Same Body Detection using Accelerometers

Department of Computer Science and Engineering
Undergraduate Researchers: Zachary Stence, Torry Johnson
Mentor: Dr. Bin Wang

NSF WRIGHT STATE UNIVERSITY

This project is sponsored by the NSF Research Experience for Undergraduates Site Award "Cyber Security Research at Wright State University" (CNS 1560315).

Purpose and Objective

Producers of wearable devices often skimp on security and authentication techniques in order to cut costs. With the increasing prevalence of wearable devices in our society, this has potential to cause great harm.

Our goal is to reliably determine whether two readings from accelerometer sensors came from the same person or different people wearing a device. This could later be used as a framework for providing easy authentication techniques using accelerometers, a very cheap sensor already present in most devices today. Machine learning classifiers from Python's scikit-learn package were used [3].

Datasets

Two datasets were used for this project, UniMiB SHAR and a dataset we collected and created ourselves.

- UniMiB SHAR - A Dataset for Human Activity Recognition Using Acceleration Data from Smartphones [1]
  - 30 people performing 17 activities (running, walking, going up/down stairs, falling), 2 trials each
  - Short streams, only a few seconds each trial
  - Mainly used walking and running data for our project
- Our data
  - 2 people, each walking 12 times for about a minute each trial

Features

Mean	Standard deviation	Variance
Power	Interquartile Range	Energy
	Mean absolute deviation	

Same features used in [3]

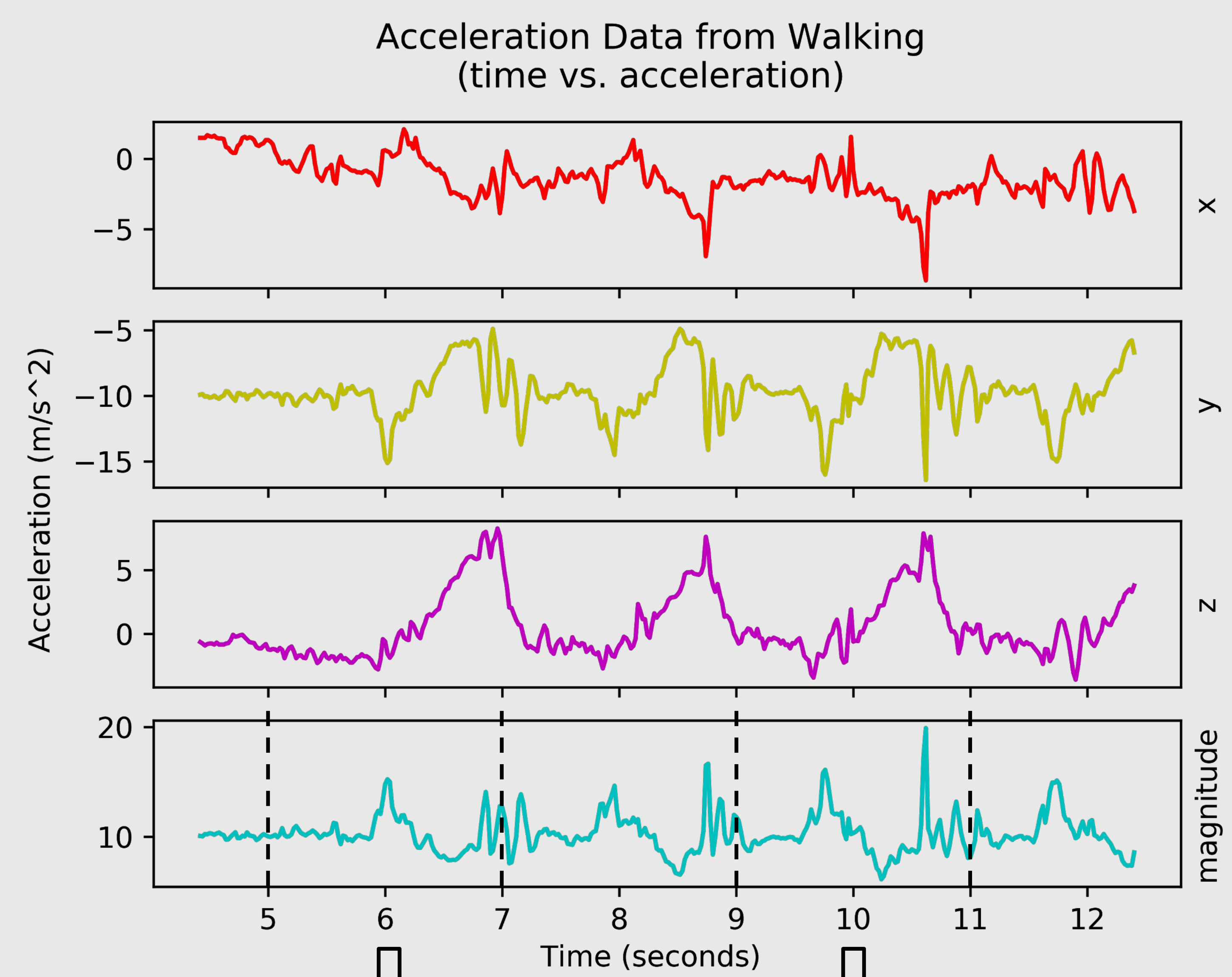
Models Used

k-Nearest Neighbors	Random Forest	Support Vector Machine
Gradient Boosting Classifier	Decision Tree	Neural Network

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.

Labelling and Using Data



$F_1 = \{f_1, f_2, f_3, \ldots\}$ $F_2 = \{f_1, f_2, f_3, \ldots\}$ $F_3 = \{f_1, f_2, f_3, \ldots\}$ $\ldots$

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

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

Figure 1: Example accelerometer streams from the UniMiB SHAR dataset and how they are manipulated into useful mathematical representations for use with machine learning.

Labelling Raw 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.

Testing With Classifiers

```
# randomly split data into training and testing portions (80% split)
X_train, y_train, X_test, y_test = get_train_test(data, 0.8)

# fit a classifier
clf.fit(X_train, y_train)

# get classifier's predictions
y_pred = clf.predict(X_test)

# evaluate the confusion matrix
cm = confusion_matrix(y_test, y_pred)
```

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

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	118	78
	F	27	172

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

All of the source code written for the project can be found at `github.com/zachstence/SameBodyDetection`