Gearheads

Gesture recognition of doorknobs by a Smartwatch and its application towards smart home

INDEX

TEAM GEARHEADS

01

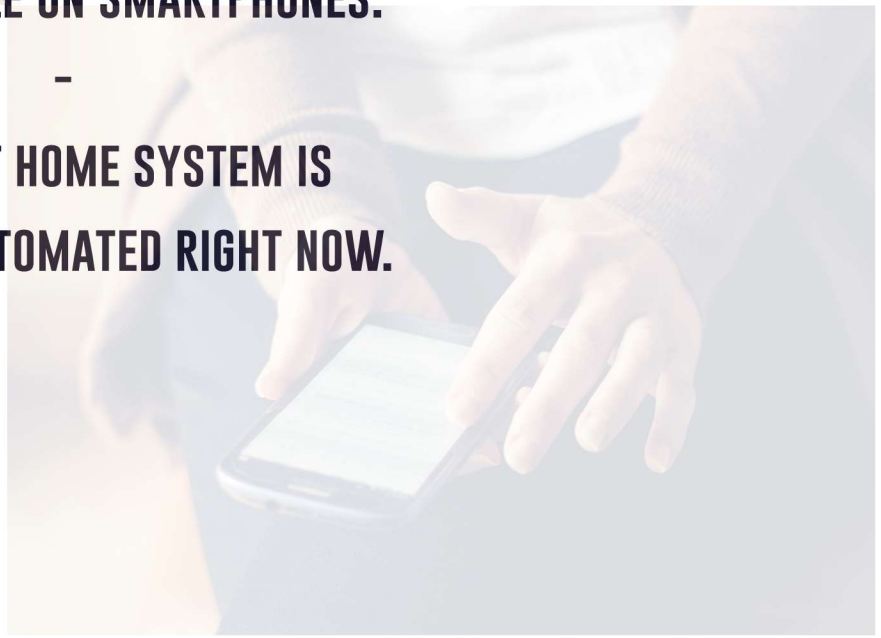
PROBLEM

The Problem of Modern SmartHome System

Most of the functions of smart watches
are available on smartphones.

-

The smart home system is
not fully automated right now.



01

PROBLEM

The Problem of Modern SmartHome System

Limitation of Gears

The number of home appliances that are directly connected

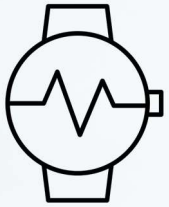
-

Limitation of Smartphone

Limitations to recognizing user behavior

I D E A

So we do



Motion
Recognize

Control
On/Off

Without lifting a finger
Triggered by only your motion

Let be Free

03

FEASIBILTY

BE AVAILABLE FOR APPLICATION

LIGHT

AIR CONDITIONER

MUSIC

IT CAN BE APPLIED IN ANY OTHER APPLIANCE

NOT ONLY LIGHT BULBS

04

MACHINE LEARNING

HOW TO RECOGNIZE MOTION

SENSOR DATA ACCESS

no	x_mean	x_stddev	y_mean	y_stddev	z_mean	z_stddev	x_diff	y_diff	z_diff				
1	-4.83073	0.58111	-5.61051	2.60353	5.75238	1.56259	3.72802	8.2361	5.62792				
2	-4.65784	0.855771	-5.41122	2.56741	6.09231	1.55135	4.72583	10.7198	6.25962	3.00455	11.5035	10.7502	1
3	-5.42003	0.725816	-4.97062	2.10173	6.07477	1.20334	3.75195	7.7671	4.85743	2.35248	11.0454	11.4948	1
4	-5.46667	0.842872	-5.57786	2.02088	5.44852	1.39858	3.42892	7.43689	5.22353	2.3471	11.0264	10.9152	1
5	-5.83317	0.67905	-4.45488	2.30496	5.9442	1.37338	2.80199	8.48256	5.202	2.92706	10.3991	11.7774	1
6	-5.62692	0.759104	-5.35244	2.31011	5.41156	1.51391	2.90728	9.04248	5.22353	2.62611	10.764	11.0385	1
7	-5.79105	0.810506	-4.75942	2.35115	5.75356	1.35029	2.89053	7.17607	4.58943	2.88636	10.513	11.5446	1
8	-5.80271	0.93745	-4.89206	2.15215	5.72938	1.1824	3.19442	6.21416	4.87897	2.7475	10.6214	11.5321	1
9	-5.90264	0.760777	-4.99385	2.04991	5.63987	1.09106	3.97687	7.85803	4.05823	2.24091	10.6337	11.5425	1
10	-5.84036	0.860814	-5.14108	2.70795	5.23732	1.29696	3.30927	7.78624	4.29272	3.00491	10.3784	11.0777	1
11	-5.58292	0.806653	-5.15403	2.66049	5.51262	1.33364	3.09392	8.40838	4.54397				
12	-5.64827	0.744618	-5.7377	2.2806	4.97102	1.72674	5.63031	8.83909	6.6999				
13	-5.48668	0.982986	-5.10589	2.73155	5.60389	1.28203	3.18245	7.85564	4.46979				
14	-5.63705	0.781236	-5.77111	2.41639	4.81825	1.84567	3.1657	9.0664	6.43191				
15	-5.69071	0.926935	-5.3493	2.69162	5.06645	1.80043	3.59402	9.40379	6.62333	2.99035	10.4157	10.7572	1
16	-5.89022	0.869711	-4.84841	3.09238	5.17177	1.5334	5.97249	10.6552	5.98445	3.297	10.0202	11.062	1
17	-5.82232	0.724589	-5.44775	2.48496	5.03683	1.53397	2.84507	9.2339	5.11585	2.61221	10.4846	10.8592	1
18	-5.69429	0.83076	-5.11165	2.96764	5.143	1.79492	3.10588	9.41576	5.69492	3.26112	10.2547	10.8373	1

DATA COLLECTING

MACHINE LEARNING

```
In [29]: %matplotlib inline
         import pandas as pd
         import numpy as np
         import scipy.stats as stats
         import matplotlib.pyplot as plt
         import sklearn
         from sklearn.cross_validation import train_test_split
         from sklearn.metrics import classification_report
                                      nNB
                                      e

                                      ', header=None, skiprows=1)

In [31]: train, test = train_test_split(dataset, train_size=0.7, random_state=0)
         X_train, X_test = train.iloc[:, 1:13].values, test.iloc[:, 1:13].values
         Y_train, Y_test = train.iloc[:, 13].values, test.iloc[:, 13].values

In [32]: print (Y_train.shape, X_train.shape)
         print (Y_test.shape, X_test.shape)

         (152,) (152, 12)
         (66,) (66, 12)

In [33]: model = GaussianNB()
         model.fit(X_train, Y_train)

Out[33]: GaussianNB(priors=None)

In [34]: predicted= model.predict(X_test)
         accuracy_score(predicted, Y_test)

Out[34]: 1.0
```

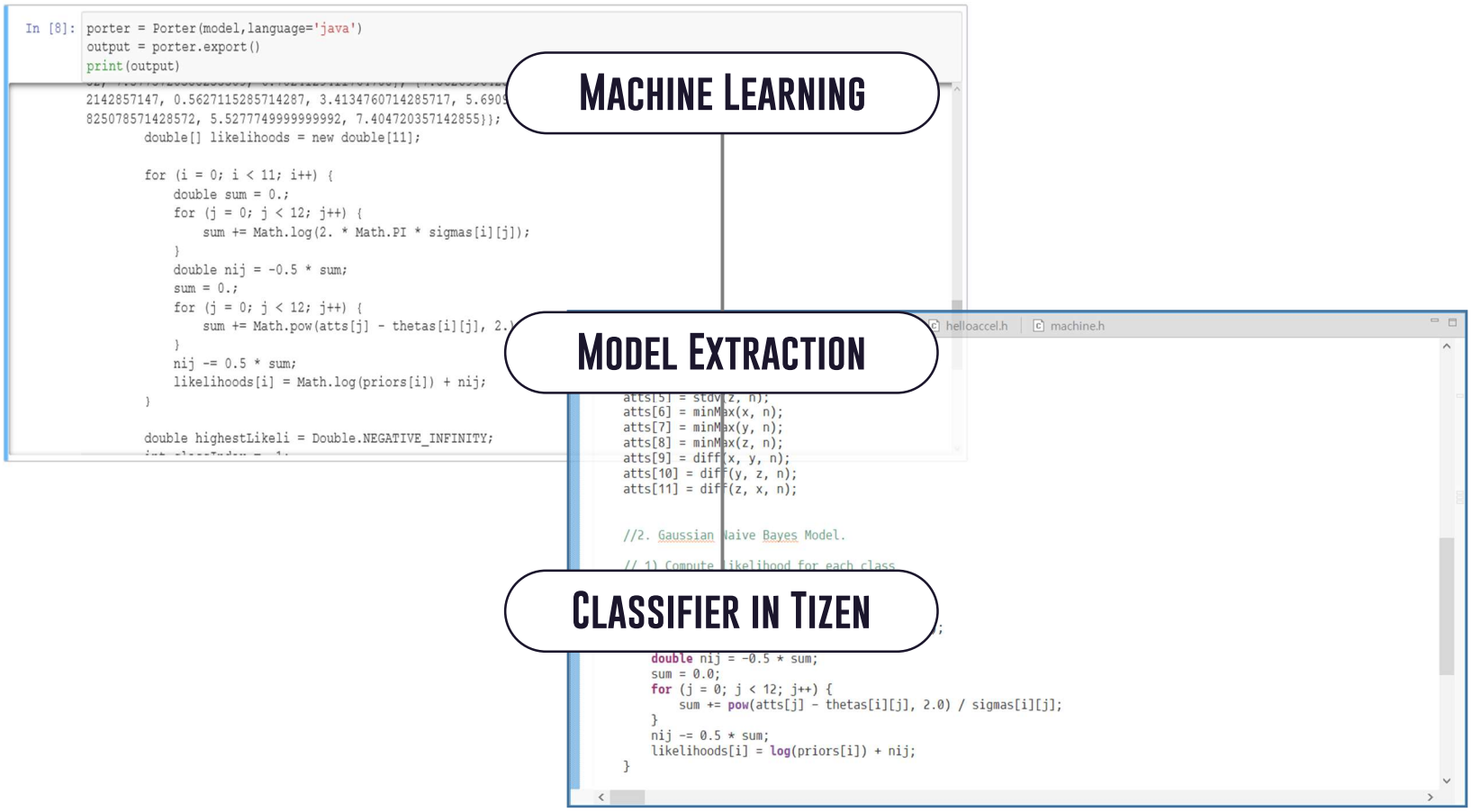
TEAM GEAR HEADS

G

04

MACHINE LEARNING

HOW TO RECOGNIZE MOTION



TEAM GEAR HEADS

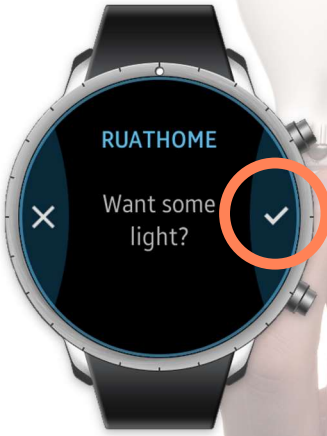
05

IMPLEMENT

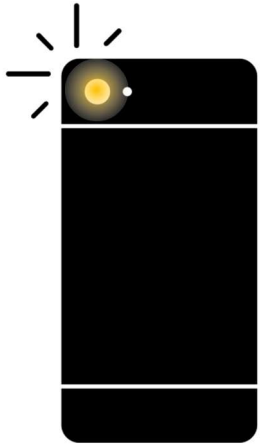
IN SMARTWATCH AND SMARTPHONE

WAIT STATE

MOTION RECOGNITION



AMBIGUOUS SITUATION



LIGHT ON

TEAM GEAR HEADS

G

06

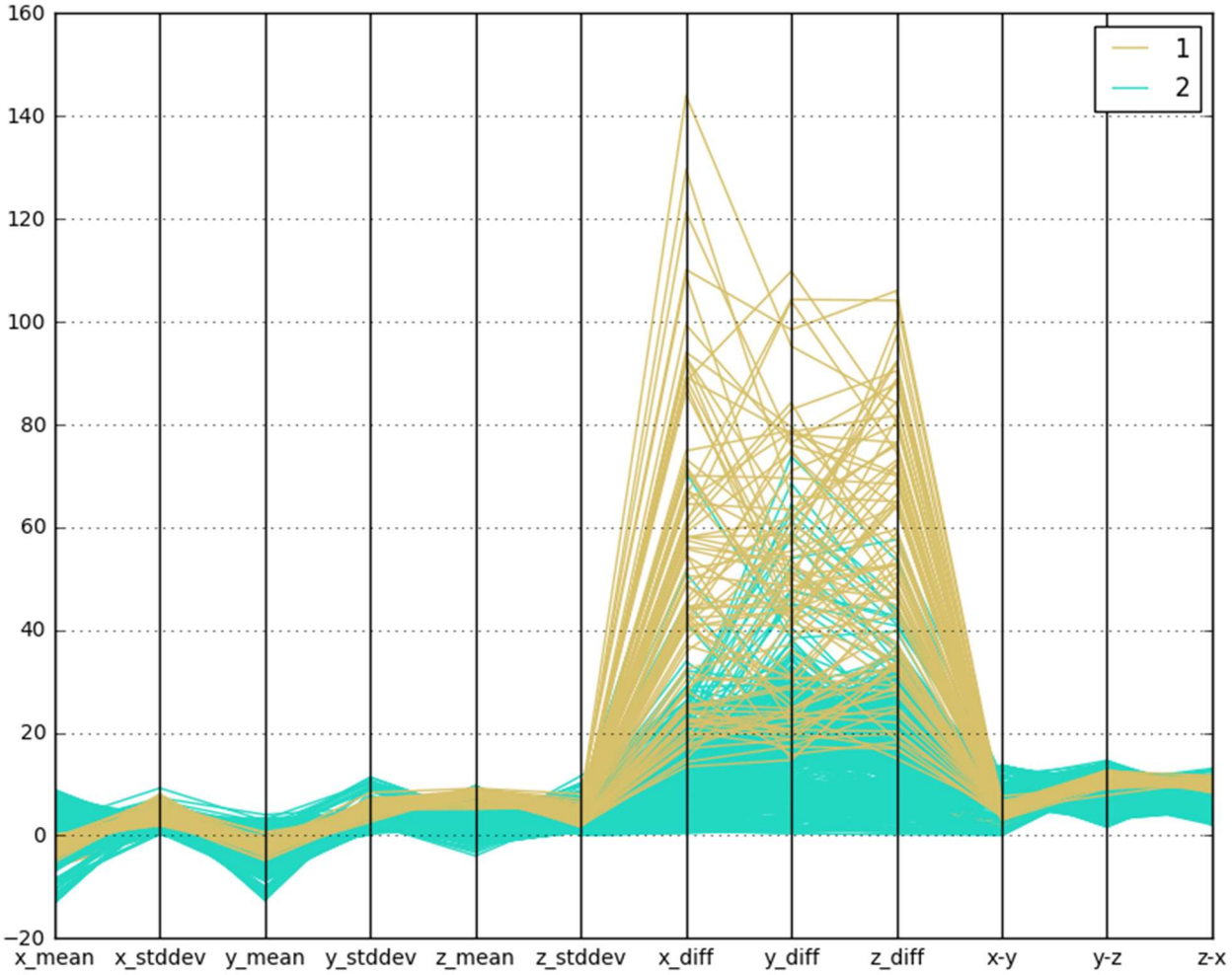
Demo Video

See how it works

07

R E S U L T

Correctness and Possibility

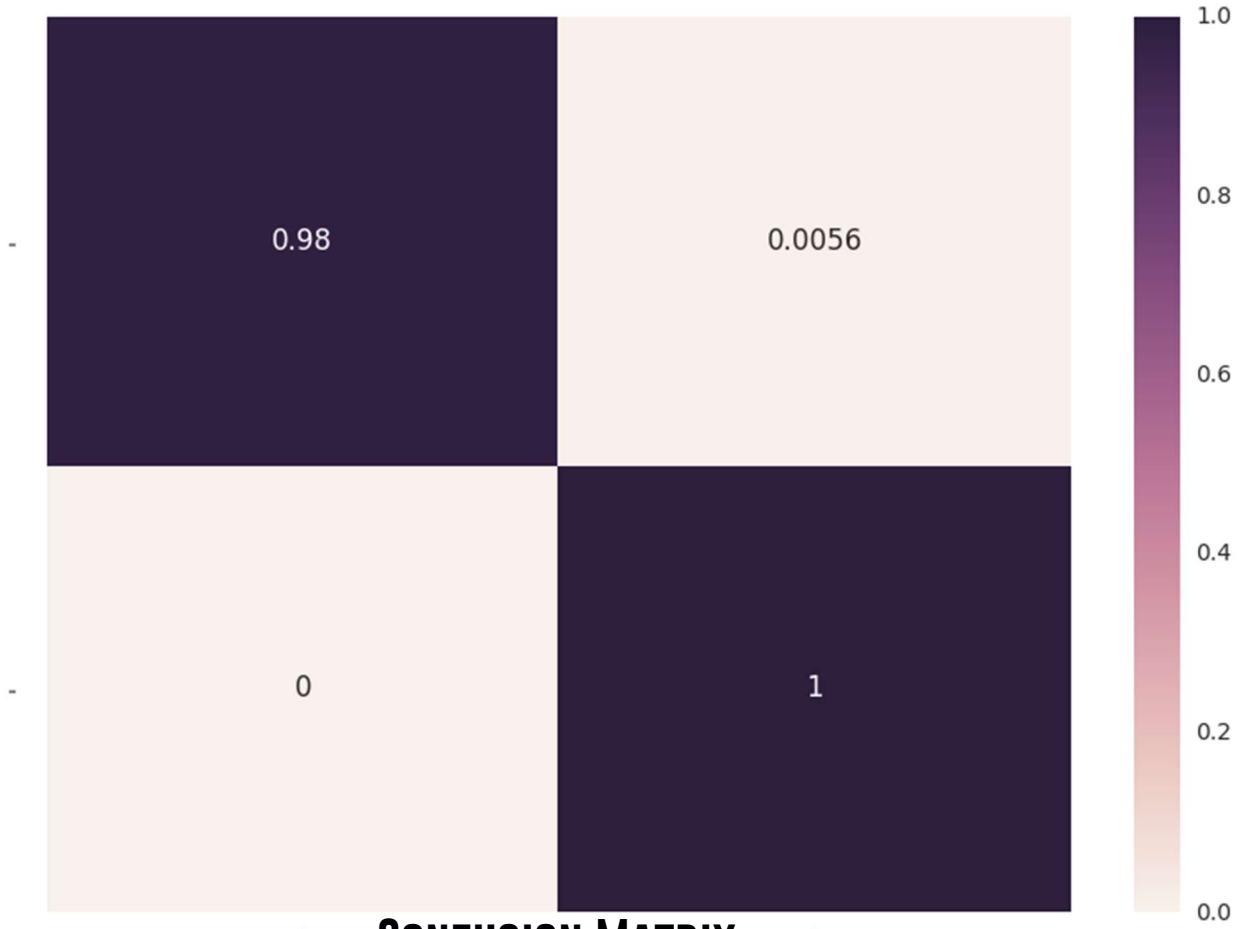


Parallel Coordinates Plot
(Yellow is open motion, Mint is others)

07

RESULT

Correctness and Possibility



Confusion Matrix

(Tp is 0.98, Tn is 1.0)

TEAM GEAR HEADS

G

07

RESULT

Correctness and Possibility

Freedom from Inconvenience

-

Fully automated smarthome

-

Not anymore smartwatch as white elephant

Thank You

Team Gearheads