

# Locus Determination

REN HANFEI

任含菲

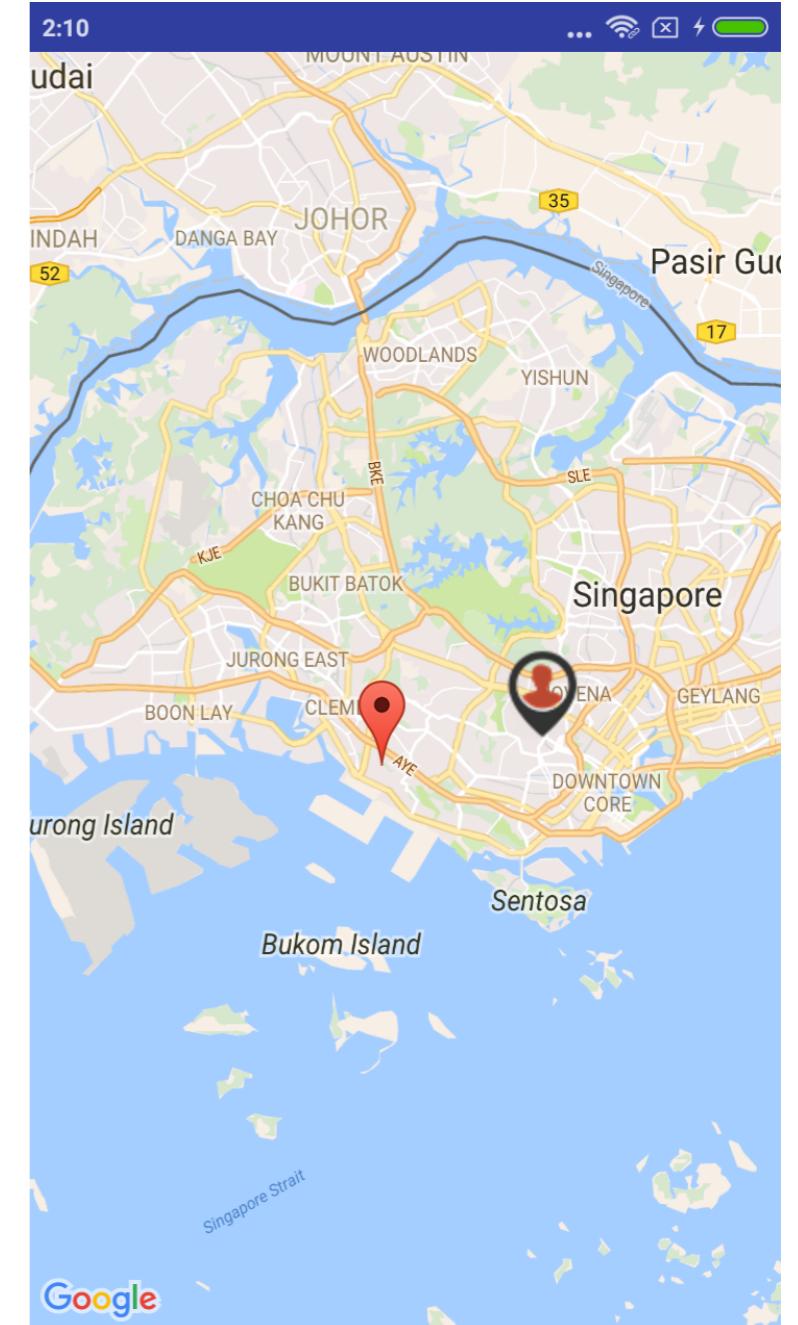
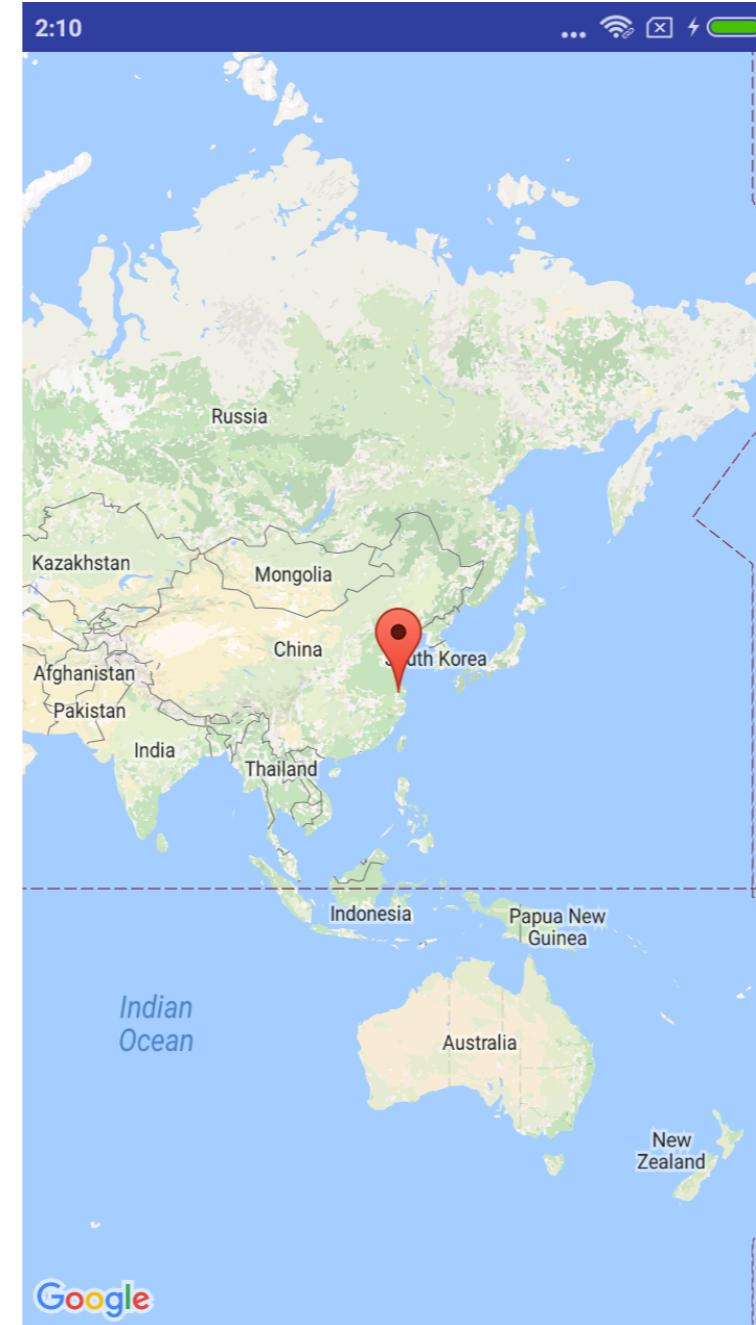
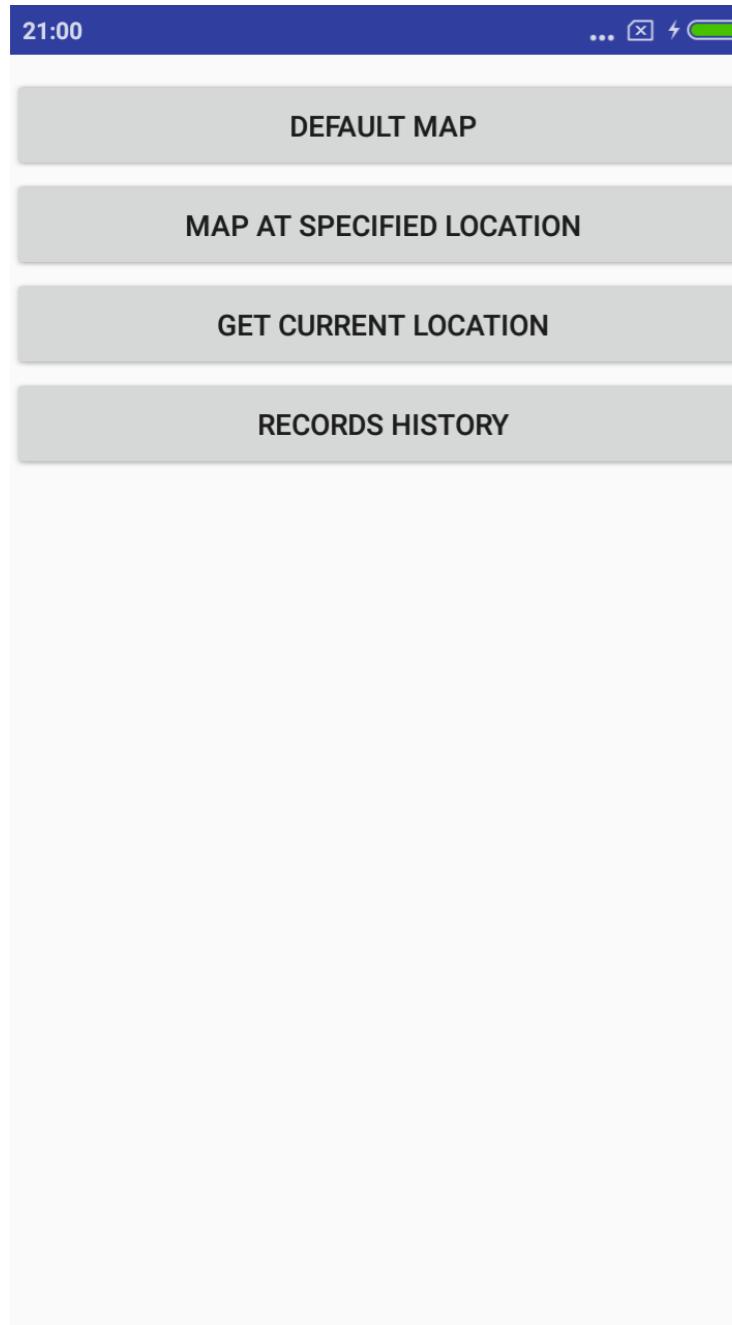
# GPS Error Determination

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

# Database Structure

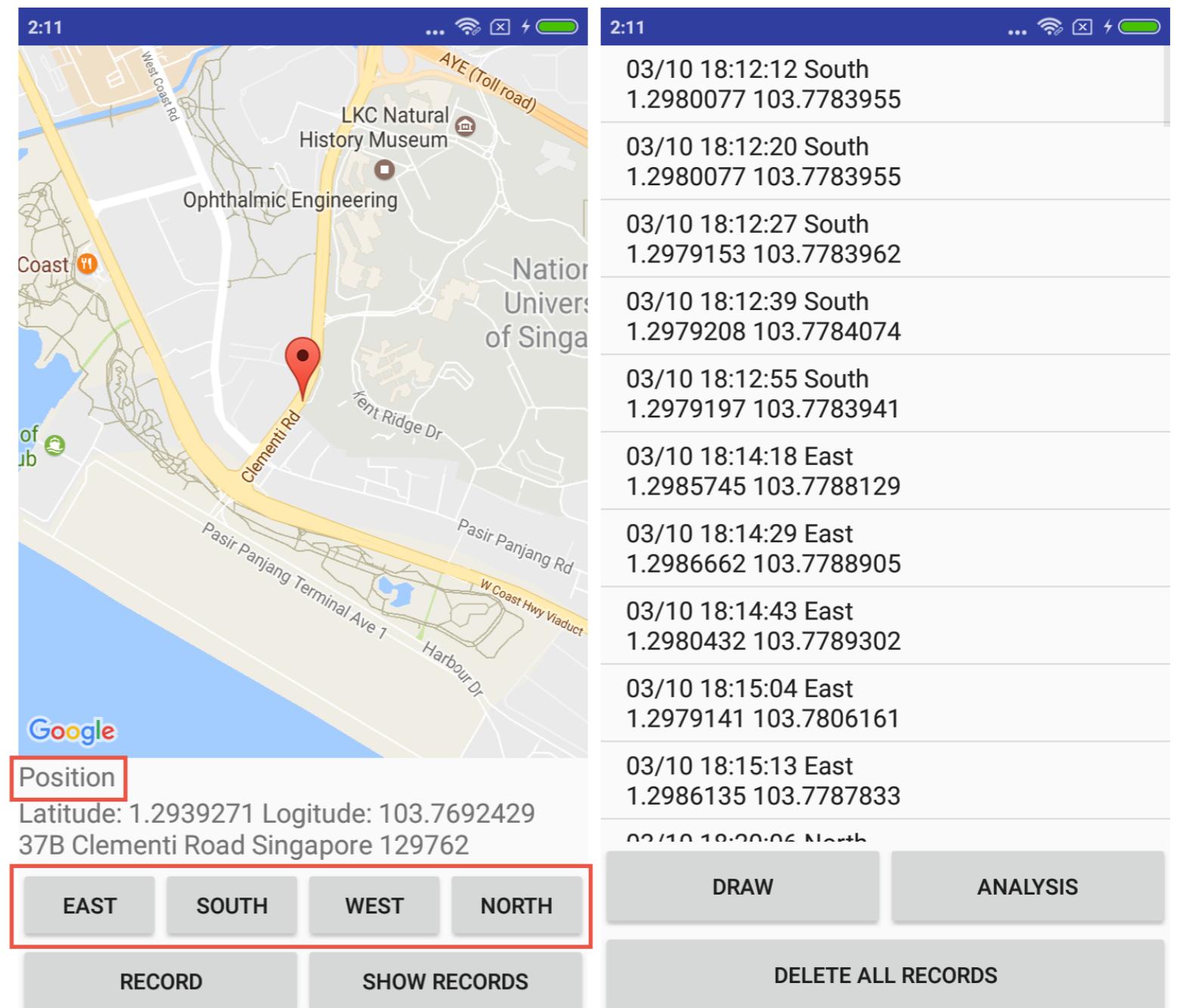
Column	Type	About
id	long	auto increment
time	text	YY-MM-DD HH:mm:ss
position	text	east / south / west / north
latitude	text	ensure precision
longitude	text	
address	text	

# User Interface



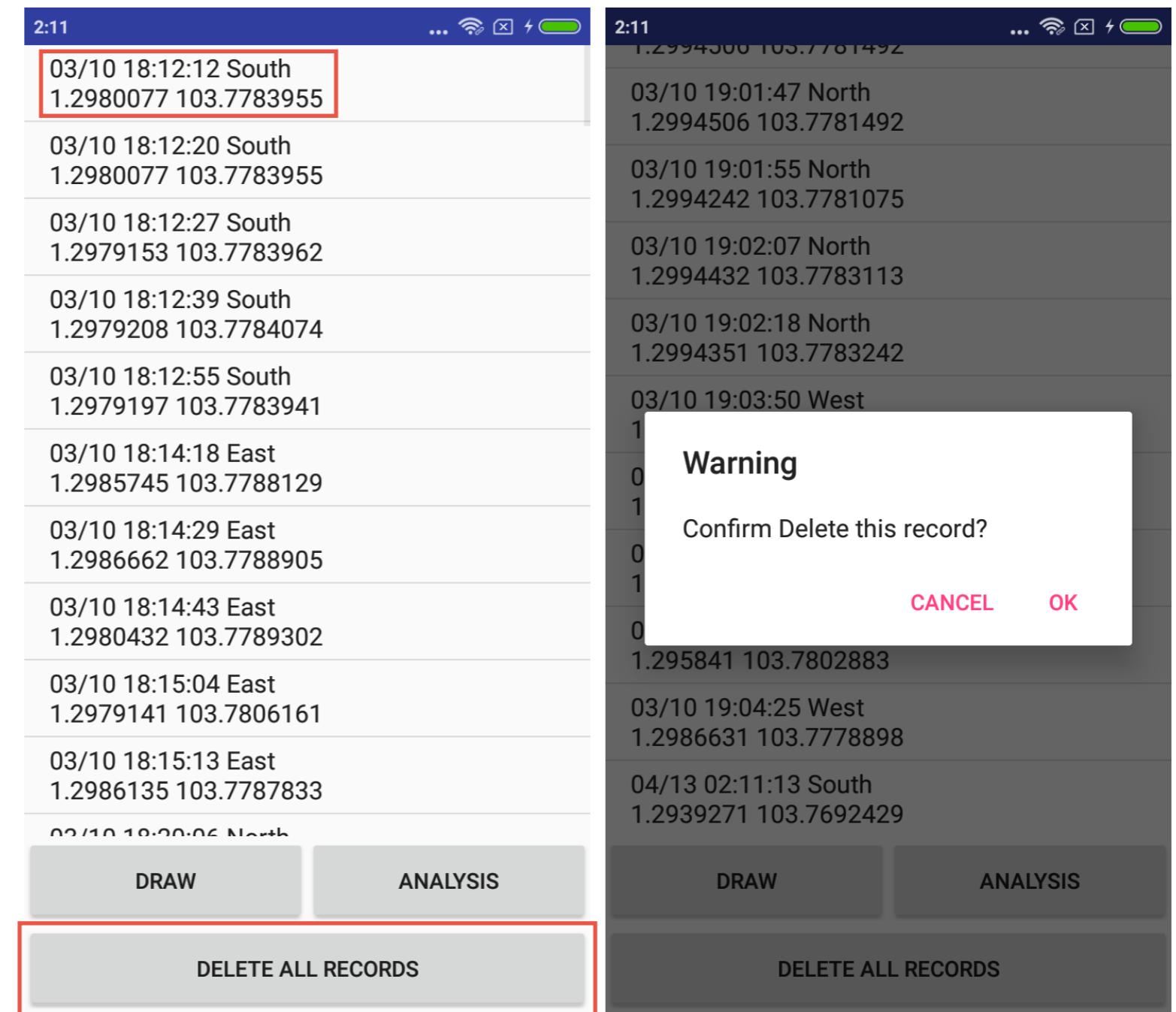
# Record Interface

- 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.



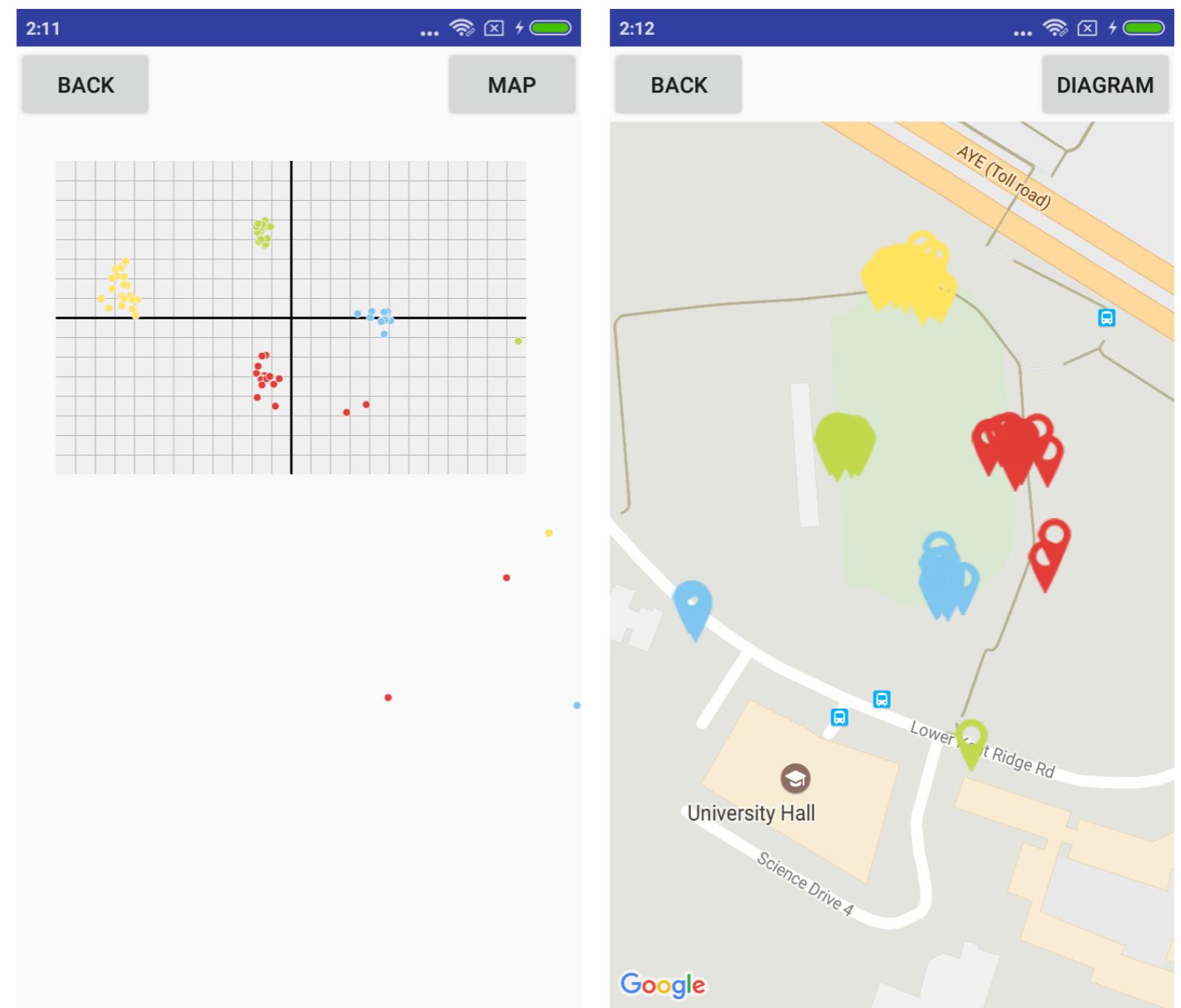
# History Interface

- 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.



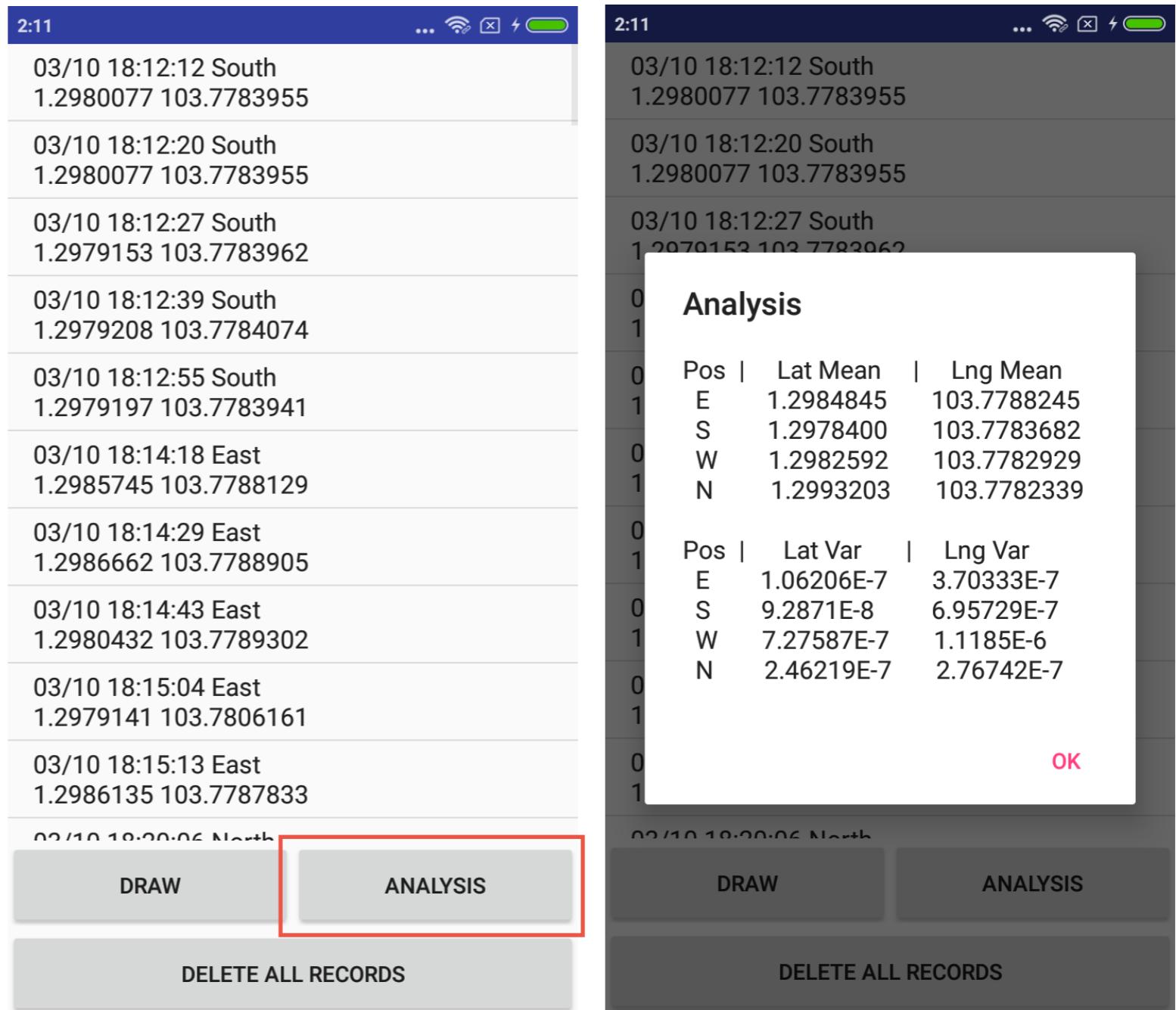
# Plot Interface

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

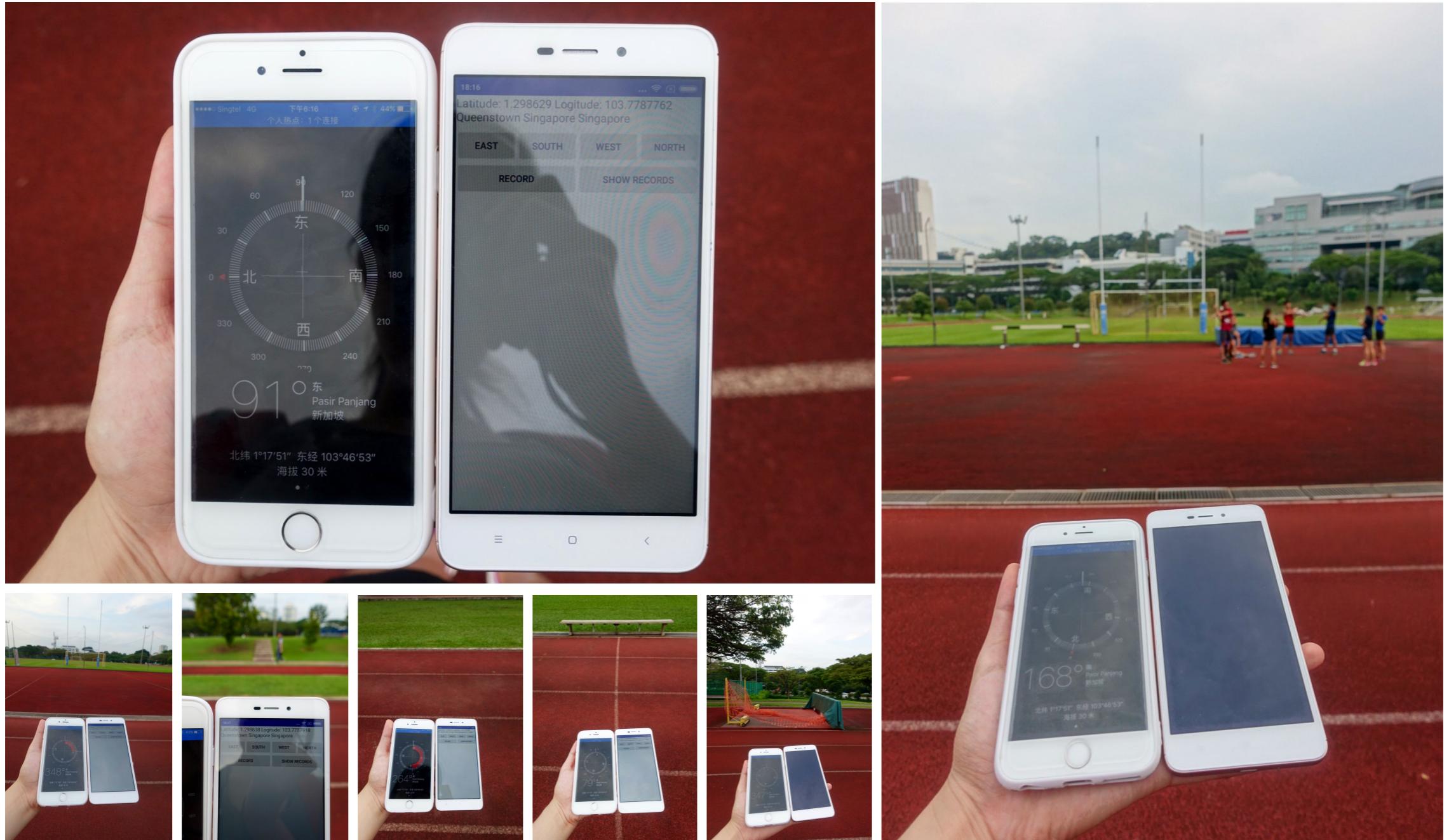


# Analysis Interface

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



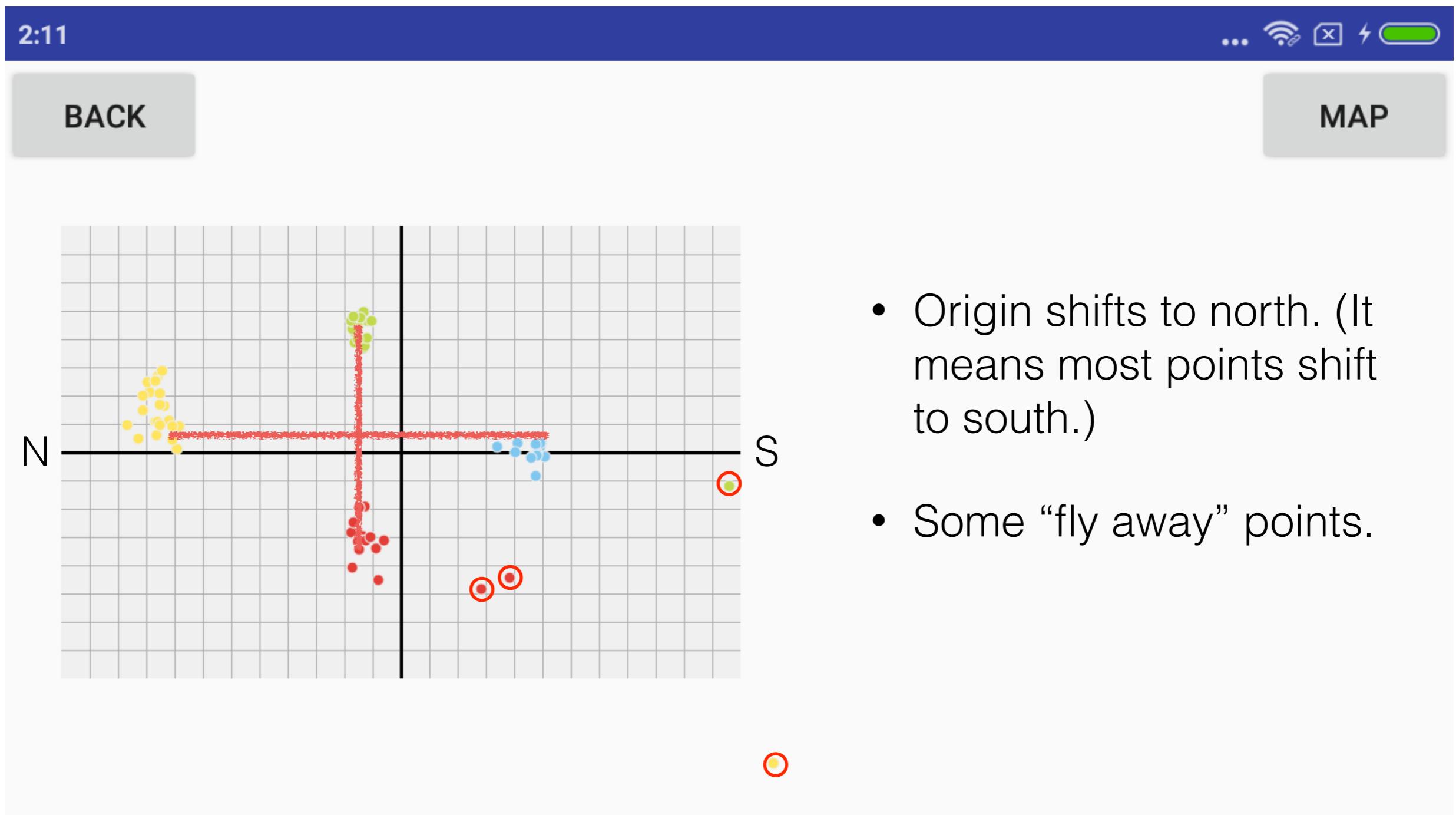
# Experimental Process



# Calculation

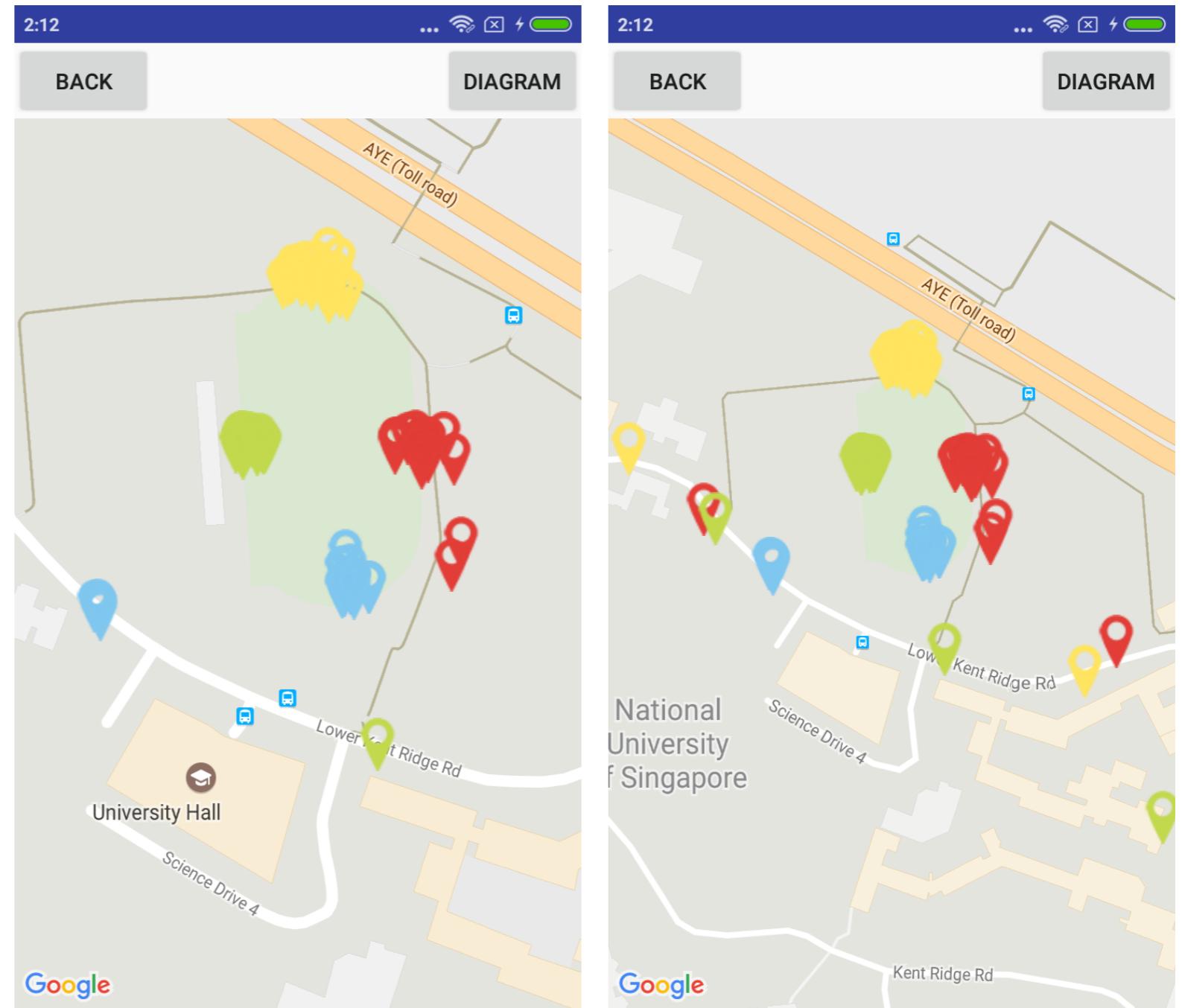
- `origin = {mean(lat), mean(lng)};`
- `kilometerPerLat = 111;`  
`kilometerPerLng = 111 * cos(origin[0] * PI / 180);`
- `latDiff = (origin[0] - geo[0]) * kilometerPerLat;`  
`lngDiff = (origin[1] - geo[1]) * kilometerPerLng;`
- `rect[0] = (int) (origin[0] + latDiff * 1000 * pixelPerMeter);`  
`rect[1]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

# Result — Diagram



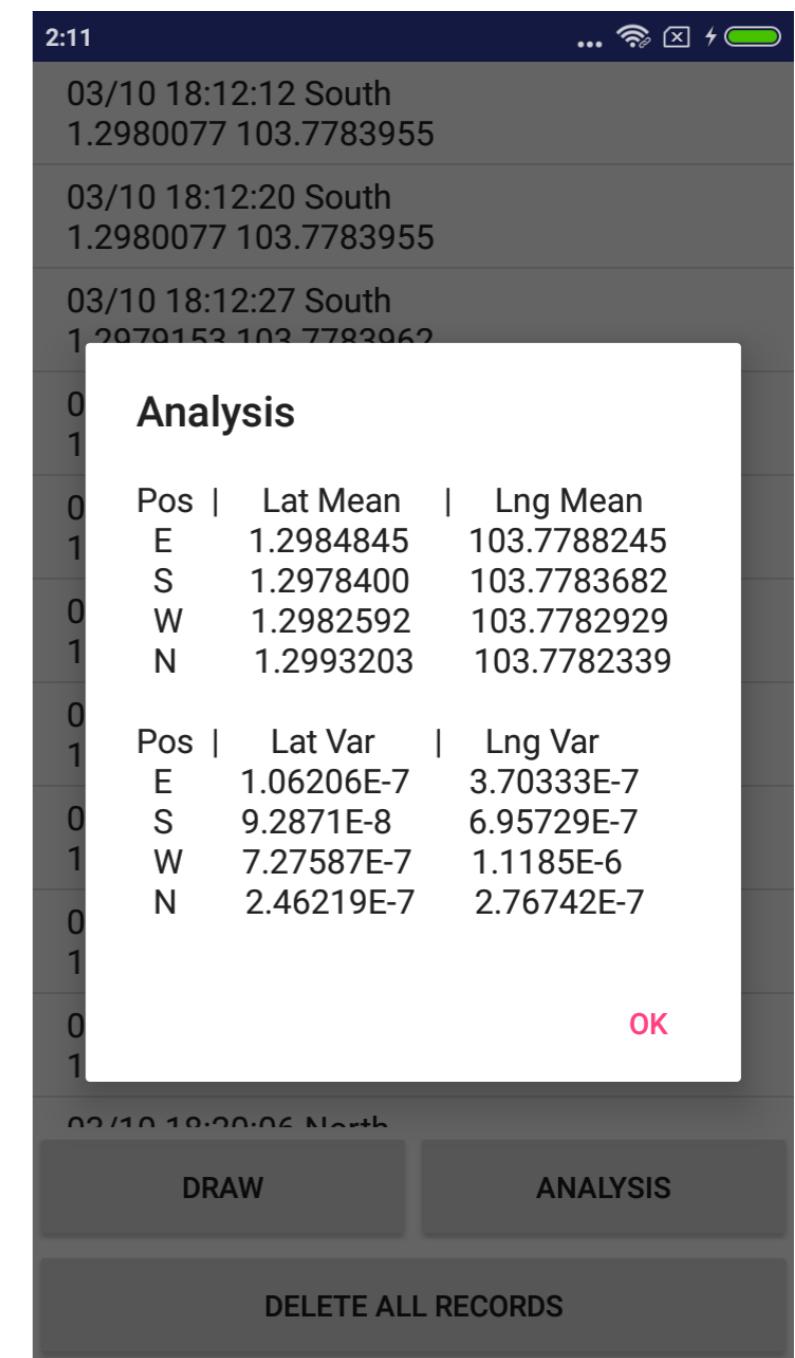
# Result — Map View

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



# Result — Mean & Variance

<b>Pos</b>	<b>Lat Mean</b>	<b>Lng Mean</b>
E	1.2984845	103.7788245
S	1.2978400	103.7783682
W	1.2982592	103.7782929
N	1.2993203	103.7782339
<b>Pos</b>	<b>Lat Var</b>	<b>Lng Var</b>
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

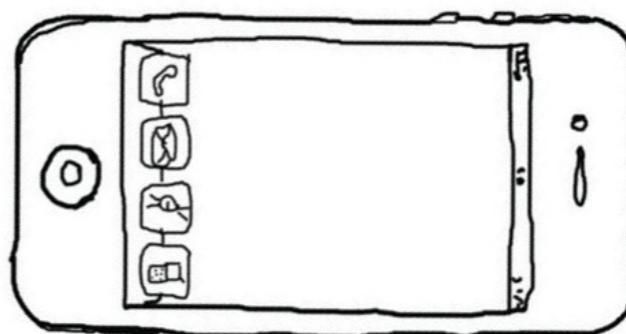
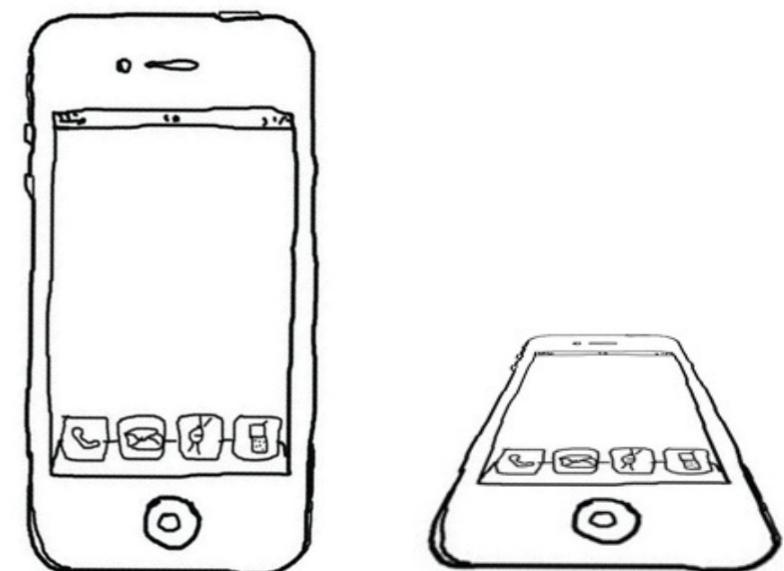


# Accelerometer Error Determination

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

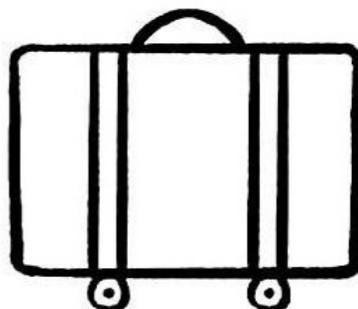
# Experiment Design

- 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)

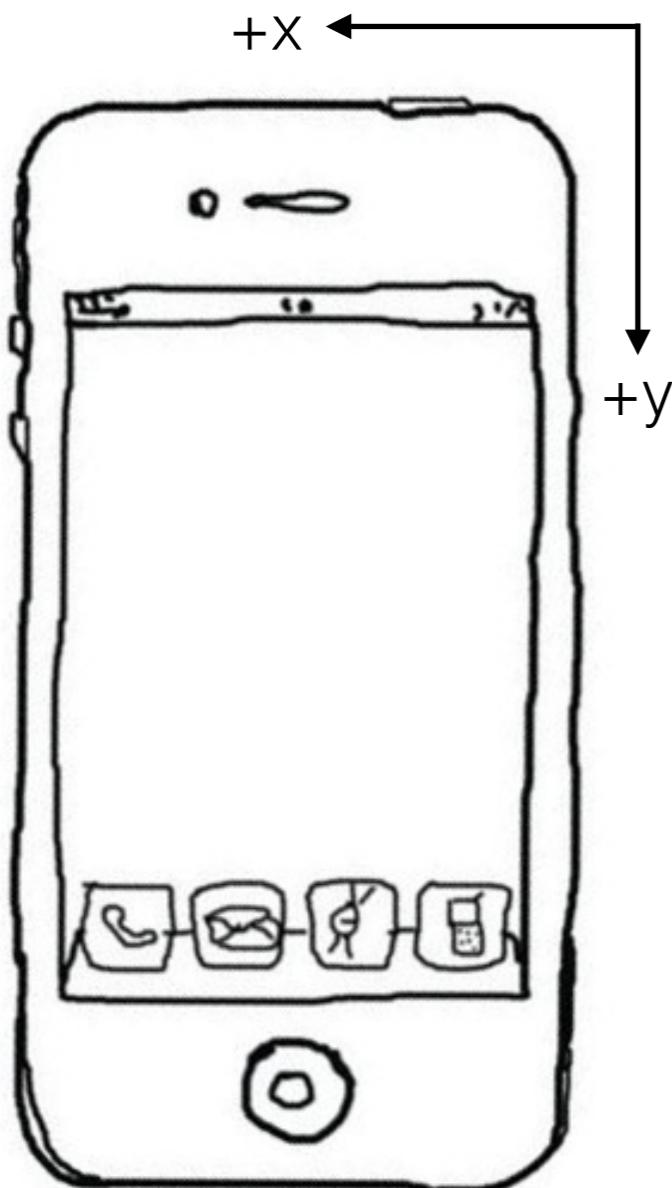


# Algorithm — Database Structure

- 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]
  - $a = \text{old reading} - \text{current reading}$  [onSensorChanged]
  - $\text{trueAcc} = 2 * \text{distance} / (\text{time}^2)$   
 $\text{accDrift} = \text{meanMeasuredAcc} - \text{trueAcc}$   
update record [onClick\_stopRecord]
- Minor direction
  - $\text{accError} = \text{meanMeasuredAcc}$

# User Interface



## Instruction

It is a 20m walking test.

**instruction**

Forward: -y Left: +x

Given the sensibility, you may need to shake the phone first when starting a new record.

## Accelerometer Data

X: 0.23

**real-time readings**

Y: -0.09

## Record Condition

Not Start

0s

+X	-X	+Y	-Y
START	STOP		
SHOW RECORDS			



## Instruction

It is a 20m walking test.

Forward: -y Left: +x

Given the sensibility, you may need to shake the phone first when starting a new record.

## Accelerometer Data

X: 0.04

Y: -0.03

## Record Condition

Recording ... go xminus ...

**status**

1.1 s **timer**

+X	-X	+Y	-Y
START	Start recording.	STOP	
SHOW RECORDS			



## Instruction

It is a 20m walking test.

Forward: -y Left: +x

Given the sensibility, you may need to shake the phone first when starting a new record.

## Accelerometer Data

X: -0.05

Y: 0.00

## Record Condition

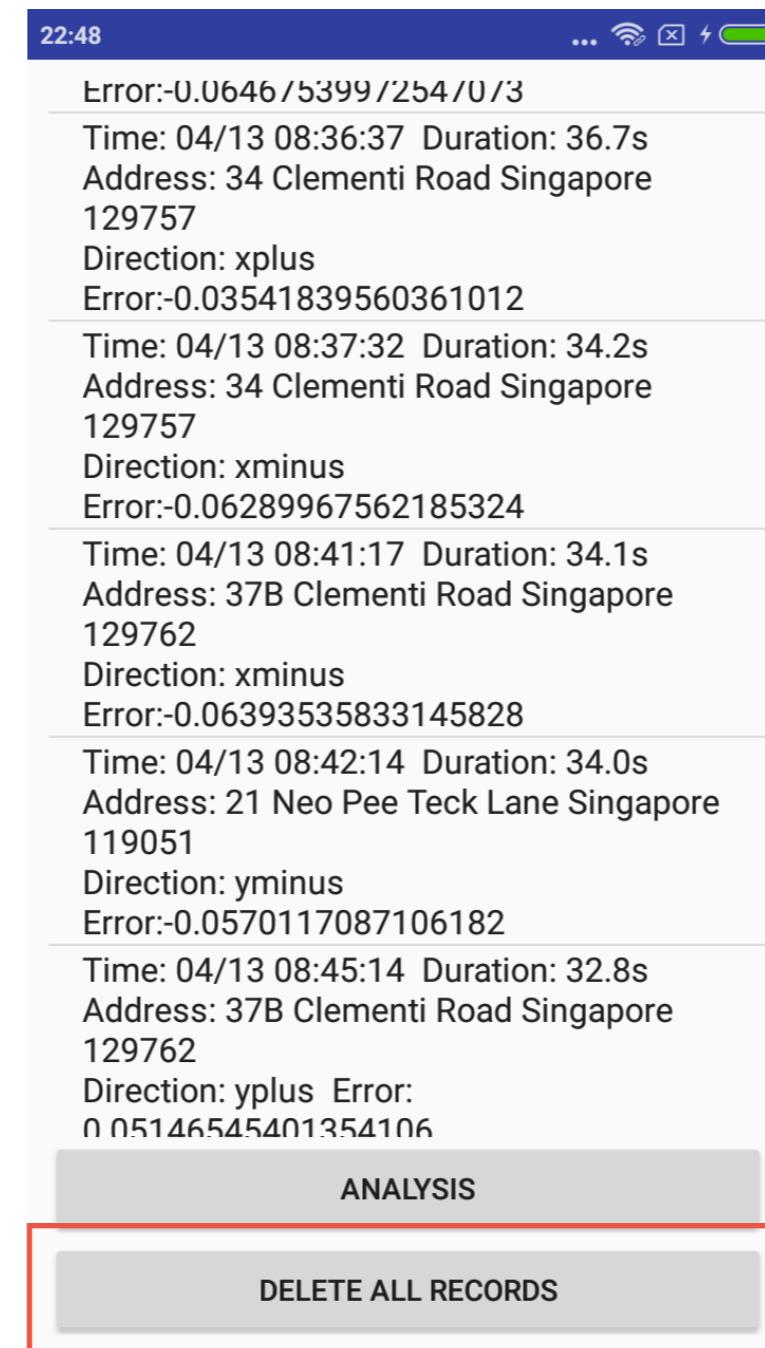
Stopped

7.0 s

+X	-X	+Y	-Y
START		STOP	
SHOW RECORDS			

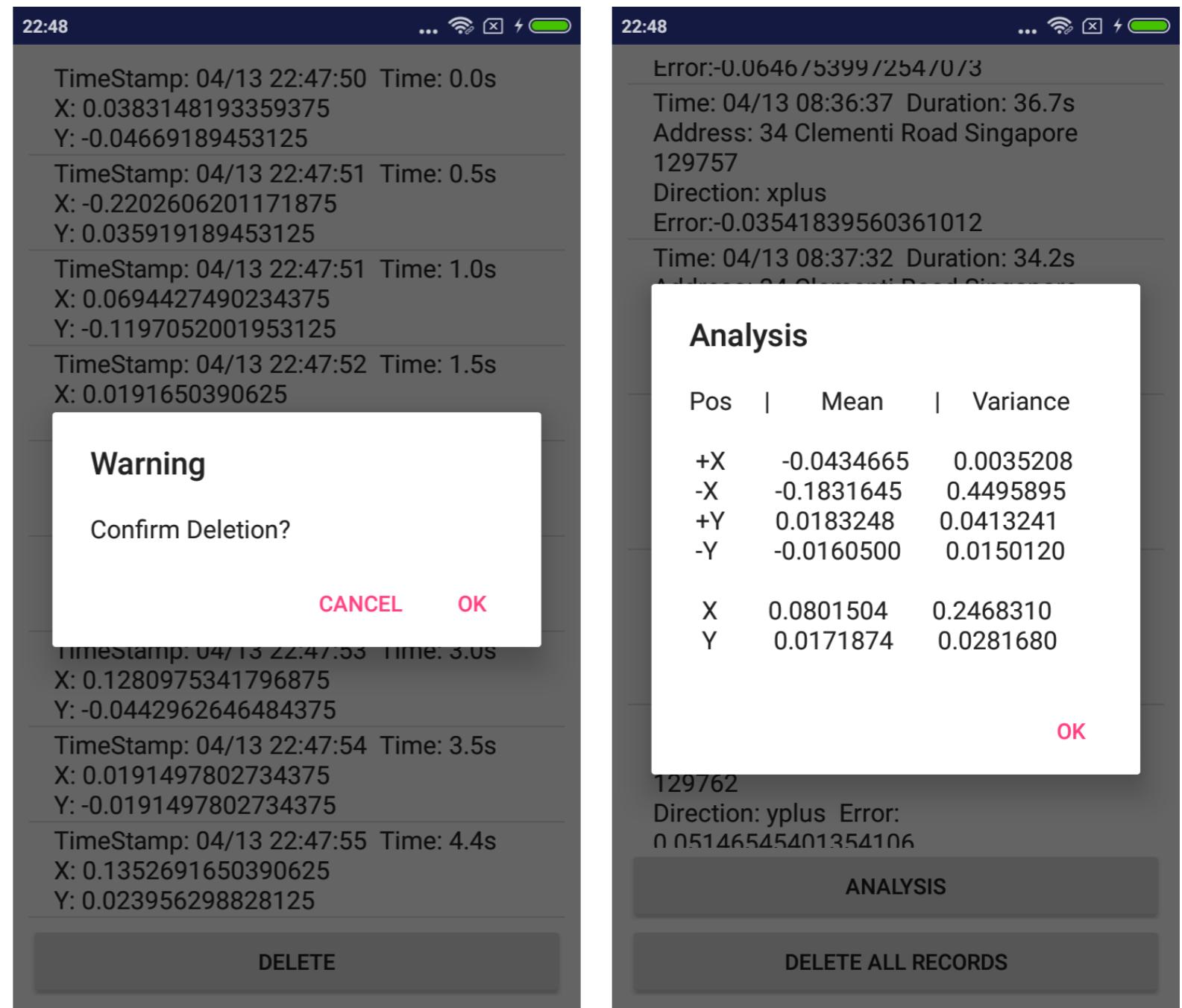
# History Interface

- 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

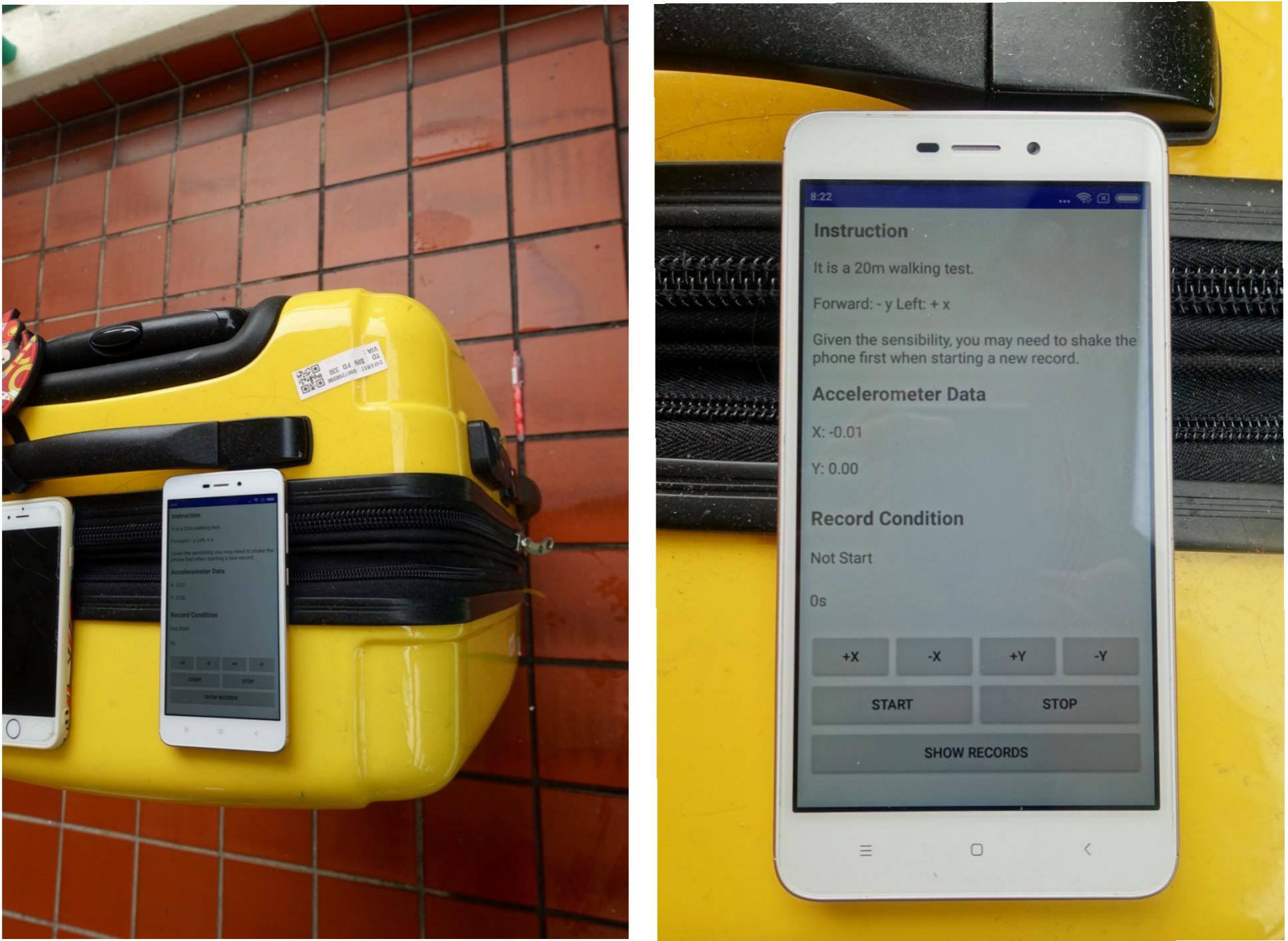
- Dorm Corridor
- With Check
- Acceptable Bump
- Convenience
- Get 20m
- Use tape
- Along check



# Marking Process

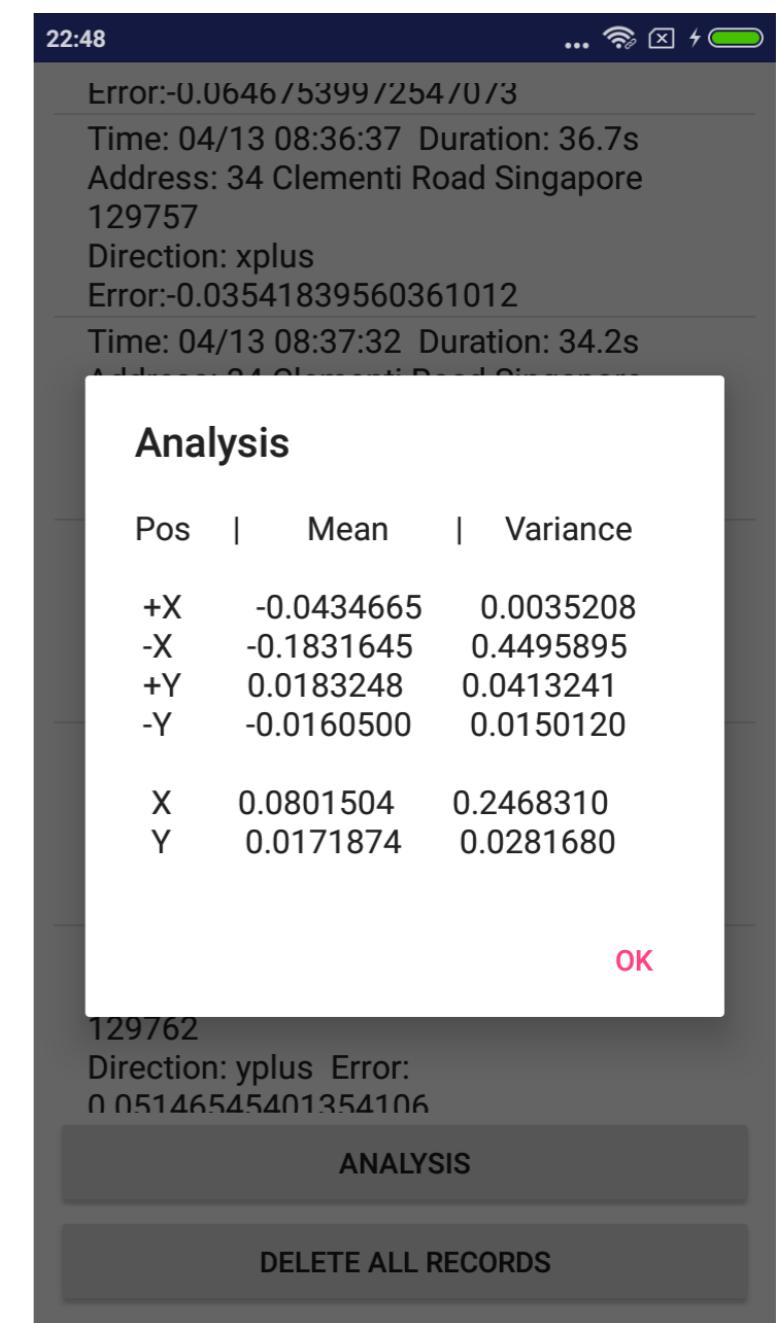


# Recording Process



# Result — Mean & Variance

<b>Pos</b>	<b>Mean</b>	<b>Variance</b>
+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)

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

$z_0, z_1, \dots$

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  
 $xy = \text{measurement}$   
 $v = 0$

2 Compute

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

3 Update

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

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

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

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

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

P

error covariance matrix  
suppose an initial value  
(invertible matrix)

Q

process noise covariance matrix  
guess and tune

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

$z_0, z_1, \dots$

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

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

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

How to get v from a?  
 $curV = prevV + (prevA + curA)/deltaT$

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

# Get R

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

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

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

11:09

... ✎

## 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

11:30

... ✎

11:47

... ✎

## Measurement Noise Covariance Matrix

175.74293, 4.0259116, 0.0, 0.0  
4.0259116, 90.301509, 0.0, 0.0  
0.0, 0.0, 7.732136E-5, 2.0560208E-5,  
0.0, 0.0, 2.0560208E-5, 8.3577526E-4,

## Measurement Noise Covariance Matrix

38.688155, 11.441456, 0.0, 0.0  
11.441456, 61.841524, 0.0, 0.0  
0.0, 0.0, 7.732136E-5, 2.0560208E-5,  
0.0, 0.0, 2.0560208E-5, 8.3577526E-4,

# Implement Kalman Filter

- Class KalmanStatus, Class KalmanFilter

```
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}
};

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

    // control input
    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

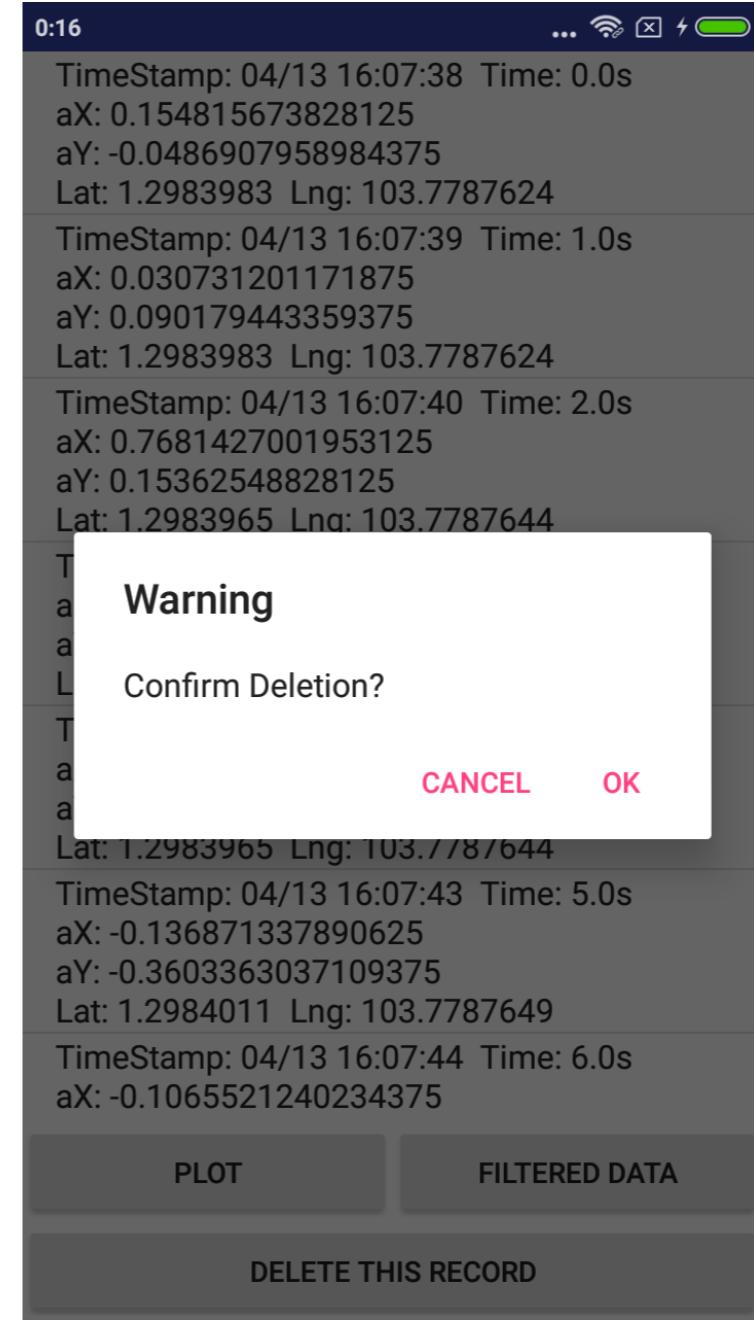
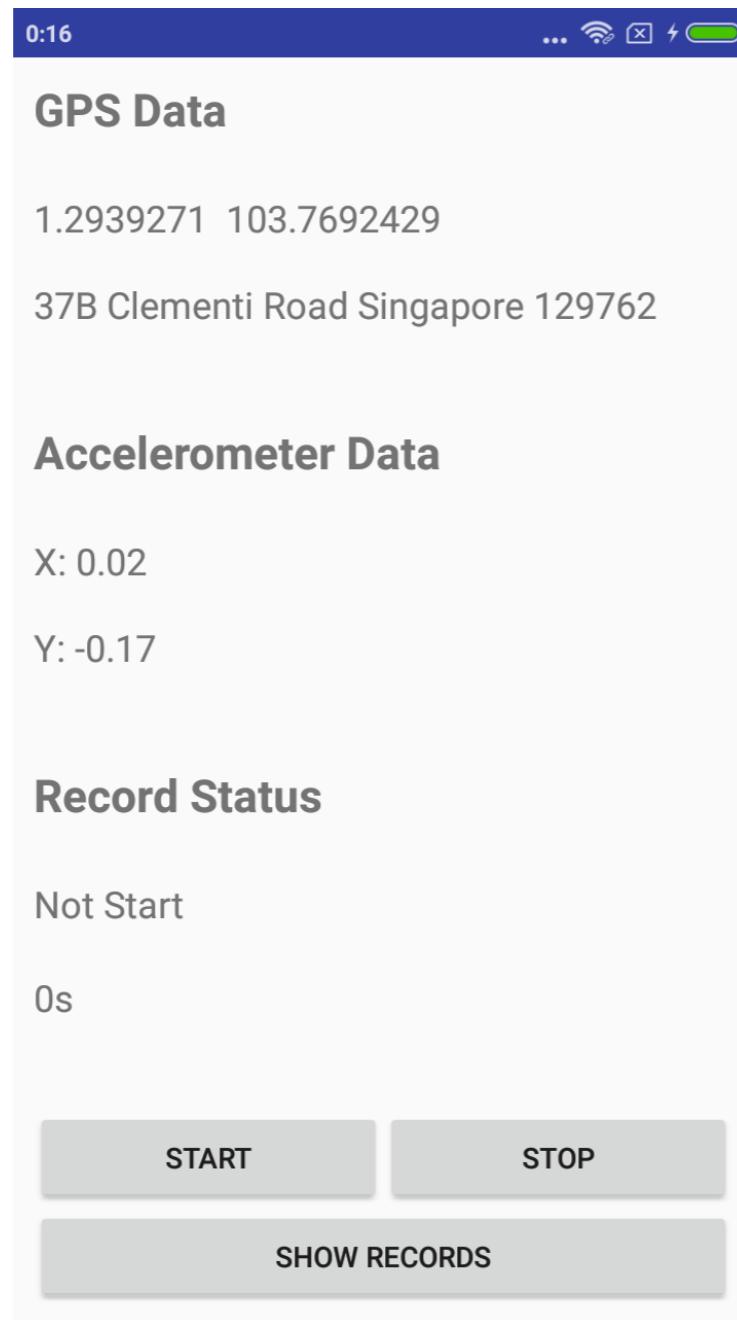
$$A^{-1} = \frac{1}{\det(A)} \text{adj}(A).$$

- When to filter

Record and filter at the same time

— Save time and space, but cause lag

# User Interface



# History Interface

0:15 ...

Time: 04/13 09:37:31 Duration: 417.6s  
Address: 1207 Dover Road Singapore 139654

---

Time: 04/13 09:47:08 Duration: 333.8s  
Address: 21 Lower Kent Ridge Road  
Singapore 119077

---

Time: 04/13 09:53:11 Duration: 319.0s  
Address: 25 Lower Kent Ridge Road  
Singapore 119081

---

Time: 04/13 09:59:00 Duration: 311.4s  
Address: 21 Lower Kent Ridge Road  
Singapore 119077

---

Time: 04/13 10:04:26 Duration: 322.5s  
Address: 21 Lower Kent Ridge Road  
Singapore 119077

---

Time: 04/13 15:35:13 Duration: 379.9s  
Address: 21 Lower Kent Ridge Road  
Singapore 119077

---

Time: 04/13 15:42:10 Duration: 350.9s  
Address: 21 Lower Kent Ridge Road  
Singapore 119077

---

Time: 04/13 15:54:33 Duration: 341.0s  
Address: 21 Lower Kent Ridge Road  
Singapore 119077

---

Time: 04/13 16:01:01 Duration: 353.9s  
Address: 21 Lower Kent Ridge Road  
Singapore 119077

---

Time: 04/13 16:07:38 Duration: 388.0s

**DELETE ALL RECORDS**

0:16 ...

TimeStamp: 04/13 16:07:38 Time: 0.0s  
aX: 0.154815673828125  
aY: -0.0486907958984375  
Lat: 1.2983983 Lng: 103.7787624

---

TimeStamp: 04/13 16:07:39 Time: 1.0s  
aX: 0.030731201171875  
aY: 0.090179443359375  
Lat: 1.2983983 Lng: 103.7787624

---

TimeStamp: 04/13 16:07:40 Time: 2.0s  
aX: 0.7681427001953125  
aY: 0.15362548828125  
Lat: 1.2983965 Lng: 103.7787644

---

TimeStamp: 04/13 16:07:41 Time: 3.0s  
aX: 0.670379638671875  
aY: -0.358734130859375  
Lat: 1.2983965 Lng: 103.7787644

---

TimeStamp: 04/13 16:07:42 Time: 4.0s  
aX: 0.792877197265625  
aY: 0.0343170166015625  
Lat: 1.2983965 Lng: 103.7787644

---

TimeStamp: 04/13 16:07:43 Time: 5.0s  
aX: -0.136871337890625  
aY: -0.3603363037109375  
Lat: 1.2984011 Lng: 103.7787649

---

TimeStamp: 04/13 16:07:44 Time: 6.0s  
aX: -0.1065521240234375

**PLOT**

**FILTERED DATA**

**DELETE THIS RECORD**

0:15 ...

TimeStamp: 04/13 15:42:10 Time: 0.0  
X: 29.04870000000348m  
Y: 40.04961019882311m  
filteredX: 29.04870000000348m  
filteredY: 40.04961019882311m  
Vx: 0.0845947265625m/s  
Vy: -0.07022857666015625m/s

---

TimeStamp: 04/13 15:42:11 Time: 1.0  
X: 29.04870000000348m  
Y: 40.04961019882311m  
filteredX: 29.04874821646712m  
filteredY: 40.04954074638813m  
Vx: 0.2027130126953125m/s  
Vy: -0.12609100341796875m/s

---

TimeStamp: 04/13 15:42:12 Time: 2.0  
X: 29.126399999999553m  
Y: 40.06070734859138m  
filteredX: 29.126213751639618m  
filteredY: 40.06043758274566m  
Vx: -0.28530120849609375m/s  
Vy: -0.1460418701171875m/s

---

TimeStamp: 04/13 15:42:13 Time: 3.0  
X: 28.893299999986688m  
Y: 40.016318752672305m  
filteredX: 28.89399634182064m  
filteredY: 40.01660983908572m  
Vx: -1.2026216125516875m/s

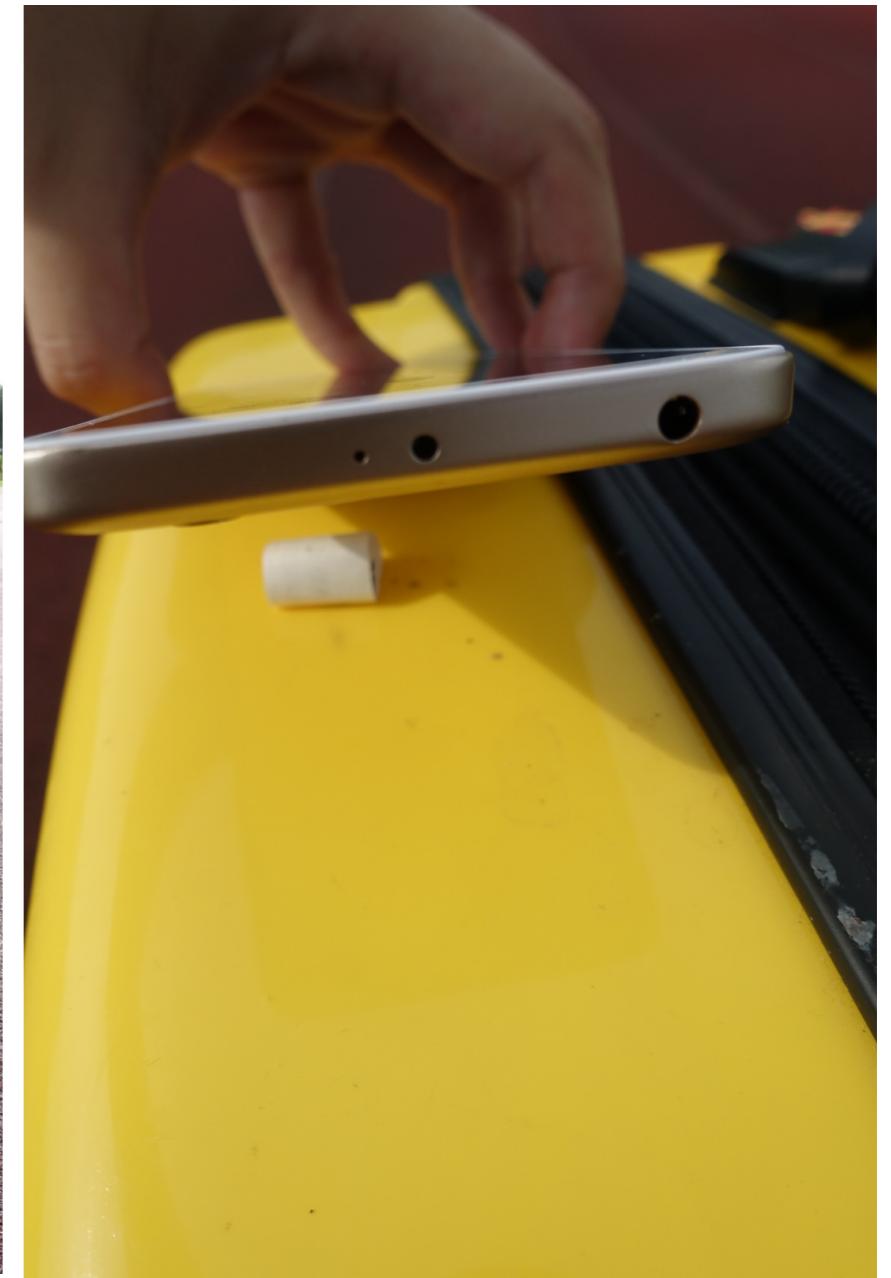
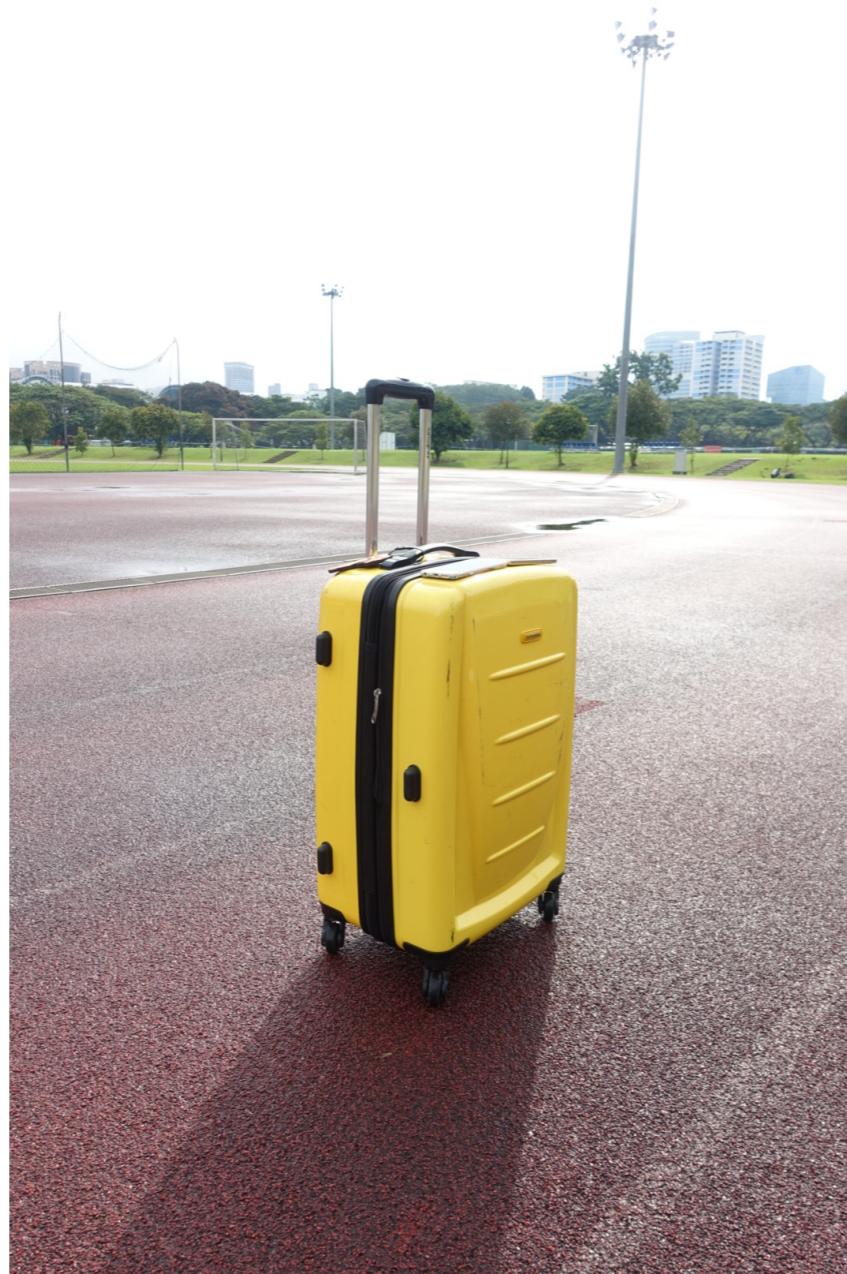
**BACK**

**PLOT**

**UPDATE FILTERING**

# Process

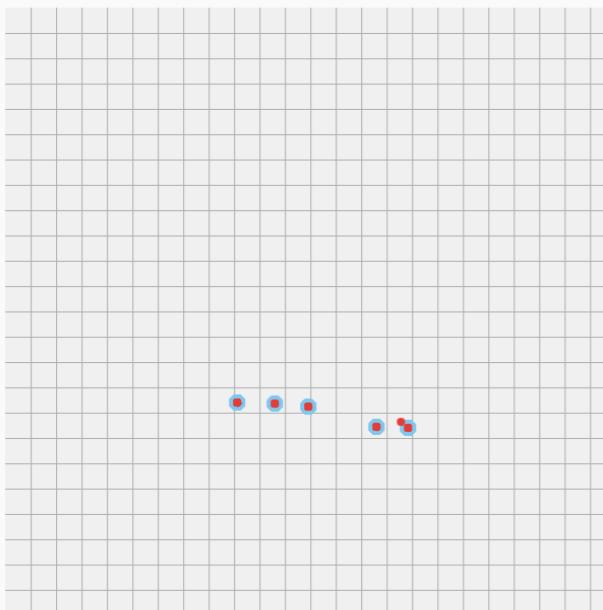
- Stick
- Reduce bump
- Prevent flip
- Adjust interval
- Get more acceleration for a more accurate v



# Accident

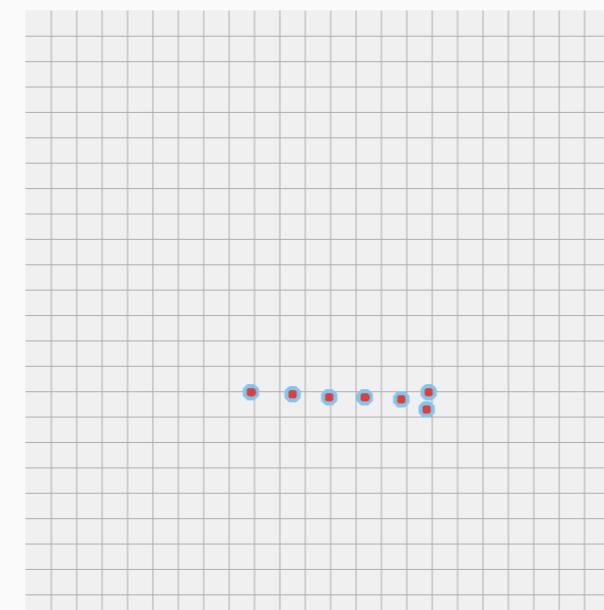
18:18 ... ⚡

BACK



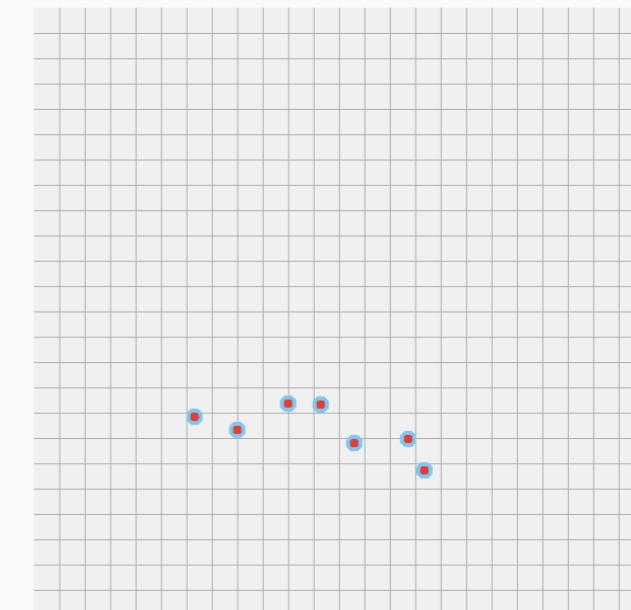
18:18 ... ⚡

BACK



18:18 ... ⚡

BACK

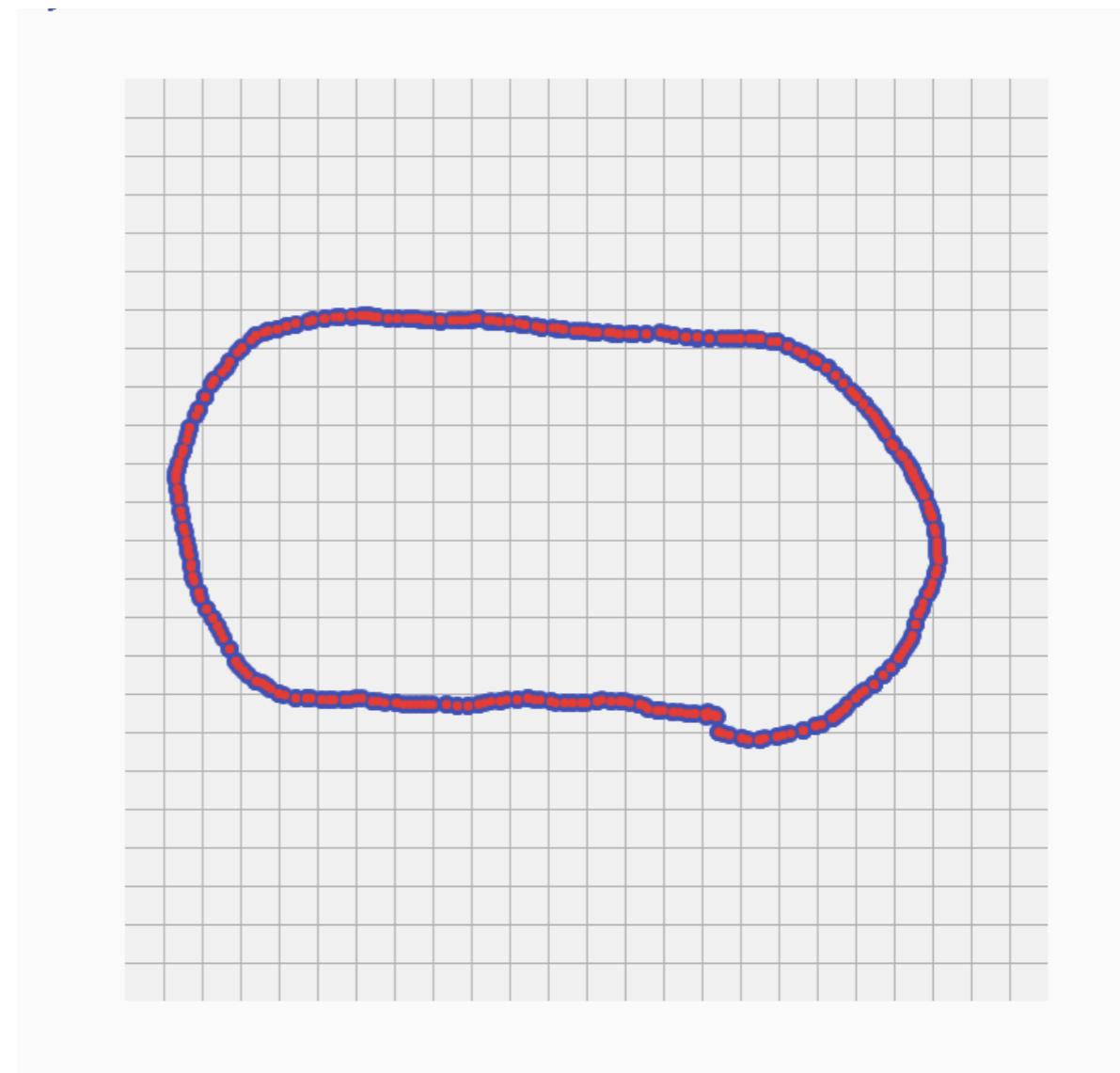


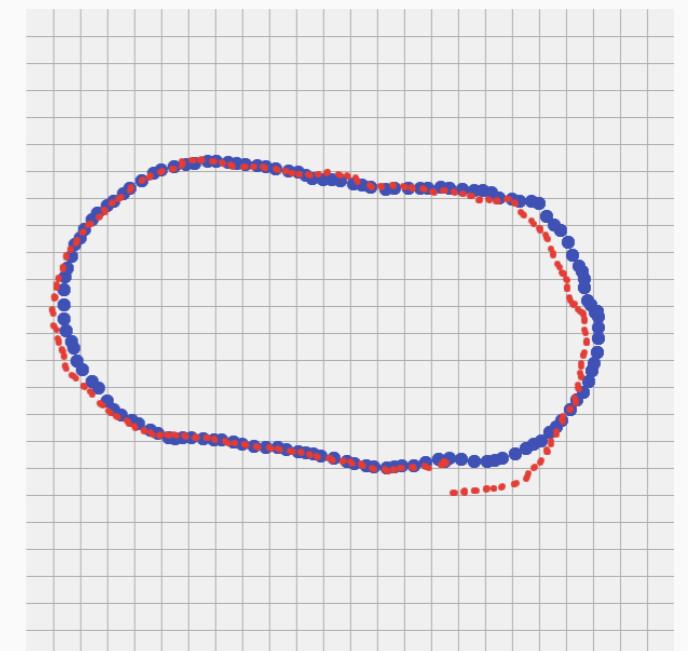
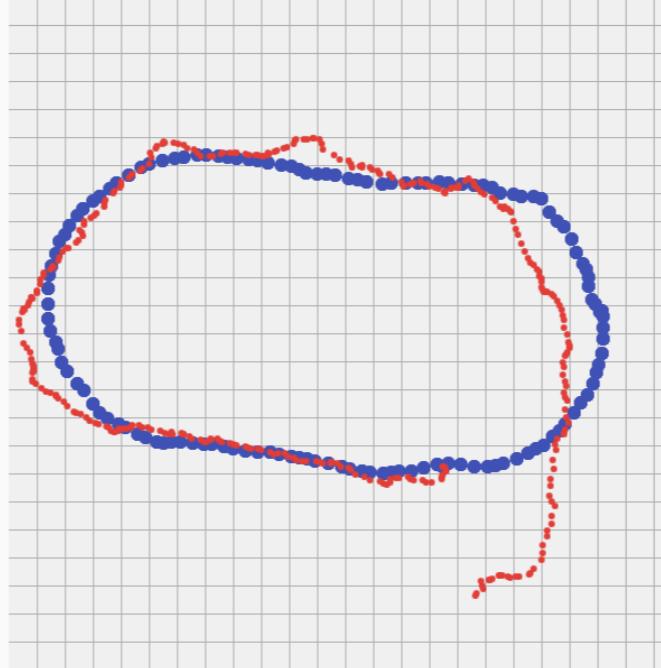
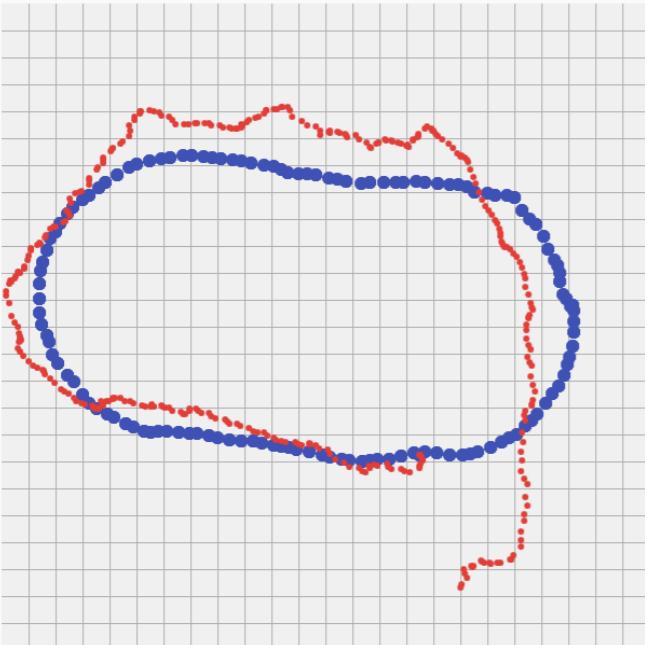
Lose many records

Weak network leads to location listener lagging

# Result

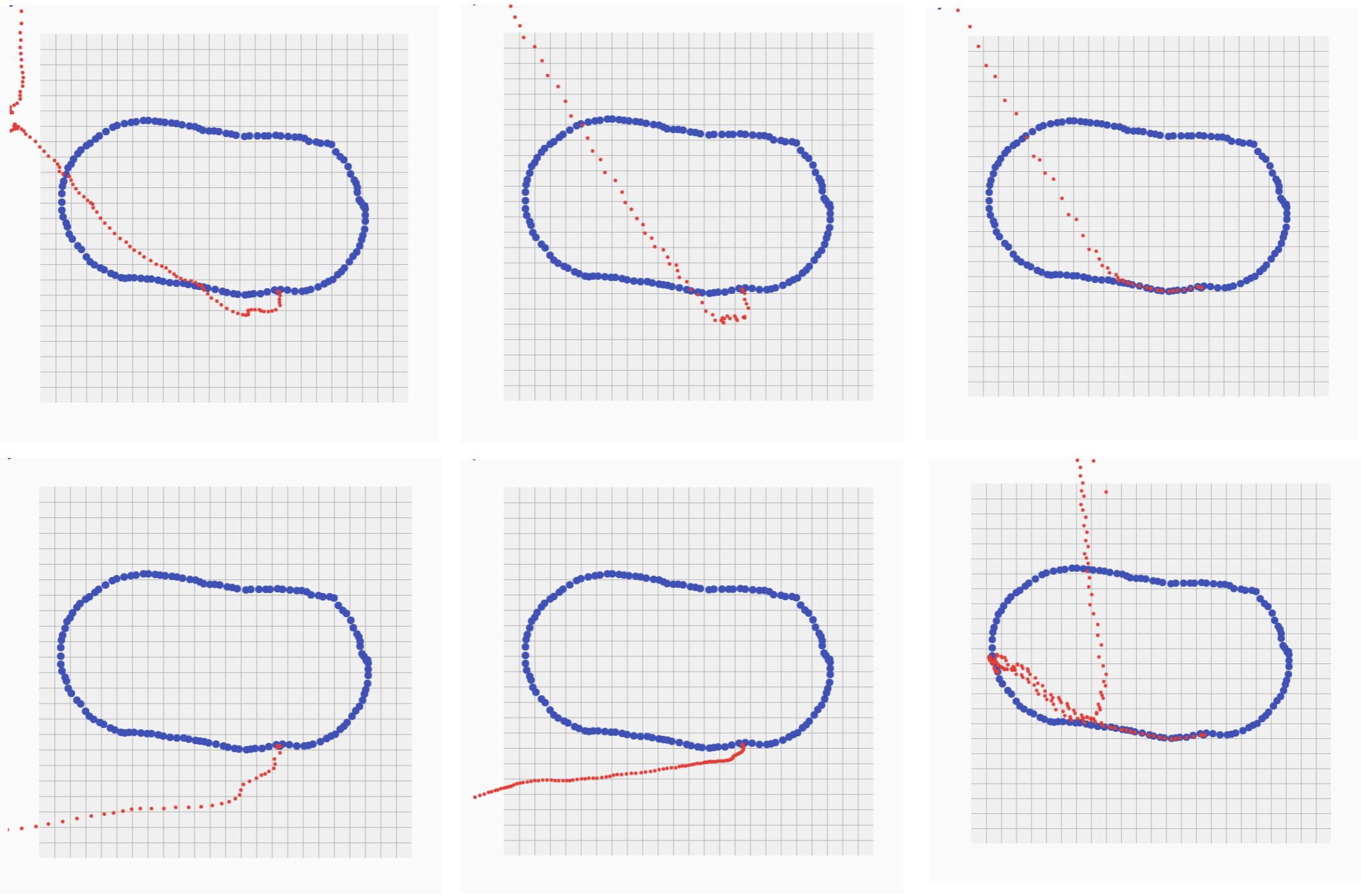
- Blue — raw data
- Red — filtered data
- Nothing wrong, just cannot see “filtered”
- Try tuning Q and change diagonal element first ( $x, y, vx, vy$ )





## Q — Tuning

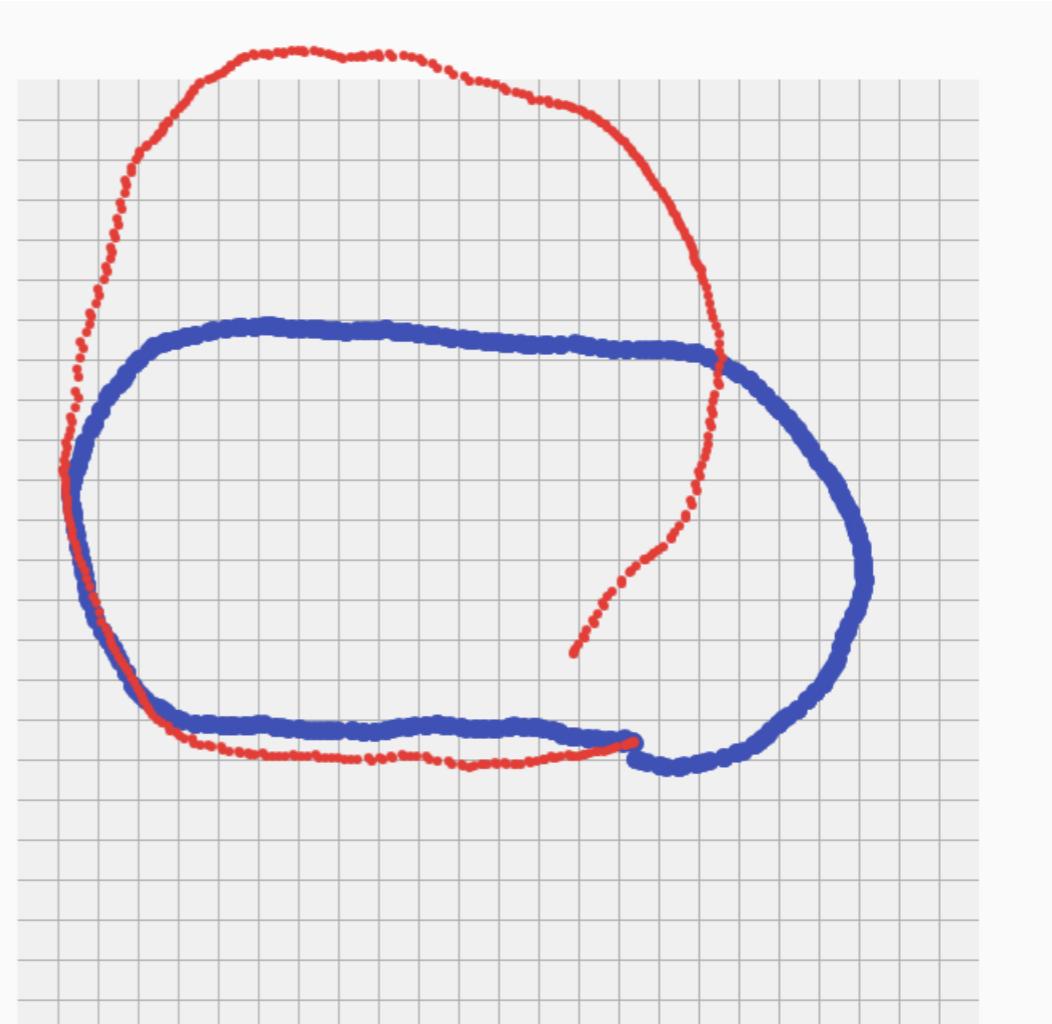
- Change the first two diagonal element
- Track becomes more bumping
- Reduce the elements, a little effect on filtering, but disconnect start and end of the track



- 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)

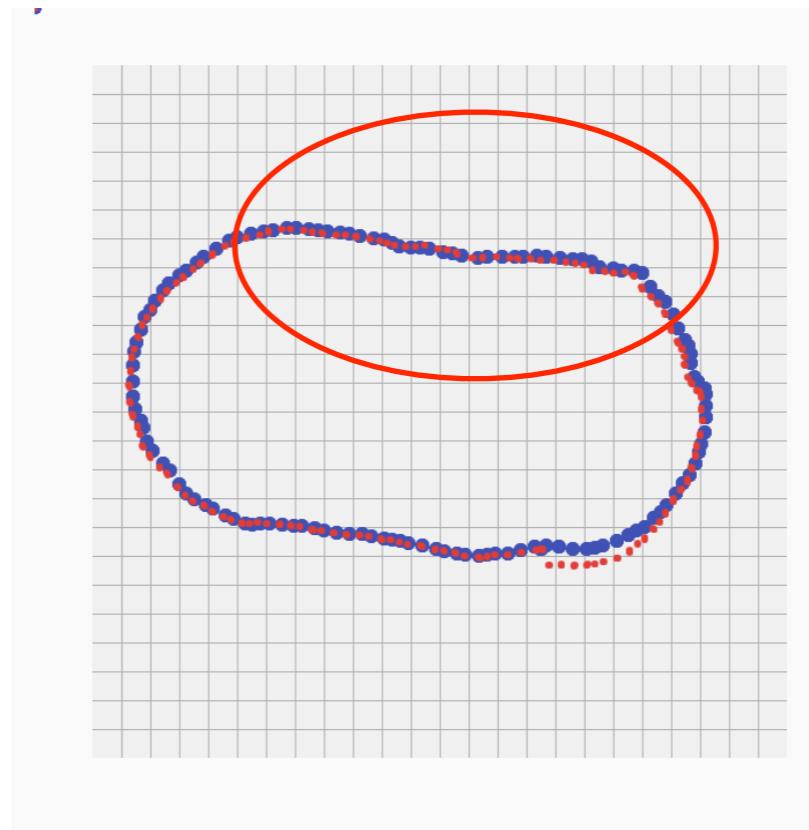
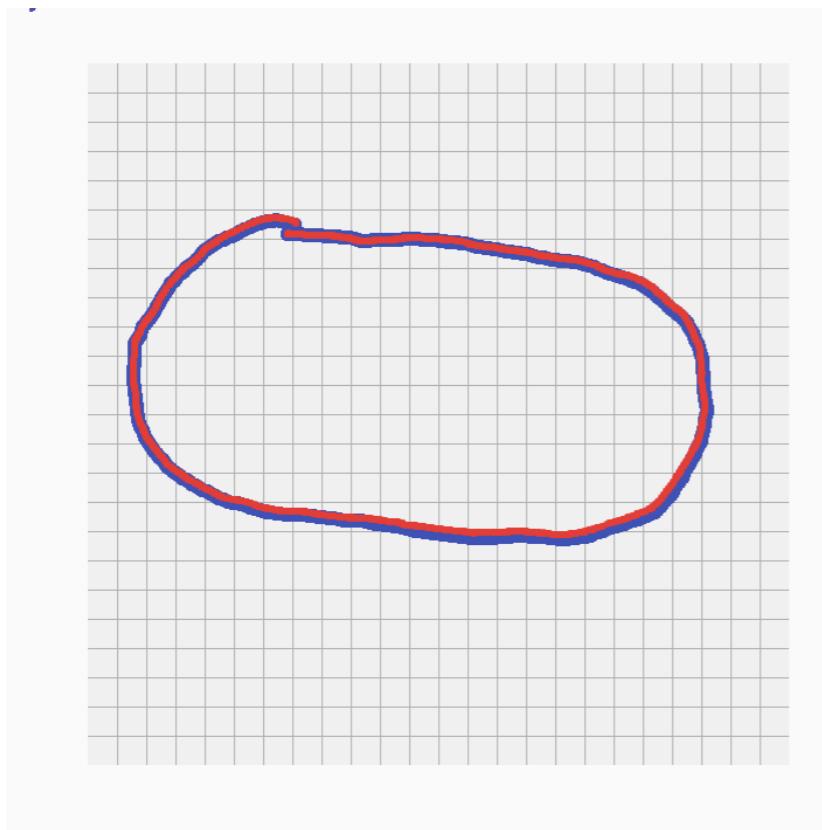
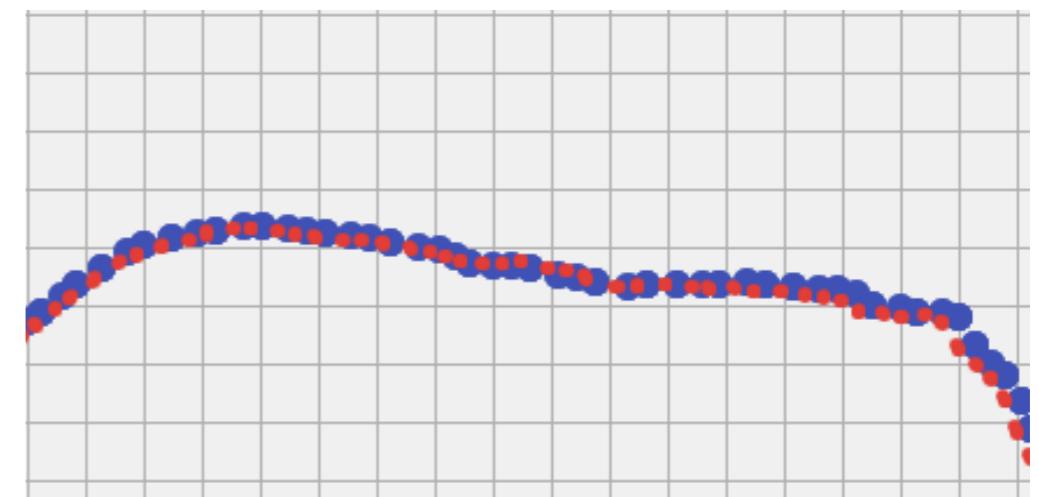
# Tuning

- 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