Determining whether Two Devices are on the Same Person using Accelerometers

Final Presentation

Zachary Stence and Torry Johnson

Purpose and Objective

- Producers of wearable devices skimp on security and authentication to cut costs
- Increasing prevalence of wearable devices in our society, has potential to cause great harm
- Reliably determine whether two readings came from the same person or different people
- Could later be used as a framework for providing authentication techniques
- Accelerometers are very cheap sensors already present in most devices today

Features

Mean

Standard deviation

Variance

Power

Interquartile Range

Energy

Mean absolute deviation

Models Used

k-Nearest Neighbors

Random Forest

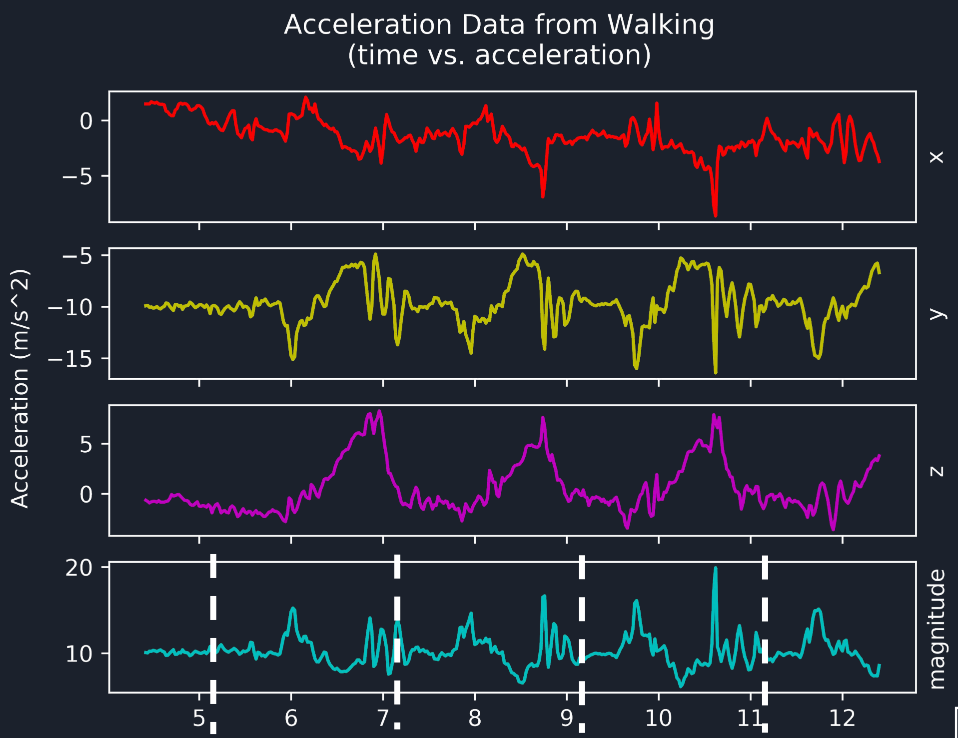
Support Vector Machine

Gradient Boosting Classifier

Decision Tree

Neural Network

Processing/Labelling Data



1. Split a magnitude stream into windows of length *w*.
2. Extract a feature vector from each window.
3. Combine the feature vectors to form a feature matrix, *A*.
4. Repeat steps 1-3 with another magnitude stream to obtain a feature matrix, *B*.
5. Calculate the normalized coherence of each feature of *A* and *B* across a window *c* to obtain a coherence matrix.
6. Split this coherence matrix into rows and label them corresponding to where the original magnitude streams came from.

$$\text{magnitude} = \sqrt{x^2 + y^2 + z^2}$$

$F_2 = \{f_1, f_2, f_3, \ldots\}$

$F_1 = \{f_1, f_2, f_3, \ldots\}$ $F_3 = \{f_1, f_2, f_3, \ldots\}$

feature vectors

$$A = \begin{bmatrix} \vdots & \vdots & \vdots & \\ F_1 & F_2 & F_3 & \ldots \\ \vdots & \vdots & \vdots & \end{bmatrix}$$

$$B = \begin{bmatrix} \vdots & \vdots & \vdots & \\ F_1 & F_2 & F_3 & \ldots \\ \vdots & \vdots & \vdots & \end{bmatrix}$$

labelled data

$$\begin{bmatrix} C(f_{A_1}^{1\ldots1+c}, f_{B_1}^{1\ldots1+c}) & \ldots \\ C(f_{A_1}^{2\ldots2+c}, f_{B_1}^{2\ldots2+c}) & \ldots \\ \vdots & \end{bmatrix} \begin{bmatrix} 0 \text{ or } 1 \\ 0 \text{ or } 1 \\ 0 \text{ or } 1 \end{bmatrix}$$

Splitting into Training and Testing Data

Generated datasets

$$\begin{bmatrix} [[C_1, C_2, \ldots, C_7], 0 \text{ or } 1], \\ [[C_1, C_2, \ldots, C_7], 0 \text{ or } 1], \\ \vdots \end{bmatrix}$$

random 80%

$$\begin{bmatrix} [[C_1, C_2, \ldots, C_7], 0 \text{ or } 1], \\ \vdots \end{bmatrix}$$

random 20%

$$\begin{bmatrix} [[C_1, C_2, \ldots, C_7], 0 \text{ or } 1], \\ \vdots \end{bmatrix}$$

X_train

$$\begin{bmatrix} [C_1, C_2, \ldots, C_7], \\ [C_1, C_2, \ldots, C_7], \\ \vdots \end{bmatrix}$$

y_train

$$\begin{bmatrix} 0 \text{ or } 1, \\ 0 \text{ or } 1, \\ \vdots \end{bmatrix}$$

X_test

$$\begin{bmatrix} [C_1, C_2, \ldots, C_7], \\ [C_1, C_2, \ldots, C_7], \\ \vdots \end{bmatrix}$$

y_test

$$\begin{bmatrix} 0 \text{ or } 1, \\ 0 \text{ or } 1, \\ \vdots \end{bmatrix}$$

Results

- We achieved roughly the same results with both sets of data.

- around 72% success when classifying accelerometer streams. These results aren't bad, but could definitely be improved.

UniMiB-SHAR	
Accuracy	0.716
Precision	0.678
F1 Score	0.748
Train Time (s)	0.004
Test Time (s)	0.003

Confusion matrix		Predicted	
		T	F
Actual	T	118	78
	F	27	172

Our data	
Accuracy	0.729
Precision	0.725
F1 Score	0.730
Train Time (s)	0.082
Test Time (s)	0.193

Confusion matrix		Predicted	
		T	F
Actual	T	3564	1365
	F	1297	3598

Conclusion

- Our program can classify two accelerometer streams as same body or different body reasonably accurately
- Parameters play a huge role in the performance of the models (window length, coherence window length, etc)
- Better accuracy can be achieved with other models, but there is a tradeoff with time
- In the future...
  - A more carefully collected dataset with more people
  - Higher quality accelerometers could be used
  - Different features could be used and optimized
- This seems to be a very promising way of authenticating users without requiring new sensors or complicated systems

References

1. Daniela Micucci, Marco Mobilio, Paolo Napoletano. “UniMiB SHAR: A Dataset for Human Activity Recognition Using Acceleration Data from Smartphones.” Applied Sciences, vol. 7, no. 10, 24 Oct. 2017. Applied Sciences, doi:10.3390/app7101101.
2. Cory T Cornelius, David F Kotz. “Recognizing Whether Sensors Are on the Same Body.” Pervasive and Mobile Computing, vol. 8, no. 6, Dec. 2012, pp. 822–836. ScienceDirect, doi:10.1016/j.pmcj.2012.06.005.
3. “Scikit-Learn: Machine Learning in Python.” Scikit-Learn, Oct. 2017, scikit-learn.org/stable/index.html.