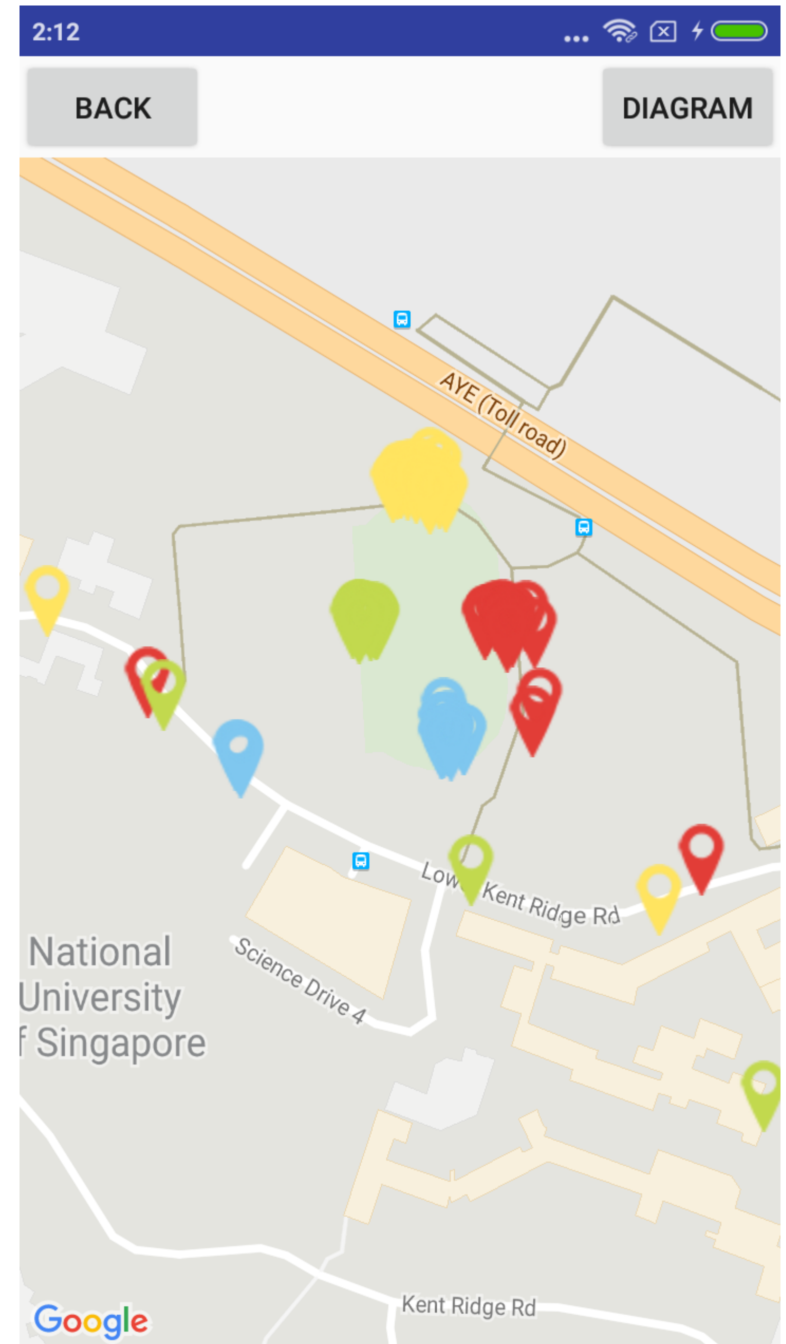


- Origin shifts to north. (It means most points shift to south.)
- Some “fly away” points.

Result — Map View

- Map view. Most points are at their right locations.
- Some “flyaway” points along the southern road.

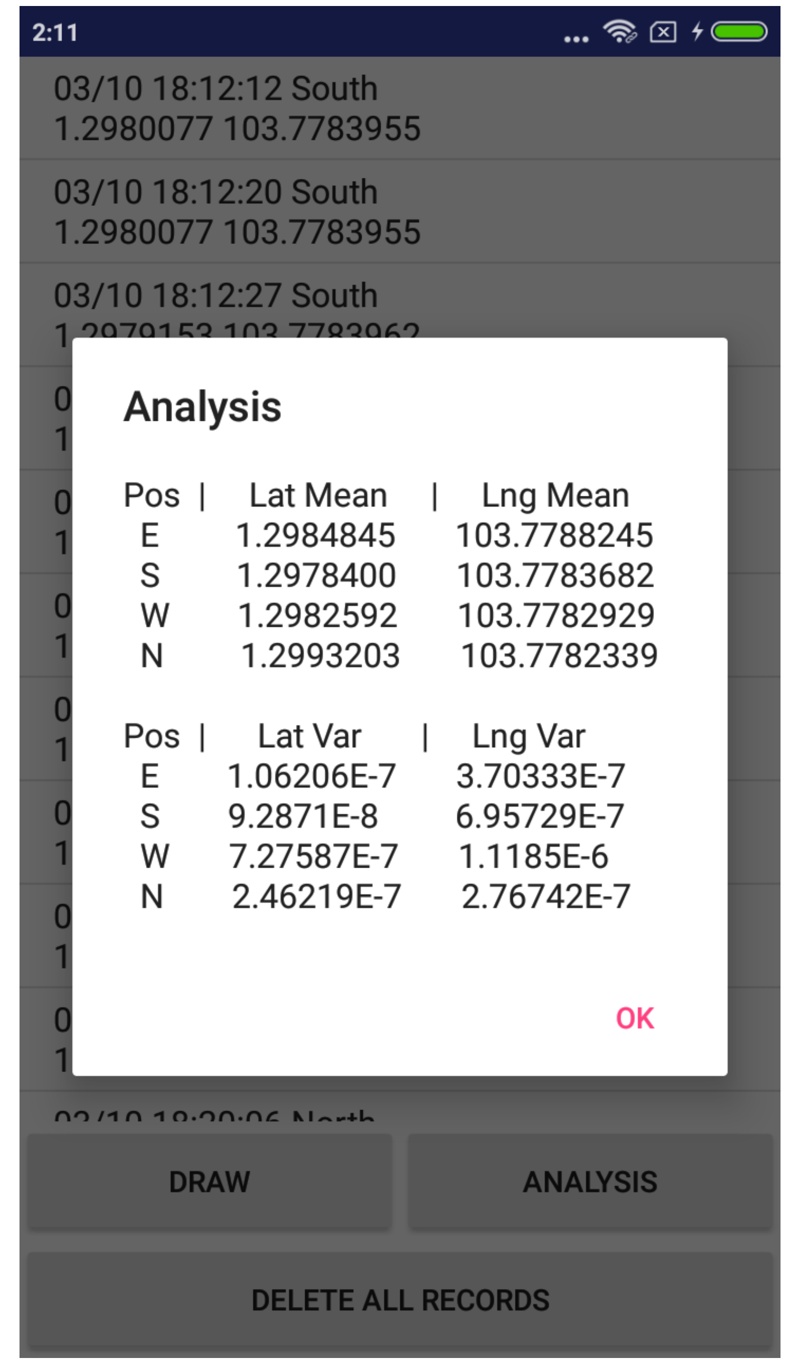




Result — Mean & Variance

Pos	Lat Mean	Lng Mean
E	1.2984845	103.7788245
S	1.2978400	103.7783682
W	1.2982592	103.7782929
N	1.2993203	103.7782339

Pos	Lat Var	Lng Var
E	1.06206E-07	3.70333E-07
S	9.2871E-08	6.95729E-07
W	7.27587E-07	1.1185E-06
N	2.46219E-07	2.76742E-07

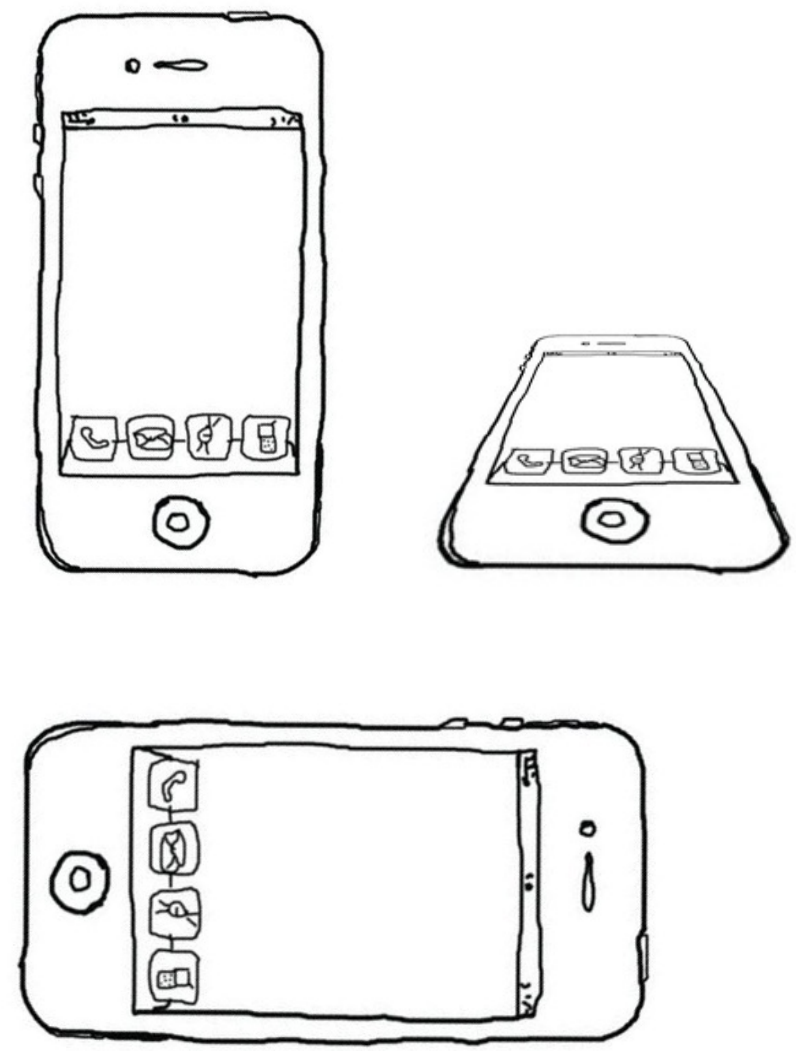


Accelerometer Error Determination

- Experiment Design
- Algorithm
  - Database Structure
  - Calculation
- User Interface
- Experimental Process
- Result

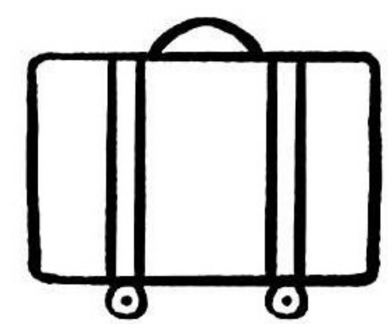
Experiment Design

- Still Experiment
  - Put the phone in different posture
  - Advantage:
    - easy to implement
    - easy to calculate
  - Disadvantage
    - hard to get true “horizontal”
    - ignore friction effect



Experiment Design

- Dynamic Experiment
  - Move straight with phone, record acceleration, do calculation and compare the results with the distance
  - Advantage:
    - more precise
  - Disadvantage
    - hard to implement (how to move steadily)



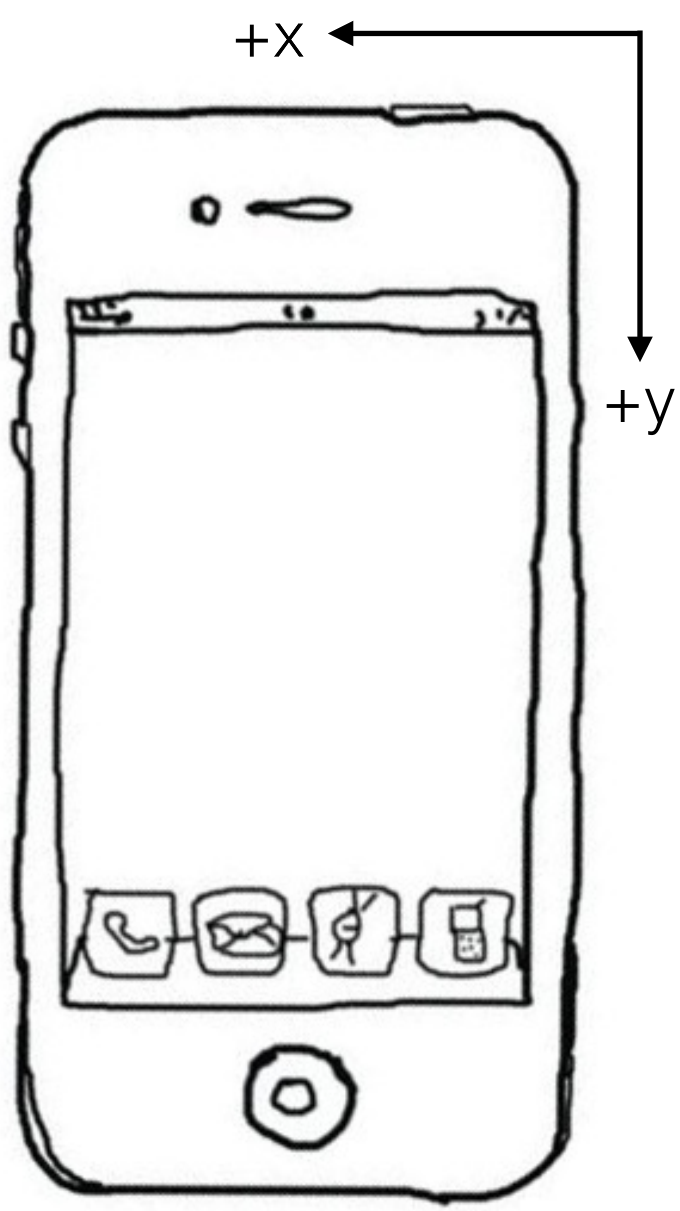
    - hard to calculate (how to deal get more precise results from changing acceleration)

Algorithm — Database Structure

- Must calculate credible errors from experiment — get data from multiple records.
- Two tables — records & item.

Column	About
id	
timestamp	YY-MM-DD HH-mm-ss
time	from start (s)
address	record for reference
direction	+x / -x / +y / -y
error	error in main direction
otherError	error in the other direction

Algorithm — Database Structure

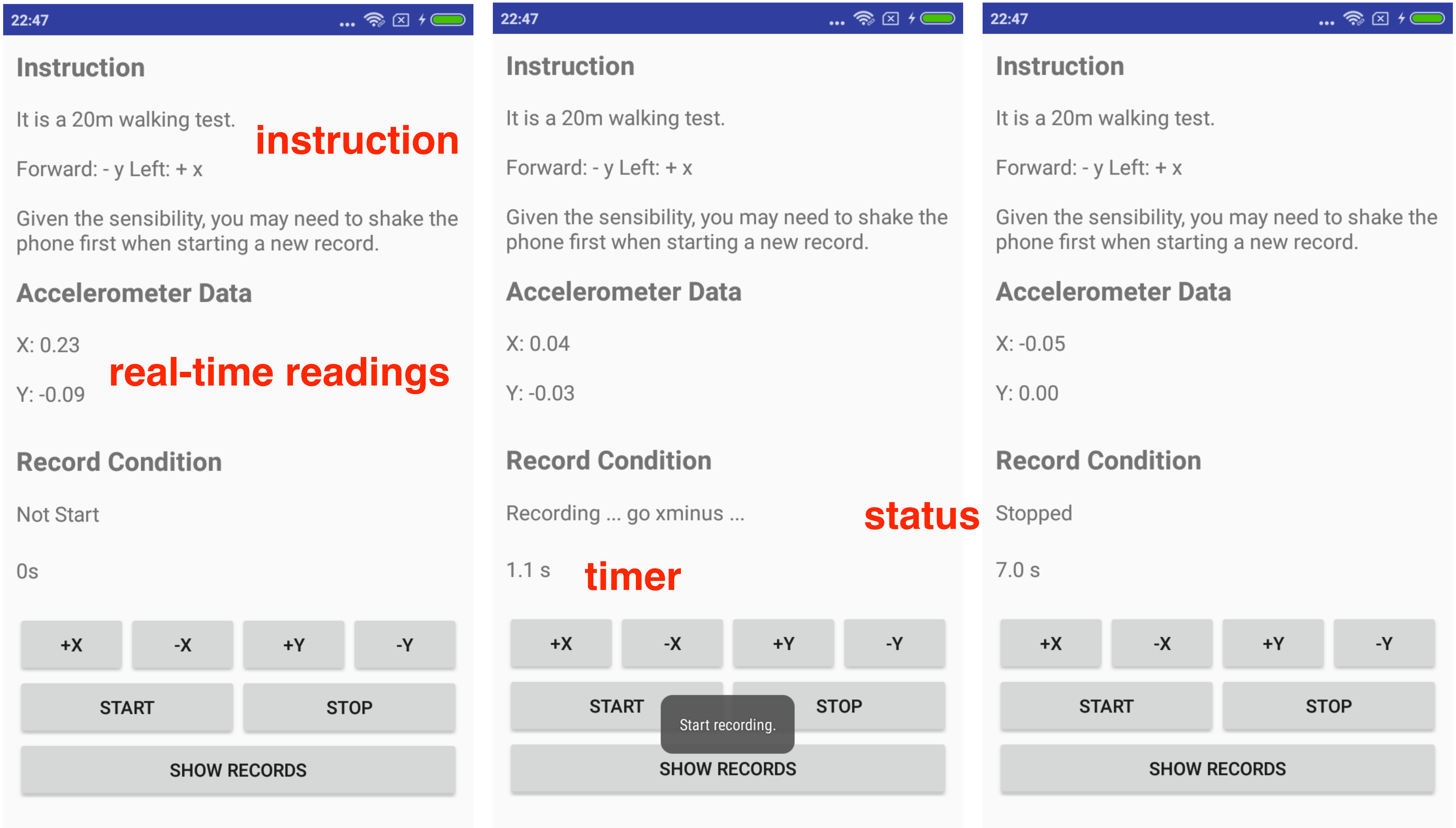


Column	About
id	
timestamp	YY-MM-DD HH-mm-ss
time	duration
lat	(planned to help correct error)
lng	
ax	acceleration in x
ay	acceleration in y
recordID	belong to which record

Algorithm — Calculation

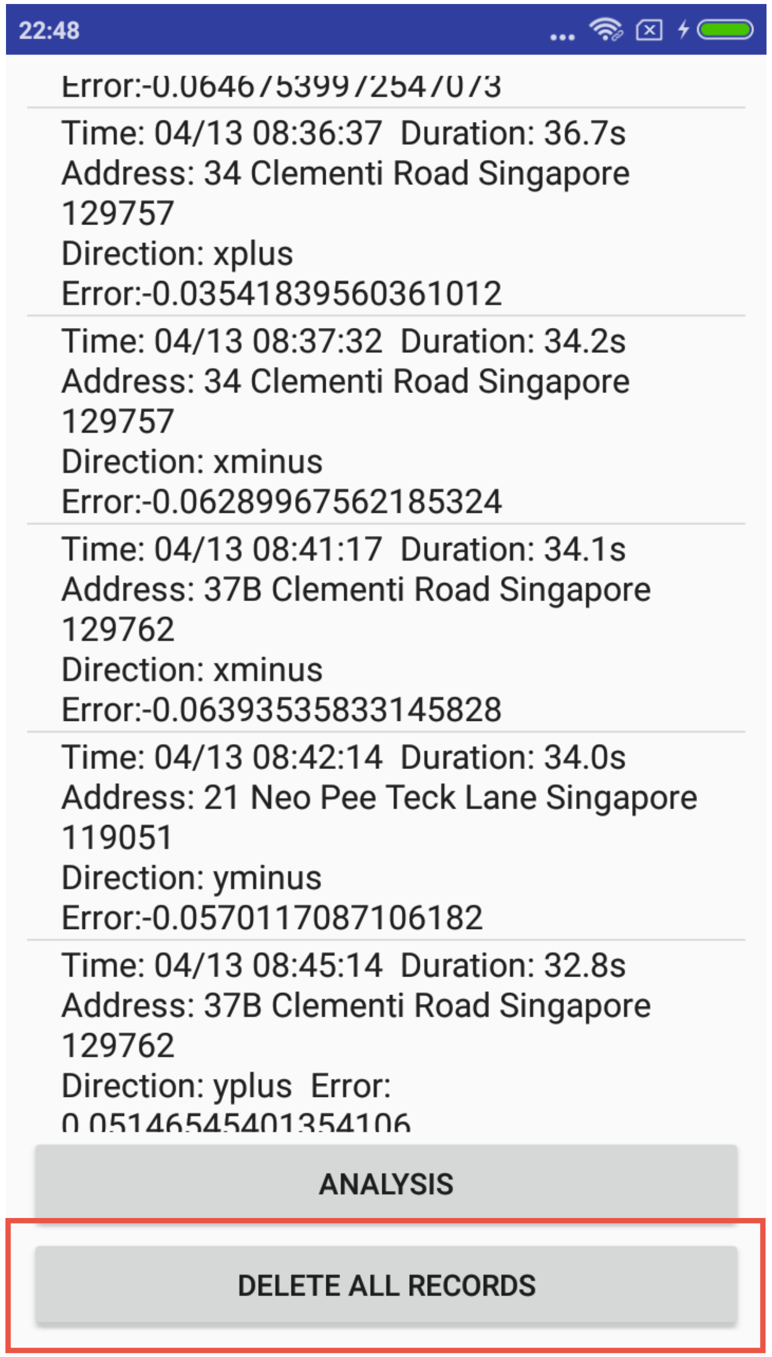
- Main direction
  - add new record [onClick_startRecord]
  - a = old reading - current reading [onSensorChanged]
  - trueAcc = 2*distance/(time^2)
    accDrift = meanMeasuredAcc - trueAcc
    update record [onClick_stopRecord]
- Minor direction
  - accError = meanMeasuredAcc

User Interface



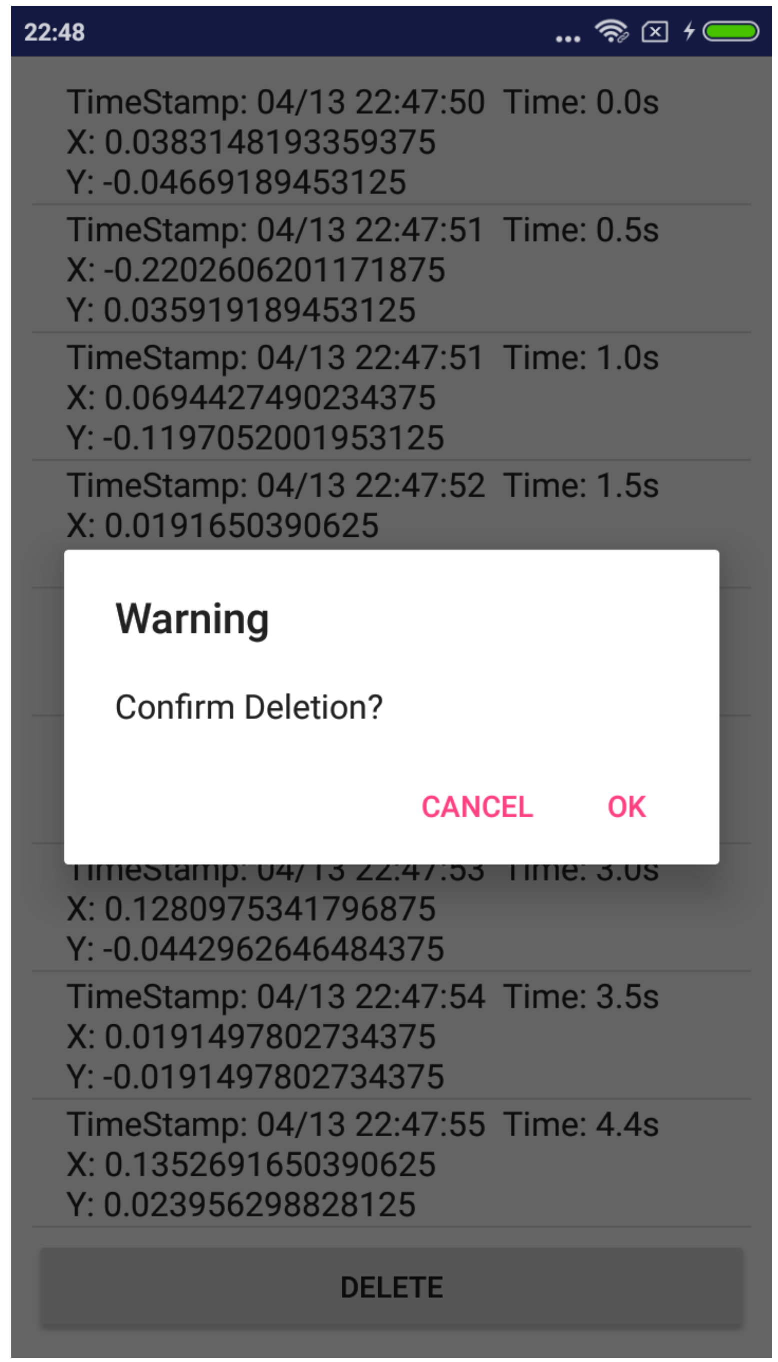
History Interface

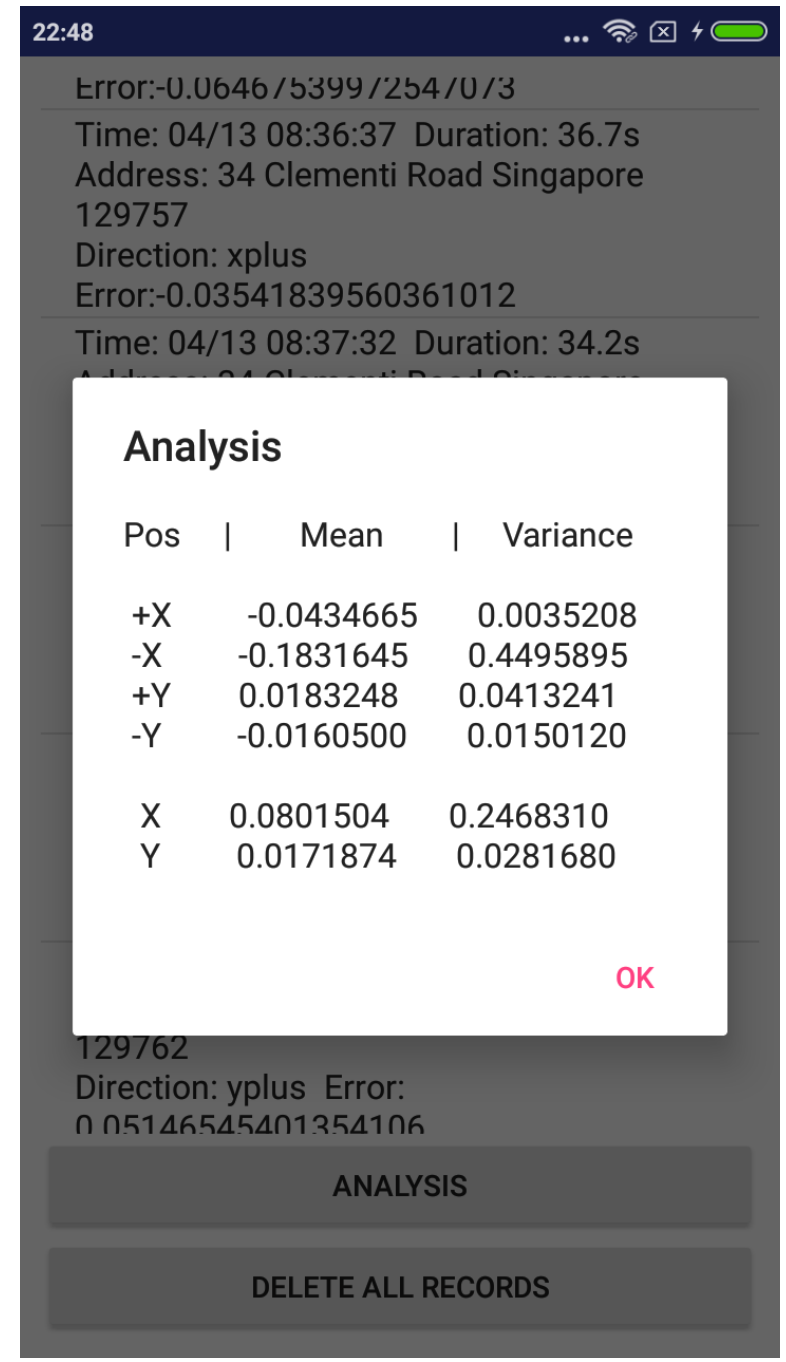
- Records are shown in listview.
- Click on certain row to view detailed items of a certain record.



Dialog Interface

- Deletion warning dialog.
- Analysis dialog.





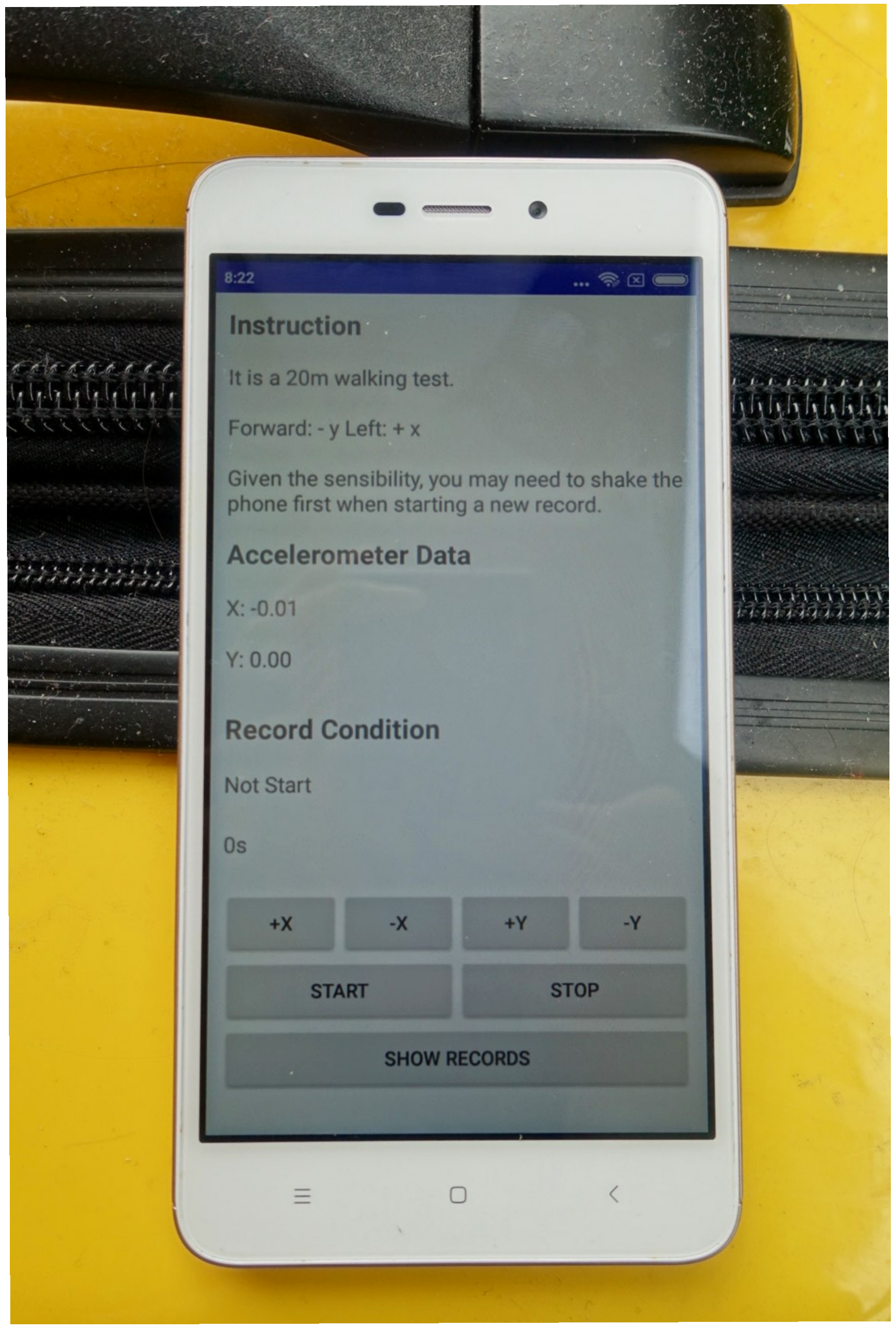
Experimental Process

- Dorm Corridor
- With Check
- Acceptable Bump
- Convenience

- Get 20m
- Use tape
- Along check

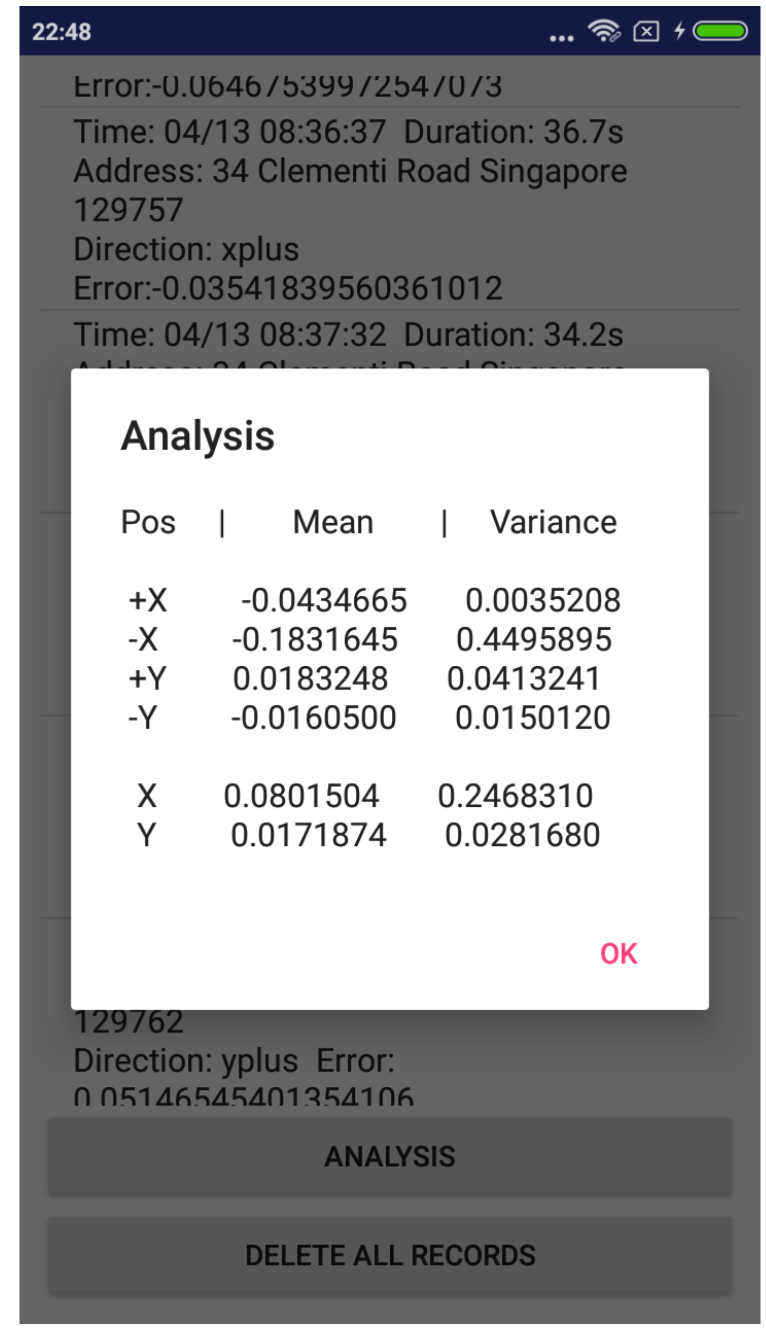
Marking Process

Recording Process



Result — Mean & Variance

Pos	Mean	Variance
+x	-0.0434665	0.0035208
-x	-0.1831645	0.4495895
+y	0.0183248	0.0413241
-y	-0.0160500	0.0150120
x	0.0801504	0.2468310
y	0.0171874	0.0281680



Locus Determination

- Understand Kalman Filter
- Implement Kalman Filter
- User Interface
- Experimental Process
- Result

Understand Kalman Filter

H
transformation matrix
maps the state vector parameters into the measurement domain
(Identical Matrix)

R
measurement noise covariance matrix
(from former 2 experiments)

$$\begin{bmatrix} e_x^2 & e_x e_y & e_x e_{v_x} & e_x e_{v_y} \\ e_x e_y & e_y^2 & e_y e_{v_x} & e_y e_{v_y} \\ e_y e_{v_y} & e_y e_{v_x} & e_{v_x}^2 & e_{v_x} e_{v_y} \\ e_x e_{v_y} & e_x e_{v_x} & e_{v_x} e_{v_y} & e_{v_y}^2 \end{bmatrix}$$

2 Compute

$$K_k = P_k^- H_k (H_k P_k^- H_k + R_k)^{-1}$$

$z_0, z_1, \dots$

measurement matrix

$$\begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x_k \\ y_k \\ v_{x_k} \\ v_{y_k} \end{bmatrix}$$

estimation
initialization
xy = measurement
v = 0

1

$$\hat{x}_{k+1}^- = \phi_k \hat{x}_k + u_t B_t$$

$$P_{k+1}^- = \phi_k P_k \phi_k + Q_k$$

cannot get true state ax ay
measurement - mean error

3 Update

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-)$$

$$u_t = \begin{bmatrix} a_{x_t} & 0 & 0 & 0 \\ 0 & a_{y_t} & 0 & 0 \\ 0 & 0 & a_{x_t} & 0 \\ 0 & 0 & 0 & a_{y_t} \end{bmatrix}$$

$$B_t = \begin{bmatrix} \left(\frac{\Delta t}{2}\right)^2 \\ \left(\frac{\Delta t}{2}\right)^2 \\ \Delta t \\ \Delta t \end{bmatrix}$$

P
error covariance matrix
suppose an initial value
(invertible matrix)

Q
process noise covariance matrix
guess and tune

$\hat{x}_0, \hat{x}_1, \dots$

4 Compute

$$P_k = [1 - K_k H_k] P_k^-$$

1 actually represents identical matrix

Understand Kalman Filter

$$\begin{bmatrix} x_k \\ y_k \\ v_{x_k} \\ v_{y_k} \end{bmatrix}$$

which xy?
latlng or xy?
xy: 1 lat unequals 1 lng

2 Compute
$$K_k = P_k^- H_k (H_k P_k^- H_k + R_k)^{-1}$$

$z_0, z_1, \ldots$

1
$$\hat{x}_{k+1}^- = \phi_k \hat{x}_k$$
$$P_{k+1}^- = \phi_k P_k \phi_k + Q_k$$

3 Update
$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H_k \hat{x}_k^-)$$

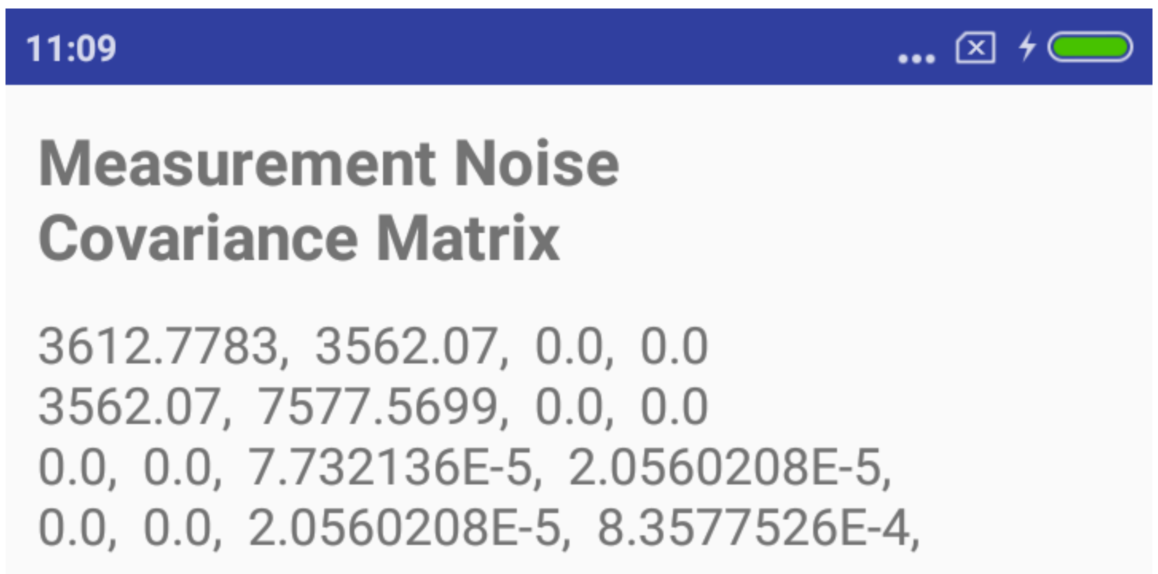
How to get v from a?
curV = preV + (preA + curA)/deltaT

$\hat{x}_0, \hat{x}_1, \ldots$

4 Compute
$$P_k = [1 - K_k H_k] P_k^-$$

Get R

$$\begin{bmatrix} e_x^2 & e_x e_y & e_x e_{v_x} & e_x e_{v_y} \\ e_x e_y & e_y^2 & e_y e_{v_x} & e_y e_{v_y} \\ e_y e_{v_y} & e_y e_{v_x} & e_{v_x}^2 & e_{v_x} e_{v_y} \\ e_x e_{v_y} & e_x e_{v_x} & e_{v_x} e_{v_y} & e_{v_y}^2 \end{bmatrix} \quad \begin{bmatrix} e_x^2 & e_x e_y & 0 & 0 \\ e_x e_y & e_y^2 & 0 & 0 \\ 0 & 0 & e_{v_x}^2 & e_{v_x} e_{v_y} \\ 0 & 0 & e_{v_x} e_{v_y} & e_{v_y}^2 \end{bmatrix} \quad \begin{bmatrix} e_x^2 & e_x e_y & 0 & 0 \\ e_x e_y & e_y^2 & 0 & 0 \\ 0 & 0 & e_{a_x}^4 & e_{a_x}^2 e_{a_y}^2 \\ 0 & 0 & e_{a_x}^2 e_{a_y}^2 & e_{a_y}^4 \end{bmatrix}$$



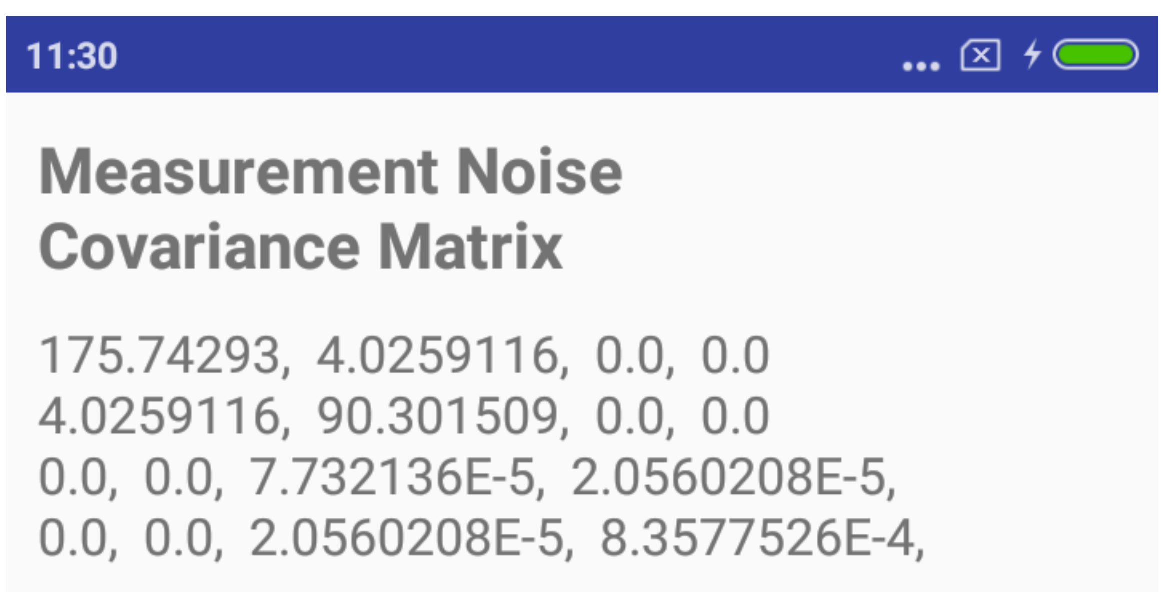
Covariance matrix for xy error is too big

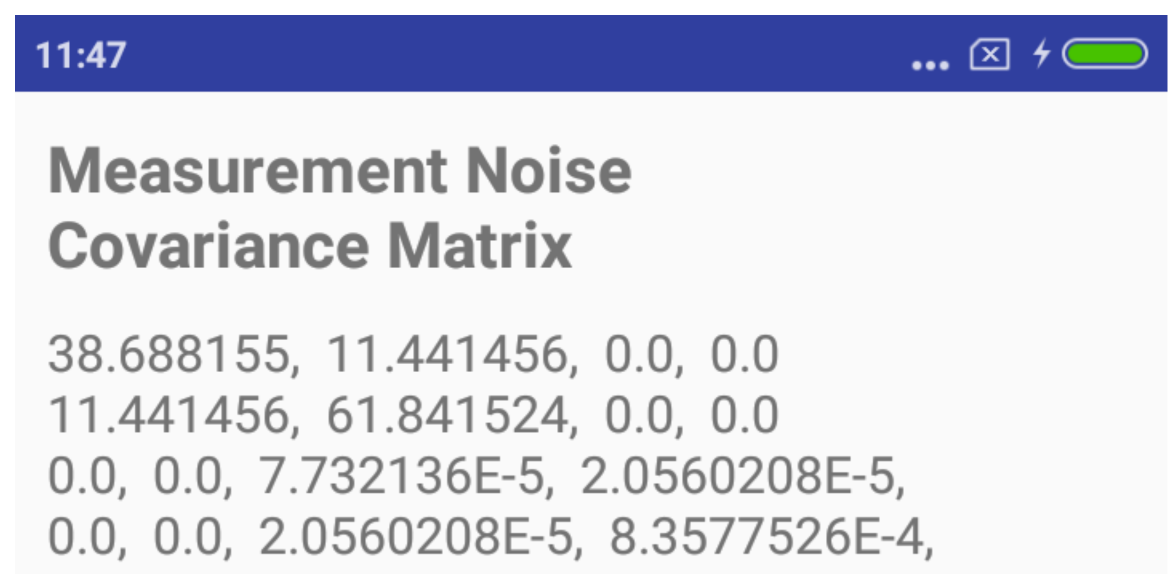
- "flyaway" points affect the results

  set threshold, do not calculate "flyaway" points

- mean values are seen as true values

  set threshold, calculate mean values again





Implement Kalman Filter

- Class KalmanStatus, Class KalmanFilter

```java
private final double deltaT = 1; // unit: s

private final double[][] stateTransitionM = {
        {1, 0, deltaT, 0},
        {0, 1, 0, deltaT},
        {0, 0, 1, 0},
        {0, 0, 0, 1}
};

private final double[][] measurementM = {
        {1, 0, 0, 0},
        {0, 1, 0, 0},
        {0, 0, 1, 0},
        {0, 0, 0, 1}
};

private final double[][] processNoiseCov = {
        {0.002, 0, 0, 0},
        {0, 0.003, 0, 0},
        {0, 0, 0.001, 0},
        {0, 0, 0, 0.001}
};

private final double[][] measureNoiseCov = {
        {38.688155, 11.441456, 0, 0},
        {11.441456, 61.841524, 0, 0},
        {0, 0, 7.732136e-5, 2.0560208e-5},
        {0, 0, 2.0560208e-5, 8.3577526e-4}
};
```

```java
public class KalmanStatus {
    public double[] advanceEstimation;
    public double[] estimation;
    public double[][] advanceError; // P-
    public double[][] error; // P
    public double[] measurement;
    public double[][] kalmanGain;
    public double time; // from start unit:s
```

```java
// control iput
double accX = 0;
double accY = 0;

double accXMeanError = 0.0086473;
double accYMeanError = 0.0171874;

double[] controlVec = {deltaT * deltaT / 2, deltaT * deltaT / 2, deltaT, deltaT};
double[][] controlM = {
        {accX - accXMeanError, 0, 0, 0},
        {0, accY - accYMeanError, 0, 0},
        {0, 0, accX - accXMeanError, 0},
        {0, 0, 0, accY - accYMeanError}
};
```

Implement Kalman Filter

- Handwriting matrix operations

In relation to its adjugate [edit]

The adjugate of a matrix A can be used to find the inverse of A as follows:

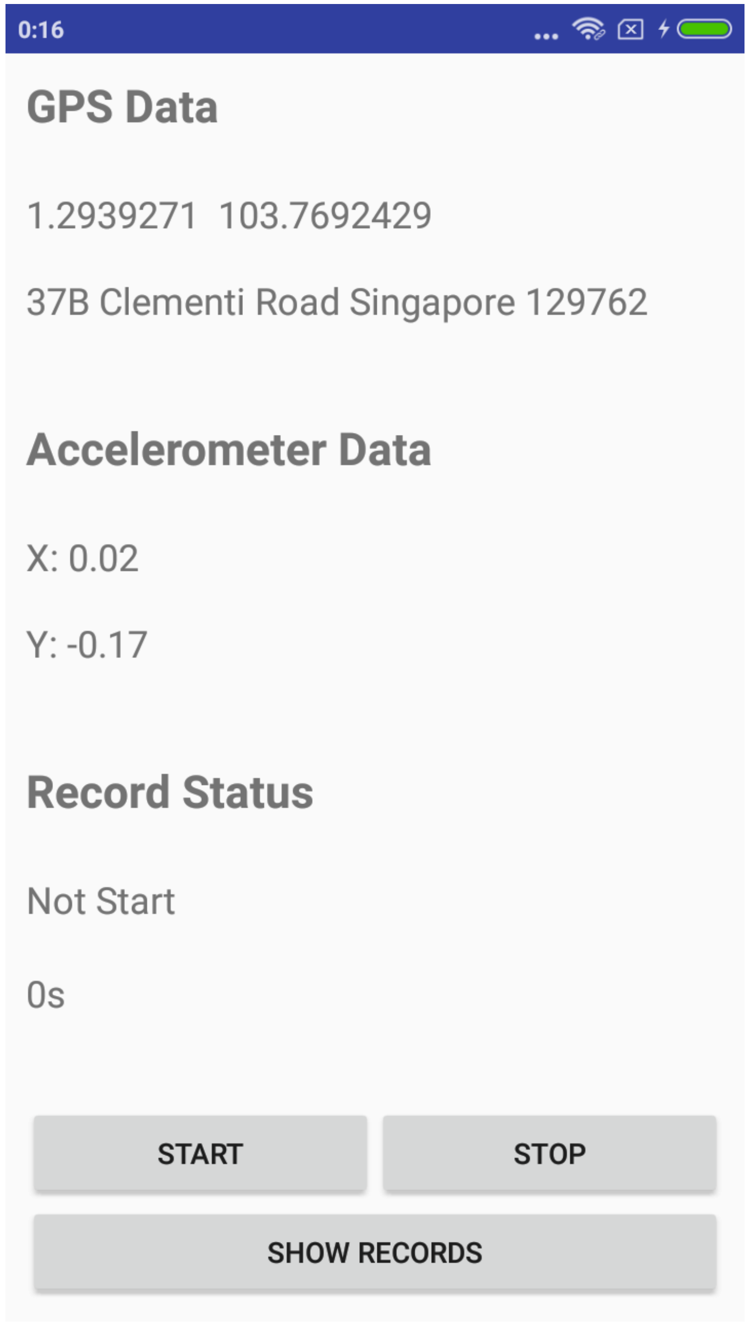
If A is an $n \times n$ invertible matrix, then

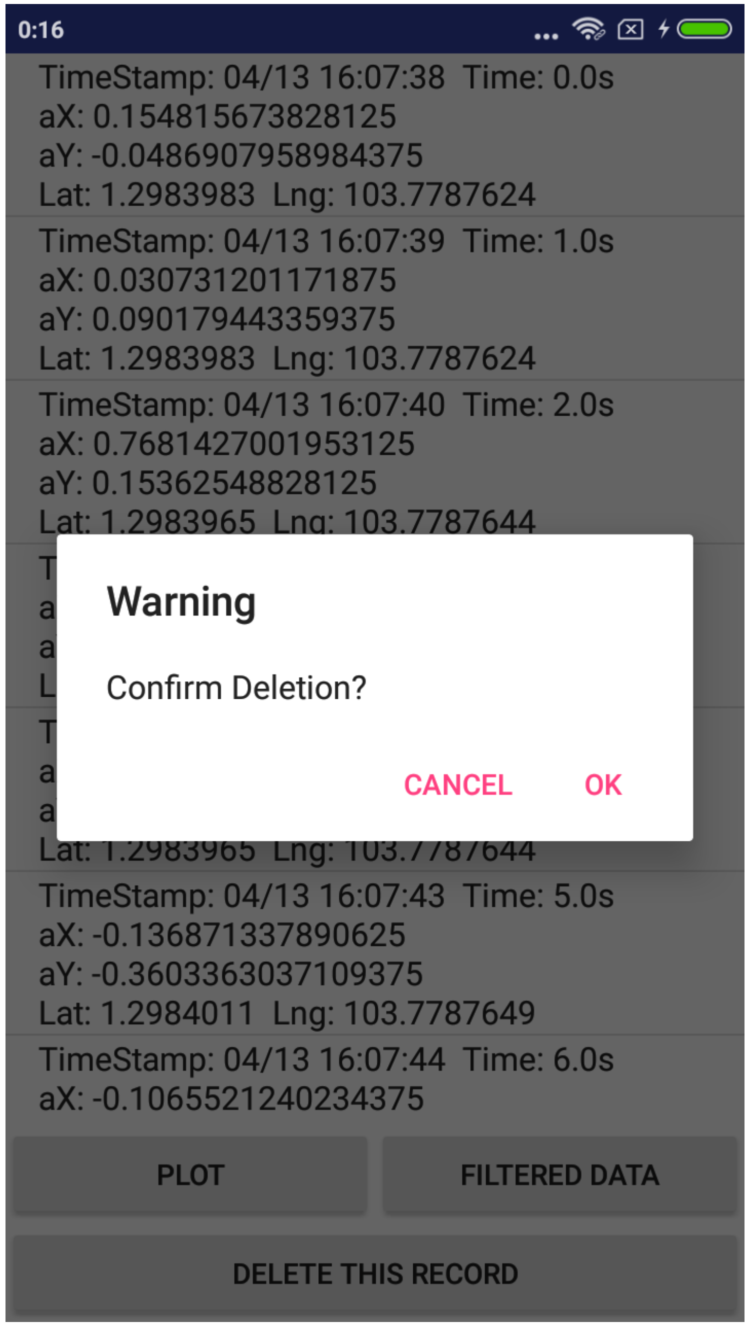
$$A^{-1} = \frac{1}{\det(A)} \operatorname{adj}(A).$$

- When to filter

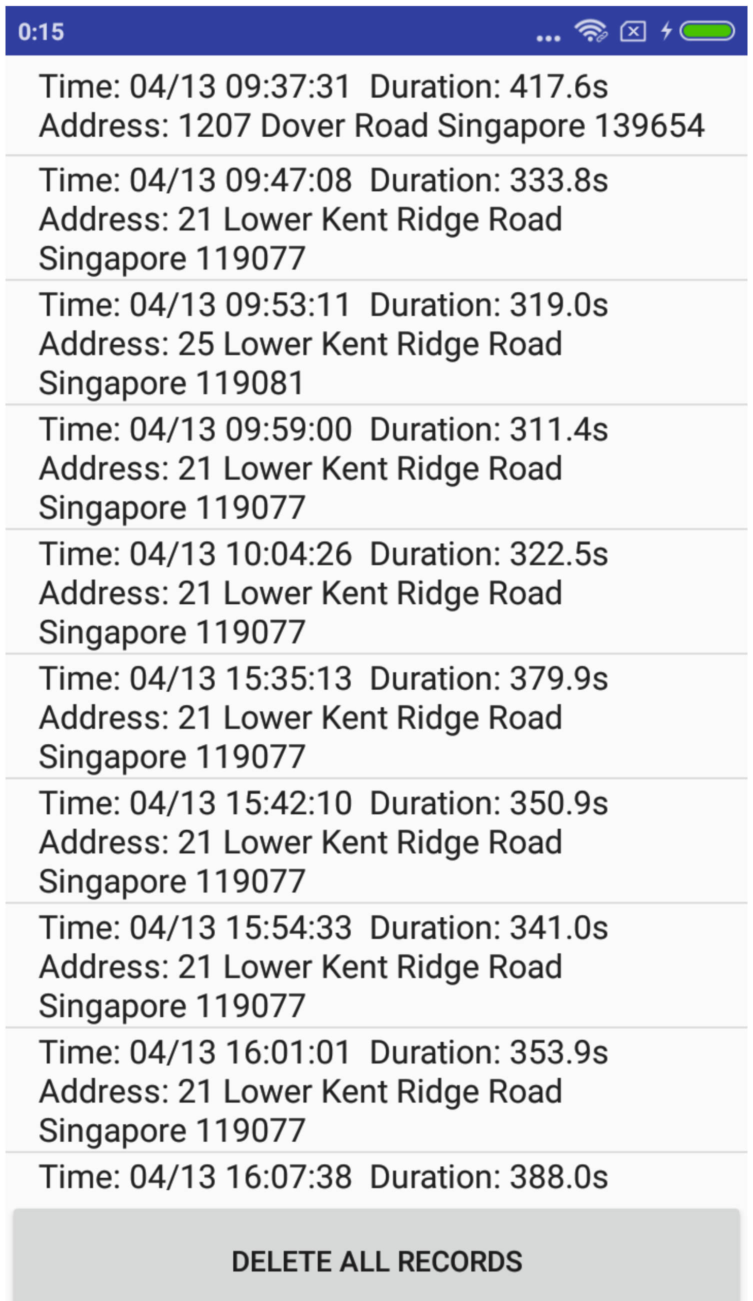
Record and filter at the same time
— Save time and space, but cause lag

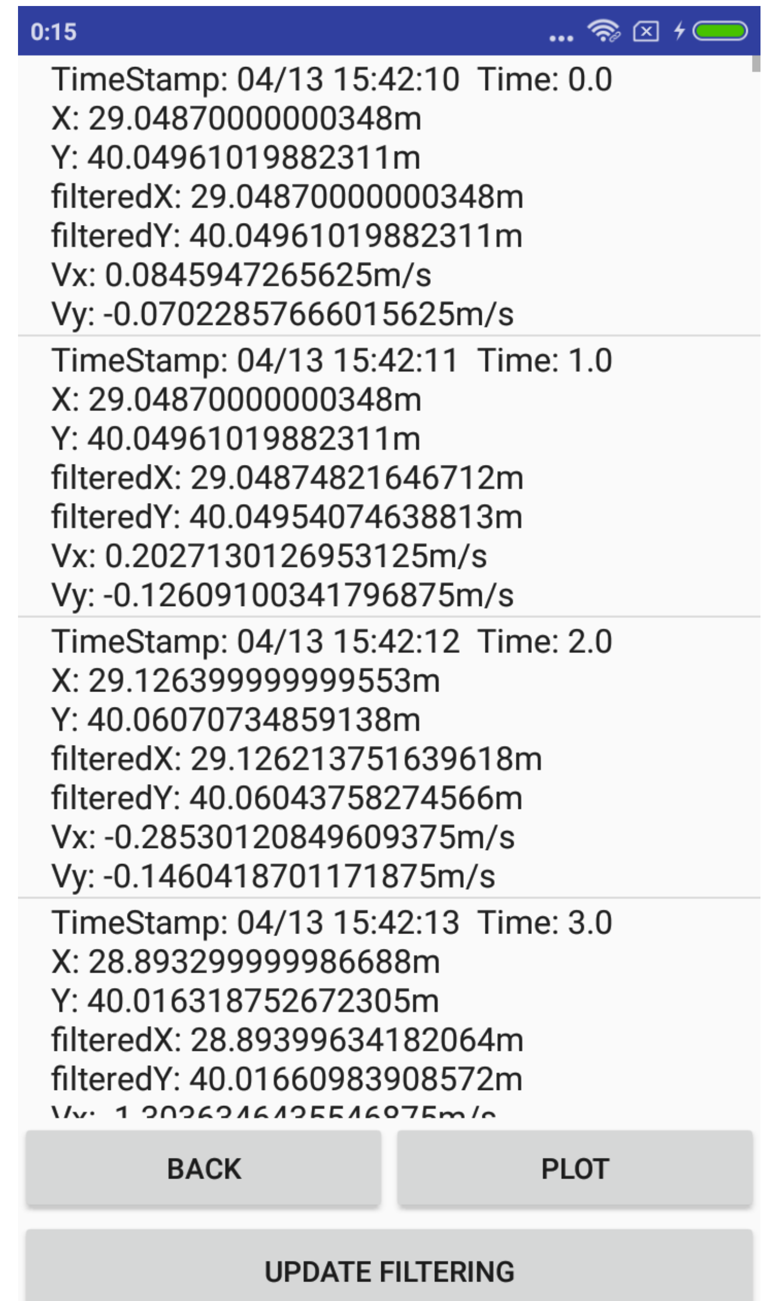
User Interface





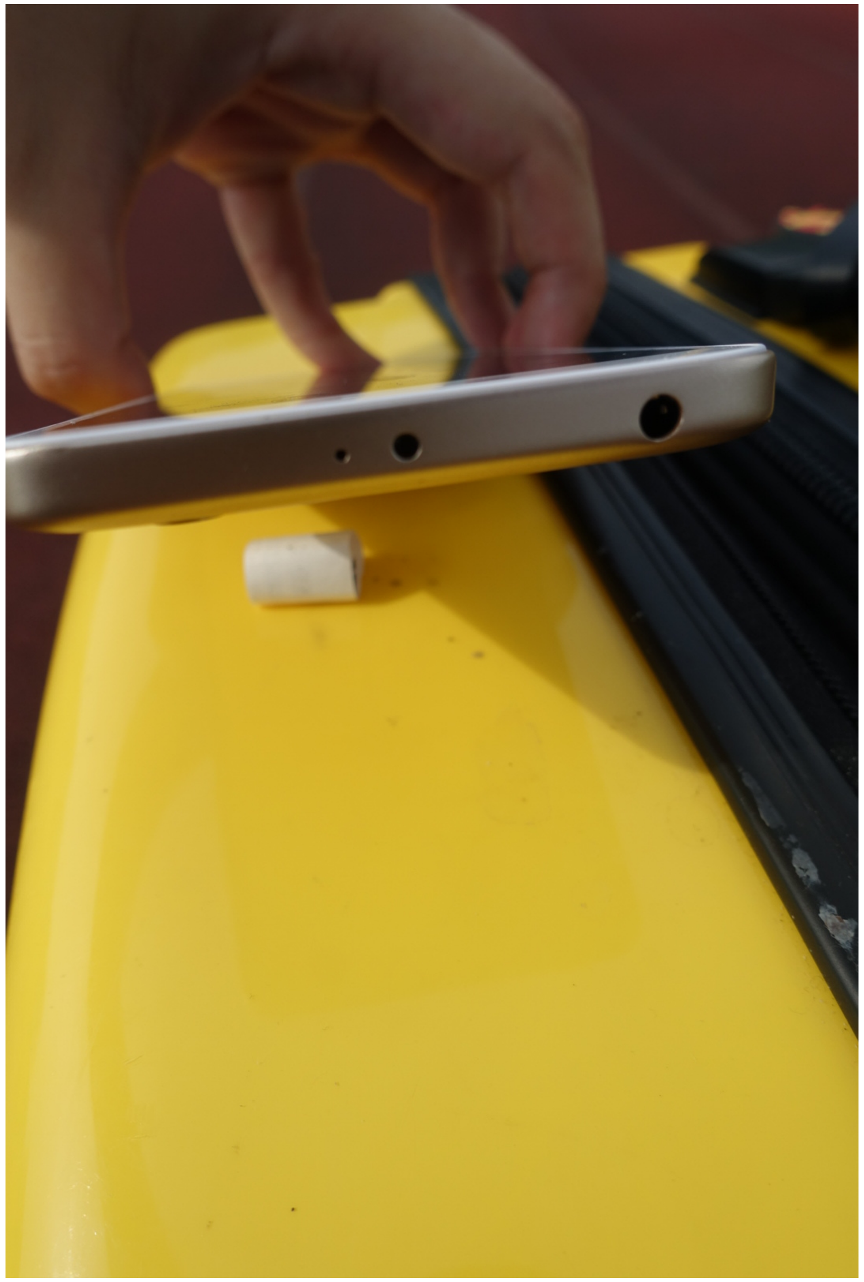
History Interface



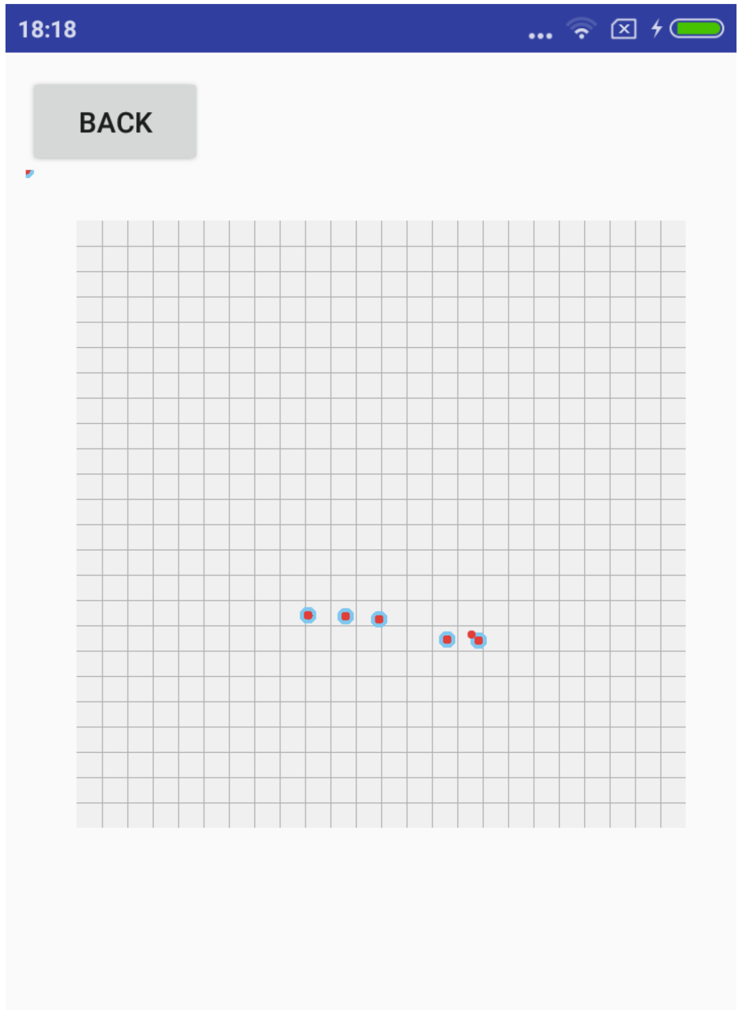


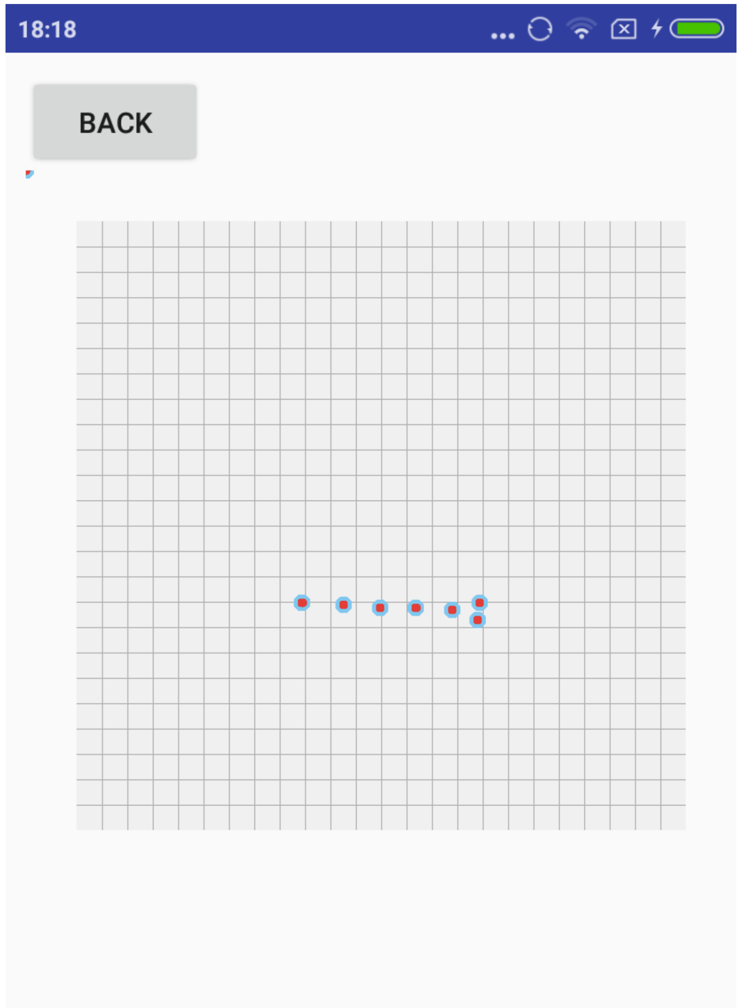
Process

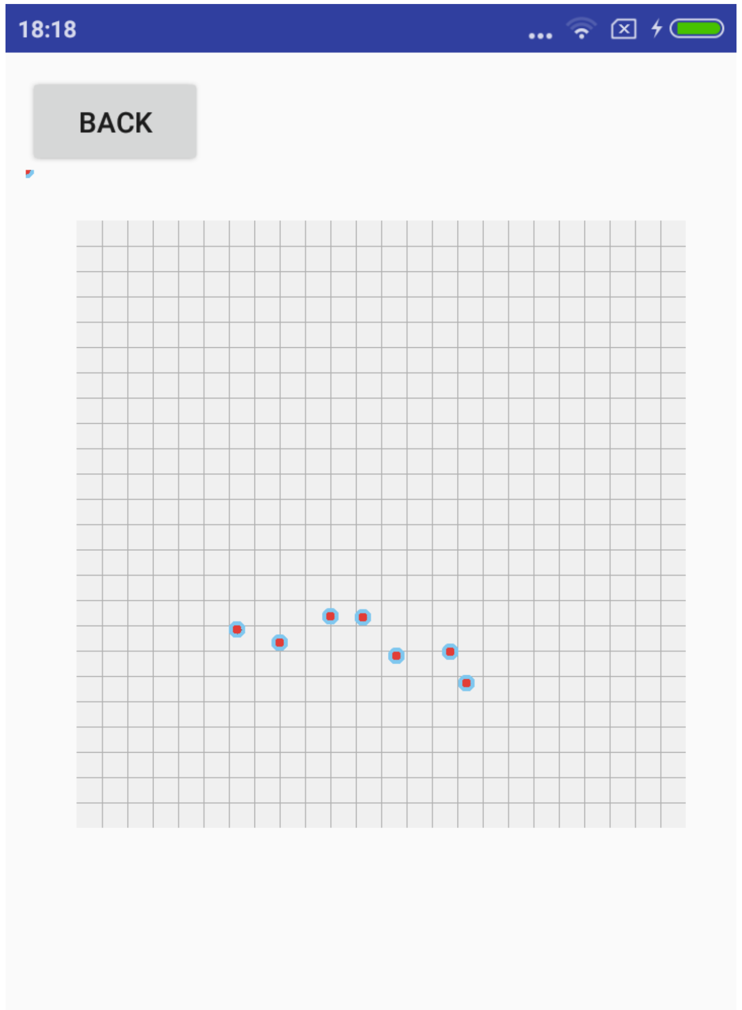
- Stick
  - Reduce bump
  - Prevent flip
- Adjust interval
  - Get more acceleration for a more accurate v



Accident





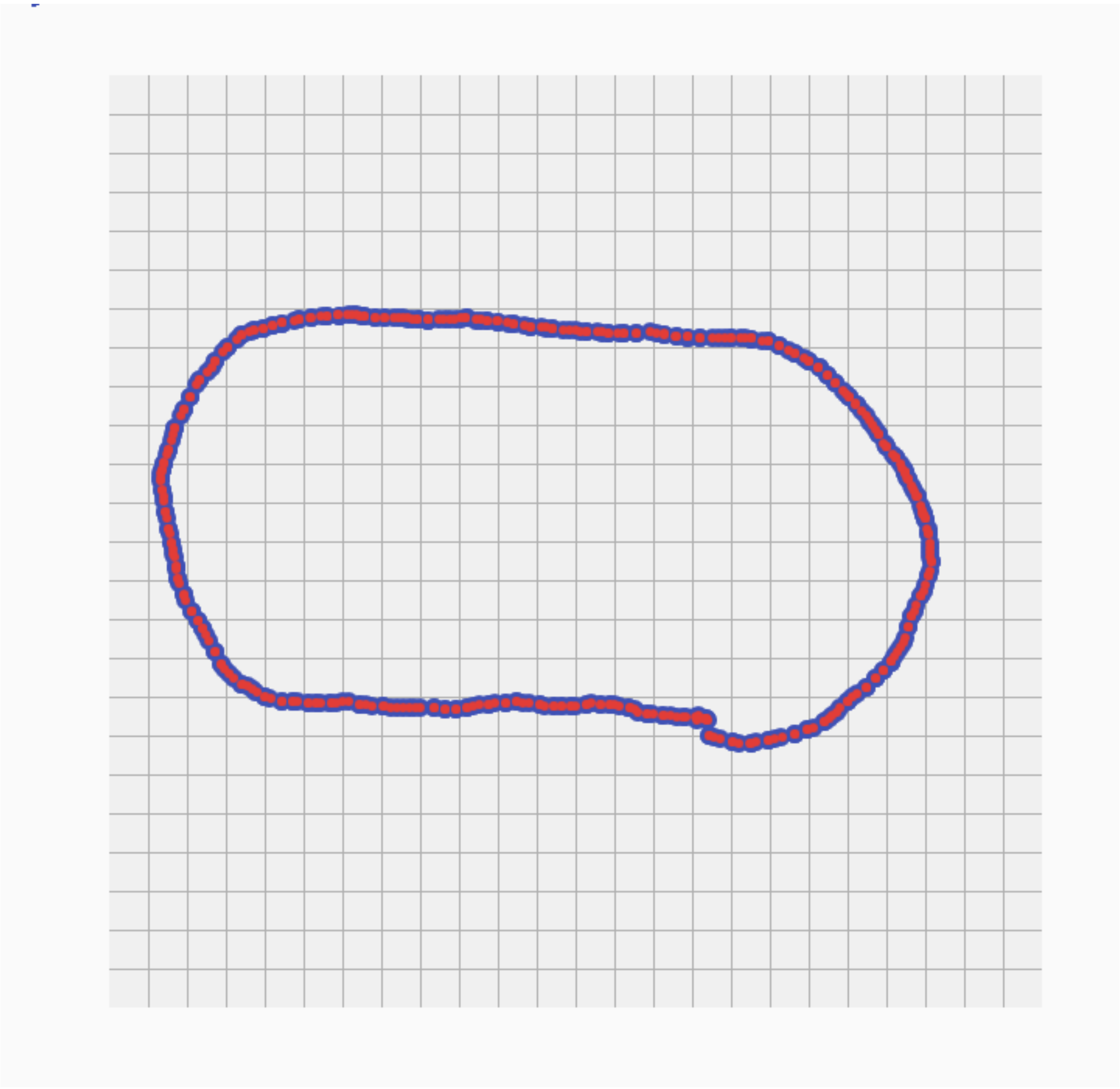


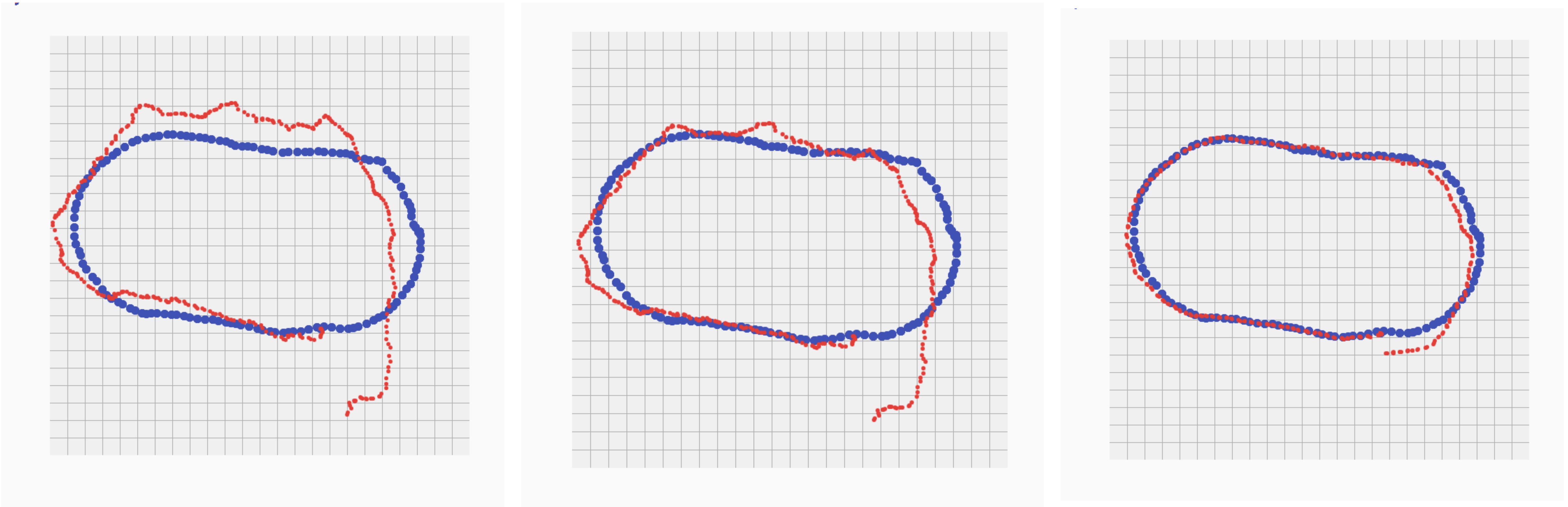
Lose many records

Weak network leads to location listener lagging

Result

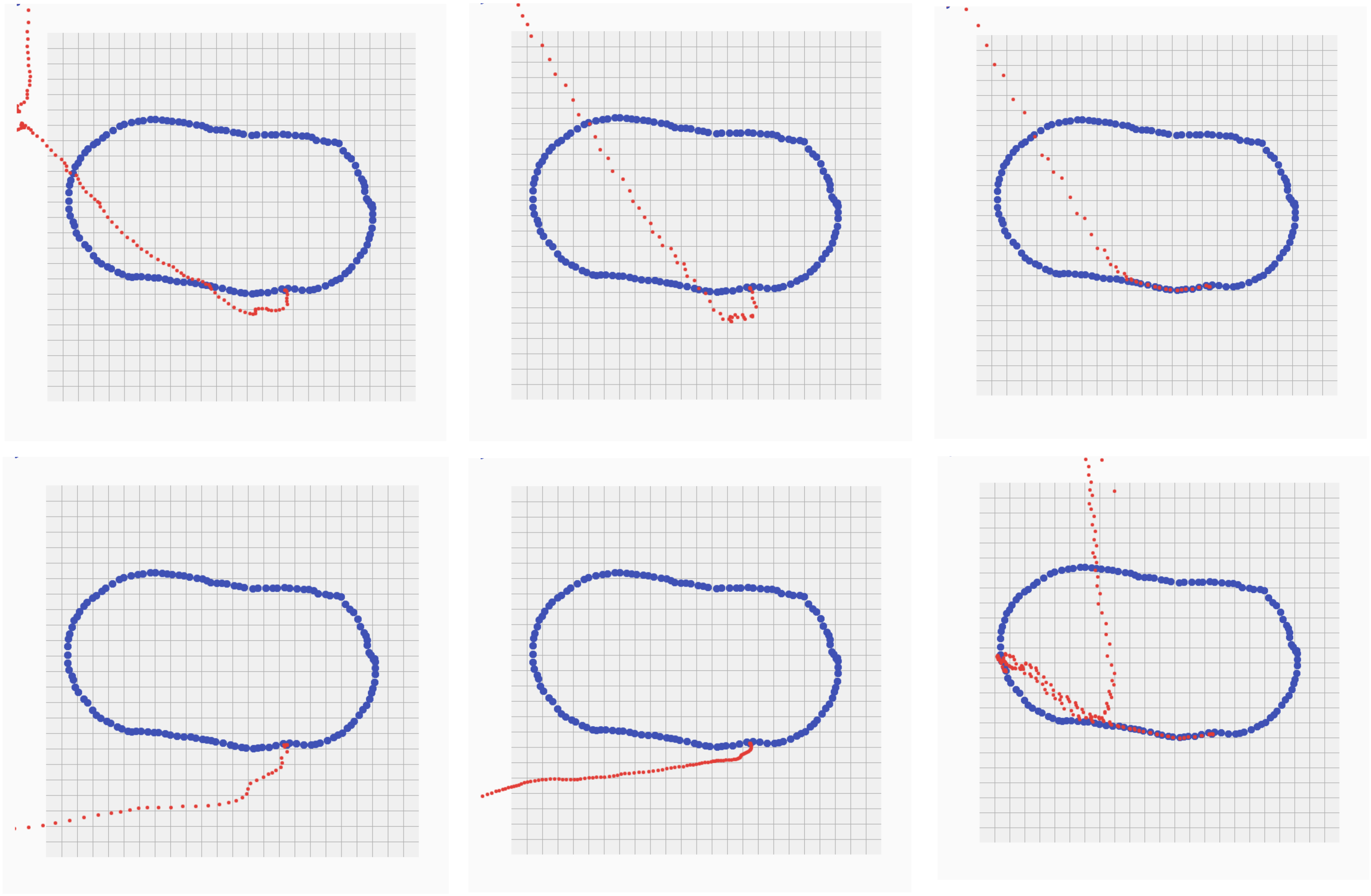
- Blue — raw data
- Red — filtered data
- Nothing wrong, just cannot see “filtered”
- Try tuning Q and change diagonal element first (x, y, vx, vy)





Q — Tuning

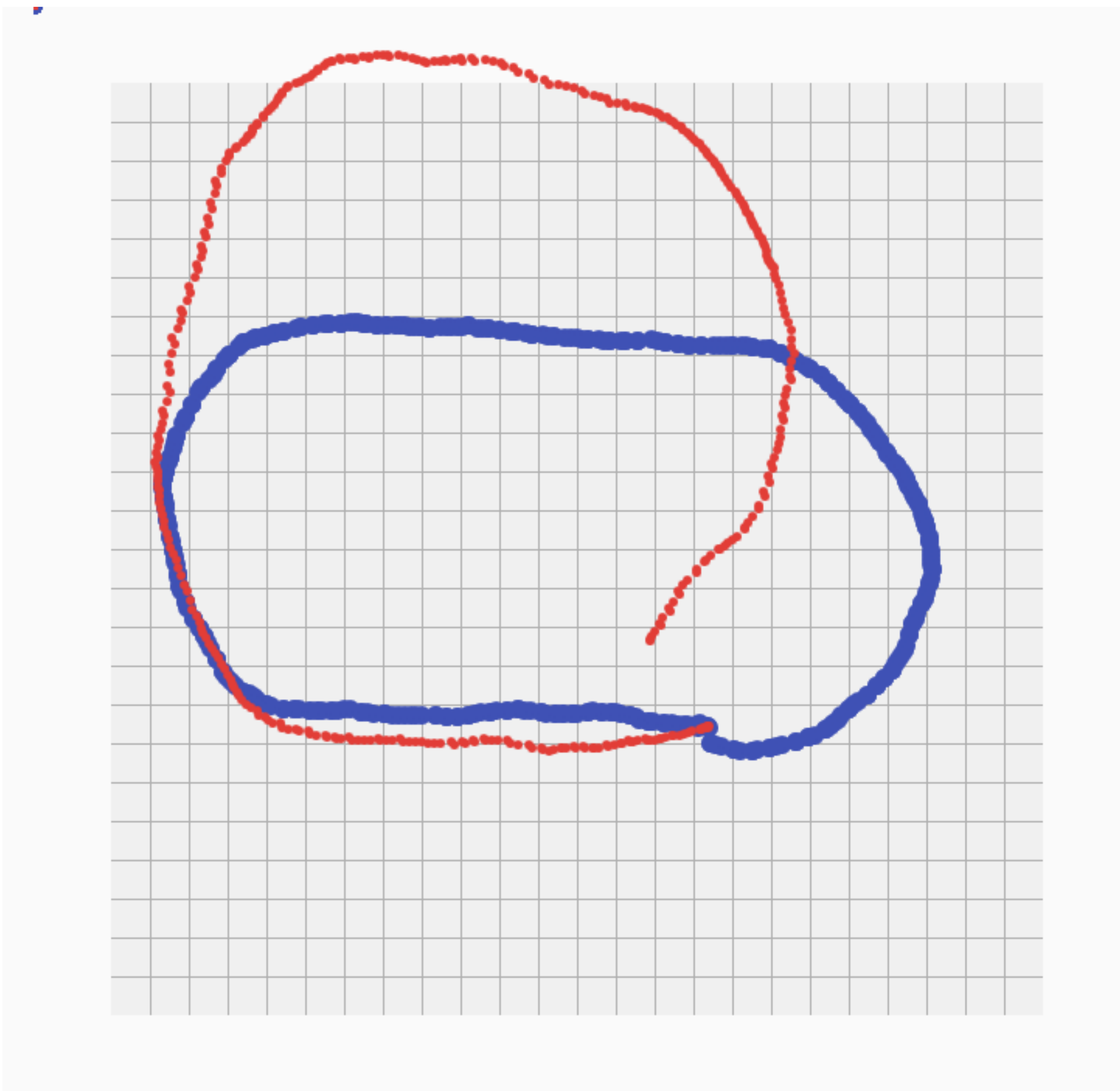
- Change the first two diagonal element
- Track becomes more bumping
- Reduce the elements, a little effect on filtering, but disconnect start and end of the track



- Change the last two diagonal element
- One make the upper track and the other make the lower track
- Hard to find the balance (the last picture)

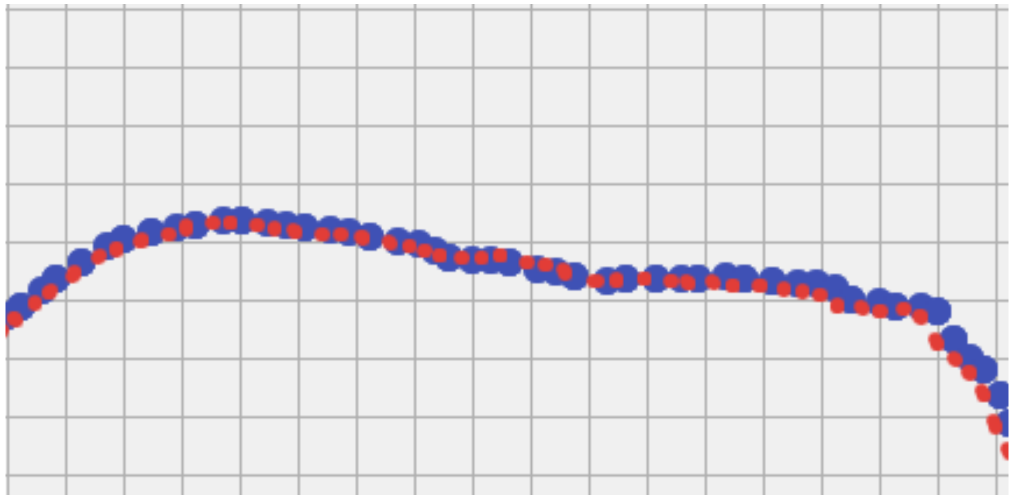
Tuning

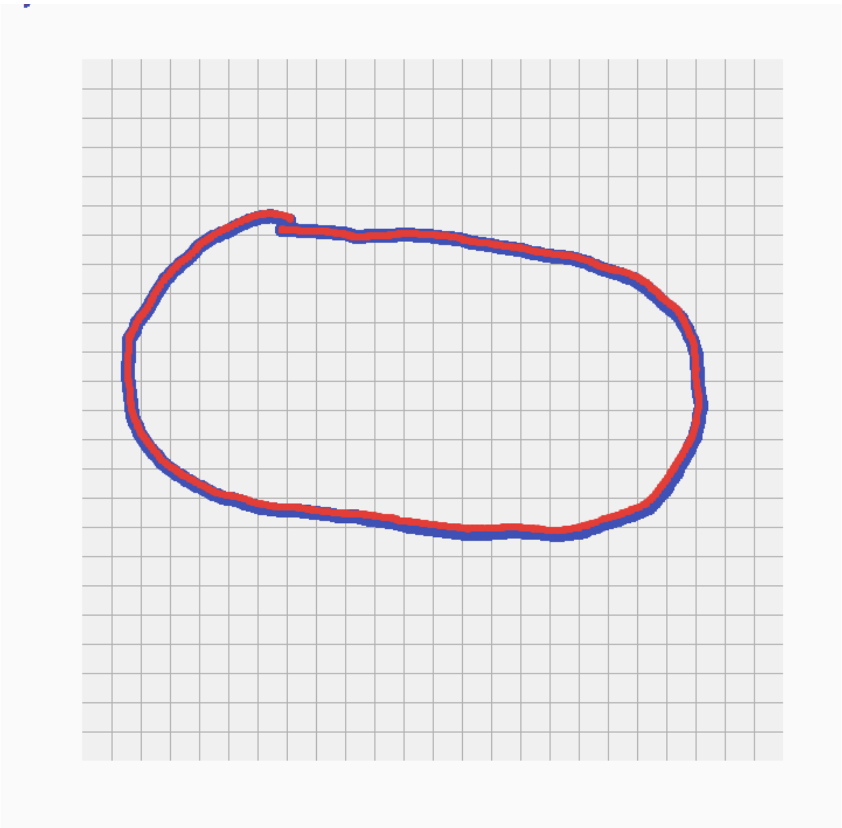
- Recalled the adjustment on R
- Tried the untrimmed R
- More shifting

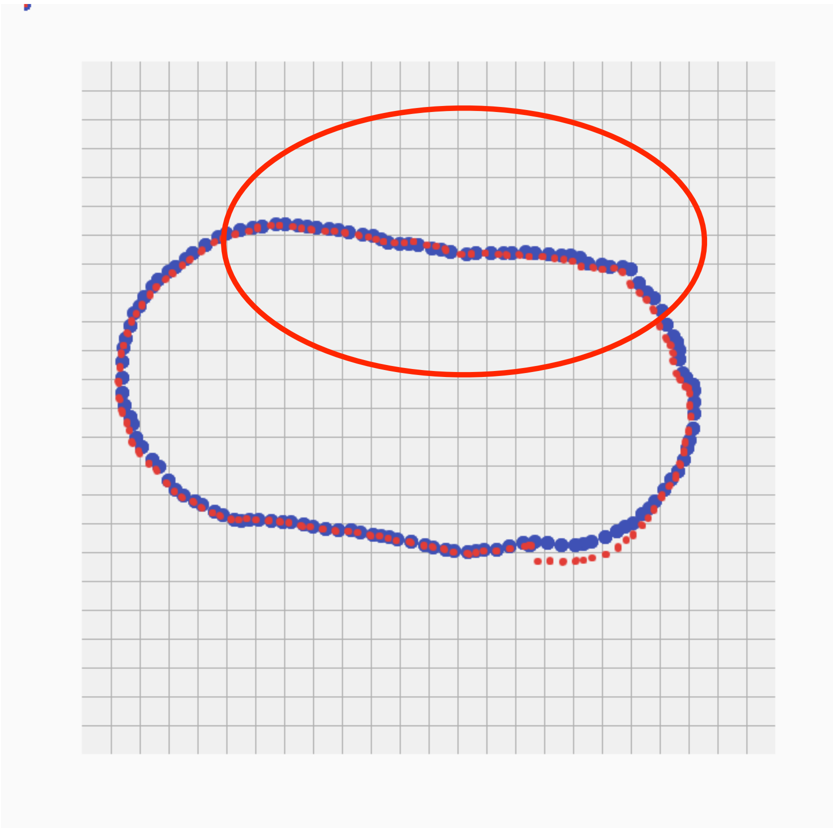


Determination

$$\begin{bmatrix} 0.002 & 0.000 & 0.000 & 0.000 \\ 0.000 & 0.003 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.001 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.001 \end{bmatrix}$$







A little filtering effect

Improvement

- Multithread Recording
  - Prevent lag recording due to weak network signal
- More accurate acceleration recording
  - Continuously (more frequently) record acceleration
  - Use mean acceleration during delta t to calculate v

Thank you