8-2012

Track Prediction for Border Security Using a Discrete Kalman Filter

Jonathan William Hickerson
jhicker2@utk.edu

To the Graduate Council:

I am submitting herewith a thesis written by Jonathan William Hickerson entitled "Track Prediction for Border Security Using a Discrete Kalman Filter." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Engineering.

Qing Cao, Major Professor

We have read this thesis and recommend its acceptance:

Lynne E. Parker, Hairong Qi

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Track Prediction for Border Security Using Kalman Filtering

A Thesis Presented for the
Master of Science Degree
The University of Tennessee, Knoxville

Jonathan William Hickerson
August 2012

ACKNOWLEDGEMENTS

Many people have made this possible and are owed a great deal of gratitude. My advisor, Dr. Qing Cao has been an enormous help throughout graduate school and the preparation of this thesis, as have my committee members, Dr. Lynne Parker and Dr. Hairong Qi.

I also owe a great deal of gratitude to my colleagues and mentors at Oak Ridge National Laboratory, including Jim Younkin, Nathan Rowe, Brad Stinson, Mike Kuhn, and Chris Pickett, without whom I would not have had the resources at my disposal to perform this research.

I also want to thank my parents for their love and support throughout my educational career, and my fiancée, Katie, for all of her support.

ABSTRACT

Border security systems are a critical component in the prevention of the transnational trafficking of unlawful goods. For economic, environmental, and public safety related reasons, it is obviously desirable to prevent the transfer of materials such as illicit drugs, nuclear material, and unlawfully present individuals across borders; however, the deployment of trafficking countermeasures is a daunting task, wrought with unique challenges, such as unfamiliar terrain, long distances, and a potentially harsh environment.

The purpose of this thesis is to suggest a method to combine three different types of sensor systems to accurately track an individual as he crosses a field of these sensors as well as to accurately predict the path that the individual will take beyond the scope of these sensors. Such a system could be used protect a border from trafficking by enabling the interception of those that would try to cross unauthorized.

The combination of the data from these sensors is accomplished using a modified Kalman filter and algorithms used in other multisensor tracking systems. While the field of multisensory tracking has been explored fairly extensively, these algorithms have not been applied to the sensors available for this research.

After the theoretical application of these algorithms is discussed, experimental data is presented to clarify the benefits and disadvantages of applying this paradigm to this set of sensors.

TABLE OF CONTENTS

CHAPTER I Introduction and General Information....1
Border Security....1
Goals....2
System Overview....2
Organization of the Thesis....3
CHAPTER II Literature Review....5
Sensor Noise Filtering and the Kalman Filter....5
Multisensor Tracking....6
Sensor Systems and the Test Bed....7
Optasense Fiber Optic Acoustic Sensor....10
Southwest Microwave Electromagnetic Interference Sensor....10
Textron Wireless Seismic Sensor....11
CHAPTER III Materials and Methods....13
The Kalman Filter Algorithm....13
Implementation Overview....13
Combining Data from Multiple Sensors....16
Implementation Specifics....18
Experimental Procedure....22
Data Simulation....22
Data Collection....23
Kalman Filter Algorithm Implementation....24
CHAPTER IV Results and Discussion....26
Runs 1-10....27
Runs 11-20....33
Runs 21-30....39
Simulated Data Runs....45
Figure 14 - Projected Actual Path and Predicted Path for a Simulated Run....47
Analysis....49
A Possible Better Approach – Predicted Velocity Averaging....49
Overall Prediction Accuracy....52
Simulated Results Observations....52
CHAPTER V Conclusions and Recommendations....54
Future Work and Suggestions....54
Conclusion....55
LIST OF REFERENCES....56
APPENDIX....59
Matlab Code....60
Measurement Data....74
Instructions....74
Run 1....74
Run 2....75

Run 3 76
Run 4 77
Run 5 77
Run 6 79
Run 7 80
Run 8 80
Run 9 81
Run 10 82
Run 11 83
Run 12 83
Run 13 84
Run 14 85
Run 15 85
Run 16 86
Run 17 86
Run 18 86
Run 19 87
Run 20 87
Run 21 88
Run 22 88
Run 23 88
Run 24 89
Run 25 89
Run 26 89
Run 27 89
Run 28 90
Run 29 90
Run 30 90
Vita 91

LIST OF TABLES

Table | Page

Table 1. Future Predicted Path Error for Runs 1-10..32
Table 2 Future Predicted Path Error for Runs 11-20...38
Table 3. Future Predicted Path Error for Runs 21-30...44
Table 4 - Simulated Data Error 48
Table 5. Path Prediction Improvement...51

LIST OF FIGURES

Figure 1 - Sensor Test Bed at Oak Ridge National Laboratory 9
Figure 2 - Cross Section of the Southwest Microwave System [6] 11
Figure 3 - Kalman Filter Algorithm Prediction/Correction Cycle [1] 16
Figure 4 - Path Walked By Subject in Runs 1 - 10 28
Figure 5 - The Actual Path, Predicted Path, and Measurements for Run 1 30
Figure 6 - Projected Actual Path and Predicted Path for Run 1 31
Figure 7 - Path Walked By Test Subject for Runs 11-20 34
Figure 8 - The Actual Path, Predicted Path, and Measurements for Run 13 36
Figure 9 - Projected Actual Path and Predicted Path for Run 13 37
Figure 10 - Path Walked By Subject in Runs 1 - 10 40
Figure 11 - The Actual Path, Predicted Path, and Measurements for Run 27 42
Figure 12 - Projected Actual Path and Predicted Path for Run 27 43
Figure 13 - The Actual Path, Predicted Path, and Measurements for a Simulated Run 46
Figure 14 - Projected Actual Path and Predicted Path for a Simulated Run 47

CHAPTER I

INTRODUCTION AND GENERAL INFORMATION

Border Security

Cooperative border security relies on a combination of technology and the active participation of a state to prevent the trafficking of illegal goods across borders. This area is of special interest to those in the field of nuclear nonproliferation, where the transfer of radiological material across borders undermines the integrity of the systems in place to prevent the proliferation of nuclear weapons. This thesis research was started at Oak Ridge National Laboratory with this application in mind [4].

While preventing an agent from crossing a border may be a suitable solution to the issue of border security in some cases, creating sufficient barriers to entry at a border may be impractical. For this reason, it is desirable to create a solution that involves tracking and intercepting individuals crossing a border instead of depending upon prevention alone.

Throughout the process of designing a border security system, additional metrics have been identified as being important to the success of any particular security system. It is beneficial to minimize the time required to deploy a system and to integrate it into the preexisting border security infrastructure within a state. For this reason, it is highly desirable to use commercial, off-the-shelf hardware whenever possible. This generally increases the speed and ease with which a system may be deployed. The ability to sufficiently fulfill these requirements is what has led to the need to integrate the three sensor systems present in this research [3][4].

Goals

The goal of this research is to determine an accurate and efficient way to combine tracking data from three different types of ground sensors to track a target as it moves across a field of the sensors as well as to accurately predict the path the target will take beyond the field of the sensors so that an interception vector may be established. Tracking a target as it is moving across a field of sensors is largely trivial, however, predicting the path and velocity a target will take after leaving this field with any degree of certainty is a more complicated problem wherein lies the crux of this research.

System Overview

The approach used in this research to aggregate data from the three types of sensors and predict a track for someone crossing their field is based upon the Kalman Filter. The Kalman Filter is simply an estimator which uses system inputs and the previous Kalman Filter estimate to determine the most likely state of the system in the next time interval [1]. Computationally speaking, this type of filter is convenient since it does not need to store data relating to all previous system states; instead, the previous state is an accurate amalgam of every system input and prediction since the first moment the system is implemented.

In terms of this research, if someone is being tracked across a field of sensors, one only needs to keep track of the next position and velocity estimations and the current position measurement instead of keeping track of all past position measurements.

Using Kalman Filtering in this track creation and prediction system has several benefits, namely that prediction is already incorporated into the creation of a track. This is to say that once someone being tracked leaves the field of sensors, a prediction of his next position and velocity will already exist. This prediction may be extrapolated to provide a position and time at which the individual may be intercepted along with the variance of the prediction.

While it would be nice to have a real-time graphic user interface upon which to view path predictions for individuals crossing the sensors, this research focuses simply on the filtering methods needed to create an accurate prediction. The creation of an easy-to-use graphic user interface implementing the path prediction methods discussed in this paper should be implemented in the future if these methods prove to be useful.

Organization of the Thesis

This thesis consists of five chapters. The first chapter provides a high-level introduction to the problem in questions and the proposed solution. Chapter II presents a brief review of literature pertaining to this study and useful for a complete understanding of it. The specifics of the methods and algorithms implemented within this study are presented in Chapter III. The results of implementing and testing these methods are described in Chapter IV, and Chapter V presents final conclusions and recommendations regarding the methods used in this study as well as areas of interest for future work on the subject.

CHAPTER II

LITERATURE REVIEW

Multisensor tracking and noise filtering have existed for more than thirty years [1]. Because of this, a great deal of material already exists on these topics which proved to be very valuable for the sake of this project. What follows is a brief summary of the papers and literature from which guidance and knowledge was drawn for this research and which should give the reader an adequate understanding so as to be able to follow the methods of the research in this thesis.

Sensor Noise Filtering and the Kalman Filter

Because the sensors being used in this research are not perfect instruments, there will always be a degree of noise in whatever measurements are achieved by their use. The goal of filtering is to eliminate this noise as much as possible to achieve the best possible picture of what is actually going on in the sensor system. The Kalman Filter is the optimal filtering tool for a large class of problems [1].

Material from the University of North Carolina at Chapel Hill entitled "An Introduction to the Kalman Filter" aided in achieving a suitable understanding of how a Kalman filter operates, how it is applicable to this research, and how to create a simple Kalman filter implementation [1]. Much of the material related to multitarget multisensor tracking takes as granted the knowledge of how a Kalman filter works, and the author of this thesis, who previously did not have such knowledge, found this resource to be very helpful.

A tremendous amount of additional information regarding the Kalman Filter and other types of filters for dealing with stochastic processes is available since the field has existed for multiple decades. Sources [11], [12], [13], and [14] all provide useful insight into the operation of this filter, and while numerous other sources exist regarding the topic, the author of this thesis has found these papers to be the most useful in terms of understanding the operation of the Kalman Filter.

Multisensor Tracking

One of the primary concerns of this thesis is how to accurately combine measurements from multiple sensors to achieve a single measurement that is more accurate than any of the individual sensor measurements. Similar problems have been dealt with in the past in great depth.

On resource that was useful in this pursuit has been Yaakov Bar-Shalom's series of books on the topic [2][7]. These books go into great depth regarding the different ways one might combine data from multiple sensors and the advantages and disadvantages of each. One of the main pieces of information gleaned from these works is the general truth that it is more effective to combine information from sensors at a central location and then create a track rather than creating individual tracks at each sensor and then trying to combine them. This knowledge was useful because two of the three sensors used in this research have the capability of creating a track for an individual crossing them. This capability was largely forsaken by this thesis in favor of taking the raw sensing data and generating a custom track from it. This approach should yield the best results.

Many additional papers exist and were perused before the creation of the system used in this thesis. While information from some of these sources was not used directly, useful information and understanding was extracted from each of them. These include [8], [9], and [10]. It is also worth noting that a great number of these papers deal with the issue of multitarget tracking, which is a subject this thesis does not deal with but which likely should be dealt with before implementing a system such as this in the field.

Sensor Systems and the Test Bed

This research centers on a border monitoring system which combines three types of ground sensors. These sensors are commercial, off-the-shelf products, which is desirable because they are less expensive than proprietary sensor systems and because they generally implement established protocols for communication, making it easier to aggregate their data. This section is divided into sections for the three types of sensors, although some sources contain valuable information for more than one of them. Additionally, some of the sources used to understand these sensors, especially their communication protocols, are not publically available, and the author may not disclose them to the reader, although the vender may provide them if asked.

An aerial picture of the test bed where these sensors are arranged with the sensors superimposed on the picture can be seen in Figure 1. While this test bed is small, it has plenty of space to adequately test the performance of the sensors and their abilities under different conditions. This test bed is located at Oak Ridge National Lab and is dedicated almost entirely to border security.

Figure 1 - Sensor Test Bed at Oak Ridge National Laboratory

Optasense Fiber Optic Acoustic Sensor

The Optasense Fiber Optic Acoustic Sensor (FOAS) uses a buried fiber optic cable to detect the presence of seismic or acoustic disturbances in the vicinity of the sensor. The Optasense Distributed Acoustic Sensing technology works by monitoring light traveling through the fiber optic cable. Impurities in the cable cause the light to be scattered or reflected in a certain way. When the cable is stressed by a seismic or acoustic disturbance, the way the light is scattered is changed. Using coherent optical time domain reflectometry, the FOAS system uses multiple light pulses to detect phase shifts caused by this strain and determines where along the cable a disturbance has taken place [5][6]. This system has shown to have a very high probability of detection for someone walking across it even if they are trying to move stealthily. In testing at Oak Ridge National Laboratory, no test subject was able to circumvent detection by this system by moving stealthily [6]. The advertising literature for this system is useful for understanding its capabilities [15].

Southwest Microwave Electromagnetic Interference Sensor

The second type of sensor utilized in this research is the Southwest Microwave Electromagnetic Interference Sensor. This sensor uses two buried coaxial cables to create an electromagnetic field. Whenever something disturbs this field, the position of the disturbance along the cables is reported. The cable is divided into cells of about 1.5 meters length, so the resolution of the of the crossing individual's position is limited by that fact. Information regarding the performance of this system was achieved through [4] and [6], and again, a general overview of the system's capabilities is available in literature from Southwest Microwave [16]. A depiction of a cross section of the system can be seen in Figure 2.

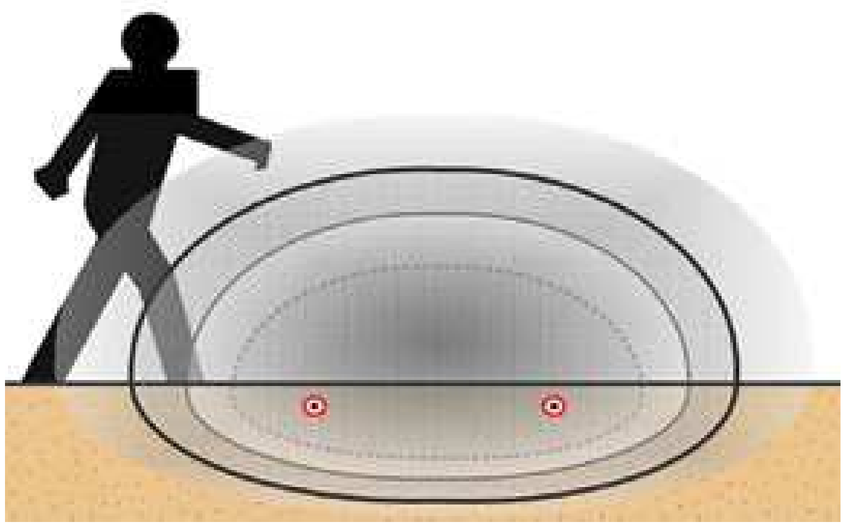

Figure 2 - Cross Section of the Southwest Microwave System [6]

Textron Wireless Seismic Sensor

The Textron Wireless Seismic Sensor utilizes a series of individual buried nodes to sense an individual walking among them or any other kind of seismic activity. This sensor is unique compared to the other two in that it operates as a number of unique nodes instead of a buried cable. This has several benefits and drawbacks compared to the other sensors. Each node listens for seismic disturbances, which could indicate that someone has entered the detection radius of the node. Any detected disturbances are reported wirelessly to a “gateway” node, which may in turn relay the information to a computer and to software for analysis. According to [4] and [6], there is a delay of around 15 seconds between the time activity occurs around one of these sensors and the time it is reported. This delay needs to be taken into account for any system seeking to harmonize the data of this sensor with any other sensors. Additional information regarding the Textron Wireless Seismic Sensor is available at [17].

CHAPTER III
Materials and Methods

The Kalman Filter Algorithm

The knowledge and information disseminated in this section was achieved primarily through sources [1], [11], [12], and [13]. If specific pieces of information are derived from a specific work, they will be cited; any other information that would not be considered general knowledge is likely common to all of these sources. Variable names in equations do vary between sources, and this paper uses the notation of [1].

Implementation Overview

The Kalman filter is a recursive filtering algorithm designed to determine the actual state of a system given some number of noisy measurements relating to the state of that system. This set of equations essentially operates by keeping track of the best state estimate it has up to the present, estimating the state in the next time step, and then comparing the state measured by sensors to the predicted state. By weighting the validity of the predicted state and the measured state, one may adequately account for noise in sensor measurements and achieve a more accurate idea of the actual system state.

The Kalman filter consists of two distinct and repeated steps, state prediction and state correction. First, a state prediction is derived from the previous corrected state prediction. This is modeled in the following equation:

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \qquad (1)$$

Where x_k represents the prediction of the next step, x_{k-1} represents the previous step, u_{k-1} represents a controlled input to the state of the system, A is a matrix relating the previous step to the current step, and B is a matrix relating the control input to the current step.

In the prediction step, it is also necessary to compute the predicted error covariance matrix for the current step. The error covariance matrix at time k represents the variance in each element of the state, x_k, as well as the covariance between the errors in each pair of two elements in the state. The equation for predicting this covariance matrix is:

$$P_k^- = AP_{k-1}A^T + Q \qquad (2)$$

In this equation, P_k is the error covariance matrix, A is the is the matrix relating the previous state to the current state (from equation 1), Q is the process noise covariance, a measure of the noise in the operation of the system and the change of the state moment-to-moment, and the T superscript represents the transposition of the matrix to which it is adjacent.

In the state correction step, the values achieved during state prediction are corrected using measurement data from sensors observing the state of the system. The first value to be calculated in this step is the Kalman Gain. This value is a representation of how heavily to weigh the data from sensors during this time step. This value is calculated using the following equation:

$$k_k = \frac{P_k^- H^T}{H P_k^- H^T + R} \qquad (3)$$

In this equation, the variables that have not already been discussed are k_k, the Kalman Gain, H, a matrix for mapping the state to the measured data, and R, the measurement noise covariance, which is similar to Q, but relating to the measurement noise rather than the state noise.

After calculating the Kalman Gain, the predicted state, $\hat{x}_k^-$, may be corrected using the Kalman Gain and the sensor data. This is achieved by using equation 4.

$$\hat{x}_k = \hat{x}_k^- + k_k (z_k - H\hat{x}_k^-) \qquad (4)$$

In this equation, the only variable that has not already been highlighted is z_k, which is a matrix of the state measurements for the current time increment. The difference between the measured state and the predicted state is multiplied by the Kalman Gain, and the sum is added to the prediction, correcting it in accordance with the weight of the state measurements.

Finally, the Kalman Gain may be used to correct the predicted covariance. This is necessary to include the measurement information in the formation of the next predicted covariance and state. The corrected covariance is calculated using equation 5, where I represents an identity matrix.

$$P_k = (I - K_k H) P_k^- \qquad (5)$$

This entire cycle may be best understood visually. The following chart, Figure 2, demonstrates the entire prediction/correction cycle [1].

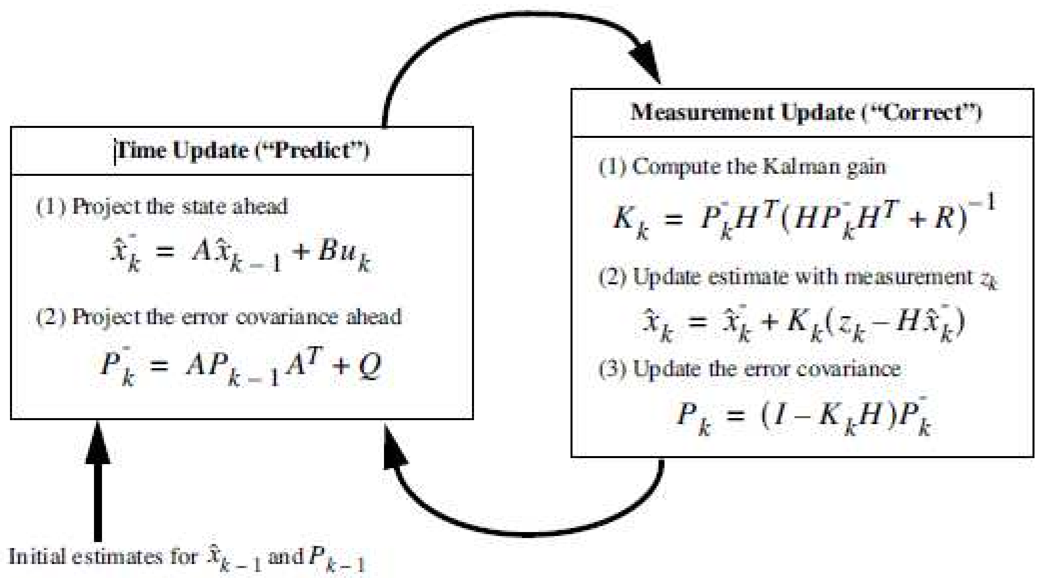

Figure 3 - Kalman Filter Algorithm Prediction/Correction Cycle [1]

Combining Data from Multiple Sensors

When dealing with noisy measurements, it is reasonable to think that given a larger number of measurements, a more accurate picture of the true system state may be determined. While this is true, actually combining the measurements may be complicated. This research deals with this problem by creating a weighted average of all of the measurements where they are weighted by their relative variance, as suggested in [11]. For instance, when fusing two measurements for latitude, the following equations would be used to determine the overall measurement and variance.

$$lat_{total} = \frac{\sigma_a^2 lat_b + \sigma_b^2 lat_a}{\sigma_a^2 + \sigma_b^2} \qquad (6)$$

$$\sigma_{total}^2 = \frac{\sigma_a^2 \sigma_b^2}{\sigma_a^2 + \sigma_b^2} \qquad (7)$$

In these equations, lat_a and lat_b are two individual latitude measurements, and σ^2_a and σ^2_b are their variances, respectively. The results, lat_{total} and σ^2_{total} represent the overall measurement and variance when these two measurements are combined. This allows each sensor to influence the gross measured state in accordance with how noisy that measured state is compared to the others.

If multiple measurements are available from the same sensor at a given time, σ^2_a and σ^2_b will be equal. This causes equation 6 to reduce to simply taking the average of the two measurements and leaving the variance unchanged, which is the intuitive solution to combining two equally reliable measurements from the same sensor.

While not all time steps will contain data from multiple sensors, if a single time step does contain multiple measurements, they are combined in this fashion and considered a single sensor state measurement for the remainder of the Kalmin Filter algorithm.

Implementation Specifics

While the previous section outlines the basic algorithm used to filter noisy data with the Kalman Filter, it would be of little use without the specific parameters used in this research. This section should resolve any questions to that regard.

Each of the parameters in the Kalman Filter algorithm centers on the definition of $x_{k,}$ the system state. The system in question for this research concerns itself with the movement of an individual in two dimensions, latitude and longitude. While the earth is a roughly elliptical object, and any individual moving along its surface is actually moving in a three dimensional arc, the definitions of latitude and longitude take these factors into account. Additionally, on any scale that will be relevant in this research (generally less than a few miles of movement), the individual might as well be moving on a flat surface, since ignoring the curvature of the earth does not lead to any significant error. With this said, x_k for this system is defined below.

$$x_k = \begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \\ y_k \\ \dot{y}_k \\ \ddot{y}_k \end{bmatrix}$$

In this definition, x and y represent the individual's position measured in latitude and longitude, respectively. The first and second derivatives of these values represent velocity and acceleration, respectively. Additionally, x_0 is initialized to be equal to the first measured data point with no velocity or acceleration. This initialization is not perfect, but it does not need to be. The algorithm will quickly adjust the predicted state in accordance with the additional measurements.

The most intuitive value to look at after recognizing the structure of x_k is A, the matrix which relates the previous state matrix to the current one. Because the system state is defined in terms of position, velocity, and acceleration, this matrix is derived from the Newtonian Mechanics equations which relate these terms.

$$p_t = p_0 + v_0 t + a_0 \frac{t^2}{2} \qquad (8)$$

$$v_t = v_0 + a_0 t^2 \qquad (9)$$

$$a_t = a_0 \qquad (10)$$

Based upon equations 8, 9, and 10, matrix A is defined below.

$$A = \begin{bmatrix} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The constant T in this matrix represents the time increment for the Kalman Filtering algorithm, which has been selected to be equal to 1 second.

The remaining terms in equation 1 are B and u_k, the control input. Because there is no control input to the movement of the individual in this research, $u_k=0$, and there is no reason to define B. This yields the specific form of equation 1 used in this thesis.

$$\begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \\ y_k \\ \dot{y}_k \\ \ddot{y}_k \end{bmatrix} = \begin{bmatrix} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ \dot{x}_{k-1} \\ \ddot{x}_{k-1} \\ y_{k-1} \\ \dot{y}_{k-1} \\ \ddot{y}_{k-1} \end{bmatrix} \tag{11}$$

One may easily verify that by multiplying out these matrices, equations 8, 9, and 10 are obtained for x and y. This represents an accurate depiction of how the system state at any particular time should relate to the state at the previous time interval.

P_k, the error covariance matrix, begins as a 6x6 matrix populated entirely by zeros. This is a justifiable starting point since it quickly adjusts and moves towards a constant value for the set of data.

The process noise covariance, Q, is a constant matrix representing the covariance in the movement of the person across the field of sensors. In reality, it is very difficult to know the exact value of this matrix, since the covariance of the noise between any two elements of the state is impossible to know. For the purposes of implementing a Kalman Filter, this value seems to be typically assigned a value and altered slightly until the performance of the system is adequate. For this implementation of the Kalman Filter Algorithm, Q is defined to be a 6x6 identity matrix multiplied by a constant 0.0001 degrees.

The matrix containing the measurement data, $z_{k,}$ contains only two values. This is due to the fact that only latitudinal and longitudinal positions are measured by the sensors in this system. The velocity and acceleration are not directly measureable in this case.

The matrix which maps the sensor data to the system state, H, is defined below.

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

This definition of H may seem confusing at first, but it is quite intuitive when you consider how it relates the measurement data to the system state in equation 12.

$$z_k = Hx_k \qquad (12)$$

$$z_k = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ \dot{x}_k \\ \ddot{x}_k \\ y_k \\ \dot{y}_k \\ \ddot{y}_k \end{bmatrix} \qquad (13)$$

The final parameter that has not been discussed is R, the measurement noise covariance. Because three unique sensors are being used with three distinct variances, there are actually three measurement noise covariance matrices, hereafter labeled RQ, RT, and RS for the QinetiQ (Optasense), Textron, and Southwest sensors, respectively. These are each 2x2 matrices, and are initialized as a scalar value multiplied by a 2x2 identity matrix. In each case, the scalar value selected for this thesis was 0.01 degrees. The sensor generally have similar noise, however, it may be worth varying these values in the future to see if more accurate results may be achieved. It is also worth noting that the particular values of R and Q are much less important than the ratio of R to Q. In fact, as long as the ratio remained constant, very similar results were obtained.

Experimental Procedure

The algorithm under discussion in this paper is meant to be implementable in real time or with a slight lag behind real time. Without this feature, it would not be useful for the applications discussed in the introduction. However, for the purposes of testing the algorithm extensively and being able to change input frequently, data was collected from the sensors en masse, and then Kalman Filtering was applied to it after the fact.

Data Simulation

To ensure that the methods used on this data are not fundamentally flawed, sensor data was simulated to perfectly match the implemented values of

R and Q. This was accomplished by declaring an actual path in Matlab, adding white Gaussian noise to it according to the value of Q, and creating sensor "measurements" by adding white Gaussian noise to the actual path according to the value of R. By treating these measurements as real and applying the Kalman filter to them, a decent understanding can be obtained of whether the methods described in this paper are useful and whether any apparent shortcomings may come from inappropriately estimating the values of Q and R for each of the sensors.

Data Collection

Data collection was achieved by recording data from each of the sensors simultaneously as an individual walked a fixed path through the test bed. The path of the individual was determined ahead of time, and his starting coordinates, ending coordinates, start time, and end times were recorded as accurately as possible. This information is important because it allows for the reconstruction of the actual path taken. Without knowledge of the actual path, the Kalman Filter results could not be usefully compared to anything. Unfortunately, the only reasonable way to measure the starting and ending coordinates is by using a handheld Global Positioning System (GPS). Since this is just another noisy measurement device, a degree of inaccuracy may occur in the individual's true path. In spite of this, it is possible to recreate the true path of the individual to gain a general idea of the accuracy of the system. Since the point of this system is to be able to predict the individual's path beyond the field of sensors, the ability to accurately determine the individual's true path is paramount.

The paths chosen for the individual to take in his traversal of the field were selected to adequately vary the sensor density in the path of the subject. Varying the sensor density is important because it will have a dramatic effect on the accuracy of the system as a whole. The sensor density is likely the single most important variable in terms of the accuracy of the system. Additionally, enough samples were taken to adequately see the effects of filtering the noisy sensor data for path prediction purposes.

Kalman Filter Algorithm Implementation

The Kalman Filter Algorithm was implemented in Matlab. Matlab was chosen for speed, simplicity, and understandability. The particular code used is available in the appendix and modeled after the example in [14].

The Matlab script takes input from three texts files each containing data from one of the three sensors. These files simply contain the name of the sensor, a time stamp of when the data was taken, and a pair of coordinates representing where the individual was detected. It also has variables set at the beginning of the script for each of the constants and variables discussed previously. While these are hard coded values, they are easily changeable to quickly modify the performance of the filter.

The filtering process is discussed above. The time increment selected is a single second. If sensors have data from the chosen time interval, it is used as the measurement for that instant. If no data is available for that instant, the previous estimate is used instead until the next instant for which sensor data is available. As filtering occurs, the measurements available and the state predictions from each instant are recorded for graphing.

Viewing the actual tracking data in any intelligible way is difficult. The Matlab script attempts this by graphing in three dimensions, with time on the x axis, latitude on the y axis, and longitude on the z axis. Plots for the actual path, the sensor measurements, and the predicted path are drawn on these axes for comparison. In a separate graph, which is drawn in the same manner, the projection of the predicted path and the projection of the true path are available for comparison. This final comparison is the subject of most of the scrutiny in the Conclusion section of this paper, since the ability to predict a path into the future is the desired deliverable from this thesis.

CHAPTER IV

RESULTS AND DISCUSSION

The results of the experimental procedures used to test this system will be evaluated in three groups according to the density of sensors available in the path of the test subject. In all, 30 runs were recorded and will be examined. For each series of runs, one run has been selected to represent the series visually, and for that run, the subject's actual path, Kalman Filter predicted path, and measurement coordinates will be examined. As mentioned previously, the actual path is determined by taking the starting and ending coordinates of the run with a handheld GPS. This path will then be extended five minutes past the end of the data collection period by extrapolating the actual path. For each of the measured runs, a three dimensional graph has been produced in Matlab to observe the data. While exact numbers are important to this analysis, the ability to visually evaluate the data allows for a more intuitive examination.

Incidentally, these runs were all performed under conditions involving wet soil. Previous research indicates that a minor decrease in sensor quality may occur under these conditions for the Textron seismic sensor but is unlikely to affect the Optasense or Southwest Microwave sensors at all.

In each graph in this section, the x axis represents the time in seconds measured from the beginning of the run. The y axis is latitude, measured in feet past 35 degrees 56 minutes north latitude. The z axis is the distance in feet measured past 84 degrees 16 minutes west longitude. The values along the z axis are negative in keeping with the convention of longitudes west of the Prime Meridian being negative. It is worth noting here that latitude and longitude do not uniformly convert to feet in different geographical locations, however, conversion factors for the coordinates of the test bed were calculated because displaying distances as degrees, minutes, and seconds or decimal degrees is so unintuitive.

The results are split into three sections according to the sensor density for that group of runs. If it is desirable for some reason to look at more graphs than the ones displayed in this section of this paper, the Matlab code, raw data, and instructions for running the Matlab script are all available in the Appendix.

Runs 1-10

These runs were performed in an area with high sensor density. A medium number of Textron seismic sensors were along the path, and the path ran parallel to both the Optasense and Southwest Microwave sensors. This allowed for all sensors to take many data points, giving generally favorable results. These runs consisted of an individual walking between the coordinates c_1, (35.93798 N, 84.27489 W), and c_2, (35.93833 N, 84.27476 W). In odd numbered runs, the subject progresses from c_1 to c_2, and in even numbered runs, the subject progresses from c_2 to c_1. The path of the subject through the field of sensors is available in Figure 4.

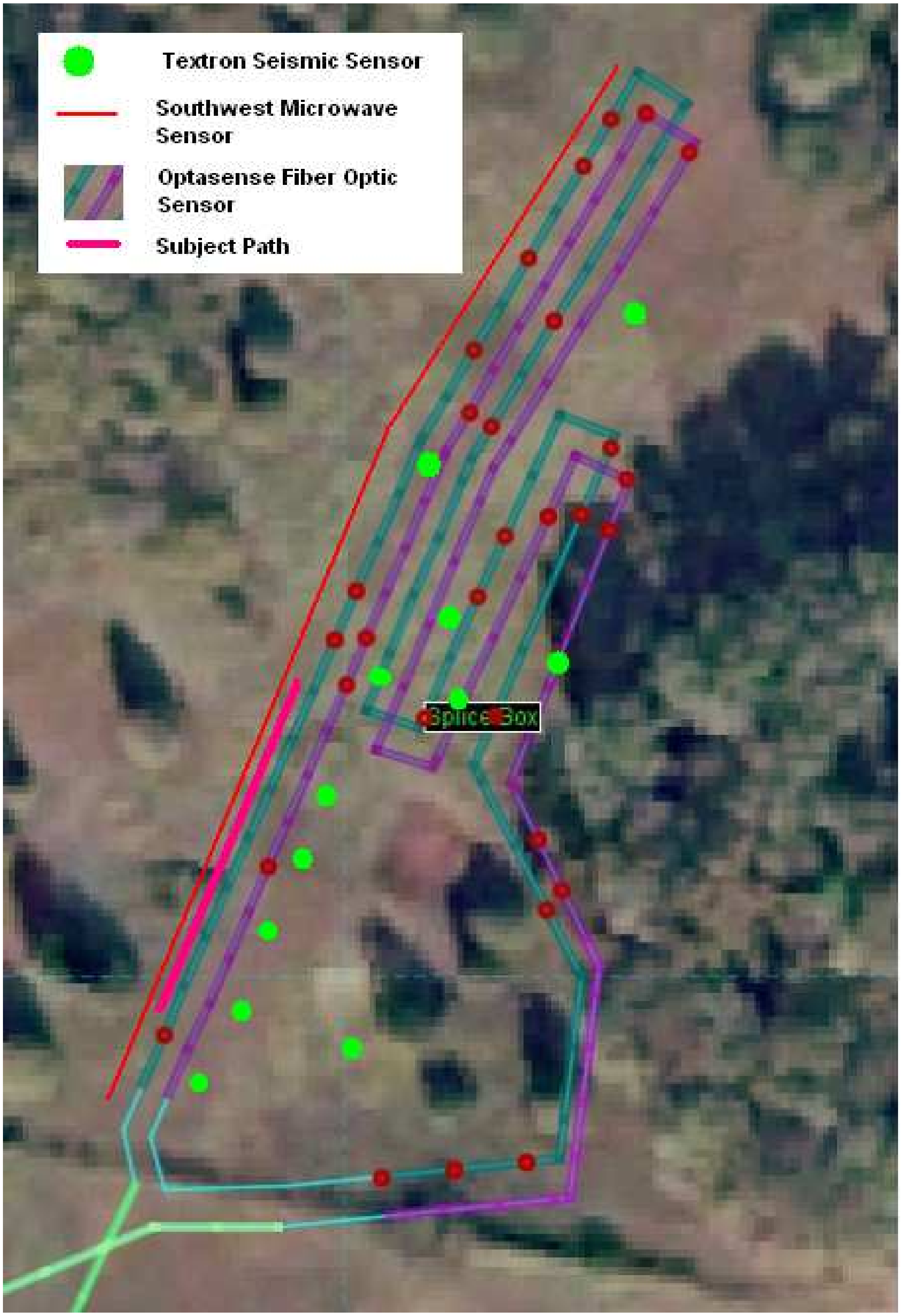

Figure 4 - Path Walked By Subject in Runs 1 - 10

In Figures 5 and 6, the results from Run 1 are displayed which was selected as a token sample of runs 1-10. In Figure 5, one may notice that while the sensor measurements are centered on the actual path, there is a significant amount of noise, which warps the predicted path significantly. It is possible that given a longer field of sensors and more measurements, this path would straighten out, but this is not apparent in this run.

One would expect that the predicted future path of these runs would be very accurate due to the abundance of sensor measurements; however, this is not necessarily the case, as shown in Figure 6. After five minutes beyond the time of the last measurement, the predicted path is a full 2467 feet away from the actual path, more than half a mile.

Table 1 displays the amount of error in each run when the predicted path has been projected 5 minutes past the time of the last measurement (an approximate time for the interception of an individual) as well as the angle between the predicted path and the actual path, as well as the error in the speed of the predicted as compared to the speed of the actual path. There is a large amount of variance in this amount of error between runs. Additionally, it appears as though a much more accurate prediction occurs when the path of the subject is from c_2 to c_1 rather than vice versa. On average, the amount of error in the angle of these measurements is 18.3°, and the avera ge amount of error in the speed was 5.465 ft/s.

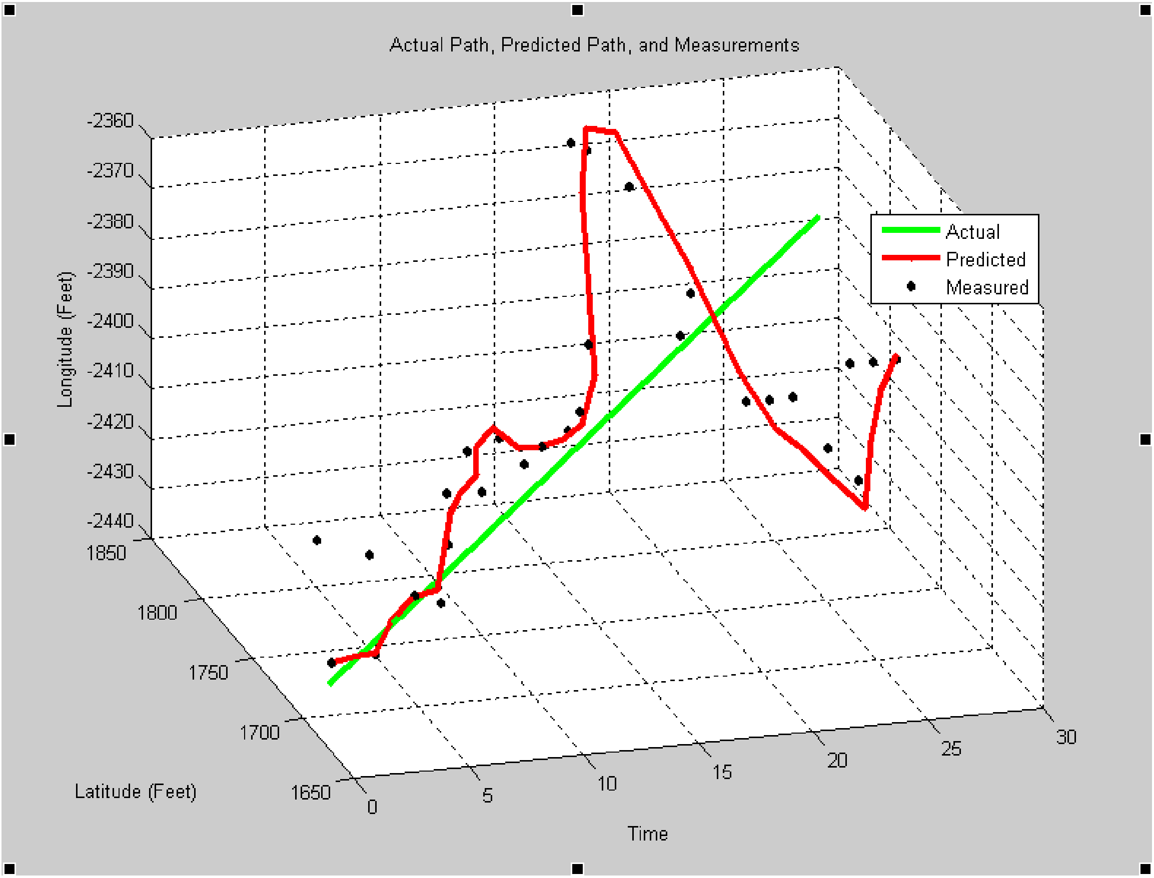

Figure 5 - The Actual Path, Predicted Path, and Measurements for Run 1

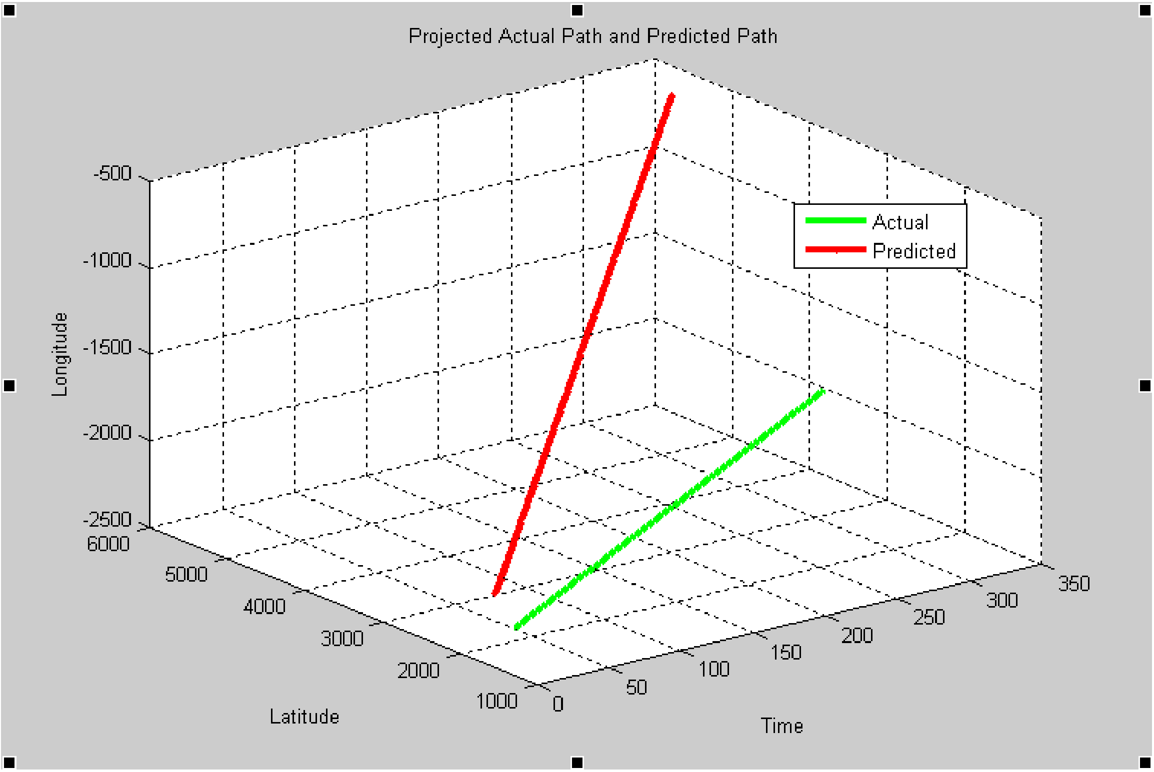

Figure 6 - Projected Actual Path and Predicted Path for Run 1

Table 1. Future Predicted Path Error for Runs 1-10.

Run Number	*Error After Five Minutes (Feet)*	*Angle Error (Degrees)*	*Speed Error (Feet per Second)*
Run 1	2730	5.777	6.801
Run 2	879	29.510	4.645
Run 3	5172	7.392	26.376
Run 4	1838	36.824	1.196
Run 5	3854	1.426	0.403
Run 6	2810	5.570	4.944
Run 7	4124	13.092	2.060
Run 8	1023	27.709	3.863
Run 9	3313	50.898	2.436
Run 10	1266	4.805	1.930
Average	2700.9	18.30	5.465

Runs 11-20

These runs were performed in an area with medium sensor density. A sufficient number of Textron sensors were available to record data, and the Optasense sensor doubles back on itself multiple times, creating several points where good measurements can be made. The Southwest Microwave sensor runs perpendicular to the path of the subject in these cases, so it is only able to provide a few measurements. These runs consisted of an individual walking between the coordinates c_1, (35.93851N, 84.27474W), and c_2, (35.93837 N, 84.17453 W). In odd numbered runs, the subject progresses from c_1 to c_2, and in even numbered runs, the subject progresses from c_2 to c_1. The path of the subject through the field of sensors is available in Figure 7.

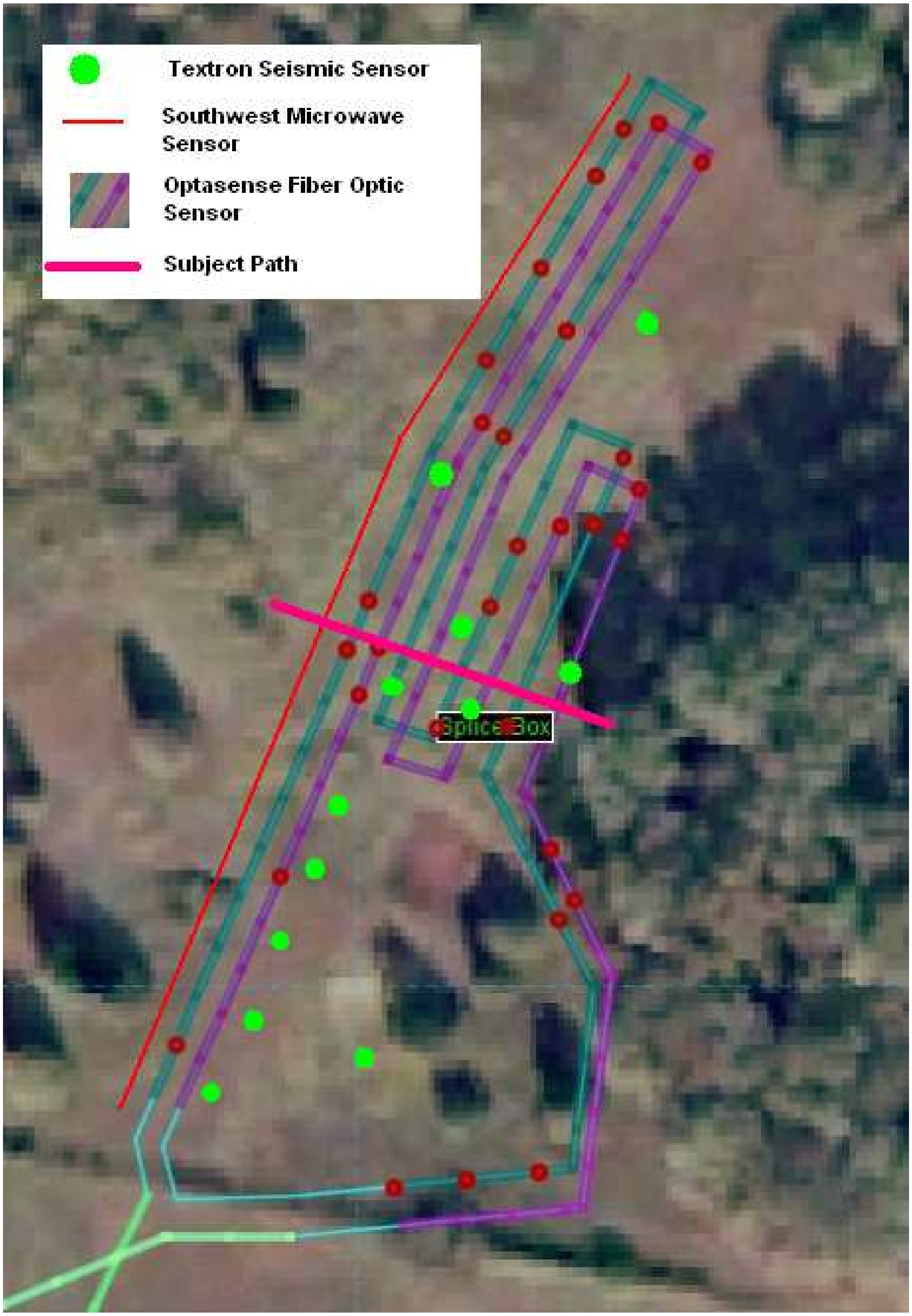

Figure 7 - Path Walked By Test Subject for Runs 11-20

The specific runs from this group are interesting. Again, a few of them will be pulled out for examination. In Figure 8, it appears that the predicted path for Run 13 approximates the actual path reasonably well. There are a few measurements that are significantly erroneous, but the predicted path is not significantly deterred by them for the most part, indicating that the values of RQ, RS, RT, and Q are appropriate or very close to appropriate for this sensor configuration.

In Figure 9, the predicted path and actual path are projected five minutes beyond the last recorded sensor measurement. One can easily see that there is significant error in the predicted path. At five minutes past the last recorded measurement, the coordinates of the predicted path are 3726 feet away from the projected straight true path. The average angle error of 34.22° is significantly higher than the average angle error of runs 1-10. The speed error, however, is very similar to that of runs 1-10 at 5.80 ft/s.

Graphs of the rest of the data in runs 11-20 look fairly similar to these two, with a predicted path that matches the actual path reasonably well during the data collection period but which diverges from the actual path quickly in the time after the measurement period. This information is available in Table 2.

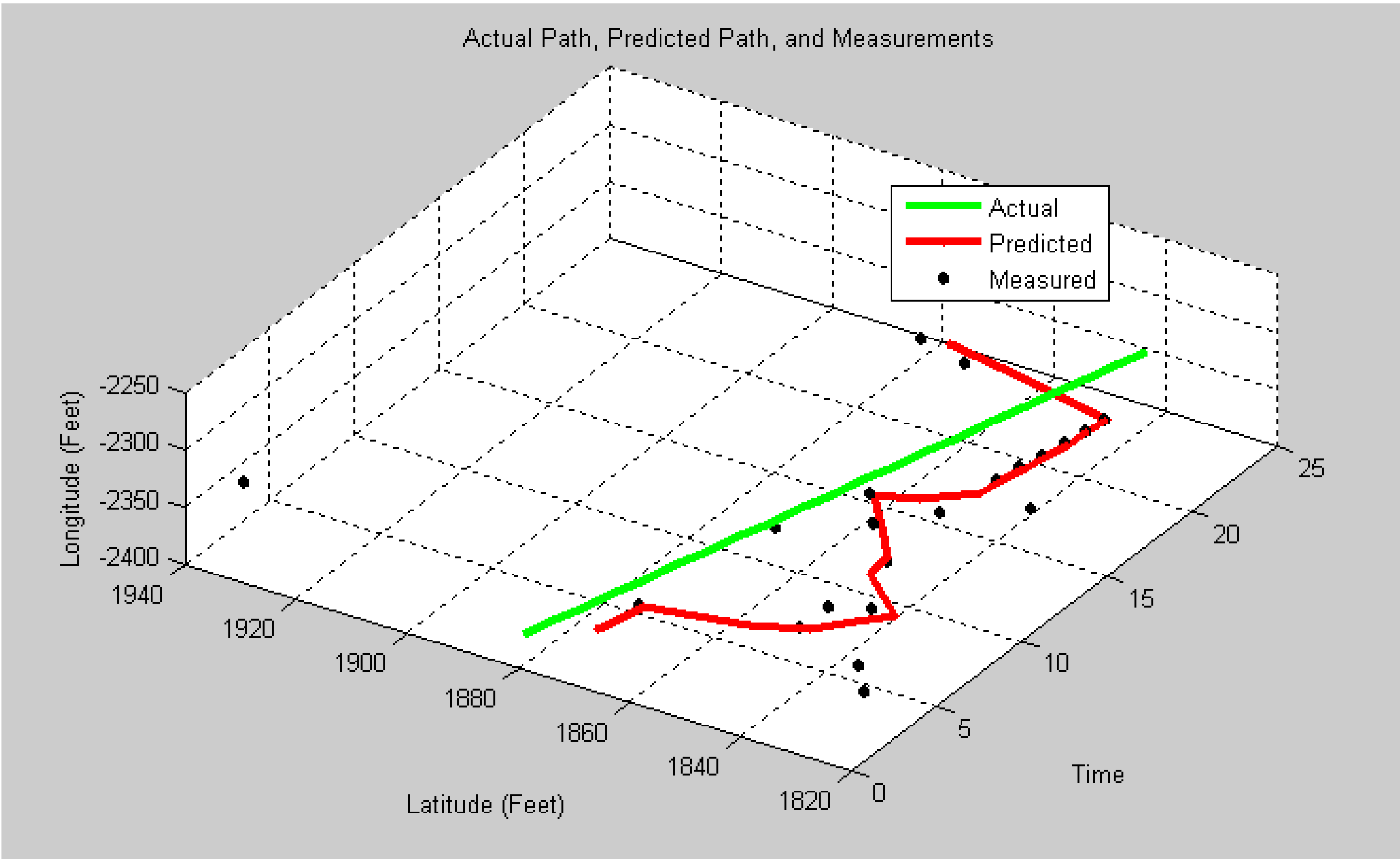

Figure 8 - The Actual Path, Predicted Path, and Measurements for Run 13

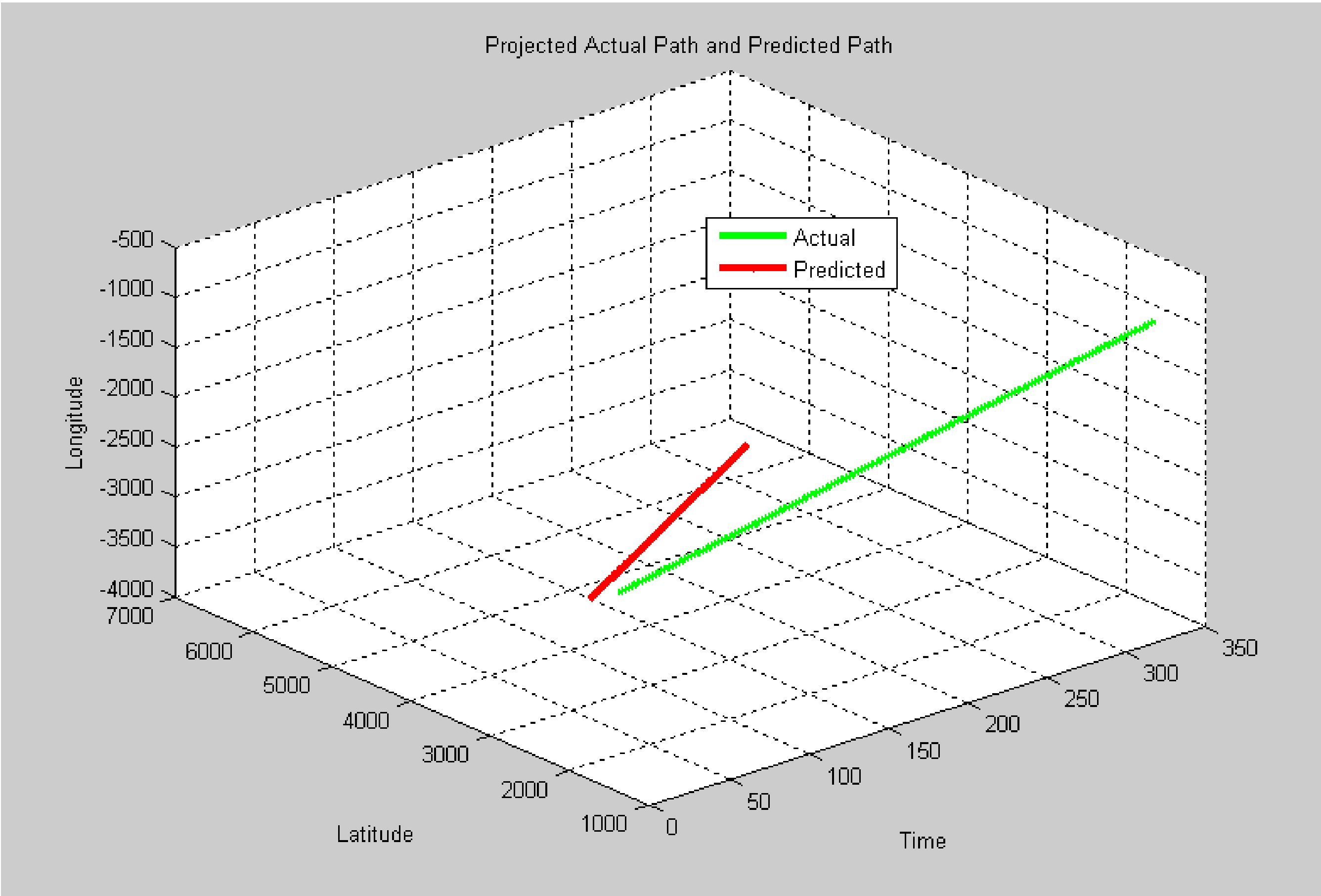

Figure 9 - Projected Actual Path and Predicted Path for Run 13

Table 2 Future Predicted Path Error for Runs 11-20.

Run Number	***Error After Five Minutes (Feet)***	***Angle Error (Degrees)***	***Speed Error (Feet per Second)***
Run 11	6466.139	51.858	14.839
Run 12	1603.784	48.681	0.050
Run 13	6391.677	21.913	8.868
Run 14	4975.669	40.590	8.260
Run 15	4161.436	29.387	1.734
Run 16	2724.849	14.724	5.845
Run 17	3786.659	31.340	7.650
Run 18	2050.178	29.782	3.233
Run 19	3004.005	25.635	6.199
Run 20	2100.270	48.302	1.361
Average	3726.467	34.221	5.804

Runs 21-30

The final series of measurement runs were performed in an area with very low sensor density. A few Textron seismic sensors were near the path, and the path crossed the Optasense and Southwest Microwave sensors in only one spot. This resulted in runs containing only a few data points from each sensor. In two cases (Run 22 and Run 28), the Textron sensors did not detect the subject during any part of the run. One would expect this to cause predicted paths that may stray far from the actual path.

These runs consisted of an individual walking between the coordinates c_1, (35.93806 N, 84.27793 W), and c_2, (35.93813 N, 84.27462 W). In odd numbered runs, the subject progresses from c_1 to c_2, and in even numbered runs, the subject progresses from c_2 to c_1. The path of the subject through the field of sensors is available in Figure 10.

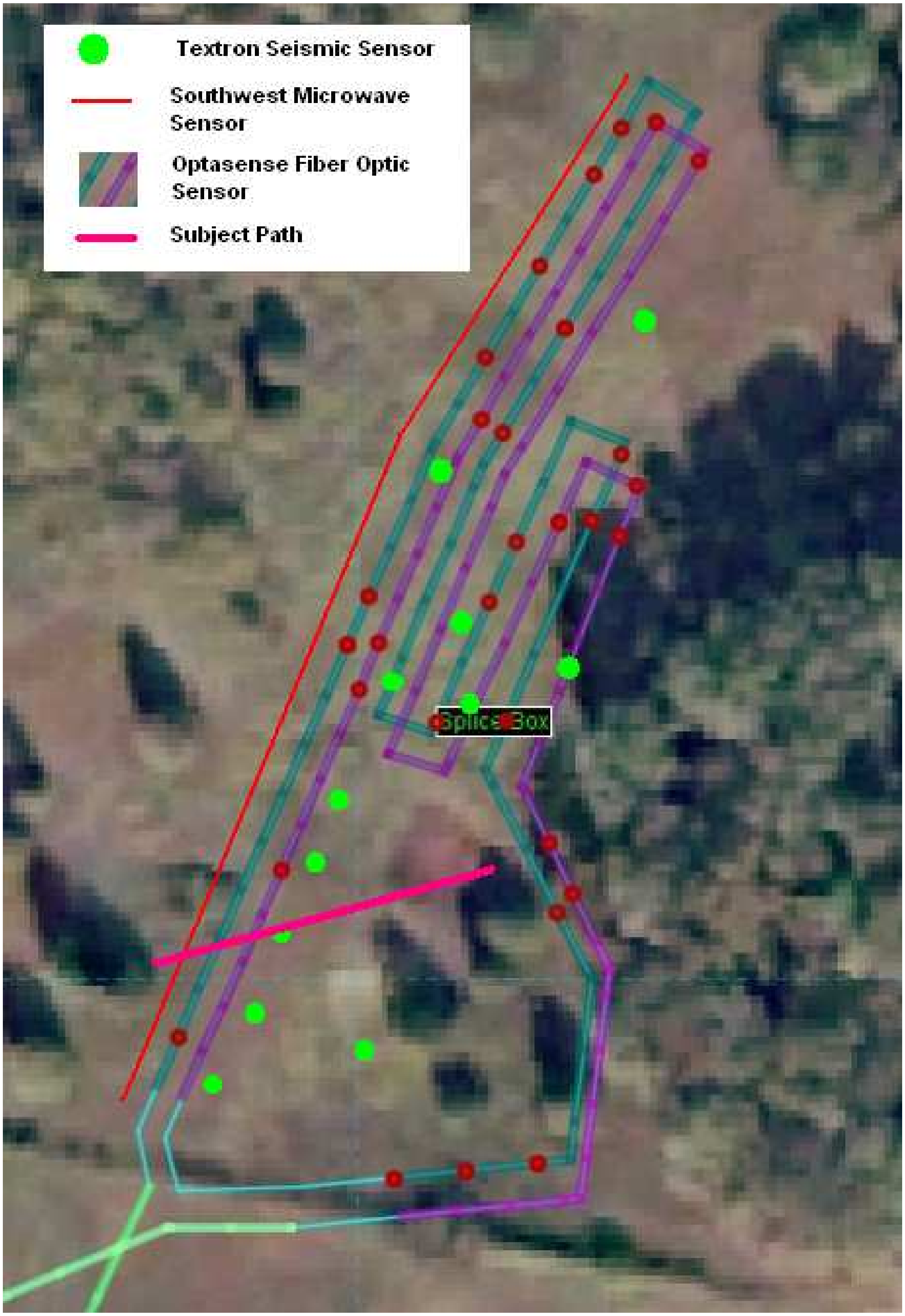

Figure 10 - Path Walked By Subject in Runs 1 - 10

In Figures 11 and 12, the results from run 27 are displayed which was selected as a token sample of runs 21-30. Unsurprisingly, the predicted path is far from the actual subject path. The very low number of sensor measurements can be blamed for this. Additionally, the Optasense and Southwest Microwave sensors are at essentially at a single point in the path of the subject. This means that even if several data points were obtained from these sensors, they would not provide much meaningful information.

Despite the fact that the predicted path is very far from the actual path, it runs somewhat parallel to the actual path. This should result in a relatively low angle error, although since the predicted starting position is so far off, this low error may not be very helpful over relatively short distances.

Table 3 displays the amount of error in each run when the predicted path has been projected 5 minutes past the time of the last measurement as well as the angle error and speed error. The angle error was surprisingly lower than that of runs 11-20 at 23.8°, which may be partly because of the near parallelism between the actual path and the predicted path. The error in speed, however, is higher than any of the earlier runs at 6.4 ft/s.

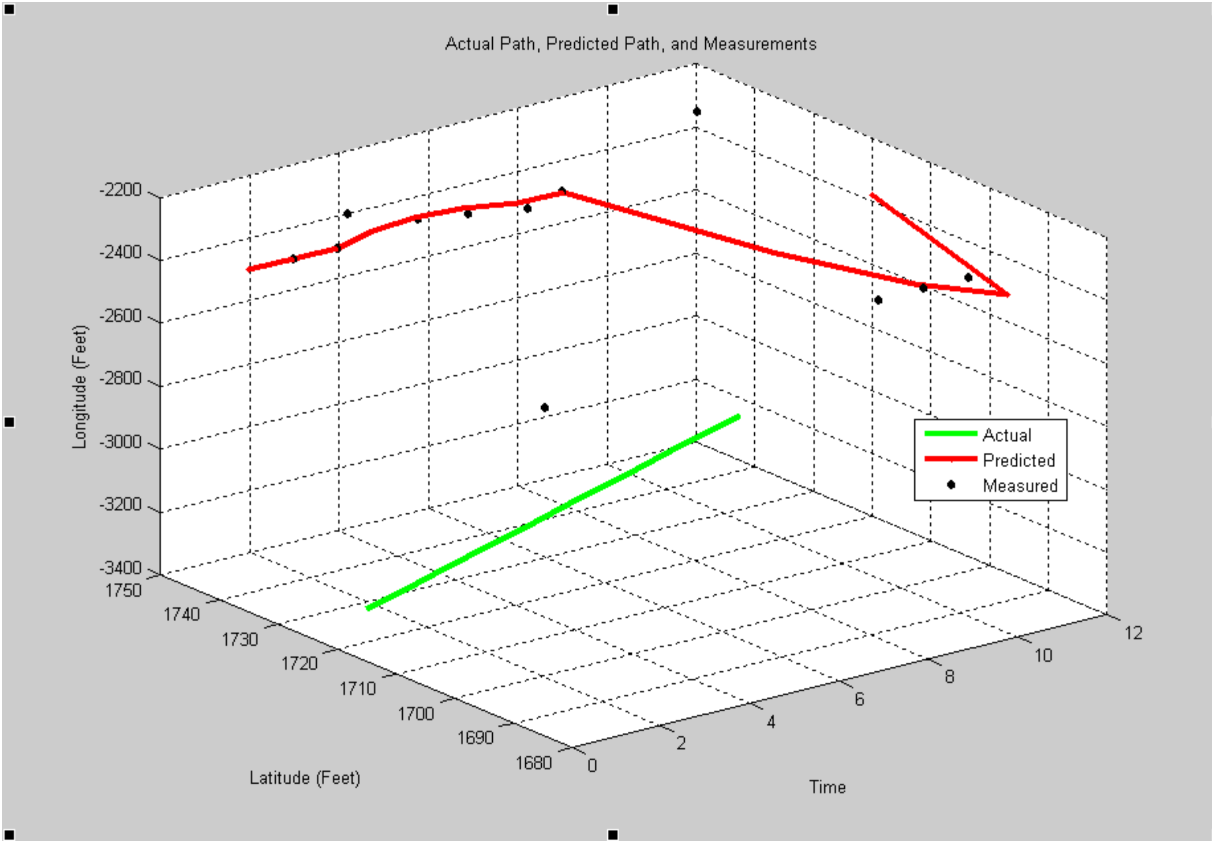

Figure 11 - The Actual Path, Predicted Path, and Measurements for Run 27

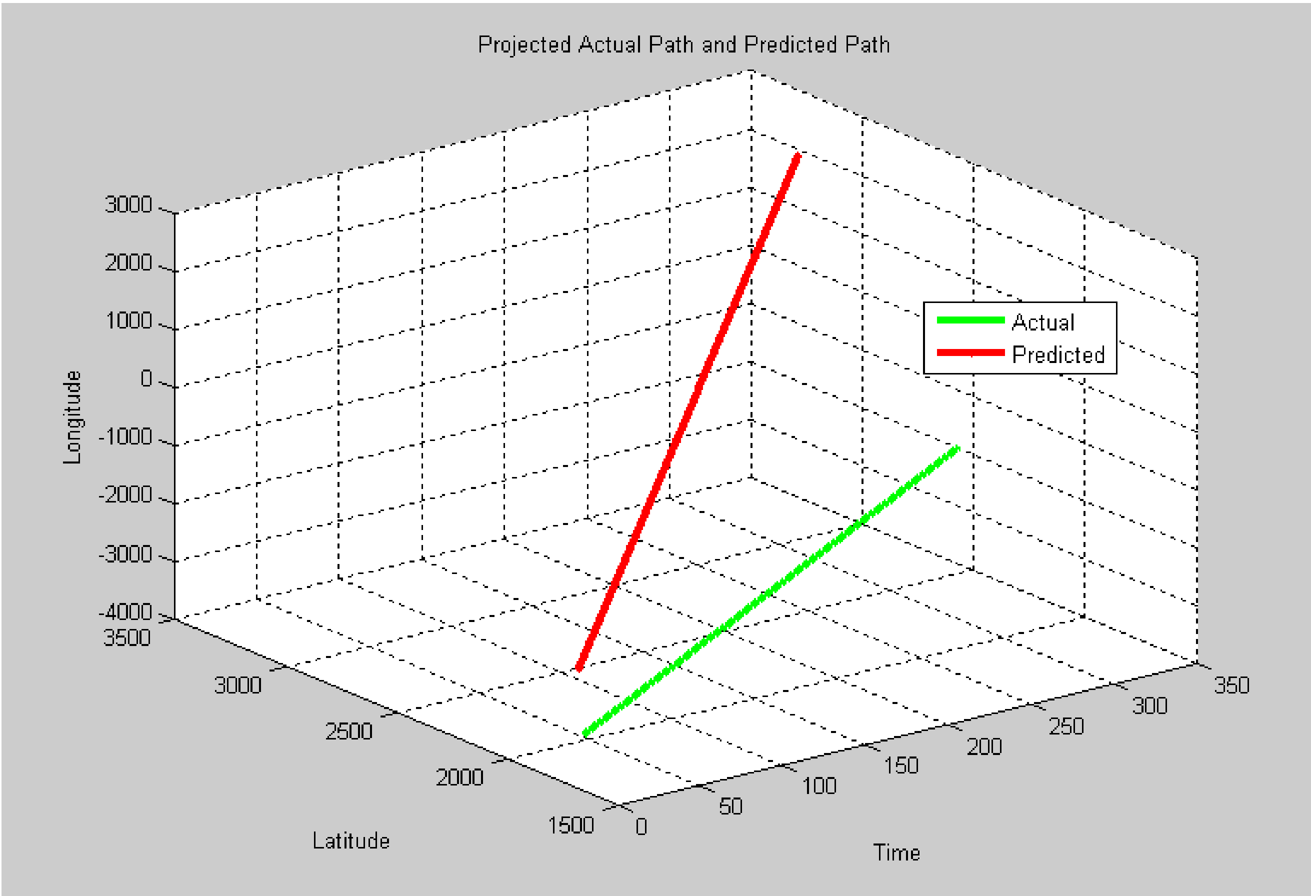

Figure 12 - Projected Actual Path and Predicted Path for Run 27

Table 3. Future Predicted Path Error for Runs 21-30.

Run Number	*Error After Five Minutes (Feet)*	*Angle Error (Degrees)*	*Speed Error (Feet per Second)*
Run 21	4501.349	6.345	8.868
Run 22	2615.562	43.122	3.429
Run 23	6694.143	25.455	9.938
Run 24	1776.637	24.131	0.745
Run 25	2869.021	20.168	5.539
Run 26	2500.973	32.924	5.541
Run 27	4599.085	0.358	9.582
Run 28	3136.783	49.535	1.523
Run 29	5190.611	4.625	11.071
Run 30	4145.494	31.363	8.096
Average	3802.966	23.803	6.433

Simulated Data Runs

As mentioned in the methods section, several runs of simulated data were achieved to test the system assuming that the value of Q and R are perfectly estimated. These runs each assumed that the individual was walking between the points c_1, (35.93798 N, 84.27489 W), and c_2, (35.93833 N, 84.27476 W). These are the same points used in runs 1 through 10, but which points are used is completely inconsequential.

Table 4 lists the error in each simulated run in the same manner as the actual runs above. Additionally, Figure 13 and Figure 14 graph the "measured" data and the predicted data after the measurement period. Of the ten simulated runs in Table 4, the average angle error is only 2.7°, and the average speed error is only 0.515 ft/s. This is quite a deviation from the operation of the Kalman filter on the measured data. Averaging 1000 simulated runs produced an average angle error of 2.87° and an average speed error of 0.086 ft/s.

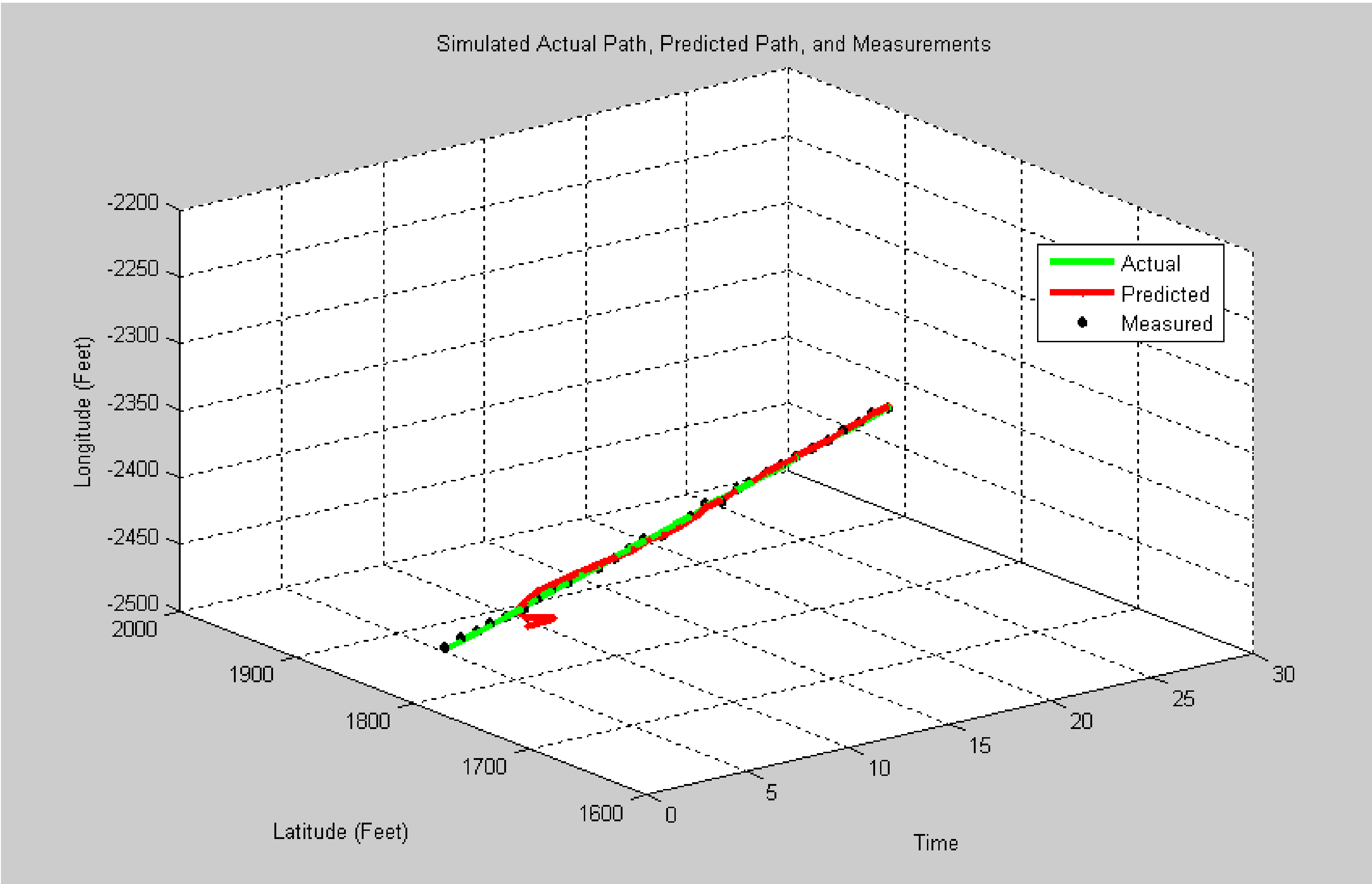

Figure 13 - The Actual Path, Predicted Path, and Measurements for a Simulated Run

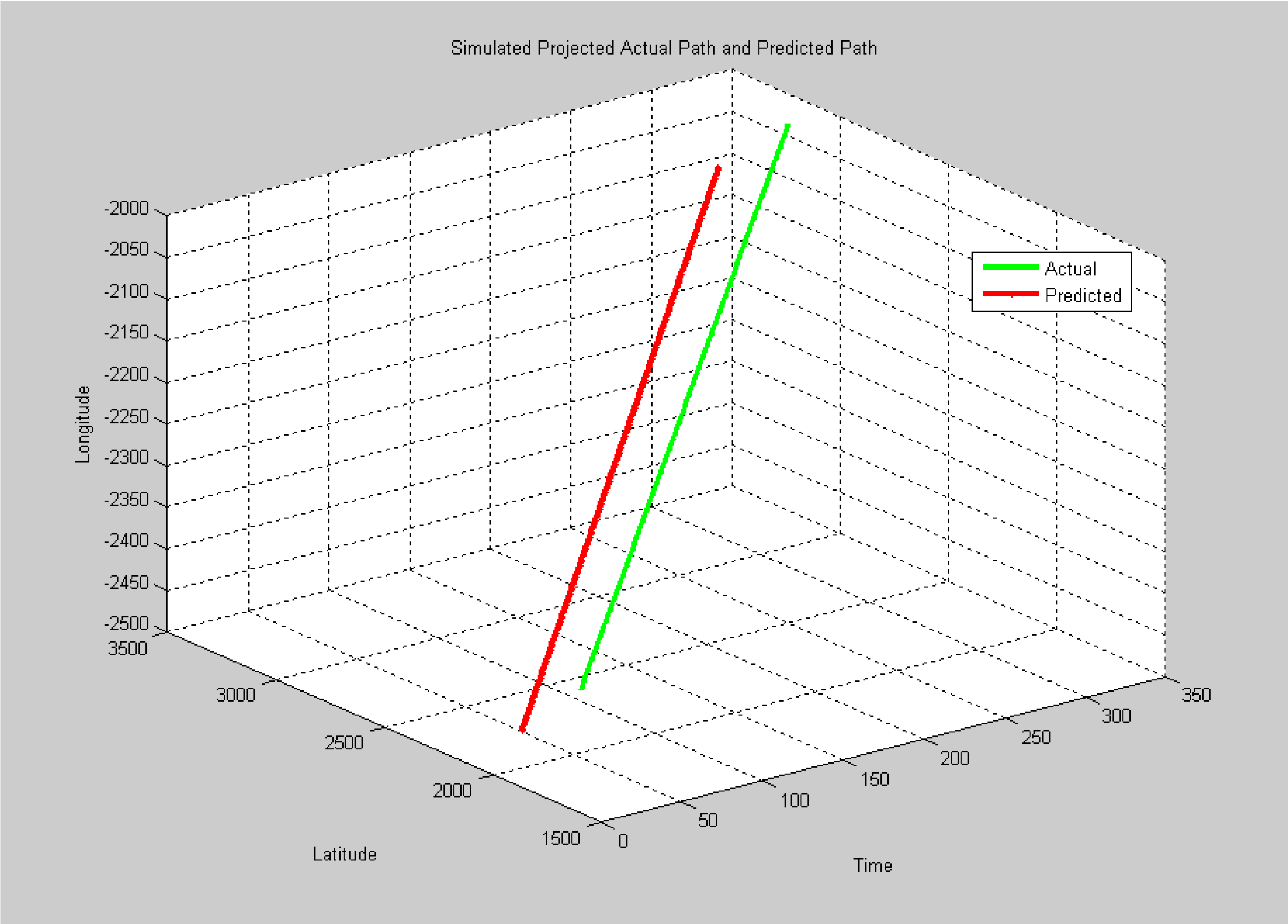

Figure 14 - Projected Actual Path and Predicted Path for a Simulated Run

Table 4 - Simulated Data Error

Run Number	***Error After Five Minutes (Feet)***	***Angle Error (Degrees)***	***Speed Error (Feet per Second)***
Run 1	664.889	3.783	0.132
Run 2	650.987	1.882	0.124
Run 3	510.822	2.858	0.049
Run 4	887.269	0.748	1.561
Run 5	551.958	1.515	0.519
Run 6	614.833	0.252	0.946
Run 7	382.926	3.429	0.368
Run 8	252.442	5.837	0.742
Run 9	777.566	5.449	0.674
Run 0	357.673	1.285	0.038
Average	565.137	2.704	0.515

Analysis

While the Kalman Filter seems to accurately produce the path of a moving individual given a sufficient number of noisy measurements, its performance as a path prediction tool is only useful in very specific and nearly ideal circumstances. Even with a very high sensor density, the path of a subject was only predictable within about a half mile after five minutes.

A Possible Better Approach – Predicted Velocity Averaging

Since the predicted path beyond the last measurement is based upon the last predicted position and velocity, it is possible that it is being skewed by error in the last measurement. A possible improvement of the prediction part of this algorithm is to project the future path based upon the average predicted velocity instead of the final predicted velocity. By changing the way the future path is predicted in this manner, a significant improvement can be observed in runs with a high or moderate amount of sensor coverage, which can be seen in Table 5. An average increase in accuracy of 81.7% was caused in the angle error of runs 1-10 by changing this part of the algorithm, and an average increase of 61.14% in runs 11-20. In runs 1-10, an increase of 52.9% was produced in the accuracy of the speed, and in runs 11-20, and increase of 64.4% was produced. This leaves an average error after 5 minutes in runs 1-10 of 923 feet, and an average error after 5 minutes in runs 11-20 of 1105.8 feet. These numbers are much more manageable when their intended purpose is the interception of an individual. By using this prediction algorithm instead, a Kalman Filter based approach would actually be useful for predicting the path of an individual crossing a border.

In runs 21-30, the runs with poor sensor coverage, the angle error actually increased from 23.8° to 37.4°. This may indicate that while the new velocity prediction technique is a definite improvement in areas with sufficient sensor coverage, it may not be productive in areas where there is poor coverage anyway. Additionally, this may indicate that the results for runs 21-30 which appeared to be more accurate predictors than runs 11-20 may have actually just been a large amount of variance in a relatively small sample size.

The downside of using this new approach is that it forsakes one of the prime benefits of Kalman Filtering. Now, rather than being able to keep track of only the current prediction and the current measurement, the predicted velocity for the entire measurement period must be remembered. While this costs some additional computational overhead, the improvement in performance is undoubtedly worth it.

Table 5. Path Prediction Improvement.

Run Number	*Original Error After Five Minutes (Feet)*	*Original Angle Error (Degrees)*	*Original Speed Error (Feet Per Second)*	*Corrected Error After Five Minutes (Feet)*	*Corrected Angle Error (Degrees)*	*Corrected Speed Error (Feet Per Second)*
Run 1	2467.098	5.777	6.801	1303.322	3.519	3.862
Run 2	1571.152	29.510	4.645	690.702	1.062	2.137
Run 3	9826.442	7.392	26.376	922.641	5.759	3.296
Run 4	2155.929	36.824	1.196	800.548	2.763	2.679
Run 5	3555.313	1.426	0.403	1364.325	6.228	3.962
Run 6	2583.995	5.570	4.944	840.330	4.227	1.314
Run 7	3031.171	13.092	2.060	899.470	1.382	2.538
Run 8	1741.265	27.709	3.863	870.461	4.723	1.121
Run 9	2165.033	50.898	2.436	651.656	0.011	1.813
Run 10	830.209	4.805	1.930	888.415	3.795	3.021
Run 11	6466.139	51.858	14.839	2092.664	45.825	4.787
Run 12	1603.784	48.681	0.050	919.771	6.985	1.286
Run 13	6391.677	21.913	8.868	1421.443	1.444	3.727
Run 14	4975.669	40.590	8.260	425.113	1.426	2.502
Run 15	4161.436	29.387	1.734	1115.082	6.927	2.947
Run 16	2724.849	14.724	5.845	1145.022	7.810	0.718
Run 17	3786.659	31.340	7.650	1011.910	15.014	1.637
Run 18	2050.178	29.782	3.233	1015.311	8.660	1.192
Run 19	3004.005	25.635	6.199	1431.693	22.675	0.804
Run 20	2100.270	48.302	1.361	480.463	16.205	1.048
Run 21	4501.349	6.345	8.868	2425.744	50.147	2.483
Run 22	2615.562	43.122	3.429	1962.801	27.720	4.357
Run 23	6694.143	25.455	9.938	1978.184	43.252	3.858
Run 24	1776.637	24.131	0.745	1944.128	28.909	2.835
Run 25	2869.021	20.168	5.539	1440.585	26.056	3.444
Run 26	2500.973	32.924	5.541	2354.610	37.599	6.211
Run 27	4599.085	0.358	9.582	2018.655	42.851	3.377
Run 28	3136.783	49.535	1.523	2332.472	49.535	3.839
Run 29	5190.611	4.625	11.071	2358.132	36.556	0.579
Run 30	4145.494	31.363	8.096	2070.454	31.307	2.387

Overall Prediction Accuracy

After modifying the algorithm to predict the future path beyond the final state measurement, this approach appears to be good enough for border security applications. With an angle error of less than 15° with good sensor coverage, it should be possible to intercept an individual crossing a field of these sensor with significantly more easy than if one simply had a binary indication of when and where the individual had crossed. Narrowing the search for an intruder to such a small area would greatly reduce the resources needed to find that individual.

Despite the potential upsides of this approach, it is clear that the density of sensors in the area being traversed by an individual needs to be sufficiently high to provide any useful information. This may be problematic if sensors are installed in a way that is not conducive to this type of algorithm. Be that as it may, it may be possible to selectively deploy additional sensors to improve the sensor density to a suitable level. Specifically, the Textron sensors can be deployed very rapidly and may be useful for filling gaps in sensor-light areas.

Simulated Results Observations

From the simulated results, it is apparent that with perfectly accurate values for Q and R, and with sufficient sensor measurements, the approach suggested in this paper is very accurate. With an angle error of less than 3°and a speed error of less than 0.1 ft/s, the predictive capabilities of this system would be invaluable. Finding better values for R and Q are critical to approaching this level of accuracy. Additionally, it is certainly possible that the noise in the movement of the individual as well as the noise in the measurements of the sensors are not truly Gaussian. If there is some systemic noise somewhere, it is worth looking for it and trying to compensate for it.

CHAPTER V

CONCLUSIONS AND RECOMMENDATIONS

Future Work and Suggestions

The path prediction algorithm demonstrated in this paper shows a great deal of promise, especially with an improved algorithm for estimating the velocity of the individual beyond the field of the sensors. Given that such a small and intuitive change was able to improve the performance of the algorithm so significantly, the future work to be explored first should be small changes in this paper's algorithms to further improve performance. Specifically, a good place to look would be achieving more exact values for the Q, RQ, RS, and RT constants in the Kalman Filter algorithm equations. The values used in this research seemed sufficient for the operation of the filter, however, since these values are nearly impossible to measure empirically, it is likely that values could be substituted which would yield more accurate results. This is also borne out in the simulated results. With perfect values for R and Q, much more accurate estimations are achievable.

Another worthy goal in the future would be to construct a system for performing this prediction algorithm in real time or close to real time. The sensor data can come in with a delay of up to 15 seconds, so a slight delay would be inevitable, however, now that it is apparent that such a system could work and be useful, constructing it would be nice.

Finally, the issue of multitarget tracking was not covered in this paper, however, if such a system were to be build, it would need to include the ability to determine whether or not a measurement belongs to the current track or should be allocated to a knew track. Algorithms to efficiently perform this operation have existed for many years and many papers have been written on the topic. Sources [2], [7], [8], [9], and [10] would be a good place to start if one desired the ability to track multiple individuals simultaneously.

Conclusion

For the purpose of border security, a system like the one examined in this paper could be very useful. Since the interception of a person trying to illegally cross a border is paramount, the ability to accurately predict their path is important. The algorithm laid forth in this paper is a very good starting point for developing a system which would adequately detect intruders and predict their direction.

While the Kalman Filter algorithm is an excellent jumping off point, using its final state estimate for the purposes of predicting a path may be insufficient. However, by averaging the system state during the entire measurement period, a much more accurate prediction is possible. It is conceivable that even more accurate methods of predicting the path are possible.

LIST OF REFERENCES

[1] Bishop, Gary, Welch, Greg. “An Introduction to the Kalman Filter.” University of North Carolina at Chapel Hill Department of Computer Science. 2001. <http://www.cs.unc.edu/~tracker/media/pdf/SIGGRAPH2001_CoursePack_08.pdf>

[2] Bar-Shalom, Yaakov. “Multitarget-Multisensor Tracking: Advanced Applications.” Artech House. 1990.

[3] Kuhn, Mike. “A Physical Protection Systems Test Bed for International Counter-Trafficking System Development.” Oak Ridge National Laboratory. 20011

[4] Stinson, Brad. “Performance Test Bed for Border Security Technologies.” Oak Ridge National Laboratory. 2010.

[5] Mestayer, J. et. al. “Field Trials of Distributed Acoustic Sensing for Geophysical Monitoring.” QinetiQ OptaSense. 2011.

[6] Stinson, Brad. Test Bed Tech Capabilities Presentation. Oak Ridge National Laboratory. 2011.

[7] Bar-Shalom, Yaakov. “Multitarget-Multisensor Tracking: Applications and Advances Volume III.” Artech House. 2000.

[8] Saha, R. K. “Track-to-Track Fusion with Dissimilar Sensors.” IEEE Transactions on Aerospace and Electronic Systems Vol 32, No. 3. July 1996.

[9] Di Lallo, A. et. al. “A Real Time Test Bed for 2D and 3D Multi-Radar Tracking and Data Fusion with Application to Border Control.” IEEE. 2006

[10] Oh, Songhwai; Sastry, Shankar.   "An Efficient Algorithm for Tracking Multiple Maneuvering Targets." IEEE. 2005

[11] Rojas, Raul. "The Kalman Filter."   Freie Universitat Berlin, Institut fur Informatik.

[12] Plessis, Roger. "Poor Man's Explanation of Kalman Filtering or How I Stopped Worrying and Learned to Love Matrix Inversion." North American Rockwell Electronics Group. June 1967.

[13] Simon, Dan. "Using Nonlinear Kalman Filtering to Estimate Signals." Embedded Systems Design, 2006.

[14] "C Code Generation for a MATLAB Kalman Filtering Algorithm." Matlab Coder. Mathworks. <http://www.mathworks.com/products/matlab-coder/examples.html;jsessionid=e6f65a2cbe1a5f22853356ce8184?file=/products/demos/shipping/coder/coderdemo_kalman_filter.html>

[15] OptaSense: a QinetiQ company. "Perimeter Security and Monitoring." Customer Service Team Advertising Literature.

[16] Southwest Microwave Advertising Literature for the Intrepid Microtrack II System. < http://www.southwestmicrowave.com/products/buried-cable-detection-systems/microtrack2/>

[17] Textron Wireless Seismic Sensor Datasheet. < http://www.textrondefense.com/assets/pdfs/datasheets/TDS%20UGS%2004-01-11m.pdf>

APPENDIX

Matlab Code

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
%testbed.m
%
%Kalman Filtering Implemented using sensors in Test Bed
%
%Jonthan Hickerson
%7-20-12
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%


clearvars -global

%%%%%%%%%%%%%%%%%%%%%%Initialize Constants%%%%%%%%%%%%%%%%%%%%%%%%%%%



%%%%%%%%%%%%%%%%%RUNS 1 - 10%%%%%%%%%%%%%%%%%%%
%%%%%%%%%ODD%%%%%%%%%%%
StartingLat = 35.93798;
StartingLon =  -84.27489;
FinalLat = 35.93833;
FinalLon = -84.27476;

%%%%%%%%%EVEN%%%%%%%%%
% FinalLat = 35.93798;
% FinalLon =  -84.27489;
% StartingLat = 35.93833;
% StartingLon = -84.27476;
%
%
% %%%%%%%%%%%%%%%%%RUNS 11 - 20%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%ODD%%%%%%%%%%%%
% StartingLat = 35.93851;
% StartingLon =  -84.27474;
% FinalLat = 35.93837;
% FinalLon = -84.27441;
%
% %%%%%%%%%EVEN%%%%%%%%%
% FinalLat = 35.93851;
% FinalLon =  -84.27474;
% StartingLat = 35.93837;
% StartingLon = -84.27441;
%
%
% %%%%%%%%%%%%%%%%%RUNS 21 - 30%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%ODD%%%%%%%%%%%%
% StartingLat = 35.93806;
% StartingLon =  -84.27793;
```

```
% FinalLat = 35.93813;
% FinalLon = -84.27462;
%
% %%%%%%%%EVEN%%%%%%%%%%%
% FinalLat = 35.93806;
% FinalLon =  -84.27793;
% StartingLat = 35.93813;
% StartingLon = -84.27462;




TraversalTime = 24;

Bearing = 125;
WalkingSpeed = 1; %fps

FeetPerDegreeLat = 364034.141885025;  %feet
FeetPerDegreeLon = 296043.3276884315;  %feet

T = 1;

A = [1 T T^2/2 0 0 0;...  %State Transition Matrix
     0 1 T 0 0 0;...
     0 0 1 0 0 0;...
     0 0 0 1 T T^2/2;...
     0 0 0 0 1 T;...
     0 0 0 0 0 1];

H = [1 0 0 0 0 0;...
     0 0 0 1 0 0];  %Measurement Matrix

Q =  0.0001 * eye(6); %Process Noise Covariance Matrix
RQ = 0.01 * eye(2);  %QinetiQ Measurement Noise Covariance Matrix
RT = 0.01 * eye(2);  %Textron Measurement Noise Covariance Matrix
RS = 0.01 * eye(2);  %Southwest Measurement Noise Covariance Matrix
%As R approaches zero, more and more weight is given to the measurement
%As Q approaches zero, more and more weight is given to the estimate

x_est = zeros(6,1); %estimated state
p_est = zeros(6,6); %estimated estimate error covariance
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




%%%%%%%%%%%%%%%%%%%Grab Data from QinetiQ text file%%%%%%%%%%%%%%%%%%%
count = 1;
fid = fopen('QinetiQ.txt');
line = fgetl(fid);

clearvars thedata;
```

```
clearvars thetextrondata;
clearvars thesouthwestdata;
thedata{1} = '';

while ischar(line)
    thedata{count} = strread(line, '%s')';     %#ok<*SAGROW,*REMFF1>
    count = count + 1;
    line = fgetl(fid);
end

fclose(fid);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




%%%%%%%%%%%%%%%Grab Data from Southwest text file%%%%%%%%%%%%%%%%%%%%%%
count = 1;
fid = fopen('Southwest.txt');
line = fgetl(fid);

clearvars thesouthwestdata;
thesouthwestdata{1} = '';

while ischar(line)
    thesouthwestdata{count} = strread(line, '%s')';
%#ok<*SAGROW,*REMFF1>
    count = count + 1;
    line = fgetl(fid);
end

fclose(fid);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




%%%%%%%%%%%%%%%%%%Grab Data from Textron text file%%%%%%%%%%%%%%%%%%%%%%
count = 1;
fid = fopen('Textron.txt');
line = fgetl(fid);

clearvars thetexttrondata;
thetextrondata{1} = '';

while ischar(line)
    thetextrondata{count} = strread(line, '%s')';
%#ok<*SAGROW,*REMFF1>
    count = count + 1;
    line = fgetl(fid);
end
```

```
fclose(fid);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




%%%%%%%%%%%%%%%%%%%%%%%%Find first time value%%%%%%%%%%%%%%%%%%%%%%%%%
if(~strcmp(thedata{1},''))
    starttimeQ = datenum([thedata{1}{2} ' ' thedata{1}{3}]);
    [j,sizeQ] = size(thedata);
else
    sizeQ = 0;
    starttimeQ = datenum(3000, 0, 0);
end
if(~strcmp(thesouthwestdata{1},''))
    starttimeS = datenum([thesouthwestdata{1}{2} ' '
thesouthwestdata{1}{3}]);
    [j,sizeS] = size(thesouthwestdata);
else
    sizeS = 0;
    starttimeS = datenum(3000, 0, 0);
end
if(~strcmp(thetextrondata{1},''))
    starttimeT = datenum([thetextrondata{1}{2} ' '
thetextrondata{1}{3}]);
    [j,sizeT] = size(thetextrondata);
else
    sizeT = 0;
    starttimeT = datenum(3000, 0, 0);
end


starttime = min([starttimeQ starttimeS starttimeT]); %#ok<*NASGU>

currentsecond = starttime;
nextsecond = starttime + (1/3600/24);

%a = datestr(currentsecond);
%b = datestr(nextsecond);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%




%%%%%%%%%Set the Measured State to be equal to the seconds of the %%%%%
%%%%%%%%%%%%%%%%%first Lat/Long with no Velocity%%%%%%%%%%%%%%%%%%%
measurement = [str2double(thedata{1}{6}); str2double(thedata{1}{7})];
%#ok<*ST2NM>
measurement(1) = (measurement(1)*60 - fix(measurement(1)*60))*60;
measurement(2) = (measurement(2)*60 - fix(measurement(2)*60))*60;

x_est = [measurement(1); 0; 0; measurement(2); 0; 0];
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



%%%%%%%%%%%%%%%%%Initialize Arrays for Graphing%%%%%%%%%%%%%%%%%%%%%%%%
predic_state_x = [];
measured_state_x = [];
predic_state_y = [];
measured_state_y = [];
future_state_x = [];
future_state_y = [];
date_array = [];
date_string_array = [];

kalman_velocity_x = [];
kalman_velocity_y = [];
variancex = [];
variancey = [];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%Initialize Variables for Looping%%%%%%%%%%%%%%%%%%%%%%%%%
keepgoing = 1;
indexQ = 1;
indexS = 1;
indexT = 1;
useQ = 0;
useS = 0;
useT = 0;

count = 1;



while keepgoing == 1

%%%%%%%%%Determine what instruments have a measurement%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%From this time%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    if((indexQ <= sizeQ) && (datenum([thedata{indexQ}{2} ' '
thedata{indexQ}{3}]) == currentsecond))
        useQ = 1;
    end

    if((indexS <= sizeS) && (datenum([thesouthwestdata{indexS}{2} ' '
thesouthwestdata{indexS}{3}]) == currentsecond))
        useS = 1;
    end

    if((indexT <= sizeT) && (datenum([thetextrondata{indexT}{2} ' '
thetextrondata{indexT}{3}]) == currentsecond))
        useT = 1;
    end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


%%%%%%%%%%%%%%%%%Fuse Measurements from same instant%%%%%%%%%%%%%%%%%
%%Import Lat/Lon and convert to Degrees/Minutes/Seconds.  Keep seconds%
%%%%%%%%%%%%%(No variation in Degrees or Minutes will occur)%%%%%%%%%%%%
    if(useQ && ~useS && ~useT)
        R = RQ;
        measurement = [str2double(thedata{indexQ}{6});
str2double(thedata{indexQ}{7})]; %#ok<*ST2NM>

        iq=1;
        while((indexQ+iq<=sizeQ) && (datenum(thedata{indexQ}{3}) ==
datenum(thedata{indexQ + iq}{3})))
            measurement = measurement + [str2double(thedata{indexQ +
iq}{6}); str2double(thedata{indexQ + iq}{7})];
            iq = iq+1;
        end

        measurement = measurement/iq;

        measurement(1) = (measurement(1)*60 -
fix(measurement(1)*60))*60;
        measurement(2) = (measurement(2)*60 -
fix(measurement(2)*60))*60;

    elseif(~useQ && useS && ~useT)
        R = RS;
        measurement = [str2double(thesouthwestdata{indexS}{6});
str2double(thesouthwestdata{indexS}{7})]; %#ok<*ST2NM>

        is=1;
        while((indexS+is<=sizeS) &&
(datenum(thesouthwestdata{indexS}{3}) ==
datenum(thesouthwestdata{indexS + is}{3})))
            measurement = measurement +
[str2double(thesouthwestdata{indexS + is}{6});
str2double(thesouthwestdata{indexS + is}{7})];
            is = is+1;
        end

        measurement = measurement/is;

        measurement(1) = (measurement(1)*60 -
fix(measurement(1)*60))*60;
        measurement(2) = (measurement(2)*60 -
fix(measurement(2)*60))*60;

    elseif(~useQ && ~useS && useT)
        R = RT;
        measurement = [str2double(thetextrondata{indexT}{6});
str2double(thetextrondata{indexT}{7})]; %#ok<*ST2NM>
```

```
        it=1;
        while((indexT+it<=sizeT) && (datenum(thetextrondata{indexT}{3})
== datenum(thetextrondata{indexT + it}{3})))
            measurement = measurement +
[str2double(thetextrondata{indexT + it}{6});
str2double(thetextrondata{indexT + it}{7})];
            it = it+1;
        end

        measurement = measurement/it;

        measurement(1) = (measurement(1)*60 -
fix(measurement(1)*60))*60;
        measurement(2) = (measurement(2)*60 -
fix(measurement(2)*60))*60;

    elseif(useQ && useS && ~useT)
        R = (RQ * RS)/(RQ + RS);

        measurementQ = [str2double(thedata{indexQ}{6});
str2double(thedata{indexQ}{7})]; %#ok<*ST2NM>

        iq=1;
        while((indexQ+iq<=sizeQ) && (datenum(thedata{indexQ}{3}) ==
datenum(thedata{indexQ + iq}{3})))
            measurementQ = measurementQ + [str2double(thedata{indexQ +
iq}{6}); str2double(thedata{indexQ + iq}{7})];
            iq = iq+1;
        end

        measurementQ = measurementQ/iq;

        measurementQ(1) = (measurementQ(1)*60 -
fix(measurementQ(1)*60))*60;
        measurementQ(2) = (measurementQ(2)*60 -
fix(measurementQ(2)*60))*60;

        measurementS = [str2double(thesouthwestdata{indexS}{6});
str2double(thesouthwestdata{indexS}{7})]; %#ok<*ST2NM>

        is=1;
        while((indexS+is<=sizeS) &&
(datenum(thesouthwestdata{indexS}{3}) ==
datenum(thesouthwestdata{indexS + is}{3})))
            measurementS = measurementS +
[str2double(thesouthwestdata{indexS + is}{6});
str2double(thesouthwestdata{indexS + is}{7})];
            is = is+1;
        end

        measurementS = measurementS/is;
```

```
        measurementS(1) = (measurementS(1)*60 -
fix(measurementS(1)*60))*60;
        measurementS(2) = (measurementS(2)*60 -
fix(measurementS(2)*60))*60;

        measurement(1) = ((measurementQ(1) * RS(1,1)) +
(measurementS(1) * RQ(1,1)))/(RS(1,1)+RQ(1,1));
        measurement(2) = ((measurementQ(2) * RS(2,2)) +
(measurementS(2) * RQ(2,2)))/(RS(2,2)+RQ(2,2));

    elseif(useQ && ~useS && useT)
        R = (RQ * RT)/(RQ + RT);

        measurementQ = [str2double(thedata{indexQ}{6});
str2double(thedata{indexQ}{7})]; %#ok<*ST2NM>

        iq=1;
        while((indexQ+iq<=sizeQ) && (datenum(thedata{indexQ}{3}) ==
datenum(thedata{indexQ + iq}{3})))
            measurementQ = measurementQ + [str2double(thedata{indexQ +
iq}{6}); str2double(thedata{indexQ + iq}{7})];
            iq = iq+1;
        end

        measurementQ = measurementQ/iq;

        measurementQ(1) = (measurementQ(1)*60 -
fix(measurementQ(1)*60))*60;
        measurementQ(2) = (measurementQ(2)*60 -
fix(measurementQ(2)*60))*60;

        measurementT = [str2double(thetextrondata{indexT}{6});
str2double(thetextrondata{indexT}{7})]; %#ok<*ST2NM>

        it=1;
        while((indexT+it<=sizeT) && (datenum(thetextrondata{indexT}{3})
== datenum(thetextrondata{indexT + it}{3})))
            measurement = measurement +
[str2double(thetextrondata{indexT + it}{6});
str2double(thetextrondata{indexT + it}{7})];
            it = it+1;
        end

        measurement = measurement/it;

        measurementT(1) = (measurementT(1)*60 -
fix(measurementT(1)*60))*60;
        measurementT(2) = (measurementT(2)*60 -
fix(measurementT(2)*60))*60;
```

```
        measurement(1) = ((measurementQ(1) * RT(1,1)) +
(measurementT(1) * RQ(1,1)))/(RT(1,1)+RQ(1,1));
        measurement(2) = ((measurementQ(2) * RT(2,2)) +
(measurementT(2) * RQ(2,2)))/(RT(2,2)+RQ(2,2));

    elseif(~useQ && useS && useT)
        R = (RS * RT)/(RS + RT);

        measurementS = [str2double(thesouthwestdata{indexS}{6});
str2double(thesouthwestdata{indexS}{7})]; %#ok<*ST2NM>

        is=1;
        while((indexS+is<=sizeS) &&
(datenum(thesouthwestdata{indexS}{3}) ==
datenum(thesouthwestdata{indexS + is}{3})))
            measurementS = measurementS +
[str2double(thesouthwestdata{indexS + is}{6});
str2double(thesouthwestdata{indexS + is}{7})];
            is = is+1;
        end

        measurementS = measurementS/is;

        measurementS(1) = (measurementS(1)*60 -
fix(measurementS(1)*60))*60;
        measurementS(2) = (measurementS(2)*60 -
fix(measurementS(2)*60))*60;

        measurementT = [str2double(thetextrondata{indexT}{6});
str2double(thetextrondata{indexT}{7})]; %#ok<*ST2NM>

        it=1;
        while((indexT+it<=sizeT) && (datenum(thetextrondata{indexT}{3})
== datenum(thetextrondata{indexT + it}{3})))
            measurement = measurement +
[str2double(thetextrondata{indexT + it}{6});
str2double(thetextrondata{indexT + it}{7})];
            it = it+1;
        end

        measurement = measurement/it;

        measurementT(1) = (measurementT(1)*60 -
fix(measurementT(1)*60))*60;
        measurementT(2) = (measurementT(2)*60 -
fix(measurementT(2)*60))*60;

        measurement(1) = ((measurementS(1) * RT(1,1)) +
(measurementT(1) * RS(1,1)))/(RT(1,1)+RS(1,1));
        measurement(2) = ((measurementS(2) * RT(2,2)) +
(measurementT(2) * RS(2,2)))/(RT(2,2)+RS(2,2));

    elseif(useQ && useS && useT)
```

```
        R = (RQ * RS)/(RQ + RS);
        R = (R * RT)/(R + RT);

        measurementQ = [str2double(thedata{indexQ}{6});
str2double(thedata{indexQ}{7})]; %#ok<*ST2NM>

        iq=1;
        while((indexQ+iq<=sizeQ) && (datenum(thedata{indexQ}{3}) ==
datenum(thedata{indexQ + iq}{3})))
            measurementQ = measurementQ + [str2double(thedata{indexQ +
iq}{6}); str2double(thedata{indexQ + iq}{7})];
            iq = iq+1;
        end

        measurementQ = measurementQ/iq;

        measurementQ(1) = (measurementQ(1)*60 -
fix(measurementQ(1)*60))*60;
        measurementQ(2) = (measurementQ(2)*60 -
fix(measurementQ(2)*60))*60;

        measurementS = [str2double(thesouthwestdata{indexS}{6});
str2double(thesouthwestdata{indexS}{7})]; %#ok<*ST2NM>

        is=1;
        while((indexS+is<=sizeS) &&
(datenum(thesouthwestdata{indexS}{3}) ==
datenum(thesouthwestdata{indexS + is}{3})))
            measurementS = measurementS +
[str2double(thesouthwestdata{indexS + is}{6});
str2double(thesouthwestdata{indexS + is}{7})];
            is = is+1;
        end

        measurementS = measurementS/is;

        measurementS(1) = (measurementS(1)*60 -
fix(measurementS(1)*60))*60;
        measurementS(2) = (measurementS(2)*60 -
fix(measurementS(2)*60))*60;

        measurementT = [str2double(thetextrondata{indexT}{6});
str2double(thetextrondata{indexT}{7})]; %#ok<*ST2NM>

        it=1;
        while((indexT+it<=sizeT) && (datenum(thetextrondata{indexT}{3})
== datenum(thetextrondata{indexT + it}{3})))
            measurement = measurement +
[str2double(thetextrondata{indexT + it}{6});
str2double(thetextrondata{indexT + it}{7})];
            it = it+1;
        end
```

```
        measurement = measurement/it;

        measurementT(1) = (measurementT(1)*60 -
fix(measurementT(1)*60))*60;
        measurementT(2) = (measurementT(2)*60 -
fix(measurementT(2)*60))*60;

        measurement(1) = ((measurementQ(1) * RS(1,1)) +
(measurementS(1) * RQ(1,1)))/(RS(1,1)+RQ(1,1));
        measurement(2) = ((measurementQ(2) * RS(2,2)) +
(measurementS(2) * RQ(2,2)))/(RS(2,2)+RQ(2,2));

        measurement(1) = ((measurement(1) * RT(1,1)) + (measurementT(1)
* R(1,1)))/(R(1,1)+RT(1,1));
        measurement(2) = ((measurement(2) * RT(2,2)) + (measurementT(2)
* R(2,2)))/(R(2,2)+RT(2,2));
    else
        %If no data from any sensor, tell it that the measured data is
        %exactly what is expected, pushing us forward to the next time
        %step.  Use last R for simplicity
        R = R;
        measurement = [y(1); y(2)];
    end



   %Predict
   x_prd = A * x_est;
   p_prd = A * p_est * A' + Q;


   %Correct
   S = H * p_prd' * H' + R;
   B = H * p_prd';

   K = (S\ B)';

   x_est = x_prd + K * (measurement - H*x_prd);
   p_est = p_prd - K * H * p_prd;

   y = H * x_est;



   measured_state_x = [measured_state_x; measurement(1)]; %#ok<*AGROW>
   measured_state_y = [measured_state_y; measurement(2)];
   predic_state_x = [predic_state_x; y(1)];
   predic_state_y = [predic_state_y; y(2)];

   kalman_velocity_x = [kalman_velocity_x; x_est(2)]
   kalman_velocity_y = [kalman_velocity_y; x_est(5)]
```

```
   variancex = [variancex; p_est(2,2)]
   variancey = [variancey; p_est(5,5)]

 %%%%%%%%%%%%%%%%%%%%Update measurement indices%%%%%%%%%%%%%%%%%%%%%%%
   if(useQ)
       indexQ = indexQ + iq;
   end
   if(useS)
       indexS = indexS + is;
   end
   if(useT)
       indexT = indexT + it;
   end

   if(indexQ > sizeQ && indexS > sizeS && indexT > sizeT)
       keepgoing = 0;
   end

   useQ = 0;
   useT = 0;
   useS = 0;

   currentsecond = datenum(datestr(datenum(datestr(currentsecond)) +
(1/3600/24)));
   a = datestr(currentsecond);

   count = count + 1;
   endtime=currentsecond;
end

%%%%%%%%%%%%%Use the final x_est to predict future movement%%%%%%%%%%%

velocityx=mean(kalman_velocity_x);
velocityy=mean(kalman_velocity_y);

for(j = count : T : count + 300)
    Hfuture = [1 j 0 0 0 0;...
               0 0 0 1 j 0];  %Measurement Matrix

    y = Hfuture * x_est;  %[x_est(1); velocityx; 0; x_est(4);
velocityy; 0];


    future_state_x = [future_state_x, y(1)];
    future_state_y = [future_state_y, y(2)];


end



StartingLatSeconds = (StartingLat*60 - fix(StartingLat*60))*60;
```

```
StartingLonSeconds = (StartingLon*60 - fix(StartingLon*60))*60;
FinalLatSeconds = (FinalLat*60 - fix(FinalLat*60))*60;
FinalLonSeconds = (FinalLon*60 - fix(FinalLon*60))*60;


slopelat = (FinalLatSeconds - StartingLatSeconds)/(count-1);
slopelon = (FinalLonSeconds - StartingLonSeconds)/(count-1);


FiveMinPathError = ((future_state_x(301)/3600*FeetPerDegreeLat)-
(slopelat*(301)+StartingLatSeconds)/3600*FeetPerDegreeLat)^2 +
((future_state_y(301)/3600*FeetPerDegreeLat)-
(slopelon*(301)+StartingLonSeconds)/3600*FeetPerDegreeLon)^2;

FiveMinPathError = sqrt(FiveMinPathError)

actual_vector = [(301) (future_state_x(301) -
future_state_x(1))/3600*FeetPerDegreeLat (future_state_y(301) -
future_state_y(1))/3600*FeetPerDegreeLat];

predicted_vector = [(301)
((slopelat*301+StartingLatSeconds)/3600*FeetPerDegreeLat-
(slopelat*1+StartingLatSeconds)/3600*FeetPerDegreeLat)
((slopelon*301+StartingLonSeconds)/3600*FeetPerDegreeLon -
(slopelon*1+StartingLonSeconds)/3600*FeetPerDegreeLon)];

angle =
(atan2(norm(cross(actual_vector,predicted_vector)),dot(actual_vector,pr
edicted_vector))*180/2/3.14159)

%%%%%%%%%%%%%%%%%%%%%%%%%%%Graph the data%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
time = 1:T:(count-1);
xData = linspace(starttime, endtime, count-1);
scrsz = get(0,'ScreenSize');
plot3(time, (slopelat*time+StartingLatSeconds)/3600*FeetPerDegreeLat,
(slopelon*time+StartingLonSeconds)/3600*FeetPerDegreeLon, '-g',
'Linewidth',3)
hold on
plot3(time, predic_state_x/3600*FeetPerDegreeLat,
predic_state_y/3600*FeetPerDegreeLon, '-r.', 'Linewidth',3)
plot3(time, measured_state_x/3600*FeetPerDegreeLat,
measured_state_y/3600*FeetPerDegreeLon, 'kx ', 'Linewidth',3)
hold off
legend('Actual', 'Predicted', 'Measured');
axis([0, count-1, 1600, 2000, -2500, -2200]);
title('Run 27 Actual Path, Predicted Path, and Measurements');
xlabel('Time')
ylabel('Latitude (Feet)')
zlabel('Longitude (Feet)')
%axis auto
rotate3d on
grid on
```

```
overtime = count:T:count+300;
scrsz = get(0,'ScreenSize');
figure(2)
plot3(overtime,
(slopelat*overtime+StartingLatSeconds)/3600*FeetPerDegreeLat,
(slopelon*overtime+StartingLonSeconds)/3600*FeetPerDegreeLon, '-g',
'Linewidth',3)
hold on
plot3(overtime, future_state_x/3600*FeetPerDegreeLat,
future_state_y/3600*FeetPerDegreeLon, '-r.', 'Linewidth',3)
hold off
legend('Actual', 'Predicted', 'Measured');
title('Run 27 Projected Actual Path and Predicted Path');
xlabel('Time')
ylabel('Latitude')
zlabel('Longitude')
axis([count, count+300, 16, 20, -30, -26]);
axis auto
rotate3d on
grid on
measurement;
```

Measurement Data

Instructions

To use this data, save the Textron data as Textron.txt, the QinetiQ data as QinetiQ.txt, and the Southwest Microwave data as Southwest.txt. Place these three text files in the same directory as the Matlab code, and then run the code.

Run 1

QinetiQ 7/11/2012 1:24:04 PM LatLon: 35.93798904875162 -84.2748684719682 4 90
QinetiQ 7/11/2012 1:24:05 PM LatLon: 35.93808851480986 -84.27481600937807 3 60
QinetiQ 7/11/2012 1:24:07 PM LatLon: 35.93808851480986 -84.27481600937807 3 50
QinetiQ 7/11/2012 1:24:07 PM LatLon: 35.93805535944111 -84.2748334969326 4 90
QinetiQ 7/11/2012 1:24:08 PM LatLon: 35.93805535944111 -84.2748334969326 4 90
QinetiQ 7/11/2012 1:24:08 PM LatLon: 35.93808851480986 -84.27481600937807 3 60
QinetiQ 7/11/2012 1:24:09 PM LatLon: 35.93805535944111 -84.2748334969326 4 90
QinetiQ 7/11/2012 1:24:10 PM LatLon: 35.93805535944111 -84.2748334969326 4 90
QinetiQ 7/11/2012 1:24:10 PM LatLon: 35.93810690584708 -84.27483395059356 4 25
QinetiQ 7/11/2012 1:24:11 PM LatLon: 35.93810690584708 -84.27483395059356 4 30
QinetiQ 7/11/2012 1:24:13 PM LatLon: 35.938178566882236 -84.27479594345041 4 35
QinetiQ 7/11/2012 1:24:13 PM LatLon: 35.938178566882236 -84.27479594345041 4 40
QinetiQ 7/11/2012 1:24:14 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:24:15 PM LatLon: 35.938178566882236 -84.27479594345041 4 50
QinetiQ 7/11/2012 1:24:17 PM LatLon: 35.93815482549134 -84.27478103425042 4 55
QinetiQ 7/11/2012 1:24:18 PM LatLon: 35.93815482549134 -84.27478103425042 4 60
QinetiQ 7/11/2012 1:24:18 PM LatLon: 35.938214397372484 -84.27477693986506 4 60
QinetiQ 7/11/2012 1:24:19 PM LatLon: 35.938221136147014 -84.27474605907219 3 60
QinetiQ 7/11/2012 1:24:19 PM LatLon: 35.938221136147014 -84.27474605907219 4 65
QinetiQ 7/11/2012 1:24:20 PM LatLon: 35.938221136147014 -84.27474605907219 3 50
QinetiQ 7/11/2012 1:24:20 PM LatLon: 35.938221136147014 -84.27474605907219 4 70
QinetiQ 7/11/2012 1:24:21 PM LatLon: 35.938221136147014 -84.27474605907219 4 75
QinetiQ 7/11/2012 1:24:22 PM LatLon: 35.938221136147014 -84.27474605907219 4 80
QinetiQ 7/11/2012 1:24:23 PM LatLon: 35.938221136147014 -84.27474605907219 4 80
QinetiQ 7/11/2012 1:24:24 PM LatLon: 35.93825429150053 -84.27472857144544 4 80

Southwest 7/11/2012 1:24:08 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:24:09 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:24:10 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:24:25 PM LatLon 35.9381671608124 -84.2747781992666

Southwest 7/11/2012 1:24:26 PM LatLon 35.9381671608124 -84.2747781992666
Southwest 7/11/2012 1:24:27 PM LatLon 35.9381671608124 -84.2747781992666
Southwest 7/11/2012 1:24:30 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:24:31 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:24:32 PM LatLon 35.9381963059109 -84.2747692241452

Textron 7/11/2012 1:24:10 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:24:15 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:24:15 PM LatLon 35.9384433333333 -84.27469
Textron 7/11/2012 1:24:20 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:24:20 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:24:21 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:24:22 PM LatLon 35.9383733333333 -84.2745866666667

Run 2

QinetiQ 7/11/2012 1:25:05 PM LatLon: 35.938214397372484 -84.27477693986506 2 50
QinetiQ 7/11/2012 1:25:07 PM LatLon: 35.938178566882236 -84.27479594345041 3 50
QinetiQ 7/11/2012 1:25:09 PM LatLon: 35.938214397372484 -84.27477693986506 2 50
QinetiQ 7/11/2012 1:25:13 PM LatLon: 35.938178566882236 -84.27479594345041 4 25
QinetiQ 7/11/2012 1:25:14 PM LatLon: 35.93810690584708 -84.27483395059356 2 50
QinetiQ 7/11/2012 1:25:14 PM LatLon: 35.938178566882236 -84.27479594345041 4 25
QinetiQ 7/11/2012 1:25:15 PM LatLon: 35.938178566882236 -84.27479594345041 4 30
QinetiQ 7/11/2012 1:25:16 PM LatLon: 35.938178566882236 -84.27479594345041 4 30
QinetiQ 7/11/2012 1:25:17 PM LatLon: 35.93808851480986 -84.27481600937807 4 30
QinetiQ 7/11/2012 1:25:18 PM LatLon: 35.93814273635876 -84.27481494703454 4 35
QinetiQ 7/11/2012 1:25:19 PM LatLon: 35.93808851480986 -84.27481600937807 4 35
QinetiQ 7/11/2012 1:25:20 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:25:20 PM LatLon: 35.93805535944111 -84.2748334969326 4 40
QinetiQ 7/11/2012 1:25:21 PM LatLon: 35.938071075314646 -84.2748529541447 4 45
QinetiQ 7/11/2012 1:25:22 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:25:24 PM LatLon: 35.93805535944111 -84.2748334969326 4 45
QinetiQ 7/11/2012 1:25:24 PM LatLon: 35.93805535944111 -84.2748334969326 4 50
QinetiQ 7/11/2012 1:25:25 PM LatLon: 35.93805535944111 -84.2748334969326 2 50
QinetiQ 7/11/2012 1:25:25 PM LatLon: 35.93805535944111 -84.2748334969326 4 55
QinetiQ 7/11/2012 1:25:26 PM LatLon: 35.93805535944111 -84.2748334969326 4 55
QinetiQ 7/11/2012 1:25:27 PM LatLon: 35.93802220409773 -84.27485098445771 4 60
QinetiQ 7/11/2012 1:25:27 PM LatLon: 35.93798904875162 -84.2748684719682 3 50
QinetiQ 7/11/2012 1:25:28 PM LatLon: 35.93802220409773 -84.27485098445771 4 60

Southwest 7/11/2012 1:25:11 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:25:12 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:25:18 PM LatLon 35.9381671608124 -84.2747781992666
Southwest 7/11/2012 1:25:19 PM LatLon 35.9381671608124 -84.2747781992666
Southwest 7/11/2012 1:25:20 PM LatLon 35.9380214353097 -84.2748230747746

Southwest 7/11/2012 1:25:20 PM LatLon 35.9381671608124 -84.2747781992666
Southwest 7/11/2012 1:25:23 PM LatLon 35.9380214353097 -84.2748230747746

Textron 7/11/2012 1:25:24 PM LatLon 35.9379816666667 -84.2749333333333

Run 3

QinetiQ 7/11/2012 1:26:01 PM LatLon: 35.93805535944111 -84.2748334969326 2 50
QinetiQ 7/11/2012 1:26:04 PM LatLon: 35.93808851480986 -84.27481600937807 3 50
QinetiQ 7/11/2012 1:26:05 PM LatLon: 35.938071075314646 -84.2748529541447 3 60
QinetiQ 7/11/2012 1:26:07 PM LatLon: 35.93805535944111 -84.2748334969326 3 60
QinetiQ 7/11/2012 1:26:08 PM LatLon: 35.93808851480986 -84.27481600937807 3 60
QinetiQ 7/11/2012 1:26:09 PM LatLon: 35.93808851480986 -84.27481600937807 3 50
QinetiQ 7/11/2012 1:26:11 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:26:11 PM LatLon: 35.93814273635876 -84.27481494703454 4 25
QinetiQ 7/11/2012 1:26:11 PM LatLon: 35.93814273635876 -84.27481494703454 4 30
QinetiQ 7/11/2012 1:26:12 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:26:12 PM LatLon: 35.93814273635876 -84.27481494703454 4 35
QinetiQ 7/11/2012 1:26:12 PM LatLon: 35.93812167014452 -84.27479852182547 2 50
QinetiQ 7/11/2012 1:26:13 PM LatLon: 35.93812167014452 -84.27479852182547 2 50
QinetiQ 7/11/2012 1:26:13 PM LatLon: 35.93814273635876 -84.27481494703454 4 35
QinetiQ 7/11/2012 1:26:15 PM LatLon: 35.93814273635876 -84.27481494703454 4 40
QinetiQ 7/11/2012 1:26:16 PM LatLon: 35.938214397372484 -84.27477693986506 4 45
QinetiQ 7/11/2012 1:26:17 PM LatLon: 35.938178566882236 -84.27479594345041 4 50
QinetiQ 7/11/2012 1:26:19 PM LatLon: 35.938187980820544 -84.27476354666862 4 50
QinetiQ 7/11/2012 1:26:19 PM LatLon: 35.938187980820544 -84.27476354666862 2 50
QinetiQ 7/11/2012 1:26:19 PM LatLon: 35.938178566882236 -84.27479594345041 4 55
QinetiQ 7/11/2012 1:26:20 PM LatLon: 35.938221136147014 -84.27474605907219 3 50
QinetiQ 7/11/2012 1:26:20 PM LatLon: 35.938214397372484 -84.27477693986506 4 60
QinetiQ 7/11/2012 1:26:21 PM LatLon: 35.938214397372484 -84.27477693986506 4 65
QinetiQ 7/11/2012 1:26:22 PM LatLon: 35.938221136147014 -84.27474605907219 2 50
QinetiQ 7/11/2012 1:26:23 PM LatLon: 35.93825429150053 -84.27472857144544 4 65
QinetiQ 7/11/2012 1:26:23 PM LatLon: 35.938221136147014 -84.27474605907219 2 50
QinetiQ 7/11/2012 1:26:24 PM LatLon: 35.938221136147014 -84.27474605907219 4 70
QinetiQ 7/11/2012 1:26:24 PM LatLon: 35.93825429150053 -84.27472857144544 4 70

Southwest 7/11/2012 1:26:07 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:26:08 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:26:09 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:26:12 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:26:14 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:26:31 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:26:33 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:26:35 PM LatLon 35.9381963059109 -84.2747692241452

Southwest 7/11/2012 1:26:37 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:26:39 PM LatLon 35.9381963059109 -84.2747692241452

Textron 7/11/2012 1:26:19 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:26:24 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:26:30 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:26:36 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:26:37 PM LatLon 35.9383733333333 -84.2745866666667
Textron 7/11/2012 1:26:40 PM LatLon 35.938445 -84.2745416666667

Run 4

QinetiQ 7/11/2012 1:27:05 PM LatLon: 35.93825022788952 -84.27475793624663 2 50
QinetiQ 7/11/2012 1:27:07 PM LatLon: 35.938187980820544 -84.27476354666862 2 50
QinetiQ 7/11/2012 1:27:16 PM LatLon: 35.93812167014452 -84.27479852182547 2 50
QinetiQ 7/11/2012 1:27:19 PM LatLon: 35.93805535944111 -84.2748334969326 4 25
QinetiQ 7/11/2012 1:27:20 PM LatLon: 35.93805535944111 -84.2748334969326 4 30
QinetiQ 7/11/2012 1:27:20 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:27:21 PM LatLon: 35.93805535944111 -84.2748334969326 4 35
QinetiQ 7/11/2012 1:27:22 PM LatLon: 35.93802220409773 -84.27485098445771 4 35
QinetiQ 7/11/2012 1:27:23 PM LatLon: 35.93805535944111 -84.2748334969326 2 50
QinetiQ 7/11/2012 1:27:23 PM LatLon: 35.93799941428413 -84.2748909611722 4 40
QinetiQ 7/11/2012 1:27:24 PM LatLon: 35.93799941428413 -84.2748909611722 2 50
QinetiQ 7/11/2012 1:27:25 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:27:25 PM LatLon: 35.93802220409773 -84.27485098445771 4 55
QinetiQ 7/11/2012 1:27:26 PM LatLon: 35.93805535944111 -84.2748334969326 2 50
QinetiQ 7/11/2012 1:27:27 PM LatLon: 35.93805535944111 -84.2748334969326 4 60
QinetiQ 7/11/2012 1:27:27 PM LatLon: 35.93802220409773 -84.27485098445771 3 50
QinetiQ 7/11/2012 1:27:28 PM LatLon: 35.93799941428413 -84.2748909611722 4 65

Southwest 7/11/2012 1:27:04 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:27:05 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:27:06 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:27:09 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:27:10 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:27:11 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:27:22 PM LatLon 35.9380214353097 -84.2748230747746
Southwest 7/11/2012 1:27:23 PM LatLon 35.9380214353097 -84.2748230747746
Southwest 7/11/2012 1:27:34 PM LatLon 35.9380214353097 -84.2748230747746

Run 5

QinetiQ 7/11/2012 1:28:02 PM LatLon: 35.93805535944111 -84.2748334969326 3 60

QinetiQ 7/11/2012 1:28:04 PM LatLon: 35.93802220409773 -84.27485098445771 3 60
QinetiQ 7/11/2012 1:28:06 PM LatLon: 35.93808851480986 -84.27481600937807 3 60
QinetiQ 7/11/2012 1:28:07 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:28:09 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:28:10 PM LatLon: 35.93808851480986 -84.27481600937807 4 25
QinetiQ 7/11/2012 1:28:10 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:28:11 PM LatLon: 35.93808851480986 -84.27481600937807 4 30
QinetiQ 7/11/2012 1:28:11 PM LatLon: 35.93808851480986 -84.27481600937807 2 50
QinetiQ 7/11/2012 1:28:12 PM LatLon: 35.93808851480986 -84.27481600937807 4 30
QinetiQ 7/11/2012 1:28:13 PM LatLon: 35.93812167014452 -84.27479852182547 2 50
QinetiQ 7/11/2012 1:28:14 PM LatLon: 35.93815482549134 -84.27478103425042 4 35
QinetiQ 7/11/2012 1:28:15 PM LatLon: 35.93812167014452 -84.27479852182547 4 40
QinetiQ 7/11/2012 1:28:15 PM LatLon: 35.93815482549134 -84.27478103425042 4 40
QinetiQ 7/11/2012 1:28:16 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:28:16 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:28:18 PM LatLon: 35.938214397372484 -84.27477693986506 4 50
QinetiQ 7/11/2012 1:28:18 PM LatLon: 35.938187980820544 -84.27476354666862 2 50
QinetiQ 7/11/2012 1:28:19 PM LatLon: 35.938187980820544 -84.27476354666862 4 55
QinetiQ 7/11/2012 1:28:19 PM LatLon: 35.938187980820544 -84.27476354666862 4 55
QinetiQ 7/11/2012 1:28:19 PM LatLon: 35.938221136147014 -84.27474605907219 3 60
QinetiQ 7/11/2012 1:28:21 PM LatLon: 35.938221136147014 -84.27474605907219 4 60
QinetiQ 7/11/2012 1:28:21 PM LatLon: 35.938221136147014 -84.27474605907219 3 50
QinetiQ 7/11/2012 1:28:22 PM LatLon: 35.938221136147014 -84.27474605907219 4 65
QinetiQ 7/11/2012 1:28:22 PM LatLon: 35.93825022788952 -84.27475793624663 4 65
QinetiQ 7/11/2012 1:28:23 PM LatLon: 35.938221136147014 -84.27474605907219 4 70
QinetiQ 7/11/2012 1:28:24 PM LatLon: 35.938221136147014 -84.27474605907219 4 75
QinetiQ 7/11/2012 1:28:25 PM LatLon: 35.938221136147014 -84.27474605907219 4 75
QinetiQ 7/11/2012 1:28:26 PM LatLon: 35.93825429150053 -84.27472857144544 3 50
QinetiQ 7/11/2012 1:28:26 PM LatLon: 35.93825022788952 -84.27475793624663 4 80

Southwest 7/11/2012 1:28:08 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:28:09 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:28:20 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:28:31 PM LatLon 35.9381817333617 -84.2747737117067

Textron 7/11/2012 1:28:03 PM LatLon 35.9379816666667 -84.2749333333333
Textron 7/11/2012 1:28:12 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:28:15 PM LatLon 35.9383733333333 -84.2745866666667
Textron 7/11/2012 1:28:16 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:28:20 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:28:22 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:28:23 PM LatLon 35.9381366666667 -84.2747166666667

Run 6

QinetiQ 7/11/2012 1:43:01 PM LatLon: 35.93825429150053 -84.27472857144544 4 50
QinetiQ 7/11/2012 1:43:02 PM LatLon: 35.93828744682153 -84.27471108381981 4 50
QinetiQ 7/11/2012 1:43:05 PM LatLon: 35.93825022788952 -84.27475793624663 4 50
QinetiQ 7/11/2012 1:43:06 PM LatLon: 35.938221136147014 -84.27474605907219 4 60
QinetiQ 7/11/2012 1:43:10 PM LatLon: 35.938178566882236 -84.27479594345041 4 50
QinetiQ 7/11/2012 1:43:10 PM LatLon: 35.938214397372484 -84.2747693986506 4 25
QinetiQ 7/11/2012 1:43:11 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:43:11 PM LatLon: 35.93815482549134 -84.27478103425042 4 30
QinetiQ 7/11/2012 1:43:12 PM LatLon: 35.938178566882236 -84.27479594345041 4 35
QinetiQ 7/11/2012 1:43:12 PM LatLon: 35.93815482549134 -84.27478103425042 4 35
QinetiQ 7/11/2012 1:43:14 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:43:14 PM LatLon: 35.93815482549134 -84.27478103425042 4 40
QinetiQ 7/11/2012 1:43:15 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:43:15 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:43:15 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:43:16 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:43:17 PM LatLon: 35.93810690584708 -84.27483395059356 4 50
QinetiQ 7/11/2012 1:43:18 PM LatLon: 35.93812167014452 -84.27479852182547 4 55
QinetiQ 7/11/2012 1:43:19 PM LatLon: 35.93812167014452 -84.27479852182547 4 55
QinetiQ 7/11/2012 1:43:20 PM LatLon: 35.93812167014452 -84.27479852182547 4 55
QinetiQ 7/11/2012 1:43:21 PM LatLon: 35.93812167014452 -84.27479852182547 4 55
QinetiQ 7/11/2012 1:43:22 PM LatLon: 35.93803524480099 -84.27487195766702 4 55
QinetiQ 7/11/2012 1:43:23 PM LatLon: 35.93802220409773 -84.27485098445771 4 50
QinetiQ 7/11/2012 1:43:23 PM LatLon: 35.93803524480099 -84.27487195766702 4 60
QinetiQ 7/11/2012 1:43:24 PM LatLon: 35.93805535944111 -84.2748334969326 4 60
QinetiQ 7/11/2012 1:43:24 PM LatLon: 35.93798904875162 -84.2748684719682 4 65
QinetiQ 7/11/2012 1:43:25 PM LatLon: 35.93796358376027 -84.27490996466223 4 70
QinetiQ 7/11/2012 1:43:26 PM LatLon: 35.93802220409773 -84.27485098445771 4 60
QinetiQ 7/11/2012 1:43:26 PM LatLon: 35.93796358376027 -84.27490996466223 4 75

Southwest 7/11/2012 1:43:04 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:43:05 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:43:06 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:43:09 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:43:10 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:43:12 PM LatLon 35.9381817333617 -84.2747737117067

Textron 7/11/2012 1:42:59 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:43:02 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:43:15 PM LatLon 35.938115 -84.2748

Run 7

QinetiQ 7/11/2012 1:44:05 PM LatLon: 35.93802220409773 -84.27485098445771 4 60
QinetiQ 7/11/2012 1:44:07 PM LatLon: 35.93805535944111 -84.2748334969326 4 50
QinetiQ 7/11/2012 1:44:08 PM LatLon: 35.93803524480099 -84.27487195766702 4 60
QinetiQ 7/11/2012 1:44:09 PM LatLon: 35.93798904875162 -84.2748684719682 4 90
QinetiQ 7/11/2012 1:44:10 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:44:13 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:44:14 PM LatLon: 35.938071075314646 -84.2748529541447 4 25
QinetiQ 7/11/2012 1:44:14 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:44:15 PM LatLon: 35.938071075314646 -84.2748529541447 4 30
QinetiQ 7/11/2012 1:44:15 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:44:16 PM LatLon: 35.938071075314646 -84.2748529541447 4 30
QinetiQ 7/11/2012 1:44:16 PM LatLon: 35.93812167014452 -84.27479852182547 4 30
QinetiQ 7/11/2012 1:44:17 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:44:17 PM LatLon: 35.93808851480986 -84.27481600937807 4 35
QinetiQ 7/11/2012 1:44:18 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:44:18 PM LatLon: 35.93808851480986 -84.27481600937807 4 35
QinetiQ 7/11/2012 1:44:19 PM LatLon: 35.938214397372484 -84.27477693986506 4 35
QinetiQ 7/11/2012 1:44:20 PM LatLon: 35.938187980820544 -84.27476354666862 4 50
QinetiQ 7/11/2012 1:44:20 PM LatLon: 35.93814273635876 -84.27481494703454 4 40
QinetiQ 7/11/2012 1:44:21 PM LatLon: 35.93814273635876 -84.27481494703454 4 45
QinetiQ 7/11/2012 1:44:21 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:44:23 PM LatLon: 35.938214397372484 -84.27477693986506 4 45
QinetiQ 7/11/2012 1:44:23 PM LatLon: 35.938214397372484 -84.27477693986506 4 50
QinetiQ 7/11/2012 1:44:24 PM LatLon: 35.938221136147014 -84.27474605907219 4 50
QinetiQ 7/11/2012 1:44:24 PM LatLon: 35.938214397372484 -84.27477693986506 4 55
QinetiQ 7/11/2012 1:44:25 PM LatLon: 35.938214397372484 -84.27477693986506 4 60
QinetiQ 7/11/2012 1:44:26 PM LatLon: 35.93825429150053 -84.27472857144544 4 50

Southwest 7/11/2012 1:44:11 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:44:12 PM LatLon 35.937948572552 -84.2748455124666
Southwest 7/11/2012 1:44:13 PM LatLon 35.937948572552 -84.2748455124666

Textron 7/11/2012 1:44:15 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:44:19 PM LatLon 35.9383733333333 -84.2745866666667
Textron 7/11/2012 1:44:20 PM LatLon 35.938445 -84.2745416666667

Run 8

QinetiQ 7/11/2012 1:45:05 PM LatLon: 35.93828744682153 -84.27471108381981 4 50
QinetiQ 7/11/2012 1:45:07 PM LatLon: 35.938221136147014 -84.27474605907219 4 50
QinetiQ 7/11/2012 1:45:08 PM LatLon: 35.938221136147014 -84.27474605907219 4 50
QinetiQ 7/11/2012 1:45:09 PM LatLon: 35.938221136147014 -84.27474605907219 4 60
QinetiQ 7/11/2012 1:45:11 PM LatLon: 35.938187980820544 -84.27476354666862 4 60

QinetiQ 7/11/2012 1:45:13 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:45:14 PM LatLon: 35.938178566882236 -84.27479594345041 4 25
QinetiQ 7/11/2012 1:45:15 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:45:16 PM LatLon: 35.938178566882236 -84.27479594345041 4 30
QinetiQ 7/11/2012 1:45:16 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:45:17 PM LatLon: 35.93810690584708 -84.27483395059356 4 35
QinetiQ 7/11/2012 1:45:17 PM LatLon: 35.93810690584708 -84.27483395059356 4 35
QinetiQ 7/11/2012 1:45:19 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:45:19 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:45:20 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:45:21 PM LatLon: 35.93815482549134 -84.27478103425042 4 50
QinetiQ 7/11/2012 1:45:22 PM LatLon: 35.93805535944111 -84.2748334969326 4 50
QinetiQ 7/11/2012 1:45:22 PM LatLon: 35.93805535944111 -84.2748334969326 4 50
QinetiQ 7/11/2012 1:45:23 PM LatLon: 35.93802220409773 -84.27485098445771 4 60
QinetiQ 7/11/2012 1:45:24 PM LatLon: 35.93803524480099 -84.27487195766702 4 55
QinetiQ 7/11/2012 1:45:24 PM LatLon: 35.93805535944111 -84.2748334969326 4 50
QinetiQ 7/11/2012 1:45:26 PM LatLon: 35.93805535944111 -84.2748334969326 4 60
QinetiQ 7/11/2012 1:45:26 PM LatLon: 35.938071075314646 -84.2748529541447 4 65
QinetiQ 7/11/2012 1:45:27 PM LatLon: 35.93803524480099 -84.27487195766702 4 65
QinetiQ 7/11/2012 1:45:28 PM LatLon: 35.93805535944111 -84.2748334969326 4 60
QinetiQ 7/11/2012 1:45:28 PM LatLon: 35.938071075314646 -84.2748529541447 4 70
QinetiQ 7/11/2012 1:45:28 PM LatLon: 35.93805535944111 -84.2748334969326 4 69
QinetiQ 7/11/2012 1:45:29 PM LatLon: 35.93803524480099 -84.27487195766702 4 75
QinetiQ 7/11/2012 1:45:29 PM LatLon: 35.93805535944111 -84.2748334969326 4 75
QinetiQ 7/11/2012 1:45:29 PM LatLon: 35.93798904875162 -84.2748684719682 4 90
QinetiQ 7/11/2012 1:45:29 PM LatLon: 35.93805535944111 -84.2748334969326 4 60
QinetiQ 7/11/2012 1:45:30 PM LatLon: 35.93802220409773 -84.27485098445771 4 90

Southwest 7/11/2012 1:45:08 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:45:09 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:45:10 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:45:16 PM LatLon 35.9381817333617 -84.2747737117067
Southwest 7/11/2012 1:45:17 PM LatLon 35.9381817333617 -84.2747737117067

Textron 7/11/2012 1:45:05 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:45:17 PM LatLon 35.938115 -84.2748

Run 9

QinetiQ 7/11/2012 1:46:06 PM LatLon: 35.93805535944111 -84.2748334969326 4 60
QinetiQ 7/11/2012 1:46:08 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:46:10 PM LatLon: 35.93808851480986 -84.27481600937807 4 60
QinetiQ 7/11/2012 1:46:11 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:46:14 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:46:14 PM LatLon: 35.93805535944111 -84.2748334969326 4 25

QinetiQ 7/11/2012 1:46:14 PM LatLon: 35.93808851480986 -84.27481600937807 4 30
QinetiQ 7/11/2012 1:46:15 PM LatLon: 35.93808851480986 -84.27481600937807 4 35
QinetiQ 7/11/2012 1:46:15 PM LatLon: 35.93812167014452 -84.27479852182547 4 60
QinetiQ 7/11/2012 1:46:16 PM LatLon: 35.93812167014452 -84.27479852182547 4 40
QinetiQ 7/11/2012 1:46:18 PM LatLon: 35.93815482549134 -84.27478103425042 4 60
QinetiQ 7/11/2012 1:46:19 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:46:19 PM LatLon: 35.938178566882236 -84.27479594345041 4 50
QinetiQ 7/11/2012 1:46:20 PM LatLon: 35.938187980820544 -84.27476354666862 4 60
QinetiQ 7/11/2012 1:46:20 PM LatLon: 35.938178566882236 -84.27479594345041 4 55
QinetiQ 7/11/2012 1:46:21 PM LatLon: 35.938178566882236 -84.27479594345041 4 60
QinetiQ 7/11/2012 1:46:22 PM LatLon: 35.938187980820544 -84.27476354666862 4 65
QinetiQ 7/11/2012 1:46:23 PM LatLon: 35.938221136147014 -84.27474605907219 4 60
QinetiQ 7/11/2012 1:46:24 PM LatLon: 35.93825022788952 -84.27475793624663 4 70
QinetiQ 7/11/2012 1:46:24 PM LatLon: 35.938214397372484 -84.27477693986506 4 75
QinetiQ 7/11/2012 1:46:24 PM LatLon: 35.938221136147014 -84.27474605907219 4 50
QinetiQ 7/11/2012 1:46:26 PM LatLon: 35.938187980820544 -84.27476354666862 4 80
QinetiQ 7/11/2012 1:46:26 PM LatLon: 35.93825429150053 -84.27472857144544 4 50
QinetiQ 7/11/2012 1:46:27 PM LatLon: 35.938187980820544 -84.27476354666862 4 85
QinetiQ 7/11/2012 1:46:27 PM LatLon: 35.93825022788952 -84.27475793624663 4 85
QinetiQ 7/11/2012 1:46:28 PM LatLon: 35.93825022788952 -84.27475793624663 4 90
QinetiQ 7/11/2012 1:46:29 PM LatLon: 35.938286058373315 -84.27473893262695 4 90

Southwest 7/11/2012 1:46:14 PM LatLon 35.9379631451039 -84.2748410249315
Southwest 7/11/2012 1:46:15 PM LatLon 35.9379631451039 -84.2748410249315
Southwest 7/11/2012 1:46:16 PM LatLon 35.9379631451039 -84.2748410249315

Textron 7/11/2012 1:46:05 PM LatLon 35.9379816666667 -84.2749333333333
Textron 7/11/2012 1:46:07 PM LatLon 35.93882 -84.274415
Textron 7/11/2012 1:46:15 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:46:18 PM LatLon 35.9384433333333 -84.27469
Textron 7/11/2012 1:46:19 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:46:23 PM LatLon 35.9383733333333 -84.2745866666667

Run 10

QinetiQ 7/11/2012 1:47:10 PM LatLon: 35.938187980820544 -84.27476354666862 4 50
QinetiQ 7/11/2012 1:47:10 PM LatLon: 35.938187980820544 -84.27476354666862 4 25
QinetiQ 7/11/2012 1:47:10 PM LatLon: 35.93815482549134 -84.27478103425042 4 60
QinetiQ 7/11/2012 1:47:11 PM LatLon: 35.938187980820544 -84.27476354666862 4 30
QinetiQ 7/11/2012 1:47:12 PM LatLon: 35.938178566882236 -84.27479594345041 4 35
QinetiQ 7/11/2012 1:47:13 PM LatLon: 35.93810690584708 -84.27483395059356 4 40
QinetiQ 7/11/2012 1:47:13 PM LatLon: 35.93815482549134 -84.27478103425042 4 60
QinetiQ 7/11/2012 1:47:15 PM LatLon: 35.93815482549134 -84.27478103425042 4 60
QinetiQ 7/11/2012 1:47:15 PM LatLon: 35.93815482549134 -84.27478103425042 4 45
QinetiQ 7/11/2012 1:47:16 PM LatLon: 35.93808851480986 -84.27481600937807 4 50

QinetiQ 7/11/2012 1:47:16 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:47:17 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:47:17 PM LatLon: 35.93810690584708 -84.27483395059356 4 50
QinetiQ 7/11/2012 1:47:18 PM LatLon: 35.93808851480986 -84.27481600937807 4 55
QinetiQ 7/11/2012 1:47:20 PM LatLon: 35.93802220409773 -84.27485098445771 4 60
QinetiQ 7/11/2012 1:47:20 PM LatLon: 35.93802220409773 -84.27485098445771 4 50
QinetiQ 7/11/2012 1:47:21 PM LatLon: 35.93805535944111 -84.2748334969326 4 65
QinetiQ 7/11/2012 1:47:21 PM LatLon: 35.93805535944111 -84.2748334969326 4 50
QinetiQ 7/11/2012 1:47:22 PM LatLon: 35.93805535944111 -84.2748334969326 4 65
QinetiQ 7/11/2012 1:47:23 PM LatLon: 35.93808851480986 -84.27481600937807 4 50
QinetiQ 7/11/2012 1:47:23 PM LatLon: 35.93802220409773 -84.27485098445771 4 70
QinetiQ 7/11/2012 1:47:24 PM LatLon: 35.93808851480986 -84.27481600937807 4 70
QinetiQ 7/11/2012 1:47:24 PM LatLon: 35.93805535944111 -84.2748334969326 4 75
QinetiQ 7/11/2012 1:47:25 PM LatLon: 35.93805535944111 -84.2748334969326 4 80
QinetiQ 7/11/2012 1:47:26 PM LatLon: 35.93805535944111 -84.2748334969326 4 60
QinetiQ 7/11/2012 1:47:26 PM LatLon: 35.93805535944111 -84.2748334969326 4 85
QinetiQ 7/11/2012 1:47:27 PM LatLon: 35.93802220409773 -84.27485098445771 4 90
QinetiQ 7/11/2012 1:47:27 PM LatLon: 35.93802220409773 -84.27485098445771 4 90
QinetiQ 7/11/2012 1:47:28 PM LatLon: 35.93802220409773 -84.27485098445771 4 90

Southwest 7/11/2012 1:47:05 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:47:06 PM LatLon 35.9381963059109 -84.2747692241452
Southwest 7/11/2012 1:47:07 PM LatLon 35.9381963059109 -84.2747692241452

Textron 7/11/2012 1:47:17 PM LatLon 35.938115 -84.2748

Run 11

QinetiQ 7/11/2012 1:32:16 PM LatLon: 35.93837696949183 -84.27451935436109 2 50

Southwest 7/11/2012 1:32:07 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:32:08 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:32:09 PM LatLon 35.9383833674071 -84.2747030181606

Textron 7/11/2012 1:32:01 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:32:05 PM LatLon 35.9384433333333 -84.27469
Textron 7/11/2012 1:32:09 PM LatLon 35.93882 -84.274415
Textron 7/11/2012 1:32:12 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:32:15 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:32:18 PM LatLon 35.9383733333333 -84.2745866666667

Run 12

QinetiQ 7/11/2012 1:33:07 PM LatLon: 35.93837696949183 -84.27451935436109 2 50
QinetiQ 7/11/2012 1:33:10 PM LatLon: 35.93840573251469 -84.27456196823405 2 50

QinetiQ 7/11/2012 1:33:10 PM LatLon: 35.93842101580825 -84.27452385652867 3 50
QinetiQ 7/11/2012 1:33:12 PM LatLon: 35.93840573251469 -84.27456196823405 2 50
QinetiQ 7/11/2012 1:33:15 PM LatLon: 35.938388157535954 -84.27460901820079 3 50
QinetiQ 7/11/2012 1:33:15 PM LatLon: 35.938388157535954 -84.27460901820079 4 25
QinetiQ 7/11/2012 1:33:16 PM LatLon: 35.9384293803663 -84.27466291792898 4 30
QinetiQ 7/11/2012 1:33:17 PM LatLon: 35.938386912797945 -84.27465862083957 2 50
QinetiQ 7/11/2012 1:33:17 PM LatLon: 35.9384293803663 -84.27466291792898 4 35
QinetiQ 7/11/2012 1:33:18 PM LatLon: 35.93839354986539 -84.27468192163317 4 40
QinetiQ 7/11/2012 1:33:20 PM LatLon: 35.9384293803663 -84.27466291792898 4 40

Southwest 7/11/2012 1:33:17 PM LatLon 35.9383699856743 -84.2747114395112
Southwest 7/11/2012 1:33:19 PM LatLon 35.9383699856743 -84.2747114395112
Southwest 7/11/2012 1:33:20 PM LatLon 35.9383699856743 -84.2747114395112

Textron 7/11/2012 1:33:06 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:33:15 PM LatLon 35.938115 -84.2748

Run 13

QinetiQ 7/11/2012 1:34:04 PM LatLon: 35.93846332183767 -84.27463934110897 2 50
QinetiQ 7/11/2012 1:34:05 PM LatLon: 35.93834285450666 -84.2746321292002 3 50
QinetiQ 7/11/2012 1:34:07 PM LatLon: 35.93834285450666 -84.2746321292002 3 60
QinetiQ 7/11/2012 1:34:08 PM LatLon: 35.938388157535954 -84.27460901820079 3 60
QinetiQ 7/11/2012 1:34:09 PM LatLon: 35.93840573251469 -84.27456196823405 2 50
QinetiQ 7/11/2012 1:34:11 PM LatLon: 35.93842101580825 -84.27452385652867 2 50
QinetiQ 7/11/2012 1:34:11 PM LatLon: 35.93844175226674 -84.2745999833942 4 25
QinetiQ 7/11/2012 1:34:11 PM LatLon: 35.93844964747385 -84.27453762549652 4 25
QinetiQ 7/11/2012 1:34:13 PM LatLon: 35.93840573251469 -84.27456196823405 4 30
QinetiQ 7/11/2012 1:34:14 PM LatLon: 35.93840573251469 -84.27456196823405 4 35
QinetiQ 7/11/2012 1:34:14 PM LatLon: 35.93842535868399 -84.27449361804085 2 50
QinetiQ 7/11/2012 1:34:15 PM LatLon: 35.93840573251469 -84.27456196823405 4 40
QinetiQ 7/11/2012 1:34:16 PM LatLon: 35.93832858029554 -84.27454509064901 3 60
QinetiQ 7/11/2012 1:34:16 PM LatLon: 35.93837696949183 -84.27451935436109 4 40

Southwest 7/11/2012 1:34:07 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:34:08 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:34:09 PM LatLon 35.9383833674071 -84.2747030181606

Textron 7/11/2012 1:34:04 PM LatLon 35.93882 -84.274415
Textron 7/11/2012 1:34:08 PM LatLon 35.9384433333333 -84.27469
Textron 7/11/2012 1:34:15 PM LatLon 35.9383733333333 -84.2745866666667
Textron 7/11/2012 1:34:23 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:34:24 PM LatLon 35.938475 -84.2745633333333

Run 14

QinetiQ 7/11/2012 1:35:21 PM LatLon: 35.93840297006038 -84.27447949687894 3 60
QinetiQ 7/11/2012 1:35:22 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:23 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:24 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:25 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:25 PM LatLon: 35.93834285450666 -84.2746321292002 3 60
QinetiQ 7/11/2012 1:35:26 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:27 PM LatLon: 35.938386912797945 -84.27465862083957 3 50
QinetiQ 7/11/2012 1:35:27 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:28 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:28 PM LatLon: 35.938386912797945 -84.27465862083957 2 50
QinetiQ 7/11/2012 1:35:29 PM LatLon: 35.93840573251469 -84.27456196823405 4 25
QinetiQ 7/11/2012 1:35:30 PM LatLon: 35.9384293803663 -84.27466291792898 2 50
QinetiQ 7/11/2012 1:35:31 PM LatLon: 35.93846332183767 -84.27463934110897 4 25
QinetiQ 7/11/2012 1:35:32 PM LatLon: 35.93842006810803 -84.27464113315551 4 30

Southwest 7/11/2012 1:35:28 PM LatLon 35.9383699856743 -84.2747114395112
Southwest 7/11/2012 1:35:40 PM LatLon 35.9383699856743 -84.2747114395112

Textron 7/11/2012 1:35:20 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:35:23 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:35:29 PM LatLon 35.93814 -84.2746016666667

Run 15

QinetiQ 7/11/2012 1:36:22 PM LatLon: 35.938386912797945 -84.27465862083957 2 50
QinetiQ 7/11/2012 1:36:23 PM LatLon: 35.938386912797945 -84.27465862083957 3 50
QinetiQ 7/11/2012 1:36:25 PM LatLon: 35.93834285450666 -84.2746321292002 2 50
QinetiQ 7/11/2012 1:36:27 PM LatLon: 35.93840573251469 -84.27456196823405 2 50
QinetiQ 7/11/2012 1:36:29 PM LatLon: 35.93840297006038 -84.27447949687894 3 50
QinetiQ 7/11/2012 1:36:30 PM LatLon: 35.93842101580825 -84.27452385652867 2 50
QinetiQ 7/11/2012 1:36:32 PM LatLon: 35.93837696949183 -84.27451935436109 2 50
QinetiQ 7/11/2012 1:36:33 PM LatLon: 35.93832858029554 -84.27454509064901 2 50

Southwest 7/11/2012 1:36:25 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:36:26 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:36:27 PM LatLon 35.9383833674071 -84.2747030181606

Textron 7/11/2012 1:36:19 PM LatLon 35.93882 -84.274415
Textron 7/11/2012 1:36:31 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:36:31 PM LatLon 35.9384433333333 -84.27469
Textron 7/11/2012 1:36:32 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:36:36 PM LatLon 35.9383733333333 -84.2745866666667

Run 16

QinetiQ 7/11/2012 1:37:11 PM LatLon: 35.93832858029554 -84.27454509064901 3 60
QinetiQ 7/11/2012 1:37:15 PM LatLon: 35.93840297006038 -84.27447949687894 3 60
QinetiQ 7/11/2012 1:37:19 PM LatLon: 35.93834285450666 -84.2746321292002 2 50
QinetiQ 7/11/2012 1:37:19 PM LatLon: 35.93834285450666 -84.2746321292002 2 50
QinetiQ 7/11/2012 1:37:21 PM LatLon: 35.938386912797945 -84.27465862083957 2 50

Southwest 7/11/2012 1:37:21 PM LatLon 35.9383566039409 -84.2747198608589
Southwest 7/11/2012 1:37:22 PM LatLon 35.9383566039409 -84.2747198608589
Southwest 7/11/2012 1:37:24 PM LatLon 35.9383566039409 -84.2747198608589

Textron 7/11/2012 1:37:12 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:37:16 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:37:21 PM LatLon 35.938115 -84.2748

Run 17

QinetiQ 7/11/2012 1:38:03 PM LatLon: 35.93839354986539 -84.27468192163317 2 50
QinetiQ 7/11/2012 1:38:05 PM LatLon: 35.93842006810803 -84.27464113315551 2 50
QinetiQ 7/11/2012 1:38:05 PM LatLon: 35.9384293803663 -84.27466291792898 2 50
QinetiQ 7/11/2012 1:38:07 PM LatLon: 35.93834285450666 -84.2746321292002 3 60
QinetiQ 7/11/2012 1:38:13 PM LatLon: 35.93837696949183 -84.27451935436109 4 25
QinetiQ 7/11/2012 1:38:15 PM LatLon: 35.93832858029554 -84.27454509064901 2 50
QinetiQ 7/11/2012 1:38:15 PM LatLon: 35.93837696949183 -84.27451935436109 4 30
QinetiQ 7/11/2012 1:38:15 PM LatLon: 35.93832858029554 -84.27454509064901 2 50
QinetiQ 7/11/2012 1:38:16 PM LatLon: 35.93837696949183 -84.27451935436109 4 35

Southwest 7/11/2012 1:38:07 PM LatLon 35.9383699856743 -84.2747114395112
Southwest 7/11/2012 1:38:08 PM LatLon 35.9383699856743 -84.2747114395112
Southwest 7/11/2012 1:38:09 PM LatLon 35.9383699856743 -84.2747114395112

Textron 7/11/2012 1:38:08 PM LatLon 35.9384433333333 -84.27469
Textron 7/11/2012 1:38:12 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:38:12 PM LatLon 35.93882 -84.274415
Textron 7/11/2012 1:38:13 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:38:16 PM LatLon 35.9383733333333 -84.2745866666667

Run 18

QinetiQ 7/11/2012 1:39:09 PM LatLon: 35.93837696949183 -84.27451935436109 2 50
QinetiQ 7/11/2012 1:39:17 PM LatLon: 35.938388157535954 -84.27460901820079 2 50
QinetiQ 7/11/2012 1:39:18 PM LatLon: 35.938386912797945 -84.27465862083957 3 50

Southwest 7/11/2012 1:39:20 PM LatLon 35.9383699856743 -84.2747114395112

Southwest 7/11/2012 1:39:21 PM LatLon 35.9383699856743 -84.2747114395112
Southwest 7/11/2012 1:39:22 PM LatLon 35.9383699856743 -84.2747114395112

Textron 7/11/2012 1:39:08 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:39:11 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:39:13 PM LatLon 35.9381366666667 -84.2747166666667

Run 19

QinetiQ 7/11/2012 1:40:04 PM LatLon: 35.93839354986539 -84.27468192163317 2 50
QinetiQ 7/11/2012 1:40:06 PM LatLon: 35.9384293803663 -84.27466291792898 2 50
QinetiQ 7/11/2012 1:40:08 PM LatLon: 35.938388157535954 -84.27460901820079 2 50
QinetiQ 7/11/2012 1:40:11 PM LatLon: 35.93842101580825 -84.27452385652867 2 50
QinetiQ 7/11/2012 1:40:15 PM LatLon: 35.93837696949183 -84.27451935436109 2 50
QinetiQ 7/11/2012 1:40:16 PM LatLon: 35.93837696949183 -84.27451935436109 2 50

Southwest 7/11/2012 1:40:08 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:40:09 PM LatLon 35.9383833674071 -84.2747030181606
Southwest 7/11/2012 1:40:10 PM LatLon 35.9383833674071 -84.2747030181606

Textron 7/11/2012 1:40:13 PM LatLon 35.938475 -84.2745633333333
Textron 7/11/2012 1:40:13 PM LatLon 35.9384433333333 -84.27469
Textron 7/11/2012 1:40:14 PM LatLon 35.938445 -84.2745416666667
Textron 7/11/2012 1:40:14 PM LatLon 35.93882 -84.274415
Textron 7/11/2012 1:40:16 PM LatLon 35.9383733333333 -84.2745866666667

Run 20

QinetiQ 7/11/2012 1:41:05 PM LatLon: 35.93825316466917 -84.27454885445928 2 50
QinetiQ 7/11/2012 1:41:09 PM LatLon: 35.93825316466917 -84.27454885445928 4 50
QinetiQ 7/11/2012 1:41:09 PM LatLon: 35.93832858029554 -84.27454509064901 4 60
QinetiQ 7/11/2012 1:41:13 PM LatLon: 35.93840297006038 -84.27447949687894 4 60
QinetiQ 7/11/2012 1:41:15 PM LatLon: 35.93842101580825 -84.27452385652867 4 25
QinetiQ 7/11/2012 1:41:16 PM LatLon: 35.93840198398748 -84.27462567974123 4 25
QinetiQ 7/11/2012 1:41:17 PM LatLon: 35.93840198398748 -84.27462567974123 4 30
QinetiQ 7/11/2012 1:41:17 PM LatLon: 35.93843170904386 -84.27458169438681 4 50
QinetiQ 7/11/2012 1:41:18 PM LatLon: 35.9384293803663 -84.27466291792898 4 35
QinetiQ 7/11/2012 1:41:19 PM LatLon: 35.938386912797945 -84.27465862083957 4 50
QinetiQ 7/11/2012 1:41:19 PM LatLon: 35.93842006810803 -84.27464113315551 4 40
QinetiQ 7/11/2012 1:41:20 PM LatLon: 35.93842006810803 -84.27464113315551 4 40
QinetiQ 7/11/2012 1:41:21 PM LatLon: 35.93835375745535 -84.27467610852474 4 40
QinetiQ 7/11/2012 1:41:21 PM LatLon: 35.93835375745535 -84.27467610852474 4 50
QinetiQ 7/11/2012 1:41:22 PM LatLon: 35.93839354986539 -84.27468192163317 4 45

Southwest 7/11/2012 1:41:20 PM LatLon 35.9383699856743 -84.2747114395112

Textron 7/11/2012 1:41:06 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:41:14 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:41:15 PM LatLon 35.938115 -84.2748

Run 21

QinetiQ 7/11/2012 1:51:20 PM LatLon: 35.93815482549134 -84.27478103425042 4 50

Southwest 7/11/2012 1:51:25 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:51:26 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:51:27 PM LatLon 35.9379922902071 -84.2748320498564

Textron 7/11/2012 1:51:17 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:51:25 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:51:28 PM LatLon 35.93814 -84.2746016666667

Run 22

QinetiQ 7/11/2012 1:52:00 PM LatLon: 35.93815482549134 -84.27478103425042 4 25
QinetiQ 7/11/2012 1:52:16 PM LatLon: 35.938071075314646 -84.2748529541447 4 50
QinetiQ 7/11/2012 1:52:17 PM LatLon: 35.93805535944111 -84.2748334969326 4 50
QinetiQ 7/11/2012 1:52:19 PM LatLon: 35.93815482549134 -84.27478103425042 4 50

Southwest 7/11/2012 1:52:17 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:52:18 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:52:19 PM LatLon 35.9380068627585 -84.2748275623163

Run 23

QinetiQ 7/11/2012 1:53:13 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:53:15 PM LatLon: 35.938187980820544 -84.27476354666862 4 50
QinetiQ 7/11/2012 1:53:17 PM LatLon: 35.93810690584708 -84.27483395059356 4 50
QinetiQ 7/11/2012 1:53:19 PM LatLon: 35.93815482549134 -84.27478103425042 4 50

Southwest 7/11/2012 1:53:19 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:53:20 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:53:21 PM LatLon 35.9379922902071 -84.2748320498564

Textron 7/11/2012 1:53:17 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:53:19 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:53:20 PM LatLon 35.9381366666667 -84.2747166666667

Run 24

QinetiQ 7/11/2012 1:54:09 PM LatLon: 35.93812167014452 -84.27479852182547 4 50
QinetiQ 7/11/2012 1:54:11 PM LatLon: 35.93810690584708 -84.27483395059356 4 50
QinetiQ 7/11/2012 1:54:15 PM LatLon: 35.93802220409773 -84.27485098445771 4 50

Southwest 7/11/2012 1:54:11 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:54:12 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:54:14 PM LatLon 35.9380068627585 -84.2748275623163

Textron 7/11/2012 1:54:02 PM LatLon 35.93882 -84.274415

Run 25

QinetiQ 7/11/2012 1:55:05 PM LatLon: 35.93808851480986 -84.27481600937807 4 50

Southwest 7/11/2012 1:55:08 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:55:09 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:55:11 PM LatLon 35.9379922902071 -84.2748320498564

Textron 7/11/2012 1:55:07 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:55:12 PM LatLon 35.93814 -84.2746016666667
Textron 7/11/2012 1:55:13 PM LatLon 35.938115 -84.2748

Run 26

QinetiQ 7/11/2012 1:56:11 PM LatLon: 35.93802220409773 -84.27485098445771 4 50
QinetiQ 7/11/2012 1:56:15 PM LatLon: 35.93802220409773 -84.27485098445771 4 50

Southwest 7/11/2012 1:56:13 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:56:13 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:56:15 PM LatLon 35.9380068627585 -84.2748275623163

Textron 7/11/2012 1:56:03 PM LatLon 35.93882 -84.274415

Run 27

QinetiQ 7/11/2012 1:57:07 PM LatLon: 35.93812167014452 -84.27479852182547 4 60

Southwest 7/11/2012 1:57:10 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:57:11 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:57:12 PM LatLon 35.9379922902071 -84.2748320498564

Textron 7/11/2012 1:57:02 PM LatLon 35.9379816666667 -84.2749333333333

Textron 7/11/2012 1:57:05 PM LatLon 35.9381366666667 -84.2747166666667
Textron 7/11/2012 1:57:08 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:57:13 PM LatLon 35.93814 -84.2746016666667

Run 28

QinetiQ 7/11/2012 1:58:18 PM LatLon: 35.93810690584708 -84.27483395059356 4 50

Southwest 7/11/2012 1:58:19 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:58:20 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 1:58:21 PM LatLon 35.9380068627585 -84.2748275623163

Run 29

QinetiQ 7/11/2012 1:59:17 PM LatLon: 35.93810690584708 -84.27483395059356 4 60

Southwest 7/11/2012 1:59:19 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:59:20 PM LatLon 35.9379922902071 -84.2748320498564
Southwest 7/11/2012 1:59:21 PM LatLon 35.9379922902071 -84.2748320498564

Textron 7/11/2012 1:59:17 PM LatLon 35.938115 -84.2748
Textron 7/11/2012 1:59:23 PM LatLon 35.93814 -84.2746016666667

Run 30

QinetiQ 7/11/2012 2:00:13 PM LatLon: 35.93812167014452 -84.27479852182547 4 60

Southwest 7/11/2012 2:00:13 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 2:00:14 PM LatLon 35.9380068627585 -84.2748275623163
Southwest 7/11/2012 2:00:15 PM LatLon 35.9380068627585 -84.2748275623163

Textron 7/11/2012 2:00:08 PM LatLon 35.9381366666667 -84.2747166666667

VITA

Jonathan William Hickerson was born in Knoxville, Tennessee. He pursued an undergraduate degree in Electrical Engineering and Computer Engineering at Tennessee Technological University, graduating in 2010. In the Fall of 2010, he began pursuing a Master of Science degree in Computer Engineering at The University of Tennessee in Knoxville, Tennessee. He is very interested in the field of Computer Engineering and has aspirations to pursue a Ph. D. at some point in the near future and to teach Computer Engineering at the college level.