# Walking Style Authentication (WSA)

Group 5:

Jianbo Pei, Jingyuan Wang, Bowen Sun

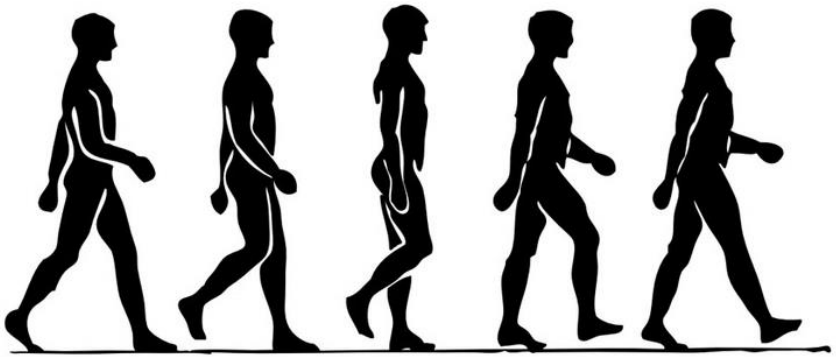
# Overview

- Idea
- Approaches
  - Data Collecting & Processing
  - Linear Regression and SVM
    - Related Works
    - Results
- Future Works


腾讯视频

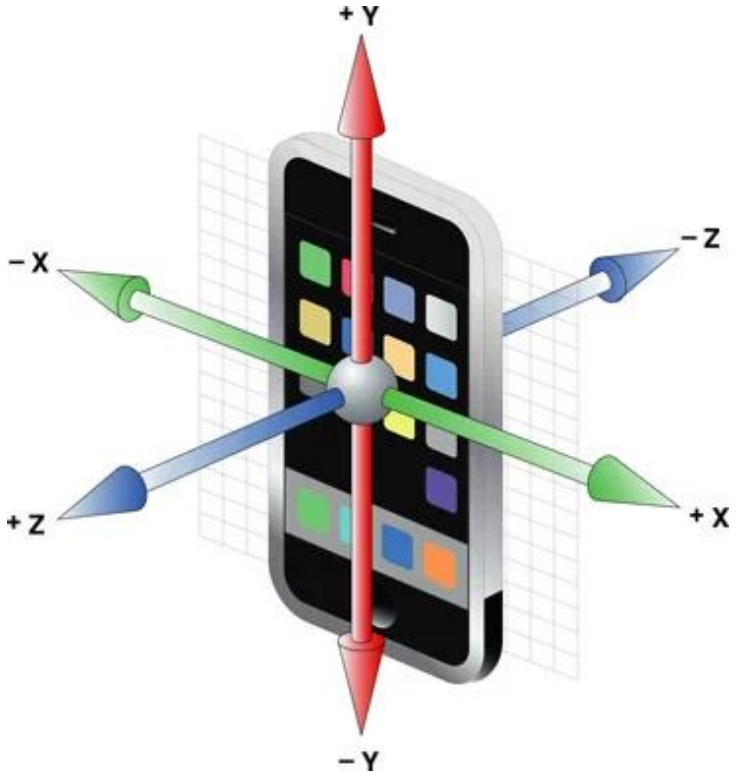
# Idea

- A new method of biometric authentication based on unique walking style of each individual.



# Motion Sensors

- step counter sensor
- linear acceleration sensor



# Approach Flow Chart

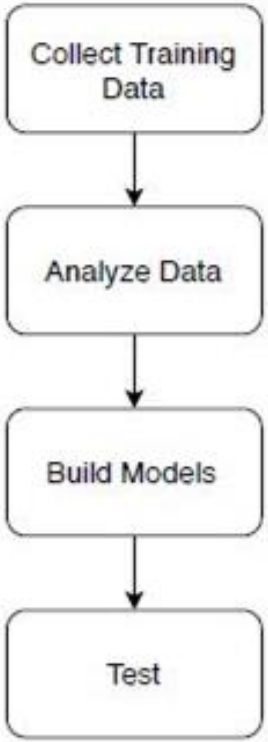



Figure 1. System Initialization

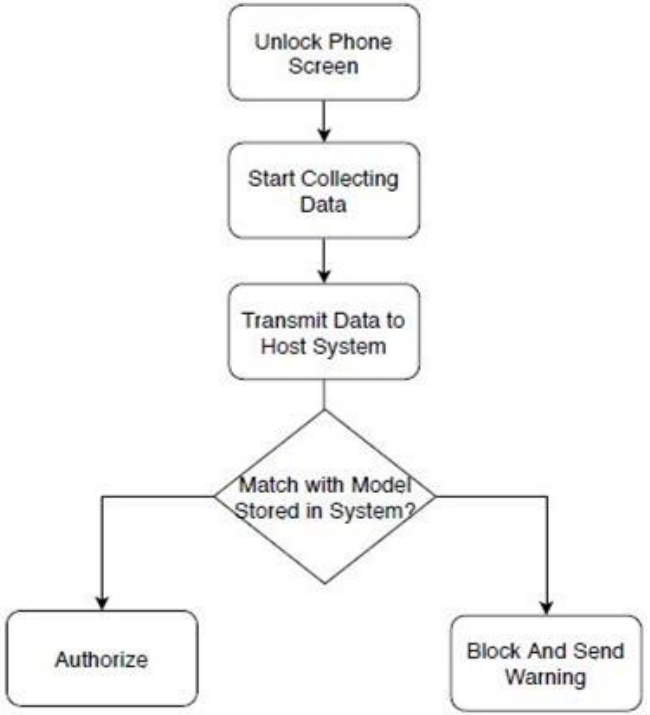
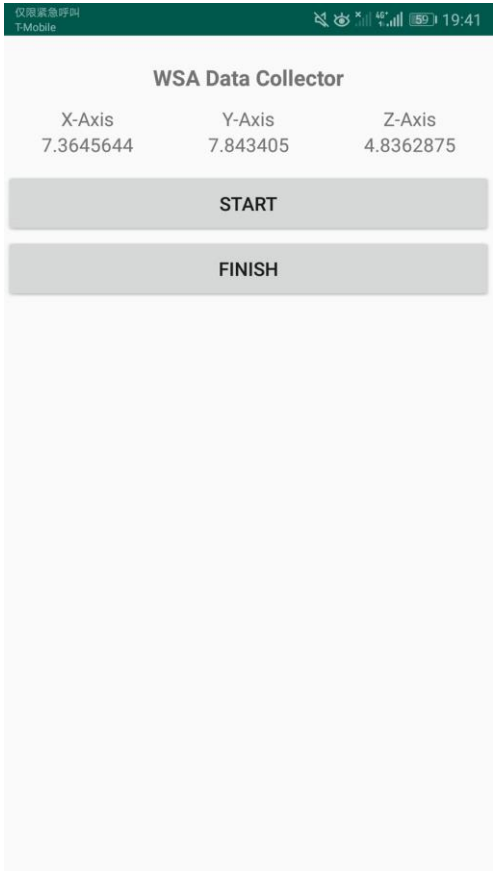



Figure 2. Authorization Process in Real Life

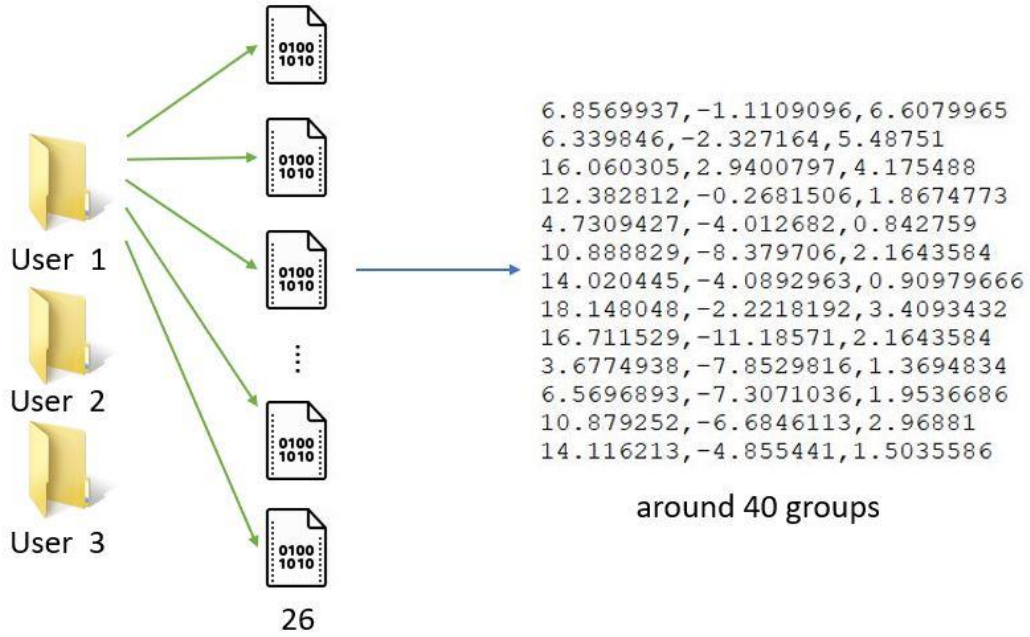
# Approach - Data Collecting

- Designed an android App using Android Studio to collect linear acceleration from x, y, and z three dimensions.
- Save the output to the external storage on the phone.
- Transmit data from phone to computer for future processing.



# Approach - Data Collecting (cont'd)

- Data collected from object walking within 10 meters.




Total: 3000 groups of data

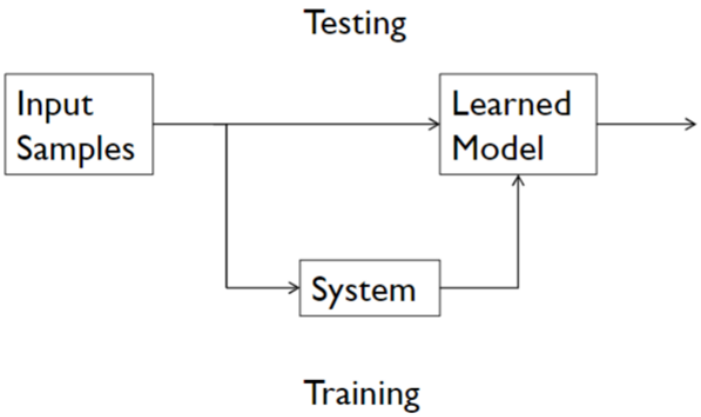
# Approach - Data Processing

- Wrote Python programs to integrate data and add labels to the data for future training and testing
- Labelling for linear regression example:

```
6.8569937,-1.1109096,6.6079965
6.339846,-2.327164,5.48751
16.060305,2.9400797,4.175488
12.382812,-0.2681506,1.8674773
4.7309427,-4.012682,0.842759
10.888829,-8.379706,2.1643584
14.020445,-4.0892963,0.90979666
18.148048,-2.2218192,3.4093432
16.711529,-11.18571,2.1643584
3.6774938,-7.8529816,1.3694834
6.5696893,-7.3071036,1.9536686
10.879252,-6.6846113,2.96881
14.116213,-4.855441,1.5035586
```

```
6.8569937,-1.1109096,6.6079965,0
6.339846,-2.327164,5.48751,0
16.060305,2.9400797,4.175488,0
12.382812,-0.2681506,1.8674773,0
4.7309427,-4.012682,0.842759,0
10.888829,-8.379706,2.1643584,0
14.020445,-4.0892963,0.90979666,0
18.148048,-2.2218192,3.4093432,0
16.711529,-11.18571,2.1643584,0
3.6774938,-7.8529816,1.3694834,0
6.5696893,-7.3071036,1.9536686,0
10.879252,-6.6846113,2.96881,0
14.116213,-4.855441,1.5035586,0
```

# Approach - Related Works




- **Supervised learning** ( $\{x_n \in R^d, y_n \in R\}_{n=1}^N$ )
  - Prediction
  - Classification (discrete labels), Regression (real values)
- **Unsupervised learning** ( $\{x_n \in R^d\}_{n=1}^N$ )
  - Clustering
  - Probability distribution estimation
  - Finding association (in features)
  - Dimension reduction
- **Semi-supervised learning**
- **Reinforcement learning**
  - Decision making (robot, chess machine)

# Approach - Linear Regression

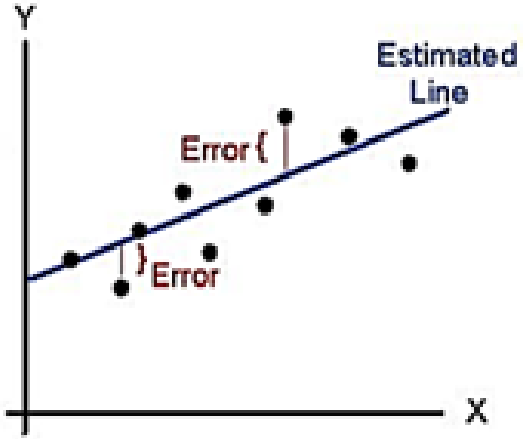
Estimated (or predicted) Y value for observation i

Estimate of the regression intercept

Estimate of the regression slope

Value of X for observation i

$$\hat{Y}_i = b_0 + b_1 X_i$$



# Approach - Linear Regression (Multiple Variables)

```
x =  [[  6.8569937   -1.1109096    6.6079965 ]
 [  6.339846    -2.327164     5.48751   ]
 [ 16.060305     2.9400797    4.175488  ]
 [ 12.382812    -0.2681506    1.8674773 ]
 [  4.7309427   -4.012682     0.842759  ]
 [ 10.888829    -8.379706     2.1643584 ]
 [ 14.020445    -4.0892963    0.90979666]
 [ 18.148048    -2.2218192    3.4093432 ]
 [ 16.711529   -11.18571      2.1643584 ]
 [  3.6774938   -7.8529816    1.3694834 ]]
```

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n$$

$$h_\theta(x) = \theta^T X$$

# Approach - Linear Regression (Multiple Variables)

**Cost Function (MSE):**

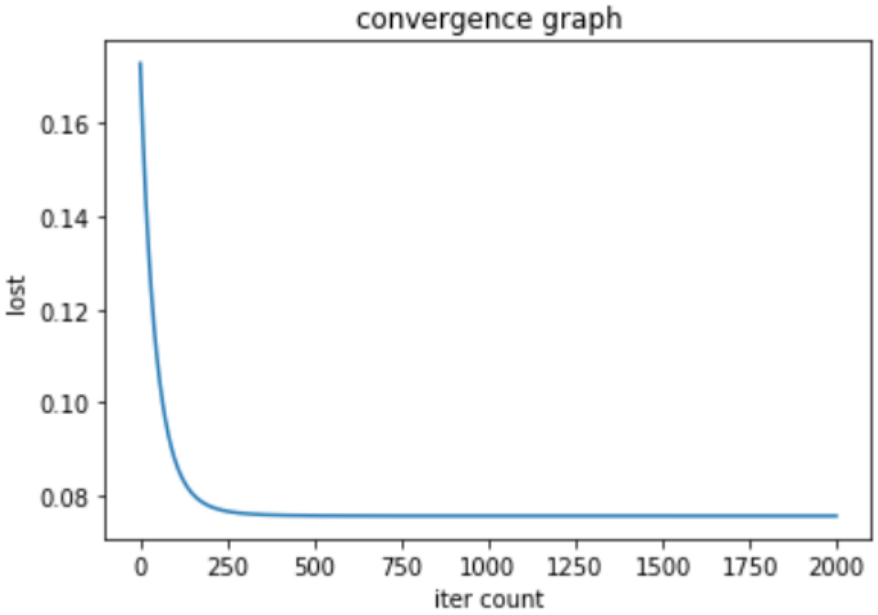
$$J(\theta_0, \theta_1 \ldots \theta_n) = \frac{1}{2m} \sum_{i=1}^{m} \left( h_\theta \left( x^{(i)} \right) - y^{(i)} \right)^2$$

**Gradient Descent:**

$$\theta_j := \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} ((h_\theta(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)})$$

$\alpha$: Learning rate

# Approach - Linear Regression (Multiple Variables)



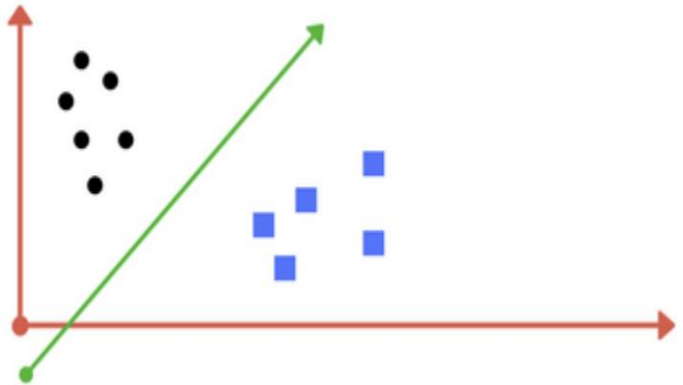

As the count of iterations increases, the lost decreases.

The accuracy is about 75%.

# Approach - SVM (Cont'd)

- Support Vector Machine (SVM)

SVM is a supervised learning algorithm. Given labeled training data, it would establish a learned model and discriminatively divide data into two parts.



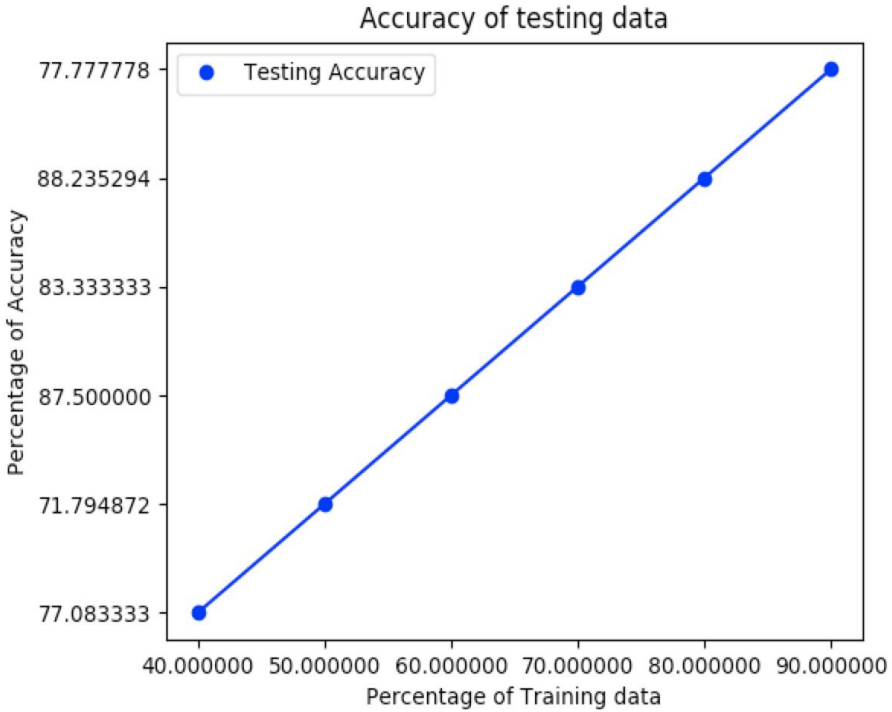
- Realization

With SVM, the accuracy of authentication has been calculated and plotted.

# Approach - SVM (Cont'd)

- Data processing
  - In the form of $[x_1, y_1, z_1,\ x_2, y_2, z_2,\ \ldots\ , x_n, y_n, z_n,\ step\_count]$ for each line.
  - Attach label 1 to people with authorization and label 0 to others.
  - Create four file.txt separately for label 0, label 1, data of label 0 and data of label 1.
- Training and Testing
  - Set parameters for SVM and run with input.
  - Divide each file for training and testing with certain percentage. Then mix all the label files and mix all the data files.
  - Change the percentage of training data and testing data to obtain accuracy.

# Approach - SVM (Cont'd)




Result:

Improve the accuracy of Linear Regression Algorithm, with 88.2% highest.

# Future Works

- Improve accuracy
  - Expand the amount of data for training
  - Try other algorithms
- Transfer data
  - Design an APP which can realize that transferring data from android to system automatically once finishing recording

# Questions?

# Thank you!