NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

AUTHENTICATION OF SMARTPHONE USER USING RSSI GEOLOCATION

by

Vincent K. Nguyen

March 2014

Thesis Advisor:	Craig Martell
Second Reader:	Mark Gondree

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** March 2014	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE
AUTHENTICATION OF SMARTPHONE USER USING RSSI GEOLOCATION

5. FUNDING NUMBERS

6. AUTHOR(S) Vincent K. Nguyen

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943–5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____NPS.2013.0058-IR-EP7-A____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

This thesis attempts to authenticate a smartphone user by pattern of life based on a smartphone user's geolocation throughout the course of a day. Current smartphone technology uses the global positioning system (GPS) as the primary source for geolocation because of its accuracy. However, services such as Google Location Service and Skyhook use Receive Signal Strength Indicator (RSSI)-based geolocation in GPS-degraded environments, such as inside a building. By using a smartphone's Wi-Fi application programming interface, a smartphone would detect all wireless access points' Wi-Fi signals and associated signal strength over a discrete time interval. A hidden Markov model is used to model various smartphone users and used as an authentication method. The resulting f-score from the experiments ranged between 0.76 and 0.80, which is well above the 0.20 baseline. It is feasible to use RSSI-based geolocation as an element in combination with other methods to continuously authenticate a smartphone user. For an acceptable authentication method, the evaluation criteria must be as close to 1.0 as possible. Future research could combine authentication from RSSI-based geolocation with gait and keystroke analysis to improve results by leveraging other sensors on a smartphone.

14. SUBJECT TERMS Machine Learning, Hidden Markov Model, Mobile Device, Mobile Phone, Smartphone

15. NUMBER OF PAGES
81

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540–01–280–5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

AUTHENTICATION OF SMARTPHONE USER USING RSSI GEOLOCATION

Vincent K. Nguyen
Lieutenant Commander, United States Navy
B.S., University of Texas at Austin, 1999

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
March 2014

Author: Vincent K. Nguyen

Approved by: Craig Martell
Thesis Advisor

Mark Gondree
Second Reader

Peter Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis attempts to authenticate a smartphone user by pattern of life based on a smartphone user's geolocation throughout the course of a day. Current smartphone technology uses the global positioning system (GPS) as the primary source for geolocation because of its accuracy. However, services such as Google Location Service and Skyhook use Receive Signal Strength Indicator (RSSI)-based geolocation in GPS-degraded environments, such as inside a building. By using a smartphone's Wi-Fi application programming interface, a smartphone would detect all wireless access points' Wi-Fi signals and associated signal strength over a discrete time interval. A hidden Markov model is used to model various smartphone users and used as an authentication method. The resulting f-score from the experiments ranged between 0.76 and 0.80, which is well above the 0.20 baseline. It is feasible to use RSSI-based geolocation as an element in combination with other methods to continuously authenticate a smartphone user. For an acceptable authentication method, the evaluation criteria must be as close to 1.0 as possible. Future research could combine authentication from RSSI-based geolocation with gait and keystroke analysis to improve results by leveraging other sensors on a smartphone.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION 1
- A. MOTIVATION 2
- B. RESEARCH QUESTION 2
- C. SIGNIFICANT FINDINGS 2
- D. THESIS STRUCTURE 3

II. PRIOR AND RELATED WORK 5
- A. RELATED RESEARCH 5
- B. AUTHENTICATION 6
- C. GEOLOCATION 7
  - 1. GPS 7
  - 2. Geometric Triangulation of Cell Towers 7
  - 3. RSSI-based 8
- D. ANDROID WI-FI MANAGER APPLICATION PROGRAMMING INTERFACE (API) 8
- E. HIDDEN MARKOV MODEL 9
  - 1. Definition of a HMM 10
  - 2. Three Fundamental Problems for HMM 11
- F. EVALUATION CRITERIA 11
  - 1. Confusion Matrix 12
  - 2. Precision 12
  - 3. Recall 12
  - 4. F-score 13

III. EXPERIMENTAL DESIGN 15
- A. HARDWARE AND SOFTWARE 15
- B. RESEARCH SUBJECTS 15
- C. LOCATION OF EXPERIMENT 16
- D. DATA COLLECTION PARAMETERS 17
- E. SPARSE MATRIX 18
- F. SPLITTING MATRIX INTO TRAINING AND TEST DATASET 19
- G. CLASSIFIER 19

IV. RESULTS AND ANALYSIS 21
- A. INITIAL PARAMETERS 21
- B. BINARY 22
- C. LOGARITHMIC VALUE OF RSSI 23
- D. VARYING THE WINDOW SIZE 24
- E. CHANGING PROPORTION OF TRAINING VERSUS TESTING DATA 25
- F. SUMMARY OF EXPERIMENTS 26

V. CONCLUSION AND FUTURE WORK 27
- A. SUMMARY 27

B. FUTURE WORK 27
C. CLOSING REMARKS..28

APPENDIX. CONFUSION MATRICES 29
A. CONFUSION MATRIX FOR INITIAL PARAMETERS.........................29
B. CONFUSION MATRIX FOR BINARY...32
C. CONFUSION MATRIX FOR LOG 3 35
D. CONFUSION MATRIX FOR LOG 5 38
E. CONFUSION MATRIX FOR LOG 7 42
F. CONFUSION MATRIX FOR WINDOW SIZE 10....................................45
G. CONFUSION MATRIX FOR WINDOW SIZE 15....................................48
H. CONFUSION MATRIX FOR WINDOW SIZE 20....................................52
I. CONFUSION MATRIX FOR 50% TRAINING, 50% TEST...................55

LIST OF REFERENCES 59

INITIAL DISTRIBUTION LIST 61

LIST OF FIGURES

Figure 1. (A) Times and locations of calls made or received and nearest antenna. (B) Approximation of antennas reception areas. (C) Lower resolution through spatial and temporal aggregation (from [7]). 6
Figure 2. Screen shots using Wi-Fi Analyzer app...9
Figure 3. Map of Naval Postgraduate School, Monterey, California (from [16])...........17
Figure 4. Screen shot of Funf Journal 18
Figure 5. Sparse matrix 19
Figure 6. Summary of average f-scores..26

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1. Example of confusion matrix 12
Table 2. Research subject's PIN, major, and number of data points 16
Table 3. Confusion Matrix for initial parameters run 1 21
Table 4. Precision, Recall, and F-score from initial parameters 22
Table 5. Precision, Recall, and F-score for binary 22
Table 6. Precision, Recall, and F-score for Log 3 23
Table 7. Precision, Recall, and F-score for Log 5 23
Table 8. Precision, Recall, and F-score for Log 7 24
Table 9. Precision, Recall, and F-score for Window Size 10 24
Table 10. Precision, Recall, and F-score for Window Size 15 25
Table 11. Precision, Recall, and F-score for Window Size 20 25
Table 12. Precision, Recall, and F-score for 50% Training, 50% Test 26
Table 13. Confusion Matrix for initial parameters run 2 29
Table 14. Confusion Matrix for initial parameters run 3 29
Table 15. Confusion Matrix for initial parameters run 4 29
Table 16. Confusion Matrix for initial parameters run 5 30
Table 17. Confusion Matrix for initial parameters run 6 30
Table 18. Confusion Matrix for initial parameters run 7 30
Table 19. Confusion Matrix for initial parameters run 8 31
Table 20. Confusion Matrix for initial parameters run 9 31
Table 21. Confusion Matrix for initial parameters run 10 31
Table 22. Confusion Matrix for binary run 1 32
Table 23. Confusion Matrix for binary run 2 32
Table 24. Confusion Matrix for binary run 3 32
Table 25. Confusion Matrix for binary run 4 33
Table 26. Confusion Matrix for binary run 5 33
Table 27. Confusion Matrix for binary run 6 33
Table 28. Confusion Matrix for binary run 7 34
Table 29. Confusion Matrix for binary run 8 34
Table 30. Confusion Matrix for binary run 9 34
Table 31. Confusion Matrix for binary run 10 35
Table 32. Confusion Matrix for log 3 run 1 35
Table 33. Confusion Matrix for log 3 run 2 35
Table 34. Confusion Mat for log 3 run 3 36
Table 35. Confusion Matrix for log 3 run 4 36
Table 36. Confusion Matrix for log 3 run 5 36
Table 37. Confusion Matrix for log 3 run 6 37
Table 38. Confusion Matrix for log 3 run 7 37
Table 39. Confusion Matrix for log 3 run 8 37
Table 40. Confusion Matrix for log 3 run 9 38
Table 41. Confusion Matrix for log 3 run 10 38
Table 42. Confusion Matrix for log 5 run 1 38

Table 43. Confusion Matrix for log 5 run 2 39
Table 44. Confusion Matrix for log 5 run 3 39
Table 45. Confusion Matrix for log 5 run 4 39
Table 46. Confusion Matrix for log 5 run 5 40
Table 47. Confusion Matrix for log 5 run 6 40
Table 48. Confusion Matrix for log 5 run 7 40
Table 49. Confusion Matrix for log 5 run 8 41
Table 50. Confusion Matrix for log 5 run 9 41
Table 51. Confusion Matrix for log 5 run 10 41
Table 52. Confusion Matrix for log 7 run 1 42
Table 53. Confusion Matrix for log 7 run 2 42
Table 54. Confusion Matrix for log 7 run 3 42
Table 55. Confusion Matrix for log 7 run 4 43
Table 56. Confusion Matrix for log 7 run 5 43
Table 57. Confusion Matrix for log 7 run 6 43
Table 58. Confusion Matrix for log 7 run 7 44
Table 59. Confusion Matrix for log 7 run 8 44
Table 60. Confusion Matrix for log 7 run 9 44
Table 61. Confusion Matrix for log 7 run 10 45
Table 62. Confusion Matrix for window size 10 run 1 45
Table 63. Confusion Matrix for window size 10 run 2 45
Table 64. Confusion Matrix for window size 10 run 3 46
Table 65. Confusion Matrix for window size 10 run 4 46
Table 66. Confusion Matrix for window size 10 run 5 46
Table 67. Confusion Matrix for window size 10 run 6 47
Table 68. Confusion Matrix for window size 10 run 7 47
Table 69. Confusion Matrix for window size 10 run 8 47
Table 70. Confusion Matrix for window size 10 run 9 48
Table 71. Confusion Matrix for window size 10 run 10 48
Table 72. Confusion Matrix for window size 15 run 1 48
Table 73. Confusion Matrix for window size 15 run 2 49
Table 74. Confusion Matrix for window size 15 run 3 49
Table 75. Confusion Matrix for window size 15 run 4 49
Table 76. Confusion Matrix for window size 15 run 5 50
Table 77. Confusion Matrix for window size 15 run 6 50
Table 78. Confusion Matrix for window size 15 run 7 50
Table 79. Confusion Matrix for window size 15 run 8 51
Table 80. Confusion Matrix for window size 15 run 9 51
Table 81. Confusion Matrix for window size 15 run 10 51
Table 82. Confusion Matrix for window size 20 run 1 52
Table 83. Confusion Matrix for window size 20 run 2 52
Table 84. Confusion Matrix for window size 20 run 3 52
Table 85. Confusion Matrix for window size 20 run 4 53
Table 86. Confusion Matrix for window size 20 run 5 53
Table 87. Confusion Matrix for window size 20 run 6 53

Table 88. Confusion Matrix for window size 20 run 7 54
Table 89. Confusion Matrix for window size 20 run 8 54
Table 90. Confusion Matrix for window size 20 run 9 54
Table 91. Confusion Matrix for window size 20 run 10 55
Table 92. Confusion Matrix for 50% training, 50% test run 1 55
Table 93. Confusion Matrix for 50% training, 50% test run 2 55
Table 94. Confusion Matrix for 50% training, 50% test run 3 56
Table 95. Confusion Matrix for 50% training, 50% test run 4 56
Table 96. Confusion Matrix for 50% training, 50% test run 5 56
Table 97. Confusion Matrix for 50% training, 50% test run 6 57
Table 98. Confusion Matrix for 50% training, 50% test run 7 57
Table 99. Confusion Matrix for 50% training, 50% test run 8 57
Table 100. Confusion Matrix for 50% training, 50% test run 9 58
Table 101. Confusion Matrix for 50% training, 50% test run 10 58

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AP	access point
API	application programming interface
CRF	conditional random fields
CSV	comma separated value
FN	false negative
FP	false positive
GPS	global positioning system
HMM	hidden Markov model
MAC	media access control
NPS	Naval Postgraduate School
PIN	personal identification number
RSSI	Received Signal Strength Indicator
SQL	structured query language
SSID	service set identifier
TN	true negative
TP	true positive
VIP	very important person

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

This thesis would not have been possible without the guidance, instruction and patience of Dr. Craig Martell and Dr. Mark Gondree. In addition, Alric Althoff, who interned at NPS for the summer of 2013, did great work in converting the MAC address and RSSI pair into a sparse matrix data structure to be inputted into Python’s HMM library.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Current smartphone technology uses the global positioning system (GPS) as a primary source for geolocation. GPS provides localization accuracy within 7.8 meters with continuous availability with 24 satellites. GPS requires line of sight acquisition of at least 3 satellites in order to calculate a receiver's current location. However, the signals from the GPS satellites could be impeded by inclement weather or obstruction such as buildings or mountains depending on the receiver's antenna gain [1]. The type of GPS receivers in smartphones varies by vendors, which results in satellite acquisition times to vary from seconds to minutes [2].

Services like Skyhook and Google Location Services, which use a form of received signal strength indicator (RSSI)-based geolocation, have gained in popularity due to their accuracy, availability, and speed for indoor geolocation without GPS coverage [3]. RSSI-based geolocation measures signal strengths of wireless access points from various locations to build a database. The location of the smartphone is calculated by first measuring the various signal strengths from surrounding wireless access points then comparing to the entry of the database. RSSI-based geolocation accuracy depends on the number of wireless access points in the database and has been shown to have accuracy within 74 meters [4]. However, Skyhook has over 50 million wireless access points in its database and reports accuracy within 10 to 20 meters [3].

Recent research has used GPS because of its availability and accuracy to link the user's' geolocation with their daily activities. Examples of activities are walking from the parking lot to the office or being at work. In this study, RSSI data will be used because of its ability to provide geolocation indoors. The RSSI data of a user's daily activity from a smartphone will be used to build a profile of the user. A hidden Markov model (HMM) will be used to classify users and ensure he or she is an authorized user of the smartphone.

A. MOTIVATION

The high-level motivation for this research was to perform preliminary experiments on methods to continuously authenticate a very important person (VIP) such as a high-ranking diplomat. Basically, during the course of a VIP's normal hour, day, or week the algorithm analyzes the RSSI from wireless AP and, based on the pattern, verifies the identity of the VIP. Conversely, if a VIP's smartphone was lost or stolen, the algorithm would detect a pattern, which is not normal and would identify the user as someone other than the VIP.

B. RESEARCH QUESTION

This thesis attempts to answer the following questions:

- Is it possible to authenticate a smartphone user by continuous RSSI-based geolocation?
- Can we use a HMM to model a user's geolocation throughout the day? If yes, can we distinguish between various individuals?

C. SIGNIFICANT FINDINGS

The result of this thesis shows the feasibility of continuously authenticating a smartphone user by modeling user-behavior based on RSSI evidence. The precision, recall, and f-score for all the experimental runs were greater than 0.7 using a HMM. Because the machine-learning algorithm must account for temporal movements from one location to another, classifiers that ignore the time domain, like clustering and Bayesian networks, will not work. Since we used a small data set and restricted our test parameters, future work is warranted.

D. THESIS STRUCTURE

This thesis is organized as follows:

- Chapter I cover the motivation, research questions, and significant findings of the research to be conducted.
- Chapter II discusses prior work as it pertains to this research.
- Chapter III describes the experimental design for this research.
- Chapter IV contains the results and analysis of the experiment
- Chapter V contains the summary of the research and recommended future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. PRIOR AND RELATED WORK

In this chapter, we first discuss prior research in the field of geolocation data from a smartphone. Next, we describe the different sources of geolocation. Finally, we discuss machine learning and the evaluation criteria for machine learning.

A. RELATED RESEARCH

Ashbrook and Starner [5] conducted two studies attempting to predict movements of people. The studies used GPS-based geolocations to model human behavior. GPS geolocation data was collected over a 4-month period in Atlanta, Georgia. Because GPS has an accuracy of approximately 15 meters, a person could be in the exact some spot yet log different locations. Ashbrook and Starner used k-means cluster algorithm to normalize the GPS error by associating all latitudes and longitudes within a half-mile radius as a single discrete location. A Markov model was then derived from the time sequenced locations. The Markov model was able to predict the probability where a person is headed based on their current location [5].

Liao et al. [6] used hierarchical conditional random fields (CRF) for GPS-based activity recognition. The study collected GPS geolocation on four users for a one-week period. The GPS locations were clustered using 10-meter segments then correlated to street locations. The bottom layer of the hierarchical CRF contained nodes from the GPS trace. The middle layer contained nodes of inferred activities such as walking, driving, or getting on the bus, while the top layer contained significant places such as home, work, or shopping. Liao et al. used the data from three users to train the data while using the fourth as the test. The study achieved above 90% accuracy for navigation activities and 85% accuracy for significant places [6].

De Montjoye et al. [7] used anonymous cellphone data for one-and-a-half million users over a 15-month period in Western Europe to find unique traces in human mobility. Each time a user made or received a call or text message, the service provider logged the time and all cellphone towers within range. Using the logs, spatial and temporal correlated information could be derived. Figure 1 depicts a sequence of calls made by a user and the area where cellphone towers were in range of the user. The study did not use machine-learning classifiers to find the traces for users. Instead, the study used set theory to extract unique traces from a set of spatial-temporal points in the mobility dataset. A unique trace is a vector of spatial-temporal points, which only appears once in the dataset. The study showed four unique spatial-temporal traces is enough to uniquely identify 95% of users [7].

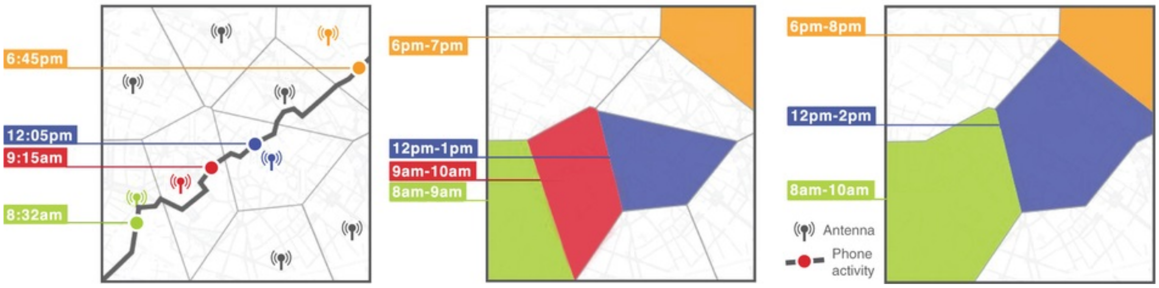


Figure 1. (A) Times and locations of calls made or received and nearest antenna. (B) Approximation of antennas reception areas. (C) Lower resolution through spatial and temporal aggregation (from [7]).

Alvarez-Alvarez et al. [8] correlated Wi-Fi position and body posture to human activity. In their experiment, they used Wi-Fi position information from four access points in a 440-square meter test environment syncing sampling rate with an accelerometer. A fuzzy rule-based classifier used the Wi-Fi geolocation to label locations such as an office, break room, or passageway. A fuzzy finite state machine used the accelerometer data to give relative posture of the person such as seated, standing, or walking. A second fuzzy finite state machine fused the relative location with relative posture to give human activity. Examples of human activities inferred in the experiment were sitting at desk, walking to the break room, or having a meeting in a co-workers office [8].

B. AUTHENTICATION

Authentication is a systematic method of verifying a set of credentials to validate an authorized user. In computer security, three general factors are used for authentication: authentication by knowledge, authentication by ownership, and authentication by biometrics. Authentication by knowledge is something a person knows, such as a password, personal identification number (PIN), or a combination lock. Examples of authentication of ownership are keys, access cards, or badges, which an individual would possess. Authentications by biometrics target physical attributes like fingerprints, iris scan, or palm reader [9].

This thesis examines the possibility of authentication by behavior using the sensors in modern smartphones. Examples of this type of authentication are gait analysis, keystroke analysis, and pattern of life. Gait analysis is studying the uniqueness of a person's motion. Keystroke analysis studies the time interval between various keys while a person types. This research will focus on pattern of life, which is a person's movement from various locations throughout a normal day.

C. GEOLOCATION

Geolocation is the process of locating the geographic location of an object, such as a smartphone or handheld GPS receiver, using electronic means. Geolocation uses positioning system such as GPS or RSSI [2].

1. GPS

GPS provides localization accuracy within 7.8 meters with continuous availability with 24 satellites. GPS requires line of sight acquisition of at least 3 satellites in order to calculate a receiver's current location. However, the signals from the GPS satellites could be impeded by inclement weather or obstruction, such as buildings or mountains, depending on the receiver's antenna gain [1]. The type of GPS receiver in smartphones varies by vendor, which results in a range of satellite acquisition times varying from seconds to minutes.

2. Geometric Triangulation of Cell Towers

Cell towers are another source of geolocation when GPS is not available. When a mobile phone user makes or receives a call, the mobile phone logs the time and cellular identification of cell towers in range. The estimated distance is calculated from the ping time between cell tower and mobile phone. Using estimated distance from multiple cell towers, a geometric triangulation calculates the approximate geolocation within two kilometers [10].

3. RSSI-based

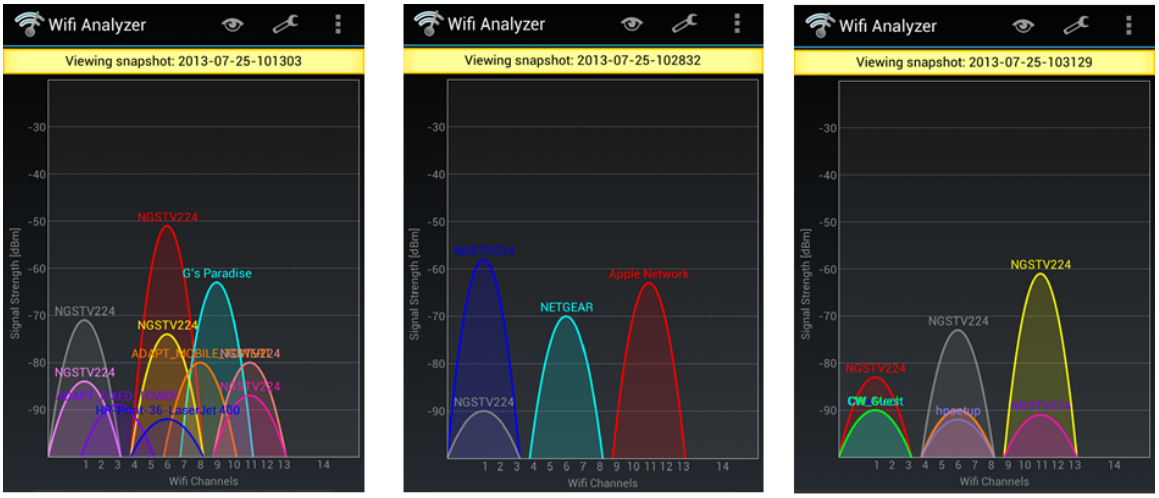
RSSI-based geolocation is often used in an indoor environment when both GPS and cell tower signals are blocked. RSSI-based geolocation measures signal strengths of wireless access points from various locations to build a database. Unlike cell tower triangulation where the distance from cell tower to mobile phone is computed, the distance from the Wi-Fi AP is not calculated from the RSSI. The RSSI is dependent on several factors to include antenna gain, atmospheric, output power, and interference. The location of the smartphone is calculated by first measuring the signal strengths from surrounding Wi-Fi AP then comparing the values to a known database. RSSI-based geolocation accuracy depends on the number of wireless access points in the database, and has been shown to have accuracy within 74 meters [4]. However, Skyhook has over 50 million wireless access points in its database and reports accuracy within 10 to 20 meters [3].

D. ANDROID WI-FI MANAGER APPLICATION PROGRAMMING INTERFACE (API)

The Android Wi-Fi Manager API [11] manages all aspects of Wi-Fi connectivity within an Android device. A smartphone user uses the Wi-Fi manager to scan for available Wi-Fi networks and the signal strength associated with each network. Once a user selects a Wi-Fi network to connect, the Wi-Fi manager initiates the require authentication handshake. The following information is received from all Wi-Fi access points within range of the mobile device:

- AP media access control (MAC) address
- Service set identifier (SSID)
- Frequency
- Channel
- RSSI
- Timestamp

Several apps are available for both Android and IOS devices. Screen shots from Wi-Fi analyzer developed by Farproc [12] are shown in Figure 2. Wi-Fi analyzer is a free app download from the Google store. The screen shots were taken from Glasgow East basement, Glasgow East third floor passageway, and the Del Monte Café, all located on the NPS campus. The screen shots show each location has a distinct fingerprint of Wi-Fi AP in relation to the Wi-Fi AP's detected and their associated RSSI even of those in the same building. This distinction is used in this thesis to model a smartphone user's pattern of life.



GE Basement

GE 3rd Floor

Del Monte Cafe

Figure 2. Screen shots using Wi-Fi Analyzer app

E. HIDDEN MARKOV MODEL

Machine learning is the process of making predictions about an unknown data set based on properties learned from a known data set used to train the system. Machine learning is sometimes incorrectly confused with data mining, which is the process of discovering unknown properties in a data set. The premise of machine learning is to take a data set with known labels and build a model. The model is then used to generalize and classify unseen data. A modern example of machine learning is the email spam itself, not spam problem. A model is built on key words and word pairs labeled by a human as either spam or not. Using the model, the classifier will label new emails as either spam or not [13]. In this thesis, a HMM, a machine learning algorithm, is used to model individual smartphone users then attempts to label those users based on unseen patterns.

A HMM [14] is a machine learning model used when the data set is dependent on the sequence of collection. A HMM is a probabilistic finite state automaton where the output is dependent on the state. For this thesis, a HMM is used because the machine learning algorithm must be able to classify smartphone user based on transitions to various locations on the NPS campus. Classifiers such as Naïve Bayes would not work because they account for the similar Wi-Fi AP's the students detect but not the changes throughout the day.

1. Definition of a HMM

The mathematical definition of a HMM is a quintuple as follows:

$$\lambda = (S, V, \pi, A, B).$$

S is the state alphabet, where N is the number of states:

$$S = \{s_1, \ldots, s_N\}.$$

V is the vocabulary alphabet for the set of symbols that may be emitted:

$$V = \{v_1, \ldots, v_M\}.$$

Q is the fixed state sequence of length T:

$$Q = q_1, \ldots, q_T.$$

O is the corresponding observations to the fixed state sequence;

$$O = o_1, \ldots, o_T.$$

A is the transition probability matrix, where a_{ij} is the probability of transitioning from state i to state j:

$$A = [a_{ij}], a_{ij} = P(q_t = s_j \mid q_{t-1} = s_i).$$

B is the emission probability matrix, where b_{ij} is the probability of emitting symbol i in state j:

$$B = [b_i(k)], b_i(k) = P(o_t = v_k \mid q_t = s_i).$$

Π is the initial probability distribution giving the probability of starting in each state:

$$\Pi = [\Pi_i], \Pi_i = P(q_1 = s_i).$$

The Markov assumption states the current state is dependent only on the previous state:

$$P(q_t \mid q_1^{t-1}) = P(q_t \mid q_{t-1}) .$$

The output-independence assumption states the observation at time t is dependent only on the current state:

$$P(o_t \mid o_1^{t=1}, q_1^t) = P(o_t \mid q_t) .$$

2. Three Fundamental Problems for HMM

There are three fundamental problems for HMM design: evaluation, decoding, and learning. Chapter 15 of Russell and Norvig [13] describes the mathematical process to solve the fundamental problems. Once the fundamental problems are solved, the HMM could be applied to numerous statistical problems. Evaluation, decoding, and learning are defined as follows:

- Evaluation: Given an observation sequence and HMM model, determine the probability of the observation sequence.
- Decoding: Given an observation sequence and HMM model, determine the optimal sequence of model states.
- Learning: Adjust the model parameters to best account for the observed signals to maximize the HMM?

F. EVALUATION CRITERIA

Machine learning algorithm uses the number of true positive (TP), false positive (FP), true negative (TN), and false negative (FN) as measurements of performance. Their definitions are as follows:

- TP: correctly identified
- FP: incorrectly identified
- TN: correctly rejected
- FN: incorrectly rejected.

1. Confusion Matrix

A confusion matrix is often used as a visualization tool showing the performance of a classifier. An example of a confusion matrix is shown in Table 1. In the example, there are 8 red, 6 blue, and 13 green. For class red, the confusion matrix yields the following results:

- 5 TP: actual red classified as red
- 1 FP: blues incorrectly classified as red
- 3 FN: red incorrectly classified as blue (2) and Green (1)
- 17 TN: remaining colors classified correctly as non-red.

Truth	**Inferred label**		
	Red	**Blue**	**Green**
Red	5	2	1
Blue	1	2	3
Green	0	4	9

Table 1. Example of confusion matrix

2. Precision

Precision is also known as the positive predictive value. Precision is the fraction of a classified class that is relevant. In our example of the confusion matrix, red would have a precision of 5/6, which is the number of red correctly identified divided by the total number inferred as red (total of the column). The formula for precision is as follows:

$$precision = \frac{TP}{TP + FP} \quad .$$

3. Recall

Recall measures the sensitivity of the algorithm. Recall is the fraction of the class correctly labeled from the actual class. In our example of the confusion matrix, red would have a recall of 5/8, which is the number of red correctly identified divided the actual number of the class (total of the row). The formula for recall is as follows:

$$recall = \frac{TP}{TP + FN} \ .$$

4. F-score

F-score is the harmonic mean of precision and recall. F-score takes into account precision and recall measuring the algorithm’s overall accuracy. In our example of the confusion matrix, red would have an f-score of 0.7. The formula for f-score is as follows [15]:

$$F - Score = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \ .$$

THIS PAGE INTENTIONALLY LEFT BLANK

III. EXPERIMENTAL DESIGN

This chapter documents the methodologies and technical approaches in developing the experimental design used in this thesis. The methodology includes the research subjects and test parameters. The technical approach covers the tools used for data collection and transforming the data to a data structure to be used in a HMM.

A. HARDWARE AND SOFTWARE

The following hardware and software were used in this project:

- Google Nexus 4 Smartphone with Android version 4.3 and 16 GB Memory
- Power Mac Dual 3 GHz Intel Xeon processor, 16 GB 667 MHz RAM Memory
- Python 2.7.5
- Funf Journal for Android
- Wi-Fi Analyzer for Android.

B. RESEARCH SUBJECTS

This thesis research used graduate students from NPS located in Monterey, California, to collect RSSI data. NPS courses use the quarter system, where each student is required to take a minimum course load of four classes each quarter. The course lectures are one hour each given Monday through Thursday with Fridays reserved for labs. Each student was assigned a randomly generated PIN to be used throughout this research in order to maintain personally identifiable information confidentiality. The students each carried a Google Nexus 4 smartphone Monday through Thursday. When the students arrived on campus at the beginning of the day, they would turn on the sensors for collection. If the students left campus for lunch or any other reason, they would turn off the sensors until their return to campus. At the end of the day, the student would turn off the sensors and lock the smartphone in a secure locker provided. In addition, the students maintained a log of times and locations on campus. The log was used to filter the data set for times when the student was off campus but forgot to turn off the sensors. Table 2 lists the pin, major and the number of data points collected for each of the nine students. The number of data points collected for each student varied according to the student's schedule. Some students stayed on campus to study while others were only on campus for lectures.

PIN	Major	# Data Points
175	Computer Science	14,784
122	Computer Science	6,021
154	National Security Affairs	17,679
112	Business	15,111
198	Information Assurance	3,906
141	Business	16,337
128	National Security Affairs	13,611
111	Information Systems	6,499
372	Computer Science	14,589

Table 2. Research subject's PIN, major, and number of data points

C. LOCATION OF EXPERIMENT

The data for this thesis was collected on the NPS campus located in Monterey, California. The NPS campus is approximately 640 acres or 2.5 square kilometers. Figure 3 is a map of the NPS campus. Approximately one-fourth of the campus houses the academic buildings, while the rest are tenant facilities for Naval Support Activity, Monterey. The yellow buildings on the map are the location of the academic buildings where a majority of the data was collected.

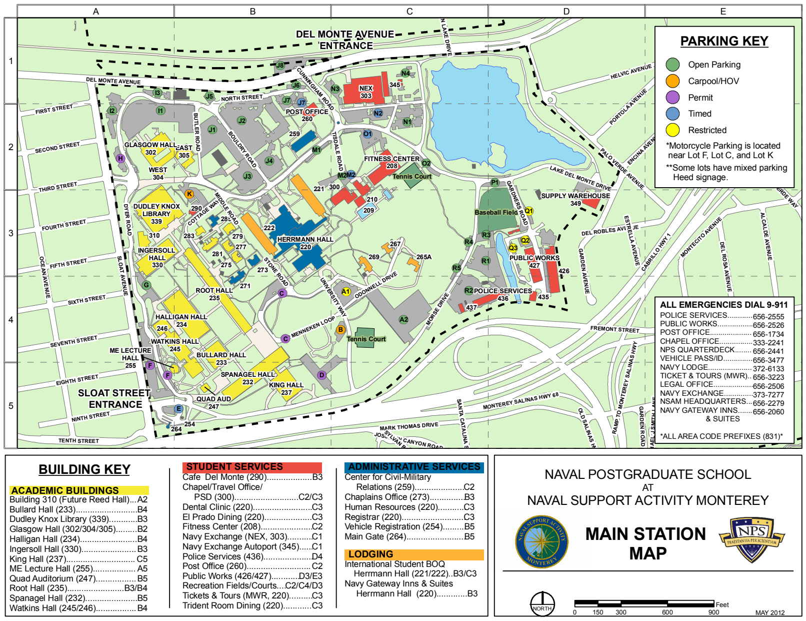
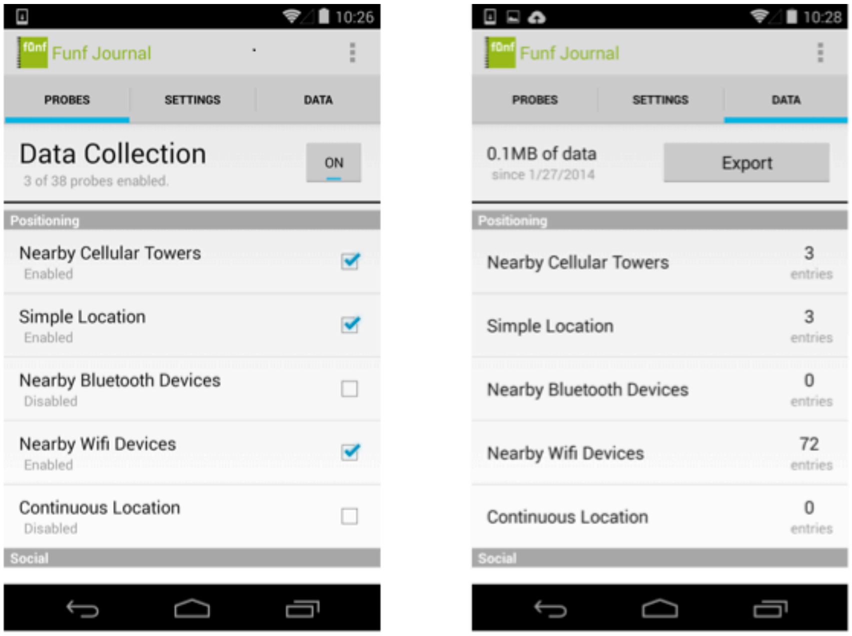

A
B
C
D
E
1
2
3
4
5
DEL MONTE AVENUE ENTRANCE
SLOAT STREET ENTRANCE
PARKING KEY
Open Parking
Carpool/HOV
Permit
Timed
Restricted
*Motorcycle Parking is located near Lot F, Lot C, and Lot K
**Some lots have mixed parking Heed signage.
ALL EMERGENCIES DIAL 9-911
POLICE SERVICES............656-2555
PUBLIC WORKS................656-2526
POST OFFICE...................656-1734
CHAPEL OFFICE................333-2241
NPS QUARTERDECK........656-2441
VEHICLE PASS/ID.............656-3477
NAVY LODGE....................372-6133
TICKET & TOURS (MWR)..656-3223
LEGAL OFFICE..................656-2506
NAVY EXCHANGE.............373-7277
NSAM HEADQUARTERS...656-2279
NAVY GATEWAY INNS.......656-2060 & SUITES
ALL AREA CODE PREFIXES (831)
GLASGOW HALL 302
EAST 305
WEST 304
DUDLEY KNOX LIBRARY 339
INGERSOLL HALL 330
ROOT HALL 235
HALLIGAN HALL 234
WATKINS HALL 245
ME LECTURE HALL 255
BULLARD HALL 233
SPANAGEL HALL 232
KING HALL 237
QUAD AUD 247
HERRMANN HALL 220
POST OFFICE 260
NEX 303
FITNESS CENTER 208
Tennis Court
Baseball Field
SUPPLY WAREHOUSE 349
PUBLIC WORKS 427
POLICE SERVICES 436
BUILDING KEY
ACADEMIC BUILDINGS
Building 310 (Future Reed Hall)...A2
Bullard Hall (233).........................B4
Dudley Knox Library (339)............B3
Glasgow Hall (302/304/305).........B2
Halligan Hall (234).........................B4
Ingersoll Hall (330)........................B3
King Hall (237)...............................C5
ME Lecture Hall (255)...................A5
Quad Auditorium (247)..................B5
Root Hall (235)........................B3/B4
Spanagel Hall (232).......................B5
Watkins Hall (245/246)..................B4
STUDENT SERVICES
Cafe Del Monte (290)..................B3
Chapel/Travel Office/ PSD (300)..........................C2/C3
Dental Clinic (220).........................C3
El Prado Dining (220)...................C3
Fitness Center (208).....................C2
Navy Exchange (NEX, 303)..........C1
Navy Exchange Autoport (345).....C1
Police Services (436)...................D4
Post Office (260)...........................C2
Public Works (426/427)...........D3/E3
Recreation Fields/Courts....C2/C4/D3
Tickets & Tours (MWR, 220).........C3
Trident Room Dining (220)...........C3
ADMINISTRATIVE SERVICES
Center for Civil-Military Relations (259)....................C2
Chaplains Office (273)..............B3
Human Resources (220)..........C3
Registrar (220)..........................C3
Vehicle Registration (254).........B5
Main Gate (264).........................B5
LODGING
International Student BOQ Herrmann Hall (221/222)..B3/C3
Navy Gateway Inns & Suites Herrmann Hall (220)............B3
NAVAL POSTGRADUATE SCHOOL AT NAVAL SUPPORT ACTIVITY MONTEREY
MAIN STATION MAP
NORTH
0 150 300 600 900 Feet
MAY 2012

Figure 3. Map of Naval Postgraduate School, Monterey, California (from [16]).

D. DATA COLLECTION PARAMETERS

Funf Journal [17] was used to collect the data for this research. Funf Journal is an open source framework that allows researchers to use Android sensors to collect and store data related to environmental and movement data. The app was downloaded from the Google store. Funf contain 38 probes enabling researchers collect data such as Wi-Fi, location, and accelerometer. Figure 4 shows screen shots of the Funf Journal positioning probes. For this research, the probes for nearby cellular towers, simple location, and nearby Wi-Fi devices were set to collect data every minute. The data is encrypted then stored in a structured query language (SQL) database on the Nexus 4. The export button allows the researcher to e-mail the encrypted files. Once on a desktop computer, the files are decrypted in a database format (.db) then converted to a comma separated value (CSV) file [17].



Data Collection
3 of 38 probes enabled.

Positioning

Nearby Cellular Towers Enabled	☑
Simple Location Enabled	☑
Nearby Bluetooth Devices Disabled	☐
Nearby Wifi Devices Enabled	☑
Continuous Location Disabled	☐

Social

0.1MB of data
since 1/27/2014

Positioning

Nearby Cellular Towers	3 entries
Simple Location	3 entries
Nearby Bluetooth Devices	0 entries
Nearby Wifi Devices	72 entries
Continuous Location	0 entries

Social

Figure 4. Screen shot of Funf Journal

E. SPARSE MATRIX

Once the data is extracted, the fields of the CSV file are parsed and filtered. The parsed file contains a list of tuples containing the timestamp, RSSI, and MAC address of all the Wi-Fi AP. A python script is used to input the list of tuples to form a sparse vector. A sparse vector and sparse matrix contains mostly zeroes [18]. The reason for transforming the tuples into sparse vector is to allow the data set to be inputted into a HMM. The script initially builds a vector of all zeros based on the MAC address. Each time an unseen MAC address is detected, a new element is created with a zero entry positioned at the sequential value based on the other MAC addresses already in the vector. Once the sparse vector is created, the script populates the sparse matrix. Each cell of the sparse matrix is RSSI values correlating to the MAC address. Within a minute sampling time, if the MAC address were detected, the RSSI value would replace the zero. The numbers of Wi-Fi AP scanned every minute varied from 1 to 20. A binary representation of the sparse matrix of test subject PIN-372 for a Wednesday from 0800 to 1700 is shown in Figure 5. The horizontal axis is the MAC addresses while the vertical axis is time interval in minutes. The sparse matrix shows the pattern as the user moves from different classrooms throughout the day.

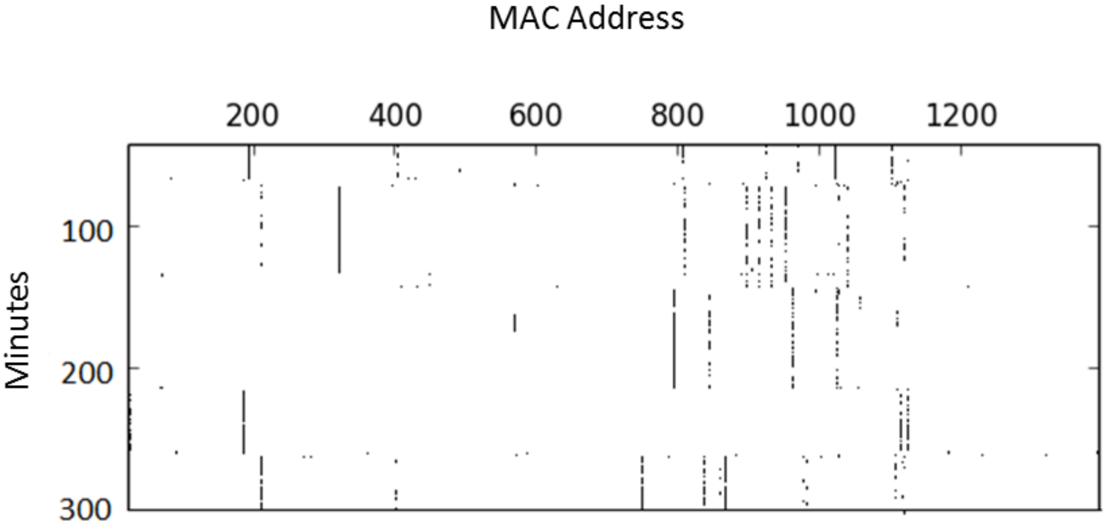


Figure 5. Sparse matrix

F. SPLITTING MATRIX INTO TRAINING AND TEST DATASET

The sparse matrix from each student was divided into window size of 25 minutes. The results were several sub matrix with 25 rows for minutes and 1154 columns for the number of total MAC address detected from all the users. Floyd's algorithm [19] for selecting random combinations of variables was used to divide the sub matrix into training and test. For the initial experiment, the algorithm randomly selected 80% of the dataset with uniform probability without replacement. The remaining 20% was used for testing. Ten runs were conducted on each experiment each randomly generating new training and test sets to provide ten-fold cross-validation.

G. CLASSIFIER

Once the dataset was randomly divided into training and test subsets, Gaussian HMM from scikit-learn [20] for python 2.7.5 was used to classify each user. The results were displayed in a confusion matrix to calculate the precision, recall, and f-score.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. RESULTS AND ANALYSIS

In this chapter, we review the results of our experiment. We first start with initial parameters of window size 25, 80% training and 20% testing, and sparse matrixes detected RSSI values. We varied our variable in each sequential experiment. For each experiment, ten runs were conducted resampling each time to provide ten-fold cross-validation. We only show the confusion matrix for the first run for the initial parameters in this chapter, the remaining confusion matrix results are in the appendix.

A. INITIAL PARAMETERS

The confusion matrix for our initial experiment is shown in Table 3. See Table 4 for the precision, recall, and f-score for each of our ten runs and the averages. For our initial parameters, we used a window size of 25, 80% training and 20% testing.

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	22	2	0	0	0	1	0	0	7
112	1	43	6	0	3	7	4	0	1
122	0	0	17	0	0	0	6	0	0
128	0	25	0	45	13	0	0	0	0
141	0	10	0	0	60	0	0	2	0
154	0	6	1	0	2	91	0	0	0
175	0	0	3	0	0	4	35	0	0
198	0	1	0	0	0	0	0	9	0
372	0	4	0	0	4	0	0	1	60

Table 3. Confusion Matrix for initial parameters run 1

	Precision	Recall	F-score
Run 1	0.81	0.77	0.77
Run 2	0.77	0.75	0.75
Run 3	0.84	0.82	0.82
Run 4	0.82	0.79	0.80
Run 5	0.83	0.81	0.81
Run 6	0.83	0.78	0.78
Run 7	0.84	0.83	0.83
Run 8	0.79	0.75	0.76
Run 9	0.81	0.79	0.79
Run 10	0.80	0.77	0.78
Avg	0.81	0.79	0.79

Table 4. Precision, Recall, and F-score from initial parameters

B. BINARY

For the binary experiment, instead of populating the sparse matrix with the RSSI value corresponding to the MAC address, a 1 was used if a Wi-Fi AP was detected otherwise the default value of zero was used. The resulting sparse matrixes only contain 0's and 1's. The purpose of this experiment is to determine if we can authenticate a user only by the Wi-Fi AP detected and not take into account the RSSI value.

	Precision	Recall	F-score
Run 1	0.81	0.77	0.77
Run 2	0.77	0.75	0.75
Run 3	0.84	0.82	0.82
Run 4	0.82	0.79	0.80
Run 5	0.83	0.81	0.81
Run 6	0.83	0.78	0.78
Run 7	0.84	0.83	0.83
Run 8	0.79	0.75	0.76
Run 9	0.81	0.79	0.79
Run 10	0.80	0.77	0.78
Avg	0.81	0.79	0.79

Table 5. Precision, Recall, and F-score for binary

C. LOGARITHMIC VALUE OF RSSI

In this experiment, we attempt to normalize the dataset by taking the logarithm value of the RSSI. The purpose for doing this is to account for the fluctuation in RSSI due to interference. The fluctuations could cause changes in the RSSI value by +/- 3 decibels. Taking the logarithm of the RSSI value clusters near values together. For example, log base 3 of RSSI values -26, -27, and -28 are 3.0 while RSSI values -29, -30, and -31 are 3.1. Table 6, Table 7, and Table 8 are the precision, recall, and f-score for log 3, log 5, and log 7, respectively.

	Precision	Recall	F-Score
Run 1	0.81	0.78	0.78
Run 2	0.79	0.79	0.79
Run 3	0.77	0.73	0.74
Run 4	0.84	0.82	0.83
Run 5	0.78	0.75	0.74
Run 6	0.79	0.77	0.77
Run 7	0.75	0.68	0.69
Run 8	0.80	0.75	0.76
Run 9	0.81	0.78	0.79
Run 10	0.82	0.81	0.80
Avg	0.80	0.77	0.77

Table 6. Precision, Recall, and F-score for Log 3

	Precision	Recall	F-Score
Run 1	0.85	0.84	0.84
Run 2	0.83	0.80	0.81
Run 3	0.79	0.74	0.74
Run 4	0.82	0.79	0.79
Run 5	0.78	0.77	0.77
Run 6	0.82	0.74	0.75
Run 7	0.80	0.74	0.75
Run 8	0.84	0.80	0.80
Run 9	0.79	0.75	0.76
Run 10	0.80	0.79	0.79
Avg	0.81	0.78	0.78

Table 7. Precision, Recall, and F-score for Log 5

	Precision	Recall	F-Score
Run 1	0.85	0.84	0.84
Run 2	0.79	0.74	0.74
Run 3	0.85	0.83	0.83
Run 4	0.84	0.80	0.81
Run 5	0.77	0.75	0.75
Run 6	0.84	0.83	0.83
Run 7	0.81	0.77	0.78
Run 8	0.81	0.80	0.80
Run 9	0.84	0.82	0.82
Run 10	0.84	0.82	0.82
Avg	0.82	0.80	0.80

Table 8. Precision, Recall, and F-score for Log 7

D. VARYING THE WINDOW SIZE

In this experiment, we vary the window size of the sparse matrix. Because NPS classes are 50 minutes long and start on the hour, varying the window size could better capture transition times when the students are moving from one class to another. Three different window sizes were used in this experiment. Table 9, Table 10, and Table 11 represent the precision, recall, and f-score for window size 10, 15, and 20, respectively.

	Precision	Recall	F-Score
Run 1	0.79	0.77	0.77
Run 2	0.76	0.75	0.74
Run 3	0.77	0.74	0.75
Run 4	0.80	0.74	0.74
Run 5	0.80	0.77	0.77
Run 6	0.76	0.74	0.74
Run 7	0.82	0.79	0.80
Run 8	0.81	0.79	0.79
Run 9	0.85	0.81	0.81
Run 10	0.78	0.76	0.76
Avg	0.79	0.77	0.77

Table 9. Precision, Recall, and F-score for Window Size 10

	Precision	Recall	F-Score
Run 1	0.81	0.75	0.76
Run 2	0.83	0.81	0.81
Run 3	0.77	0.75	0.76
Run 4	0.80	0.77	0.78
Run 5	0.84	0.83	0.83
Run 6	0.81	0.76	0.77
Run 7	0.79	0.75	0.76
Run 8	0.82	0.80	0.80
Run 9	0.80	0.74	0.76
Run 10	0.82	0.81	0.81
Avg	0.81	0.78	0.78

Table 10. Precision, Recall, and F-score for Window Size 15

	Precision	Recall	F-Score
Run 1	0.80	0.79	0.79
Run 2	0.78	0.76	0.75
Run 3	0.71	0.68	0.68
Run 4	0.80	0.77	0.77
Run 5	0.80	0.78	0.78
Run 6	0.78	0.78	0.77
Run 7	0.81	0.80	0.80
Run 8	0.85	0.83	0.83
Run 9	0.80	0.74	0.75
Run 10	0.81	0.80	0.80
Avg	0.79	0.77	0.77

Table 11. Precision, Recall, and F-score for Window Size 20

E. CHANGING PROPORTION OF TRAINING VERSUS TESTING DATA

In this experiment, we changed the proportion of training versus testing data. For machine learning algorithms, the rule of thumb is to use 80% of the data for training and building the model and reserving the remaining 20% to test against the completed model. Presented in Table 12 are the precision, recall, and f-score when only 50% of the data was used for training and the remaining 50% used for testing.

	Precision	Recall	F-Score
Run 1	0.79	0.73	0.72
Run 2	0.78	0.77	0.77
Run 3	0.82	0.70	0.71
Run 4	0.79	0.72	0.72
Run 5	0.75	0.72	0.72
Run 6	0.82	0.74	0.76
Run 7	0.81	0.77	0.77
Run 8	0.85	0.83	0.84
Run 9	0.81	0.79	0.79
Run 10	0.83	0.82	0.82
Avg	0.80	0.76	0.76

Table 12. Precision, Recall, and F-score for 50% Training, 50% Test

F. SUMMARY OF EXPERIMENTS

Figure 6 is a summary of the average f-scores from all the experiments. As expected, the worst performance was using only 50% of the dataset for training. Of note, building a binary model of Wi-Fi AP detected resulted in similar results from using the RSSI values. All variations of the experiment revealed f-scores between 0.7 and 0.8 showing a definite signal and reasonable probability of identifying a user based on RSSI-based geolocation.

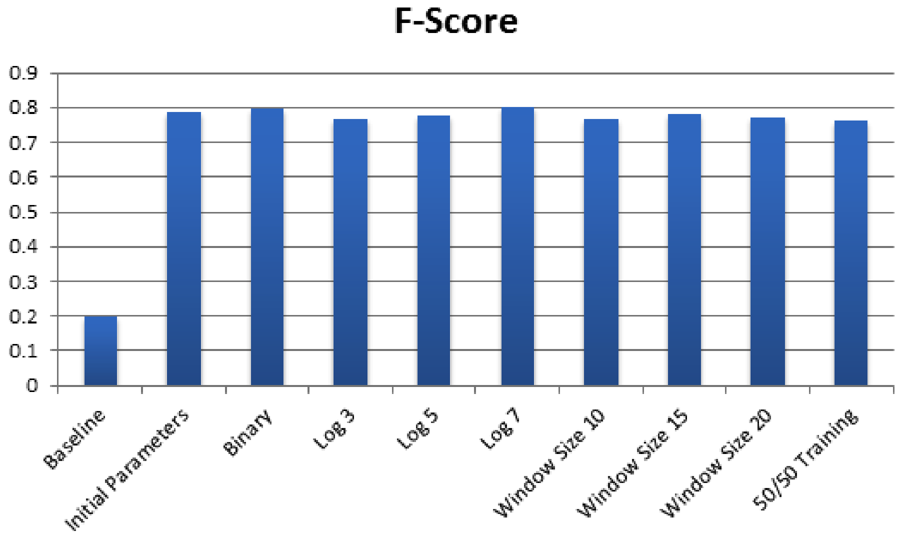


Figure 6. Summary of average f-scores

V. CONCLUSION AND FUTURE WORK

A. SUMMARY

The purpose of this thesis was to evaluate the feasibility of continuously authenticating a smartphone user using RSSI-based geolocation. Previous researches have used either GPS or cell tower geometric triangulation as geolocation sources. Our study collected RSSI data from nine NPS students each over a four-day period. The data collection was restricted to the NPS campus and filtered for times when the students were on campus. The RSSI and associated Wi-Fi AP data pair were put into a sparse matrix. The data was divided into 80% training and 20% testing. A HMM classifier was then used to model each user. The results of the experiments yield a precision, recall, and f-score between .70 and .85 for each of the test. The data shows RSSI-based geolocation could be used to continuously authenticate a smartphone user, however, results must be closer to 1.0 in order to yield the high confidence level for an authentication system.

B. FUTURE WORK

This thesis sets the foundation for future work in continuous authentication of a smartphone user. The following are recommendations for future work:

- Increase the number of research subjects. Only nine students were used during this research because the limitation on numbers of smartphones available during data collection and the requirement for each research subject to collect data for an entire week.

- Increase the diversity of the research subjects. This research focused on data collection from NPS students. The standard course load of an NPS student is four classes a day, equating to four hours a day on campus unless the student remains on campus between classes or after hours. Increasing diversity of subject pool by including professors, teaching assistants, or administrative staff could increase the data points collected per day.

- Broaden the physical parameters of the research. During this research, the data collection was restricted to NPS campus. When student left campus for lunch, medical appointments, or end of the day, students paused the data collection until return to campus. Future work could collect data outside the NPS campus for better fidelity on a subject's pattern of life throughout the day.
- Combine this research with Lieutenant William Parker's [21] "evaluation of data processing techniques for unobtrusive gait authentication" and Lieutenant Samuel Fleming's [22] "identification of a smartphone user via keystroke analysis."

C. CLOSING REMARKS

Is it possible to authenticate a smartphone user by continuous RSSI-based geolocation? With precision, recall, and f-scores above .7, it is feasible to use RSSI-based geolocation as an element in combination with other methods to continuously authenticate a smartphone user. For an acceptable authentication method, the evaluation criteria must be as close to 1.0 as possible. The research parameters in this research were very constrained, using NPS students as research subject and restricting the data collection to the NPS campus. A larger and broader data set for future work could increase the measure of performance to acceptable parameters.

Can we use a HMM to model a user's geolocation throughout the day? If yes, can we distinguish between various individuals? The result of the experiment shows a classification model which takes temporal states into consideration such as a HMM, could be used to model a user's geolocation throughout the day.

APPENDIX. CONFUSION MATRICES

A. CONFUSION MATRIX FOR INITIAL PARAMETERS

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	1	0	0	0	1	0	0	6
112	1	46	2	7	0	9	0	0	0
122	0	0	18	0	0	2	3	0	0
128	2	6	7	40	22	6	0	0	0
141	0	5	0	0	65	0	0	2	0
154	11	10	2	1	1	72	3	0	0
175	0	2	4	0	0	0	36	0	0
198	1	0	0	0	0	0	0	9	0
372	2	0	0	1	0	0	0	2	64

Table 13. Confusion Matrix for initial parameters run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	0	0	0	0	1	0	1	3
112	0	50	2	8	1	2	1	0	1
122	0	0	20	0	0	0	3	0	0
128	2	5	0	76	0	0	0	0	0
141	0	8	0	6	58	0	0	0	0
154	11	13	3	6	1	64	0	0	2
175	0	0	0	0	0	0	42	0	0
198	0	0	0	0	0	0	0	10	0
372	5	0	0	0	4	0	0	1	59

Table 14. Confusion Matrix for initial parameters run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	20	4	0	0	0	2	0	2	4
112	0	37	2	2	0	15	8	1	0
122	0	0	16	0	0	0	2	5	0
128	0	1	0	74	0	8	0	0	0
141	0	4	0	6	60	0	0	2	0
154	0	6	1	3	0	88	2	0	0
175	0	0	17	0	0	0	25	0	0
198	0	1	0	0	0	0	0	9	0
372	0	0	0	1	0	0	0	5	63

Table 15. Confusion Matrix for initial parameters run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	1	0	0	0	1	0	2	2
112	4	42	0	5	0	13	0	1	0
122	0	0	18	0	0	0	5	0	0
128	1	5	0	76	0	1	0	0	0
141	1	2	0	5	64	0	0	0	0
154	0	4	1	4	0	91	0	0	0
175	0	5	12	0	0	0	25	0	0
198	0	0	0	0	0	0	0	10	0
372	13	1	0	0	1	4	0	1	49

Table 16. Confusion Matrix for initial parameters run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	4	0	0	0	0	0	3	1
112	0	55	0	8	1	1	0	0	0
122	0	0	19	3	0	0	1	0	0
128	0	20	0	60	3	0	0	0	0
141	0	2	0	0	68	0	0	2	0
154	0	16	0	4	1	77	2	0	0
175	0	2	17	9	0	0	14	0	0
198	0	0	0	0	0	0	0	10	0
372	2	4	0	0	0	0	0	2	61

Table 17. Confusion Matrix for initial parameters run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	19	1	0	3	1	3	0	0	5
112	3	52	1	4	0	1	4	0	0
122	0	0	20	0	0	0	3	0	0
128	2	3	0	78	0	0	0	0	0
141	0	4	0	6	62	0	0	0	0
154	1	5	1	8	0	85	0	0	0
175	0	0	13	0	0	0	29	0	0
198	0	0	0	0	0	0	0	10	0
372	10	1	0	0	0	0	0	3	55

Table 18. Confusion Matrix for initial parameters run 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	1	0	1	0	0	0	1	3
112	1	41	4	9	0	9	0	0	1
122	0	0	16	0	0	0	7	0	0
128	22	7	0	50	0	3	0	0	1
141	6	8	0	0	56	0	0	2	0
154	0	8	1	9	1	80	0	0	1
175	0	4	0	0	0	0	38	0	0
198	0	1	0	0	0	0	0	9	0
372	7	4	0	0	0	0	0	1	57

Table 19. Confusion Matrix for initial parameters run 8

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	2	0	0	0	0	0	0	5
112	0	52	0	7	6	0	0	0	0
122	0	0	12	0	0	0	11	0	0
128	0	18	0	54	11	0	0	0	0
141	0	0	0	1	69	0	0	2	0
154	0	10	1	10	0	79	0	0	0
175	0	4	8	3	0	0	27	0	0
198	0	1	0	0	0	0	0	9	0
372	1	0	0	1	0	0	0	1	66

Table 20. Confusion Matrix for initial parameters run 9

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	1	0	1	0	0	0	1	5
112	5	41	3	7	0	3	4	0	2
122	0	0	22	0	0	0	1	0	0
128	2	6	2	63	0	6	0	0	4
141	1	10	0	5	53	0	0	3	0
154	0	5	3	6	0	84	0	0	2
175	0	0	14	0	0	0	28	0	0
198	0	0	0	0	0	0	0	10	0
372	9	0	0	0	0	0	0	2	58

Table 21. Confusion Matrix for initial parameters run 10

B. CONFUSION MATRIX FOR BINARY

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	20	5	0	0	0	2	0	0	5
112	1	29	4	11	4	9	5	2	0
122	0	0	23	0	0	0	0	0	0
128	0	0	0	83	0	0	0	0	0
141	0	1	0	6	64	0	0	1	0
154	0	0	2	9	0	89	0	0	0
175	0	0	11	0	0	0	31	0	0
198	0	0	0	0	0	0	0	10	0
372	6	0	0	1	0	4	0	1	57

Table 22. Confusion Matrix for binary run 1

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	1	0	0	0	0	0	0	6
112	4	27	0	4	0	21	9	0	0
122	0	0	23	0	0	0	0	0	0
128	3	20	0	57	3	0	0	0	0
141	0	4	0	1	67	0	0	0	0
154	0	6	1	5	0	88	0	0	0
175	0	0	2	0	0	0	40	0	0
198	0	1	0	0	0	0	0	9	0
372	1	0	0	0	0	4	0	0	64

Table 23. Confusion Matrix for binary run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	21	2	0	0	0	0	0	1	8
112	0	34	3	1	8	15	4	0	0
122	0	0	10	0	0	0	13	0	0
128	0	24	8	37	14	0	0	0	0
141	0	0	0	0	70	0	0	2	0
154	0	10	5	1	10	73	1	0	0
175	0	0	0	0	0	0	42	0	0
198	0	0	0	0	0	0	0	10	0
372	0	0	0	0	0	3	0	2	64

Table 24. Confusion Matrix for binary run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	2	0	2	1	0	0	0	0
112	0	47	0	15	1	1	0	0	1
122	0	0	23	0	0	0	0	0	0
128	0	5	0	77	1	0	0	0	0
141	0	2	0	1	66	0	0	3	0
154	0	14	2	8	0	75	0	1	0
175	0	2	14	0	0	0	26	0	0
198	0	0	0	0	0	0	0	10	0
372	2	4	0	0	0	0	0	0	63

Table 25. Confusion Matrix for binary run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	0	0	2	0	0	0	2	2
112	0	37	0	5	8	5	7	3	0
122	0	0	22	0	0	0	1	0	0
128	0	10	0	59	6	8	0	0	0
141	0	0	0	1	69	0	0	2	0
154	0	14	1	7	0	76	0	2	0
175	0	0	13	0	0	0	29	0	0
198	0	0	0	0	0	0	0	10	0
372	1	0	0	0	0	1	0	3	64

Table 26. Confusion Matrix for binary run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	3	0	0	0	0	0	1	3
112	2	38	1	11	8	4	0	1	0
122	0	1	19	0	0	0	3	0	0
128	0	15	0	68	0	0	0	0	0
141	0	0	0	6	63	0	0	3	0
154	0	9	1	2	1	86	0	1	0
175	0	4	2	0	0	4	32	0	0
198	0	0	0	0	0	0	0	10	0
372	1	0	0	1	0	0	0	5	62

Table 27. Confusion Matrix for binary run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	30	1	0	0	0	1	0	0	0
112	4	40	2	1	0	14	4	0	0
122	0	0	22	0	0	0	1	0	0
128	1	12	1	60	0	9	0	0	0
141	2	2	0	5	61	1	0	1	0
154	43	6	1	1	1	48	0	0	0
175	0	0	12	0	0	4	26	0	0
198	0	1	0	0	0	0	0	9	0
372	9	1	0	0	0	0	0	0	59

Table 28. Confusion Matrix for binary run 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	4	0	0	0	0	0	0	2
112	1	56	0	1	5	2	0	0	0
122	0	1	22	0	0	0	0	0	0
128	0	22	0	61	0	0	0	0	0
141	0	0	0	5	65	0	0	2	0
154	0	17	1	0	0	82	0	0	0
175	0	5	5	0	0	0	32	0	0
198	0	1	0	0	0	0	0	9	0
372	5	1	0	0	0	4	0	1	58

Table 29. Confusion Matrix for binary run 8

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	28	0	0	1	0	0	0	0	3
112	12	32	1	9	0	7	3	1	0
122	0	0	21	0	0	0	2	0	0
128	4	0	0	79	0	0	0	0	0
141	2	6	0	4	56	0	0	4	0
154	5	3	19	8	1	64	0	0	0
175	0	0	1	0	0	0	41	0	0
198	0	0	0	0	0	0	0	10	0
372	3	0	0	0	1	0	0	1	64

Table 30. Confusion Matrix for binary run 9

Truth	Inferred Labels								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	0	0	4	3	1	0	0	0
112	1	41	0	20	1	2	0	0	0
122	0	0	21	0	0	0	2	0	0
128	0	0	0	83	0	0	0	0	0
141	0	3	0	1	65	0	0	3	0
154	0	5	2	26	0	67	0	0	0
175	0	3	5	0	0	0	34	0	0
198	0	0	0	0	0	0	0	10	0
372	8	0	0	1	0	0	0	2	58

Table 31. Confusion Matrix for binary run 10

C. CONFUSION MATRIX FOR LOG 3

Truth	Inferred Labels								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	21	2	0	0	0	0	0	0	7
112	0	45	1	3	7	0	9	0	0
122	0	0	21	0	0	0	3	0	0
128	0	15	0	41	26	0	0	0	0
141	0	0	0	0	69	0	0	3	0
154	0	11	2	5	2	80	0	0	0
175	0	4	3	0	0	0	35	0	0
198	0	1	0	0	0	0	0	9	0
372	0	0	0	0	4	0	0	1	65

Table 32. Confusion Matrix for log 3 run 1

Truth	Inferred Labels								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	20	1	0	2	0	0	0	0	7
112	0	35	2	14	1	5	1	0	7
122	0	0	18	0	0	0	6	0	0
128	0	4	0	71	2	4	0	0	1
141	0	3	0	3	63	0	0	3	0
154	5	9	1	11	0	72	1	0	1
175	0	3	3	0	0	0	36	0	0
198	0	1	0	0	0	0	0	9	0
372	3	0	0	0	0	0	0	1	66

Table 33. Confusion Matrix for log 3 run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	22	1	0	0	0	1	0	1	5
112	8	36	1	8	0	5	5	2	0
122	0	0	15	0	0	7	2	0	0
128	4	24	1	50	0	2	0	1	0
141	0	10	0	6	53	0	0	3	0
154	14	9	3	3	0	71	0	0	0
175	0	1	0	0	0	0	41	0	0
198	0	0	0	0	0	0	0	10	0
372	1	4	0	0	0	0	0	2	63

Table 34. Confusion Mat for log 3 run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	1	0	1	0	0	0	0	2
112	0	42	1	14	8	0	0	0	0
122	0	1	21	0	0	0	2	0	0
128	0	9	0	72	1	0	0	0	0
141	0	0	0	0	69	0	0	3	0
154	1	12	1	9	1	75	1	0	0
175	0	3	3	0	0	0	36	0	0
198	0	1	0	0	0	0	0	9	0
372	11	0	0	0	0	0	0	1	58

Table 35. Confusion Matrix for log 3 run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	12	1	0	1	7	0	0	1	8
112	0	35	1	13	8	0	8	0	0
122	0	0	22	0	0	0	2	0	0
128	0	5	0	56	21	0	0	0	0
141	0	0	0	0	69	0	0	3	0
154	2	12	1	7	0	76	1	0	1
175	0	0	16	0	0	0	26	0	0
198	0	0	0	0	0	0	0	10	0
372	0	0	0	1	0	0	0	5	64

Table 36. Confusion Matrix for log 3 run 5

Truth	**Inferred Labels**								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	1	0	0	0	1	0	0	1
112	8	32	0	5	7	3	0	9	1
122	0	0	20	0	0	3	1	0	0
128	0	12	0	66	0	1	0	0	3
141	0	0	0	6	63	0	0	3	0
154	10	6	0	9	0	72	1	1	1
175	0	3	13	0	0	0	26	0	0
198	0	0	0	0	0	0	0	10	0
372	2	4	0	0	0	0	0	1	63

Table 37. Confusion Matrix for log 3 run 6

Truth	**Inferred Labels**								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	22	2	0	0	0	0	0	0	6
112	6	34	0	3	0	9	9	2	2
122	0	0	16	0	0	0	8	0	0
128	0	13	0	51	12	3	0	0	3
141	0	9	0	0	63	0	0	0	0
154	42	12	1	0	0	42	0	1	2
175	0	0	2	0	0	0	40	0	0
198	0	0	0	0	0	0	0	10	0
372	8	0	0	0	0	0	0	1	61

Table 38. Confusion Matrix for log 3 run 7

Truth	**Inferred Labels**								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	1	0	0	0	0	0	0	5
112	0	46	6	0	2	3	7	1	0
122	0	0	22	0	0	0	2	0	0
128	1	9	4	60	4	4	0	0	0
141	0	2	0	0	70	0	0	0	0
154	0	41	1	2	0	56	0	0	0
175	0	0	18	0	0	4	20	0	0
198	0	0	0	0	0	0	0	10	0
372	2	5	0	0	0	0	0	1	62

Table 39. Confusion Matrix for log 3 run 8

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	22	0	0	1	0	0	0	0	7
112	0	33	7	9	0	11	3	2	0
122	0	4	19	0	0	0	1	0	0
128	0	1	0	77	0	4	0	0	0
141	0	2	0	6	61	0	0	3	0
154	1	14	1	2	0	80	0	0	2
175	0	0	19	0	0	0	23	0	0
198	0	0	0	0	0	0	0	10	0
372	0	1	0	1	0	4	0	1	63

Table 40. Confusion Matrix for log 3 run 9

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	0	0	1	0	0	0	0	2
112	0	28	0	18	6	3	10	0	0
122	0	0	21	0	0	0	3	0	0
128	0	2	0	65	11	4	0	0	0
141	0	0	0	0	69	0	0	3	0
154	1	5	2	10	1	81	0	0	0
175	0	0	2	0	0	0	40	0	0
198	0	0	0	0	0	0	0	10	0
372	4	1	0	0	4	0	0	1	60

Table 41. Confusion Matrix for log 3 run 10

D. CONFUSION MATRIX FOR LOG 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	21	0	0	0	0	0	0	0	9
112	0	51	0	5	0	5	3	1	0
122	0	0	16	0	0	0	8	0	0
128	0	7	0	70	0	5	0	0	0
141	0	9	0	6	56	0	0	1	0
154	1	5	1	5	0	88	0	0	0
175	0	1	1	0	0	0	40	0	0
198	0	1	0	0	0	0	0	9	0
372	0	0	0	1	0	3	0	1	65

Table 42. Confusion Matrix for log 5 run 1

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	22	1	0	0	0	1	0	0	6
112	0	46	1	3	0	5	7	0	3
122	0	0	21	0	0	0	3	0	0
128	5	4	0	72	0	0	0	0	1
141	1	9	0	5	56	0	0	1	0
154	12	10	1	3	0	70	3	0	1
175	0	0	4	0	0	0	38	0	0
198	0	0	0	0	0	0	0	10	0
372	2	4	0	0	0	0	0	1	63

Table 43. Confusion Matrix for log 5 run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	19	1	0	0	3	0	0	2	5
112	0	46	2	3	0	1	8	5	0
122	0	0	19	0	0	0	5	0	0
128	0	25	4	26	25	1	0	1	0
141	0	9	0	0	60	0	0	3	0
154	0	12	3	3	1	81	0	0	0
175	0	0	1	0	0	0	41	0	0
198	0	0	0	0	0	0	0	10	0
372	0	0	3	0	0	0	0	1	66

Table 44. Confusion Matrix for log 5 run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	22	0	0	1	0	0	0	2	5
112	1	36	0	19	0	3	3	3	0
122	0	0	14	0	0	0	10	0	0
128	0	0	0	79	0	3	0	0	0
141	0	4	0	6	59	0	0	3	0
154	0	3	1	15	0	81	0	0	0
175	0	2	8	0	0	1	31	0	0
198	0	0	0	0	0	0	0	10	0
372	6	0	0	4	0	1	0	1	58

Table 45. Confusion Matrix for log 5 run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	20	1	0	1	0	0	0	1	7
112	1	33	10	8	0	11	0	2	0
122	0	0	9	0	0	5	10	0	0
128	4	0	0	78	0	0	0	0	0
141	0	8	0	6	54	0	0	4	0
154	0	4	1	10	0	84	1	0	0
175	0	4	10	0	0	1	27	0	0
198	0	0	0	0	0	0	0	10	0
372	0	0	0	1	0	3	0	0	66

Table 46. Confusion Matrix for log 5 run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	28	0	0	2	0	0	0	0	0
112	0	38	9	11	0	0	0	7	0
122	0	0	22	0	0	0	2	0	0
128	0	0	8	74	0	0	0	0	0
141	0	3	0	6	60	0	0	3	0
154	4	28	6	10	1	51	0	0	0
175	0	2	15	0	0	0	25	0	0
198	0	0	0	0	0	0	0	10	0
372	7	0	0	1	4	0	0	1	57

Table 47. Confusion Matrix for log 5 run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	23	2	0	0	1	0	0	0	4
112	1	53	0	1	0	3	7	0	0
122	0	0	21	0	0	0	3	0	0
128	2	22	0	51	0	7	0	0	0
141	0	4	0	6	59	0	0	3	0
154	5	45	5	0	0	45	0	0	0
175	0	0	1	0	0	0	41	0	0
198	0	0	0	0	0	0	0	10	0
372	5	0	0	0	0	0	0	1	64

Table 48. Confusion Matrix for log 5 run 7

Truth	Inferred Labels								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	0	0	1	0	0	0	0	4
112	2	46	0	15	0	0	2	0	0
122	0	0	23	0	0	0	1	0	0
128	0	2	0	80	0	0	0	0	0
141	0	4	0	6	59	0	0	3	0
154	12	8	1	18	0	61	0	0	0
175	0	1	10	0	0	0	31	0	0
198	0	1	0	0	0	0	0	9	0
372	7	1	0	0	0	0	0	1	61

Table 49. Confusion Matrix for log 5 run 8

Truth	Inferred Labels								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	1	0	0	0	0	0	2	0
112	5	34	1	4	0	5	10	6	0
122	0	0	12	0	0	0	12	0	0
128	5	5	0	68	2	0	0	0	2
141	0	10	0	0	59	0	0	3	0
154	1	13	0	2	0	80	4	0	0
175	0	0	15	0	0	0	27	0	0
198	0	0	0	0	0	0	0	10	0
372	9	0	0	1	0	4	0	1	55

Table 50. Confusion Matrix for log 5 run 9

Truth	Inferred Labels								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	19	0	0	1	0	1	0	0	9
112	3	34	7	6	0	4	9	2	0
122	0	0	12	0	0	0	12	0	0
128	1	3	7	70	0	1	0	0	0
141	0	8	0	6	57	0	0	1	0
154	1	6	2	5	1	85	0	0	0
175	0	0	0	0	0	0	42	0	0
198	0	1	0	0	0	0	0	9	0
372	4	0	0	0	0	0	0	2	64

Table 51. Confusion Matrix for log 5 run 10

E. CONFUSION MATRIX FOR LOG 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	2	0	0	3	1	0	0	0
112	0	38	0	14	0	7	0	6	0
122	0	0	22	0	0	0	2	0	0
128	0	0	0	80	2	0	0	0	0
141	0	9	0	0	63	0	0	0	0
154	2	2	2	10	1	83	0	0	0
175	0	4	1	0	0	0	37	0	0
198	0	0	0	0	0	0	0	10	0
372	8	0	0	0	0	0	0	1	61

Table 52. Confusion Matrix for log 7 run 1

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	1	0	0	0	0	0	0	2
112	0	54	0	8	1	2	0	0	0
122	0	2	15	0	0	4	3	0	0
128	0	29	0	39	10	4	0	0	0
141	0	4	0	0	68	0	0	0	0
154	0	31	0	5	0	63	1	0	0
175	0	8	3	0	0	0	31	0	0
198	0	0	0	0	0	0	0	10	0
372	6	5	0	0	0	1	0	1	57

Table 53. Confusion Matrix for log 7 run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	20	1	0	0	0	1	0	1	7
112	0	48	0	2	0	4	8	3	0
122	0	0	24	0	0	0	0	0	0
128	0	6	0	76	0	0	0	0	0
141	0	9	0	6	57	0	0	0	0
154	0	14	1	0	1	84	0	0	0
175	0	8	6	0	0	0	28	0	0
198	0	1	0	0	0	0	0	9	0
372	0	1	0	0	4	0	0	2	63

Table 54. Confusion Matrix for log 7 run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	19	1	0	1	0	0	0	0	9
112	0	56	4	5	0	0	0	0	0
122	0	0	18	0	0	0	6	0	0
128	1	2	0	79	0	0	0	0	0
141	2	10	0	4	53	0	0	3	0
154	0	25	1	6	0	68	0	0	0
175	0	2	4	0	0	4	32	0	0
198	0	0	0	0	0	0	0	10	0
372	0	6	0	1	0	0	0	1	62

Table 55. Confusion Matrix for log 7 run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	2	0	0	0	0	0	0	1
112	8	38	0	10	8	1	0	0	0
122	0	0	21	0	0	0	3	0	0
128	0	34	0	42	6	0	0	0	0
141	0	0	0	3	66	0	0	3	0
154	1	9	1	5	12	67	5	0	0
175	0	1	5	0	0	0	36	0	0
198	0	0	0	0	0	0	0	10	0
372	2	0	0	0	0	4	0	1	63

Table 56. Confusion Matrix for log 7 run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	0	0	0	0	2	0	0	3
112	2	38	0	8	9	8	0	0	0
122	0	0	21	0	0	0	3	0	0
128	4	5	0	72	0	1	0	0	0
141	0	0	0	6	63	0	0	3	0
154	0	3	1	5	1	90	0	0	0
175	0	1	13	0	0	0	28	0	0
198	0	1	0	0	0	0	0	9	0
372	1	1	0	0	0	0	0	1	67

Table 57. Confusion Matrix for log 7 run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	0	0	0	0	0	0	0	6
112	5	51	0	5	0	3	0	1	0
122	0	1	16	0	0	6	1	0	0
128	2	0	0	61	15	4	0	0	0
141	9	0	0	0	60	0	0	3	0
154	3	7	0	3	5	82	0	0	0
175	0	4	16	0	0	1	21	0	0
198	0	1	0	0	0	0	0	9	0
372	9	1	0	0	0	0	0	1	59

Table 58. Confusion Matrix for log 7 run 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	23	0	0	2	0	0	0	0	5
112	3	38	0	8	7	2	0	6	1
122	0	0	20	0	0	0	4	0	0
128	0	14	0	51	17	0	0	0	0
141	0	0	0	0	72	0	0	0	0
154	1	11	1	6	0	80	0	1	0
175	0	2	3	0	0	0	37	0	0
198	0	0	0	0	0	0	0	10	0
372	2	3	0	0	0	0	0	1	64

Table 59. Confusion Matrix for log 7 run 8

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	2	0	0	0	0	0	0	3
112	0	48	0	1	2	2	9	1	2
122	0	0	11	0	0	0	13	0	0
128	0	13	0	61	2	4	0	0	2
141	0	2	0	0	67	0	0	3	0
154	1	14	1	0	0	83	0	0	1
175	0	0	3	0	0	0	39	0	0
198	0	0	0	0	0	0	0	10	0
372	9	0	0	0	0	0	0	1	60

Table 60. Confusion Matrix for log 7 run 9

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	21	0	0	0	6	1	0	0	2
112	1	43	2	1	8	8	0	2	0
122	0	0	21	0	0	0	3	0	0
128	4	5	7	50	11	5	0	0	0
141	0	0	0	0	69	0	0	3	0
154	1	2	3	0	1	93	0	0	0
175	0	3	3	0	0	0	36	0	0
198	0	0	0	0	0	0	0	10	0
372	7	0	0	0	0	0	0	2	61

Table 61. Confusion Matrix for log 7 run 10

F. CONFUSION MATRIX FOR WINDOW SIZE 10

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	61	1	0	3	20	0	0	1	7
112	2	118	15	15	0	14	12	0	0
122	0	0	61	0	0	0	11	0	0
128	1	6	11	166	28	8	0	0	0
141	0	12	1	0	173	0	0	6	0
154	1	32	7	13	16	174	22	0	0
175	0	1	20	0	0	0	97	0	0
198	0	3	0	0	0	0	0	36	0
372	15	3	0	2	2	0	0	4	161

Table 62. Confusion Matrix for window size 10 run 1

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	48	5	1	2	8	5	0	4	20
112	6	74	24	32	0	9	19	8	4
122	0	0	36	0	0	0	36	0	0
128	0	3	11	195	3	0	0	0	8
141	2	23	0	12	147	0	0	8	0
154	1	13	16	21	0	209	0	0	5
175	0	4	0	0	0	11	103	0	0
198	1	0	0	0	0	0	0	38	0
372	0	12	0	0	0	0	1	6	168

Table 63. Confusion Matrix for window size 10 run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	65	4	0	3	0	0	0	1	20
112	16	75	24	22	0	12	0	0	27
122	0	4	59	0	0	0	9	0	0
128	12	10	13	171	0	14	0	0	0
141	11	28	0	9	138	0	0	6	0
154	6	12	16	16	0	209	0	0	6
175	0	12	24	0	0	0	82	0	0
198	2	2	0	0	0	0	0	35	0
372	0	5	0	0	0	0	0	4	178

Table 64. Confusion Matrix for window size 10 run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	2	0	2	0	0	0	1	3
112	0	41	5	12	0	1	2	3	1
122	0	1	22	0	0	0	0	0	0
128	0	10	0	73	0	0	0	0	0
141	0	6	0	5	56	0	0	4	1
154	3	14	2	13	0	62	5	1	0
175	0	1	29	0	0	0	12	0	0
198	0	0	0	0	0	0	0	10	0
372	1	0	0	0	0	0	0	1	67

Table 65. Confusion Matrix for window size 10 run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	29	1	0	0	0	0	0	1	1
112	0	40	5	6	7	3	3	0	1
122	0	0	21	0	0	0	2	0	0
128	5	29	0	45	0	4	0	0	0
141	3	0	0	3	64	0	0	2	0
154	6	6	2	5	1	80	0	0	0
175	0	0	14	0	0	0	28	0	0
198	0	0	0	0	0	0	0	10	0
372	2	0	0	1	0	0	0	0	66

Table 66. Confusion Matrix for window size 10 run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	19	0	0	0	5	3	0	2	3
112	0	26	2	14	1	9	9	4	0
122	0	0	18	0	0	0	5	0	0
128	0	3	0	61	19	0	0	0	0
141	0	10	0	0	60	0	0	2	0
154	0	4	1	8	0	85	1	1	0
175	0	0	15	0	0	0	27	0	0
198	0	0	0	0	0	0	0	10	0
372	1	1	0	0	0	0	4	1	62

Table 67. Confusion Matrix for window size 10 run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	0	0	0	0	2	0	0	5
112	0	36	9	1	0	18	0	1	0
122	0	0	13	0	0	0	5	5	0
128	0	5	8	61	7	2	0	0	0
141	0	8	0	0	62	0	0	2	0
154	0	3	6	3	1	87	0	0	0
175	0	2	1	0	0	0	39	0	0
198	0	1	0	0	0	0	0	9	0
372	2	0	0	0	0	1	0	5	61

Table 68. Confusion Matrix for window size 10 run 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	2	0	0	0	0	0	0	4
112	1	42	1	2	6	3	8	0	2
122	0	0	12	0	0	2	9	0	0
128	0	4	0	79	0	0	0	0	0
141	2	0	0	4	63	0	0	3	0
154	5	26	0	7	0	62	0	0	0
175	0	4	3	0	0	0	35	0	0
198	0	1	0	0	0	0	0	9	0
372	2	0	0	1	0	0	0	1	65

Table 69. Confusion Matrix for window size 10 run 8

Truth	**Inferred Labels**								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	2	0	0	0	0	0	1	2
112	0	51	2	1	6	2	3	0	0
122	0	1	19	0	0	0	3	0	0
128	0	17	1	54	11	0	0	0	0
141	0	0	0	0	70	0	0	2	0
154	0	24	1	0	1	74	0	0	0
175	0	1	0	0	0	4	37	0	0
198	0	1	0	0	0	0	0	9	0
372	7	0	0	0	0	0	0	2	60

Table 70. Confusion Matrix for window size 10 run 9

Truth	**Inferred Labels**								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	18	2	0	1	0	0	0	2	9
112	0	45	3	9	6	2	0	0	0
122	0	1	16	0	0	0	6	0	0
128	2	3	0	60	14	4	0	0	0
141	0	0	0	0	72	0	0	0	0
154	1	15	1	10	1	70	1	1	0
175	0	5	12	0	0	0	25	0	0
198	0	1	0	0	0	0	0	9	0
372	0	0	0	0	4	1	0	2	62

Table 71. Confusion Matrix for window size 10 run 10

G. CONFUSION MATRIX FOR WINDOW SIZE 15

Truth	**Inferred Labels**								
	111	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	38	1	0	2	0	0	0	4	14
112	2	57	6	14	12	0	14	10	0
122	0	0	21	0	0	0	14	9	0
128	0	1	14	98	27	0	0	4	0
141	0	0	0	1	122	0	0	2	0
154	1	3	8	18	1	134	0	9	0
175	0	0	16	0	0	0	61	0	0
198	0	0	0	0	0	0	0	23	0
372	3	0	0	0	0	0	0	15	103

Table 72. Confusion Matrix for window size 15 run 1

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	40	6	0	0	0	3	0	0	10
112	0	79	2	4	0	17	13	0	0
122	0	0	38	0	0	0	6	0	0
128	1	25	1	111	1	5	0	0	0
141	0	8	0	2	103	9	0	3	0
154	0	16	3	7	0	146	2	0	0
175	0	1	10	0	0	0	65	0	0
198	0	2	0	0	0	0	0	21	0
372	2	1	0	0	0	9	0	0	109

Table 73. Confusion Matrix for window size 15 run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	47	3	0	1	0	0	0	2	6
112	0	67	3	15	12	9	7	1	1
122	0	1	33	0	0	1	9	0	0
128	0	34	3	100	1	6	0	0	0
141	0	0	0	10	110	0	0	3	2
154	0	16	6	14	1	134	2	1	0
175	0	8	22	0	0	7	39	0	0
198	0	0	0	0	0	0	0	23	0
372	2	1	8	0	0	0	0	0	110

Table 74. Confusion Matrix for window size 15 run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	19	2	0	1	0	0	0	1	9
112	0	43	6	11	0	1	4	0	0
122	0	0	17	0	0	1	5	0	0
128	4	9	0	70	0	0	0	0	0
141	0	4	0	6	60	0	0	2	0
154	4	11	1	11	0	64	9	0	0
175	0	0	3	0	0	0	39	0	0
198	0	1	0	0	0	0	0	9	0
372	0	4	0	0	0	0	0	2	63

Table 75. Confusion Matrix for window size 15 run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	1	0	1	0	0	0	1	4
112	1	54	1	3	0	0	4	0	2
122	5	0	16	0	1	0	1	0	0
128	0	4	0	75	0	0	0	0	4
141	0	7	0	6	57	0	0	2	0
154	0	14	1	3	1	78	1	0	2
175	0	1	2	0	0	4	35	0	0
198	0	1	0	0	0	0	0	9	0
372	6	0	0	0	0	0	0	1	62

Table 76. Confusion Matrix for window size 15 run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	29	2	0	0	0	0	0	1	0
112	8	45	3	5	1	2	0	1	0
122	0	1	21	0	0	0	1	0	0
128	1	24	0	47	11	0	0	0	0
141	0	4	0	0	65	0	0	3	0
154	0	17	2	4	7	70	0	0	0
175	0	4	3	0	0	4	31	0	0
198	0	1	0	0	0	0	0	9	0
372	3	4	0	0	0	0	0	1	61

Table 77. Confusion Matrix for window size 15 run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	20	5	0	2	0	0	0	2	3
112	0	37	0	16	0	6	3	3	0
122	0	0	12	0	0	0	11	0	0
128	0	4	0	73	0	6	0	0	0
141	0	3	0	8	55	0	0	6	0
154	0	20	1	10	1	67	0	1	0
175	0	4	0	0	0	0	38	0	0
198	0	0	0	0	0	0	0	10	0
372	2	0	0	0	0	1	0	5	61

Table 78. Confusion Matrix for window size 15 run 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	17	1	0	0	3	2	0	3	6
112	3	40	0	8	0	5	2	7	0
122	0	0	20	0	0	0	3	0	0
128	0	1	0	81	1	0	0	0	0
141	0	10	0	3	57	0	0	2	0
154	3	6	1	14	1	72	2	1	0
175	0	4	2	0	0	0	36	0	0
198	0	0	0	0	0	0	0	10	0
372	0	0	0	1	0	0	4	2	62

Table 79. Confusion Matrix for window size 15 run 8

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	0	0	0	0	0	0	1	5
112	11	35	4	7	0	5	3	0	0
122	0	0	12	0	5	0	6	0	0
128	9	0	0	74	0	0	0	0	0
141	1	6	0	5	60	0	0	0	0
154	15	4	1	10	1	67	2	0	0
175	0	0	14	0	0	0	28	0	0
198	0	0	0	0	0	0	0	10	0
372	10	1	0	0	0	0	0	1	57

Table 80. Confusion Matrix for window size 15 run 9

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	20	3	0	1	0	1	0	0	7
112	1	41	1	7	8	7	0	0	0
122	0	0	21	0	0	0	2	0	0
128	2	1	0	64	0	16	0	0	0
141	0	0	0	6	64	0	0	2	0
154	0	5	2	7	0	86	0	0	0
175	0	5	3	0	0	1	33	0	0
198	0	1	0	0	0	0	0	9	0
372	0	1	0	0	0	2	0	1	65

Table 81. Confusion Matrix for window size 15 run 10

H. CONFUSION MATRIX FOR WINDOW SIZE 20

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	28	0	0	0	7	0	0	1	6
112	4	43	5	9	8	4	7	2	2
122	0	0	21	0	0	0	10	0	0
128	0	12	1	71	21	1	0	0	0
141	0	0	0	0	87	0	0	4	0
154	1	7	3	6	0	106	2	1	2
175	0	0	5	0	0	0	49	1	0
198	0	0	0	0	0	0	0	15	0
372	0	1	0	0	0	0	0	0	88

Table 82. Confusion Matrix for window size 20 run 1

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	28	0	0	0	6	0	0	2	6
112	0	42	1	5	6	27	0	2	1
122	0	0	19	0	0	5	7	0	0
128	3	4	1	52	28	17	1	0	0
141	0	1	0	0	90	0	0	0	0
154	1	8	2	1	1	114	0	0	1
175	0	4	7	0	0	0	44	0	0
198	0	0	0	0	0	0	0	15	0
372	3	1	0	0	0	0	0	3	82

Table 83. Confusion Matrix for window size 20 run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	1	0	0	6	1	0	2	7
112	4	49	11	11	1	2	0	6	0
122	0	0	20	0	0	0	11	0	0
128	0	7	6	56	27	10	0	0	0
141	0	13	0	0	75	0	0	3	0
154	7	40	2	3	1	72	1	2	0
175	0	4	3	0	0	6	42	0	0
198	0	0	0	0	0	0	0	15	0
372	0	1	0	0	0	2	0	3	83

Table 84. Confusion Matrix for window size 20 run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	23	0	0	1	0	0	0	0	8
112	2	40	1	10	8	3	0	0	1
122	0	0	22	0	0	0	1	0	0
128	1	1	0	49	27	4	0	0	1
141	0	0	0	0	70	0	0	2	0
154	3	3	4	5	0	83	1	0	1
175	0	2	16	0	0	0	24	0	0
198	0	1	0	0	0	0	0	9	0
372	1	2	2	0	0	0	0	0	64

Table 85. Confusion Matrix for window size 20 run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	0	0	2	0	0	0	1	2
112	2	34	6	7	7	3	2	4	0
122	0	0	18	0	0	0	5	0	0
128	0	4	8	65	0	6	0	0	0
141	0	0	0	6	66	0	0	0	0
154	8	8	3	6	0	73	1	1	0
175	0	2	5	0	0	0	34	1	0
198	0	0	0	0	0	0	0	10	0
372	6	0	0	0	0	1	0	4	58

Table 86. Confusion Matrix for window size 20 run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	0	0	0	0	3	0	0	2
112	0	30	1	10	7	7	9	0	1
122	0	0	23	0	0	0	0	0	0
128	0	8	0	70	0	5	0	0	0
141	0	0	0	6	64	0	0	2	0
154	0	4	1	5	1	87	1	0	1
175	0	4	14	0	0	4	20	0	0
198	1	0	0	0	0	0	0	9	0
372	8	0	0	0	4	0	0	1	56

Table 87. Confusion Matrix for window size 20 run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	0	0	0	0	1	0	0	6
112	1	43	8	3	8	2	0	0	0
122	0	0	18	0	0	0	5	0	0
128	0	26	0	49	2	6	0	0	0
141	0	0	0	3	69	0	0	0	0
154	0	6	2	3	1	88	0	0	0
175	0	0	4	0	0	4	34	0	0
198	0	0	0	0	0	0	0	10	0
372	6	0	0	0	0	0	0	1	62

Table 88. Confusion Matrix for window size 20 run 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	25	0	0	2	0	0	0	1	4
112	2	40	0	17	0	1	4	1	0
122	0	0	13	0	0	0	10	0	0
128	3	7	0	73	0	0	0	0	0
141	2	4	0	0	64	0	0	2	0
154	1	6	1	10	0	82	0	0	0
175	0	0	0	0	0	0	42	0	0
198	0	0	0	0	0	0	0	10	0
372	4	1	0	0	0	0	0	0	64

Table 89. Confusion Matrix for window size 20 run 8

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	21	0	0	0	0	1	0	1	9
112	7	39	0	9	5	5	0	0	0
122	0	1	20	0	0	0	2	0	0
128	30	4	0	48	0	1	0	0	0
141	6	0	0	0	64	0	0	2	0
154	11	3	5	5	0	75	1	0	0
175	0	4	15	0	0	0	23	0	0
198	0	0	0	0	0	0	0	10	0
372	0	1	0	0	0	0	0	1	67

Table 90. Confusion Matrix for window size 20 run 9

	Inferred Labels								
Truth	111	112	122	128	141	154	175	198	372
111	26	0	0	1	0	1	0	0	4
112	5	31	0	13	7	6	0	1	2
122	0	0	22	0	0	0	1	0	0
128	0	12	0	71	0	0	0	0	0
141	1	0	0	5	64	0	0	2	0
154	0	14	0	7	0	77	0	0	2
175	0	4	10	0	0	0	28	0	0
198	0	0	0	0	0	0	0	10	0
372	2	0	0	0	0	0	0	1	66

Table 91. Confusion Matrix for window size 20 run 10

I. CONFUSION MATRIX FOR 50% TRAINING, 50% TEST

	Inferred Labels								
Truth	111	112	122	128	141	154	175	198	372
111	16	5	0	0	3	1	0	1	9
112	0	42	10	0	4	4	8	0	0
122	0	0	17	0	0	0	9	0	0
128	0	45	8	28	1	4	0	0	0
141	0	1	0	2	70	0	0	2	0
154	0	8	5	0	1	89	0	0	0
175	0	0	3	0	0	0	42	0	0
198	0	1	0	0	0	0	0	12	0
372	0	0	0	0	0	5	0	2	65

Table 92. Confusion Matrix for 50% training, 50% test run 1

	Inferred Labels								
Truth	111	112	122	128	141	154	175	198	372
111	29	0	0	0	0	2	0	1	3
112	2	32	1	14	2	8	9	0	0
122	0	0	10	9	0	0	7	0	0
128	5	2	0	77	0	2	0	0	0
141	4	10	0	2	57	0	0	2	0
154	0	7	1	8	1	85	1	0	0
175	0	0	11	0	0	0	34	0	0
198	0	0	0	0	0	0	0	13	0
372	3	0	0	0	1	0	0	0	68

Table 93. Confusion Matrix for 50% training, 50% test run 2

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	1	0	5	0	0	0	0	5
112	0	57	9	0	0	0	0	1	1
122	0	0	25	0	0	0	1	0	0
128	0	25	8	53	0	0	0	0	0
141	0	10	0	6	57	0	0	2	0
154	0	29	5	0	0	69	0	0	0
175	0	7	31	0	0	0	7	0	0
198	0	1	0	0	0	0	0	12	0
372	5	5	0	0	0	0	0	1	61

Table 94. Confusion Matrix for 50% training, 50% test run 3

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	0	0	2	0	0	0	0	9
112	4	45	6	7	0	3	0	2	1
122	0	0	26	0	0	0	0	0	0
128	1	8	0	68	0	6	0	0	3
141	0	10	0	5	60	0	0	0	0
154	42	3	2	7	1	47	0	0	1
175	0	0	18	0	0	0	26	1	0
198	1	0	0	0	0	0	0	12	0
372	4	0	0	0	0	0	0	1	67

Table 95. Confusion Matrix for 50% training, 50% test run 4

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	1	0	0	0	0	0	0	4
112	9	29	4	0	5	15	2	1	0
122	0	0	17	0	0	0	6	0	0
128	1	47	0	31	0	4	0	0	0
141	0	0	0	6	66	0	0	0	0
154	1	12	1	0	2	82	2	0	0
175	0	2	8	0	0	4	28	0	0
198	0	0	0	0	0	0	0	10	0
372	1	0	0	0	0	0	0	1	67

Table 96. Confusion Matrix for 50% training, 50% test run 5

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	2	0	0	0	1	0	0	5
112	0	48	9	0	0	2	6	0	0
122	0	0	21	0	0	0	2	0	0
128	0	26	8	49	0	0	0	0	0
141	0	10	0	6	54	0	0	2	0
154	0	12	5	0	0	83	0	0	0
175	0	0	19	0	0	0	23	0	0
198	0	0	0	0	0	0	0	10	0
372	8	5	0	0	0	0	0	1	55

Table 97. Confusion Matrix for 50% training, 50% test run 6

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	27	0	0	0	0	1	0	0	4
112	2	29	9	6	8	11	0	0	0
122	0	0	17	0	0	0	6	0	0
128	2	0	8	58	0	15	0	0	0
141	1	0	0	5	63	0	0	3	0
154	1	0	5	1	0	93	0	0	0
175	0	4	10	0	0	0	28	0	0
198	0	0	0	0	0	1	0	9	0
372	3	0	0	1	0	4	0	1	60

Table 98. Confusion Matrix for 50% training, 50% test run 7

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	24	0	0	0	0	2	0	0	6
112	0	47	9	2	1	5	0	1	0
122	0	0	20	0	0	0	3	0	0
128	0	1	8	68	0	6	0	0	0
141	2	3	0	4	63	0	0	0	0
154	0	8	6	1	1	83	1	0	0
175	0	1	2	0	0	0	39	0	0
198	0	1	0	0	0	0	0	9	0
372	3	0	0	0	4	1	0	2	59

Table 99. Confusion Matrix for 50% training, 50% test run 8

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	19	3	0	2	0	0	0	0	8
112	5	32	8	8	8	0	4	0	0
122	0	0	15	0	0	0	8	0	0
128	2	2	2	77	0	0	0	0	0
141	0	0	0	6	64	0	0	2	0
154	5	7	4	10	0	74	0	0	0
175	0	0	2	0	0	0	40	0	0
198	0	1	0	0	0	0	0	9	0
372	5	1	0	0	0	0	0	1	62

Table 100. Confusion Matrix for 50% training, 50% test run 9

	Inferred Labels								
Truth	**111**	**112**	**122**	**128**	**141**	**154**	**175**	**198**	**372**
111	26	1	0	0	0	1	0	2	2
112	1	44	0	3	1	7	7	1	1
122	0	0	18	0	0	0	5	0	0
128	0	6	0	66	0	11	0	0	0
141	0	2	0	6	61	0	0	3	0
154	2	3	1	6	1	86	0	0	1
175	0	4	0	0	0	0	38	0	0
198	0	0	0	0	0	0	0	10	0
372	3	1	0	0	4	0	0	2	59

Table 101. Confusion Matrix for 50% training, 50% test run 10

LIST OF REFERENCES

[1] P. Enge and P. Misra, "Special issue on Global Positioning System," *Proc. IEEE*, vol. 87, no. 1, pp. 3-15, January 1999.

[2] G. Naik *et al.*, "Comparison of RSSI techniques in Wireless Indoor Geolocation," *2012 Nat. Conf. Computing and Commun. Syst.*, Durgapur, 2012, pp. 1-5.

[3] (2013, Mar. 15) *Skyhook Wireless, Inc*. [Online]. Available: https://www.skyhookwireless.com

[4] J. Picard and A. Weiss, "Theoretical facts on RSSI-based geolocation," *2012 IEEE 27th Conv. Elect. & Electron. Eng.*, Eilat, 2012, pp. 1-5.

[5] D. Ashbrook and T. Starner, "Using GPS to learn significant locations and predict movement across multiple users," *Personal and Ubiquitous Computing*, vol. 7, no. 5, pp. 275-286, 2003.

[6] L. Liao *et al.*, "Extracting places and activities from GPS traces using hierarchical conditional random fields," *Int. J. Robotics Research*, vol. 26, no. 1, pp. 119-134, Jan. 2007.

[7] Y. de Montjoye *et al.*, "Unique in the Crowd: The privacy bounds of human mobility," *Nature srep.*, vol. 3, Mar. 2013.

[8] A. Alvarez-Alvarez *et al.*, "Human Activity Recognition applying Computational Intelligence techniques for fusing information related to Wi-Fi positioning and body posture," *2010 IEEE Int. Conf. Fuzzy Systems*, Barcelona, 2010, pp. 1-8.

[9] S. Harris, *All in One CISSP Exam Guide*. 6th ed., P. Kumar, Ed. New York: McGraw Hill, 2013.

[10] J. Yang *et al.*, "Accuracy characterization of cell tower localization," *12th ACM Int. Conf. Ubiquitous Computing*, Copenhagen, 2010, pp. 223-226.

[11] (2013, May 5) *Wi-Fi manager: Android developers*. [Online]. Available: http://developer.android.com/reference/android/net/wifi/wifimanager.html

[12] (2013, Jun 20) *Wi-Fi analyzer*. [Online]. Available: http://www.farproc.com/wifi-analyzer

[13] S. Russell and P. Norvig, *Artificial Intelligence, a modern approach*. 3rd ed., M. Hirsch, Ed. Upper Saddle River: Pearson Education, Inc., 2010.

[14] L. Rabiner and B. Juang, "An introduction to Hidden Markov Models," *ASSP Magazine*, vol. 3, no. 1, pp. 4–16, Jan 1986.

[15] M. Sokolova *et al.*, “Beyond Accuracy, F-Score and ROC: A family of discriminant measures for performance evaluation,” *AI 2006: Advances in Artificial Intelligence*, A. Sattar and B. Kang, Eds.: Springer Berlin Heidelberg, 2006, pp. 1015-1021.

[16] (2014, Feb. 10). *Naval Postgraduate School* [Online]. Available: http://www.nps.edu/Images/Docs/NPS_Map_05-16.pdf

[17] N. Aharony *et al.*, “Social fMRI, investigating and shaping social mechanism in the real world,” *Pervasive and Mobile Computing*, vol. 7, no. 6, pp. 643-659, Dec. 2011.

[18] I. Duff *et al.*, “Sparse matrix test problems,” *ACM Trans. Math. Software*, vol. 15, no. 1, pp. 1-14, Mar. 1989.

[19] R. Floyd and J. Bentley, “Programming pearls: a sample of brilliance,” *Communications of the ACM*, vol. 30, no. 9, pp. 754-757, Sept. 1987.

[20] F. Pedregosa *et al*., “Scikit-learn: machine learning in python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.

[21] W. Parker, “Evaluation of data processing techniques for unobtrusive gait authentication,” M.S. thesis, Dept. Comput. Sci., Naval Postgraduate School, Monterey, CA, 2014.

[22] S. Fleming, “Identification of a smartphone user via keystroke analysis,” M.S. thesis, Depart. Comput. Sci., Naval Postgraduate School, Monterey, CA, 2014.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California