Air Force Institute of Technology

AFIT Scholar

3-11-2011

Perimeter Security and Intruder Detection Using Gravity Gradiometry: A Feasibility Study

Jared D. Tuinstra

**PERIMETER SECURITY AND INTRUDER DETECTION USING GRAVITY
GRADIOMETRY: A FEASIBILITY STUDY**

THESIS

Jared D. Tuinstra, Second Lieutenant, USAF

AFIT/GAE/ENY/11-M30

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY**

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GAE/ENY/11-M30

PERIMETER SECURITY AND INTRUDER DETECTION USING GRAVITY GRADIOMETRY: A FEASIBILITY STUDY

THESIS

Presented to the Faculty

Department of Aeronautics and Astronautics

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Aeronautical Engineering

Jared D. Tuinstra, BS

Second Lieutenant, USAF

March 2011

AFIT/GAE/ENY/11-M30

PERIMETER SECURITY AND INTRUDER DETECTION USING GRAVITY GRADIOMETRY: A FEASIBILITY STUDY

Jared D. Tuinstra, BS

Second Lieutenant, USAF

Approved:

_____//signed//_____	24 March 2011
Richard E. Huffman, Lt. Col., USAF (Chairman)	Date
_____//signed//_____	15 March 2011
Frederick G. Harmon, Lt. Col., USAF (Member)	Date
_____//signed//_____	17 March 2011
Christopher M. Shearer, Lt. Col., USAF (Member)	Date

AFIT/GAE/ENY/11-M30

Abstract

Changes in the mass distribution around some point on the Earth's surface induce corresponding changes to the magnitude and direction of the gravity vector at that location. The nine-tensor derivative of the gravity vector, or gravity gradient, is sensitive to very small changes in the gravity vector. With some assumptions, continuous measurement of the gravity gradient using a gravity gradiometer (GGI) is used to determine the location of a mass change in the local area near the instrument. This investigation sought to determine the effectiveness, operating characteristics, and limitations of a physical perimeter security system that uses an array of GGIs to detect and locate a human intruder. Results are obtained via computer simulations utilizing the closed form solution for calculating a gravity gradient given an object's size and mass, as well as industry-predicted future GGI performance characteristics.

Acknowledgments

This project would not have been possible without the encouragement and support of many, both past and present. Foremost, I thank my advisor, Lt Col Richard Huffman, for his help, advice, and discerning eye. He took on risk in his willingness to advise an inexperienced engineer, and I appreciate it. I thank my family for their faith in my ability to get the job done, and the enthusiasm with which they met every bit of progress. I would be remiss to not thank a very special girl from Indiana. She forfeited spending time together and got a stressed, inattentive boyfriend in return. It's not difficult to imagine why she formed the early opinion that gravity gradiometry should be better called gravity not-so-great-iometry. Finally, I thank the teachers at every level of education who offer up their wisdom and expertise with an eagerness that serves as an example for everyone.

Jared D. Tuinstra

Table of Contents

List of Figures

List of Tables

List of Equations

PERIMETER SECURITY AND INTRUDER DETECTION USING GRAVITY GRADIOMETRY: A FEASIBILITY STUDY

I. Introduction

History has shown that when man truly wants to be somewhere—whether it is the top of Mount Everest or the moon—he can get there, but only with enough time to prepare for and complete the journey. This applies to less ambitious destinations as well, like the inside of a well-stocked jewelry store after operating hours. Though buffers like well-built outer walls and bars over the windows exist to prevent unwanted intrusion, any thief could treat those devices as deterrents and simply bypass them if given enough time to do so. The key, then, to any system intending to provide security by limiting access to a given region is not only a buffer, but also an alarm system capable of detecting unwanted intrusion to limit the amount of buffer exposure time to a given threat. Continuing the example, inch thick window bars are not formidable to a dedicated burglar with a hacksaw, but they are when an alarm sounds at the start of the intrusion effort and the prospective thief has only a few minutes to work before police arrive. This principle has spawned many types of intruder detection devices, all with the goal of detecting an individual entering an area of interest and initiating some alarm before the intruder can achieve his malicious objectives.

In many cases, perimeter security is not as simple as detecting a jewel thief forcefully entering a locked building. The number of industries with an active interest in a better perimeter security system is large, and the intended application can be of tremendous magnitude and complexity. Optimizing intruder detection for the conditions and circumstances of various unique scenarios has driven the development of the perimeter intrusion detection system (PIDS), with a suite of contrasting technologies at its core. Almost every physical characteristic that defines an individual being at a location is currently used, including seismic patterns, a change in the magnetic field, electric field interference, infrared sensors, radar, motion detectors, and acoustic devices. All devices in use today typically excel in some aspects of intruder detection while falling short in others, and none is marketed as the proverbial silver bullet of perimeter security. In every case, manufacturers of a specific device recommend a combination of methods working together to achieve maximum security over the widest range of possible intrusion scenarios. While effective, joining multiple methods is expensive to design, build, and operate, and it is usually not feasible to integrate new technology into an already existing system.

So far, however, the application of gravity gradiometry as the core technology in a PIDS has remained uninvestigated. Hardly an emerging technology, gravity gradiometry seeks to measure the variations in the earth's gravitational field caused by variations in mass. Based on Newton's Law of Universal Gravitation, gravity gradiometry in a PIDS application would utilize a system of accelerometers to characterize how the 'acceleration due to gravity' term changes at a single location on the surface of the earth due to a movement in mass near the device. Because the proportion

of an intruder's mass to the earth's mass drives the change in acceleration caused by an intruding individual or vehicle, the gravity gradient for an intruder at even a modest detection range is expected to be extremely small. Due to recent successful efforts to improve the sensitivity of gravity gradiometers, the use of gravity gradiometry in a PIDS has only recently become a realistic undertaking.

Although gravity gradiometry is most likely not a standalone method for maximum intruder detection capability, it does offer some considerable advantages over its technological peers. Most notably, a gravity gradiometry instrument (GGI) is a discrete device free of infrastructure like perimeter-spanning wires or buried sensors required by many of the other technologies. Aside from freedom of the maintenance demands of a component-laden and widespread system, insertion of a GGI-based PIDS into an existing PIDS would produce immediate results. A GGI is a passive device, meaning it does not emit a signal to collect data. Passivity is not a characteristic exclusive to a GGI among the PIDS variants, but it does offer the advantage of being undetectable by a potential intruder, making tampering with or bypassing the device from a distance impossible because there is no way to know if it is monitoring a given perimeter or not. Since there is no way to block or shield the effect of mass on the gravity gradient, detection of intruding mass is possible in every direction without the possibility for camouflage.

Problem Statement

Alluded to earlier, the creation of an all-in-one PIDS is realistically impossible since a myriad of application scenarios exist, each with unique required capabilities. With

increasing terrorist activity on and around domestic and international holdings, as well as the expansion of United States military obligations to many unfriendly corners of the world, there has never been greater need for effective perimeter security. The PIDS examined in this paper are mainly useful for the detection of an intruder across some boundary through which nothing is expected to pass—a circumstance that certainly omits many potential terrorism situations, like a suicide bombing in a crowded outdoor market or a vehicle bomb parked on the street next to a high-rise building. The circumstances included, however, are considerable. Any area of interest enclosed in a perimeter with some standoff distance between the interior assets and the perimeter is a good application for a GGI-based PIDS, which includes many infrastructure level potential terrorist targets like nuclear power plants, oil refineries, water filtration plants, and research centers.

Indeed, "Protect Critical Infrastructure" is among the five overarching strategic goals identified in the U.S. Department of Homeland Security (DHS) Security Strategic Plan for 2008-2013 [1]. In a summary of how various governmental agencies and documents define and identify what is "critical infrastructure," one Congressional Research Service (CRS) report explains that while the meaning of "critical infrastructure" continues to evolve and varies from administration to administration, the essence is the same and has always included things like energy production, transmission, and distribution services; facilities using, storing or disposing of nuclear material; postal and shipping services; information and telecommunications centers; and the defense industrial base [2]. Many of these broad sectors are made up of individual facilities where a better PIDS would aid in protecting what is considered critical.

Not only useful to the Homeland Security effort, a better PIDS would have direct military application by protecting bases around the world. A few months before the terrorist destruction of the World Trade Center in September 2001, forward looking congressmen from the House of Representatives held a special Armed Services Committee hearing on "Security Against Terrorism On U.S. Military Bases." In the hearing, one base commander from every military service testified about the challenges faced in securing a large area of land. Each of the four general officers testifying identified perimeter security as their primary concern and made it clear they were aware of weaknesses in each of their respective perimeters needing improvement [3]. United States Marine Corps Major General David F. Bice, Commanding General of Marine Corps Base Camp Pendleton in California, in particular offered a telling statistic to the committee, noting about 150 illegal immigrants a year are picked up on his base and said, "It shows that anyone can get on our base if they are determined enough" [3]. A better PIDS could change the paradigm that intruder access to secure areas is unavoidable.

Some of the more devastating terrorist attacks on U.S. military assets occurred because malicious individuals were able to get close enough to their targets to inflict harm. A truck containing the equivalent of 10 tons of TNT explosive was detonated less than 100 feet from the Khobar Towers military housing complex in Saudi Arabia, killing 19 U.S. servicemen in 1996. Seventeen sailors serving on the *USS Cole* lost their lives when a small vessel set off a bomb after entering the restricted area surrounding the Navy ship while it refueled in a Yemeni port in 2000. Neither attack resulted from a PIDS failure, but they still emphasize why an effective security system is necessary; tragedy will result when anyone can access areas which are supposed to be secure.

Figure 1: Khobar Towers Following 1996 Bombing, taken from United States Air Force [4]

Development of a GGI-based PIDS promises to result in a better technique for denying one of the crucial steps in a chain of steps leading up to a terrorist attack on valuable national assets.

Research Objectives

Since there has been no previous investigation into using gravity gradiometry for perimeter security, research objectives will focus on broadly identifying whether the technology has potential as a practical method for intruder detection by designing a theoretical system. As such, the specific objectives are:

1. Generate a model demonstrating GGI integration into an area where perimeter security is needed, and identify calibration requirements.

2. Develop an algorithm utilizing GGI readings to recognize when an intrusion event has occurred.

3. Create a system for reporting intrusion anomalies and classifying the type of threat identified.

Various types of GGIs currently exist, and ongoing research continues to examine improving the technology to increase instrument sensitivity and expand mobility. GGI use within a PIDS offers the distinct advantage of low mobility requirements since the device is stationary during the data collection process. Likewise, a perimeter security application implies the size of the GGI will not be as great a concern as it might be for an aircraft navigation application. Strict volume and weight allowances on board an aircraft, for example, means the only GGIs available for use are those small and light enough—not necessarily the least noisy or easiest to use. Support considerations like power requirements and cooling systems are not of interest here since ground installation constraints are generally not strict. With these considerations in mind, the level of GGI technology used in this investigation is limited to what researchers say will be feasible in the future.

Before creating an algorithm utilizing GGI inputs to detect an intruder, the many ways an intruder can approach a secure area—and the numerous types of secure areas—must be culled to a manageable level. The representative intrusion scenarios serve to identify and demonstrate calibration procedures while maintaining broad application to specific perimeter security needs. Additionally, the devised scenarios will integrate

realistic difficulties, like terrain and weather, associated with the operation of a GGI for

perimeter security in practice rather than solely on principle.

False alarms plague all physical security systems, and a GGI-based system should

be no different. To make an effective system that limits alarms to those actually caused

by a threat, the algorithm for detecting anomalies must include some reasoning process to

determine the level of confidence that any given anomaly is a true threat. Since any threat

would most likely require immediate action, a useful perimeter security system will

provide quality information in a timely, organized fashion to allow further action.

<center>**II. Background**</center>

Existing PIDS

There are hundreds of PIDS currently on the market designed to fulfill almost every conceivable intruder scenario. To limit this discussion to a manageable level, gravity gradiometry is compared only to those technologies competing to directly fill the same security niche. With the problem statement in mind, that security scenario will loosely consist of an intruder on foot attempting to penetrate the well-defined perimeter of something defined as critical infrastructure. 'Well defined perimeter' in this case means the perimeter is clearly marked and without nearby obstructions, as might be the case around a military base. According to a researcher for Sandia National Laboratories, defining six physical and environmental factors completely characterizes any given perimeter security situation and includes aspects like topography, background noise, and the type of soil [5]. Although those aspects will eventually be important to define, for now the scenario is clear enough by saying it will be outside, cover a large area of cleared land, have a perimeter hundreds of meters long, and require a permanent system.

Several types of fence-integrated sensors exist and generally work by transferring the energy of an intruder interacting in some way with a fence to a more detectable form of energy, like a vibration sensor or a strain sensitive cable [6]. Though useful in determining whether a boundary has been compromised, it is a fundamentally different type of PIDS than the volumetric sensor category a GGI falls into. Volumetric sensors seek to track intruder movement through a secure area by using an invisible detection source, like a change in gravity gradient. Unlike a fence-mounted device, volumetric

<center>9</center>

sensors determine the presence of intruders without direct contact. So even though they fall well within the realm of perimeter intruder detection, fence sensors will not be discussed to avoid comparing unlike things. The primary categories of PIDS utilizing volumetric sensors are radar, seismic, infrared, and optical. An overview of each type of sensor follows with a focus on the weaknesses of each type of device.

Radar

Radar operates in the $100MHz$ to $300GHz$ electromagnetic energy band and works by transmitting a signal and waiting for the signal to reflect back to the receiver by striking an object in the field of view. A processor uses the pattern of reflected energy to provide information about the size and location of the object. Using radar in a PIDS application requires the signal reflected back to the radar to be greater than the noise generated by normal operation of the device. The ratio of those values is called the signal to noise ratio (SNR) and characterizes radar performance, with higher values indicating better system performance. SNR is calculated using Equation 1, where P_t is the transmitter power, G_t is the transmitter gain, G_r is the receiver gain, λ is the wavelength of the radio wave, σ is radar cross section (RCS) of the target, r is the distance from the transmitter to the target, k is Boltzmann's constant ($1.381 \cdot 10^{-23} J \cdot K^{-1}$), T is the receiver noise equivalent temperature, and B_n is the processing noise bandwidth [7].

$$SNR = \frac{P_t G_t G_r \lambda^2 \sigma}{(4\pi)^3 r^4 k T B_n}$$

Equation 1: Classic Radar Equation [7]

Of all the terms in the classic radar equation, the only terms controlled by the target are RCS value and separation distance, with the rest of the terms constants or radar device characteristics. An object's RCS value is not a measure of the true presented physical cross sectional area, but rather a measure of the object's apparent size to the radar receiver. Butler [7] identifies some typical RCS values in Table 1.

Table 1: Radar Cross Section Values for Typical Targets, taken from Butler [7]

Target	RCS (m^2)
Crawler	0.03 ~ 0.1
Pedestrian	0.3 ~ 1.5
Light Ground Vehicle	5 ~ 50
Heavy Ground Vehicle	20 ~ 1000
Small Airplane	5 ~ 20

Higher frequency devices detect objects with greater resolution, but with higher attenuation and thus lower range, while the opposite resolution and range characteristics are true for lower frequency radars. Exceedingly useful when used correctly, radar is only a reliable detection method when a clear line of sight to the secure area is available. Potential downfalls include inability to differentiate between an intruder and ground objects, heavy precipitation conditions, and blockage by buildings or other objects. Unless the area of interest is flat and objects like rocks and trees are well characterized and unchanging, radar would have a difficult time detecting an intruder moving slowly and close to the ground, particularly if steps are taken to reduce the intruder's radar signature through careful choice of clothing. One major disadvantage of some radar systems is the need for bulky antennas and numerous, large components [6, 8]. Due to its

potential detection range on the order of kilometers, using radar as an intruder early

warning system has been investigated with a focus on intruder detection in foliage. One

study successfully demonstrated detection of a human hidden $2m$ inside a brushy tree line

at a range of $500m$, a powerful capability considering many potential intruders spend time

surveilling their target before initiating intruding action [9].
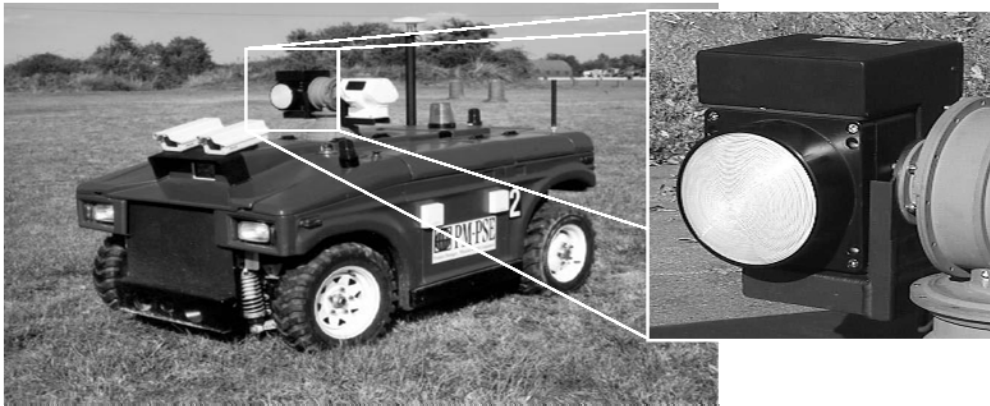
Figure 2: Millimeter-Wave Pulsed-Doppler Radar Mounted on MDARS-E Intrusion Detection System, taken from Cory, Everett, and Pastore [8]

Figure 2 shows an example of radar used in an intruder detection capacity. The

MDARS-E remotely controlled vehicle has multiple detection devices mounted to it,

including the radar shown. With a wavelength in the millimeter range, it does not require

a large antenna. However, the small wavelength results in high attenuation and a modest

$100m$ range [8].

Seismic

Seismic sensors detect intruders by measuring the seismic waves generated by the

impact of footsteps or vehicles on the ground. Using seismic waves as part of a PIDS is a

complex process made difficult because the ground supports four different types of seismic waves that each have different propagation characteristics. Shear and compression waves are three-dimensional and propagate in a spherical surface as they move away from the point of impact, so amplitude attenuation diminishes with an r^{-1} relationship, where r is the distance from the point of impact. Love waves utilize ground structure and density characteristics by travelling between channels formed in layered soil. Compression, shear, and Love waves are not useful for intruder detection because they decay too quickly and can have unpredictable motion based on the soil composition [10].

Compression Wave   **Rayleigh Wave**

Shear Wave   **Love Wave**
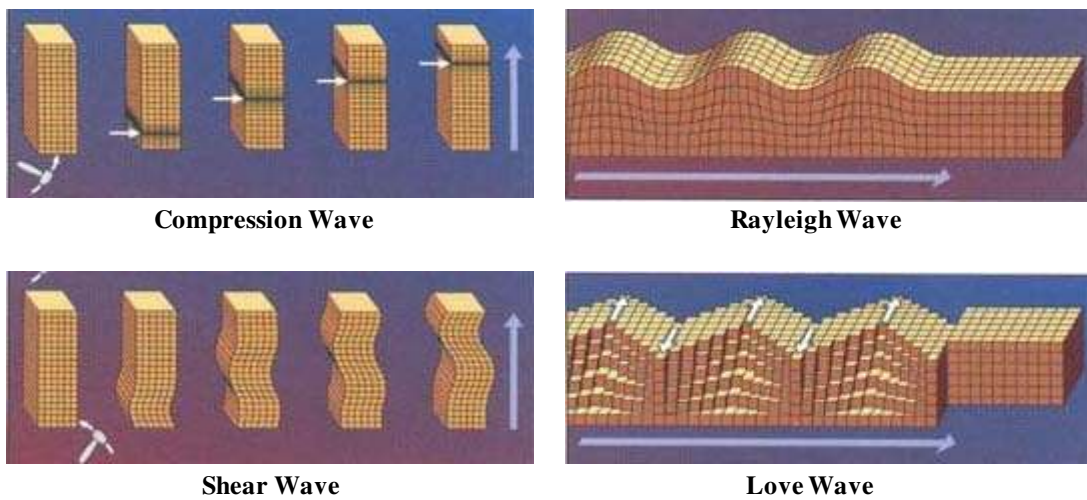
Figure 3: Illustration of Seismic Wave Types, taken from the U.S. Geological Survey [12]

The final types of seismic waves, Rayleigh waves, are confined to travel along the surface and thus exhibit amplitude decay according to an $r^{-1/2}$ relationship. This lessoned decay over distance, combined with the fact that about 70% of ground impact energy is converted to Rayleigh waves (with the remaining 30% used to create shear, compression

13

and Love waves) make them the best kind of seismic wave to detect intruder encroachment [10]. Figure 3 shows an illustration of the four types of seismic waves, and helps demonstrate why Rayleigh waves are the most useful for intruder detection. Two types of sensors can measure seismic waves—a moving coil electro-dynamic device called a geophone, or an accelerometer. The expected frequency range determines which device to use, and since an energy source like footsteps is less than $100Hz$, geophones are the typical sensor used within a PIDS [11].

Most effective when used in a sterile environment where influences contributing to ground vibration are controllable, the use of seismic activity to detect an outdoor intruder is made difficult by anything anchored to the ground and capable of transmitting wind vibrations, like trees and other vegetation, telephone poles, and fences. Geophones are typically buried to avoid direct wind impingement, but greater burial depth causes greater signal attenuation and less detection capability. The middle ground between the competing effects of wind noise and attenuation usually occurs just below the surface [11]. Even on calm days, any real world location has constant seismic noise. Since footsteps result in a relatively low energy wave, the amplitude of the wave with respect to the background noise is worthy of concern. Pakhomov and Goldburt [13] point out an improved geophone will not improve the SNR when the source of undesired noise is an environmental characteristic (as opposed to a sensor design characteristic). Those authors describe the best geophone systems as being able to operate in an environment where the SNR is 1:1 or greater, and give the resulting theoretical detection ranges for a variety of background noise conditions in Table 2 when that signal-to-noise condition is the case.

As with any of the PIDS discussed, more robust signal processing can sometimes improve the limitations resulting from a noisy environment.

Table 2: Predicted Range Capabilities for a Seismic PIDS, taken from Pakhomov and Goldburt [13]

Background Noise Level	*Range for SNR=1:1*
Very High	2 - 3*m*
High	5 - 6*m*
Medium	10 - 15*m*
Low	25 - 35*m*
Very Low	50- 70*m*
Extremely Low	70 - 90*m*

The ability for the ground to transmit a vibratory signal is heavily dependent on the type of soil, so seismic sensors are not an option in every environment, especially those areas with loose and inconsistent soil. Leet [14] notes the velocity of seismic waves can vary from 650*ft/s* in sand to 18,700*ft/s* in granite, but the most likely scenario is a heterogeneous soil composition including rocks, sand, and organic material in an unpredictable combination. Current literature rigorously defines seismic wave propagation, but most presented equations rely on a variety of assumptions precluding their application to a real world scenario. Complicating factors include acoustic noise contamination, non-geometric signal attenuation, and reflection from unique soil structure [10]. When intruder localization is as important as detection, PIDS designers use multi-component and arrayed sensors to triangulate the position of a seismic source.

That effort is made difficult by unpredictable wave reflections and velocity and can result in intruder directional position estimation error as high as 60° [15].

Figure 4: SM6-U/B Geophone, taken from Lindgren, Habberstad, and Holmberg [16]

Figure 4 shows a photograph of a geophone used in an intruder detection application. Device installation is as simple as inserting the metal stake into the ground at the desired location. The yellow box contains the bulk of the sensor and sits above the ground, while a connecting wire links the geophone to a central processor. Insight into how seismic waves travel in a particular environment typically requires a full site characterization in advance of installation, so creating a large geophone array can be involved and costly. Figure 5 shows a notional three-sensor geophone array where each geophone passes information to a processor. Geophone spacing depends on the system and site, but greater spacing typically yields more accurate intruder bearing information [6, 17].
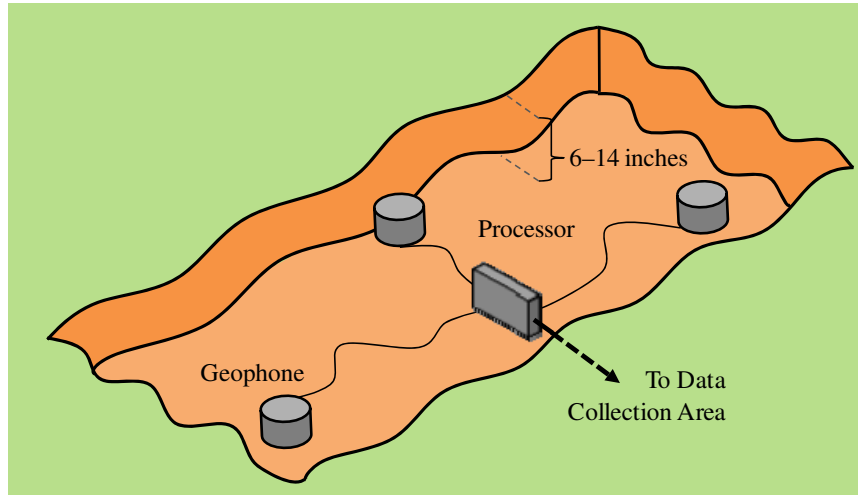
Figure 5: Notional Geophone Layout, based on Perimeter Security Sensor Technologies Handbook [6]

Infrared

The infrared region of the electromagnetic spectrum encompasses the $0.7\text{-}300\mu m$ wavelength range, bordering the visible band and the microwave band. The ability to use infrared energy in a PIDS application stems from the unique property of any object with a temperature above absolute zero to radiate energy in the infrared region according to Equation 2. In the equation, M is the radiant flux per unit area per unit time, σ is the Stefan-Boltzmann constant ($5.6704 \cdot 10^{-10} J \cdot s^{-1} m^{-2} K^{-4}$), ε is the object's emissivity, and T is the object's temperature. Emissivity is the measure of an object's relative ability to radiate absorbed energy, with a value of unity representative of a perfect emitter and called a blackbody. Doctor [18] notes clothed people have emissivities of about 0.75, meaning they are good radiators of energy.

$$M = \sigma\varepsilon T^4$$

Equation 2: The Stefan-Boltzmann Law

An infrared intruder detector works by assuming an intruder has a different temperature, and thus different radiant flux, than the surroundings. Equation 3 gives the radiant flux difference where the t subscript indicates the target and b subscript indicates the background. The equation shows a small difference in temperature between T_t and T_b will cause a disproportionately large change in radiant flux due to the fourth power dependence on temperature.

$$\Delta M = \sigma(\varepsilon_t T_t^4 - \varepsilon_b T_b^4)$$

Equation 3: Difference in Radiant Flux between Two Objects [18]

Adjusting for the change in apparent size of the target as its distance to the sensor changes requires modification of the difference in radiant flux equation. Doctor [18] gives the relationship for radiant flux density, F_d, in Equation 4 for an infrared detector capable of monitoring 2π steradians. Radiant flux density is dependent on the separation distance between the target and sensor, r, and presented surface area of the target, A. The inverse square relationship means a change in separation distance causes a non-linear change to the amount of infrared energy received by a sensor.

$$F_d = \frac{(\Delta M)A}{\pi r^2}$$

Equation 4: Radiant Flux Density of a Target Some Distance from the Sensor [18]

An infrared PIDS uses an array of optically focused sensors to create a grid pattern, with each sensor focused on a unique, typically small area dependent on the perimeter characteristics. For example, a bird's eye view of a system designed to detect four feet of movement from a 6*ft* tall and 3*ft* wide intruder at a range of 30*ft* and 75° field

of coverage would look like Figure 6. That system relies on six sets of two linked

sensors. When a heat source with enough radiant flux density passes from one of the

linked sensors to the other, the system logic determines an intruder has entered the field

of coverage.
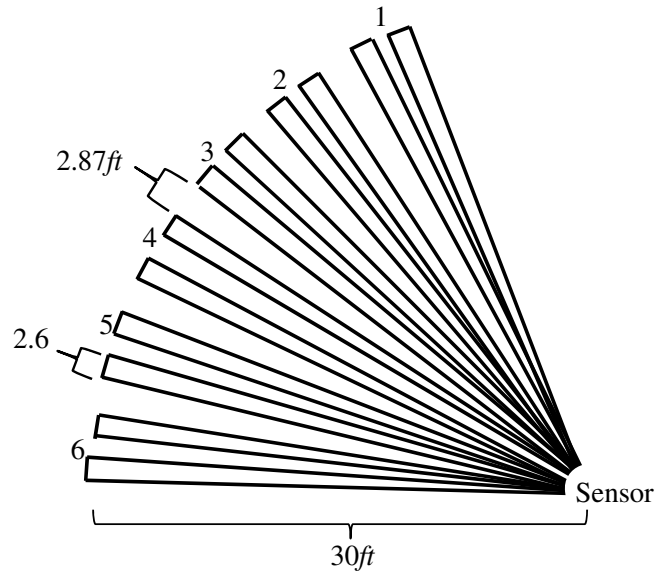
Figure 6: Bird's Eye View of an Infrared Beam Pattern, based on Doctor [18]

Bad weather can significantly attenuate the signal and cause false alarms, and the

amount and complexity of structures needed to support an array of sensors can be

significant. Additionally, the observed area must be flat and clear of obstruction to avoid

places where a slow moving intruder can hide. Passive infrared detection suffers from a

relatively short detection range of about 100ft, and can be fooled during warm days when

many objects emit infrared radiation at the same wavelengths as a human [6]. Heat

signature masking can reduce T_t enough for an intruder to blend in with the background,

and hiding behind an object opaque in the infrared energy range can prevent detection of any heat signal at all.

Figure 7 shows an example of an infrared detector used as part of the MDARS-E intruder detection system discussed previously. The infrared detector is compact, allowing high mobility on a movable arm. Combining multiple detection technologies on the vehicle allows operation in normal to low-level light conditions [8].
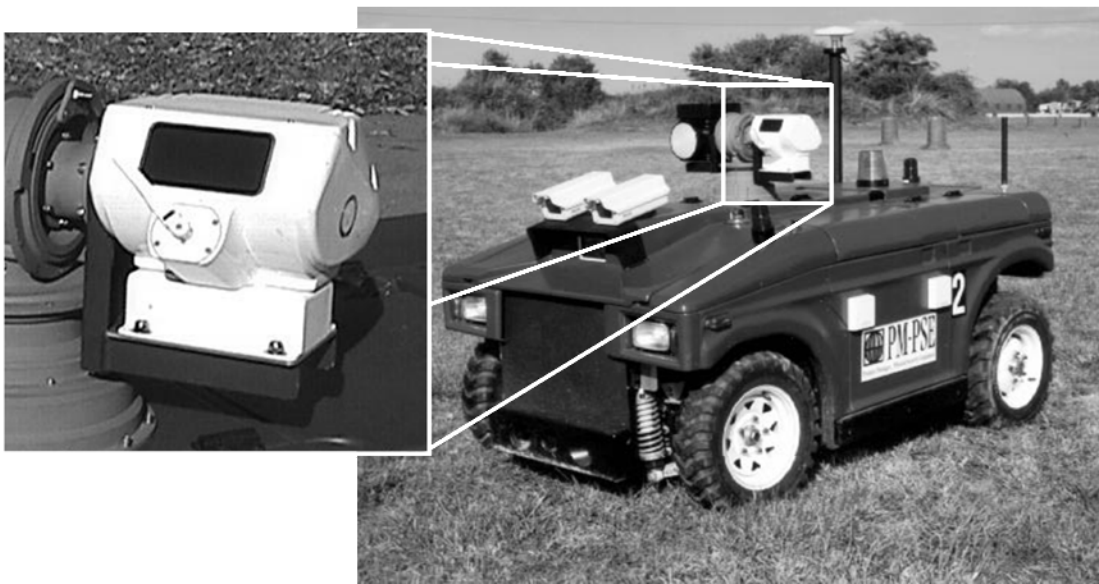
Figure 7: Infrared Detector Mounted on MDARS-E Intrusion Detection System, taken from Cory, Everett, and Pastore [8]

All previous discussion of an infrared PIDS referred to a passive infrared (PIR) device, so named because the sensor only collects energy information transmitted toward it by another source. Active infrared devices for perimeter security do exist, but work by generating and transmitting an infrared beam. An object passing between the beam generator and the receiver breaks the beam and causes an intruder alarm. Since this

device only protects a specific boundary and provides no information about intruder, it is not a true volumetric system and will not be included in this discussion.

Optical

The final exterior volumetric security method of interest is optical detection. In many cases, the human eye is still the best method of determining when an intruder enters a protected area. Labor intensive and susceptible to decreased ability caused by fatigue, a human detection presence is typically reserved for monitoring major traffic areas since the capability of automated systems is reasonable in many perimeter security situations. Motion detectors as part of a closed-circuit television system operate on a similar principle to the passive infrared detector—though over the visible light band instead of the infrared one—by comparing a current image frame to one known to be free of anomalies. When some unexpected change in the image is detected, a possible intrusion event has occurred. Inconsistent lighting and cluttering of the frame can cause false alarms or a lack of alarms depending on the situation, and a system with full coverage can be expensive to construct and monitor over time [6, 17].

Because similar principles govern energy in the visible and infrared bands, Equation 4 also describes how well an optical image sensor can see an image. As was true for infrared detectors, the ability to see an image in the visible band decreases by the inverse of the range squared. Modern video cameras typically have variable focal lengths and iris control mechanisms to maintain usefulness across a wide area in changing light conditions [17]. Figure 8 shows a photograph of a security camera mounted outdoors. It

offers an example of an optical camera capable of capturing a wide viewing angle due to a gimbal and mounting location above ground level.

Figure 8: Mounted Security Camera Enclosure, taken from Baran, O'Brien, and Fung [19]

Gravity Gradiometry

Though no GGI-based PIDS exists, familiarity with the concept allows conclusions about a theoretical system. The biggest difference between gravity gradiometry and other core technologies for intruder detection is the absence of a line of sight requirement. This is advantageous because it eases the demand for the secure area to have a specific topography and composition, and eliminates the burden of modifying large areas of landform before installing a system. Furthermore, no methods exist to cloak the change in the gravity gradient caused by the change in system mass of an approaching intruder. A GGI observes every direction at all times, reducing the chance of intrusion during radar or sensor 'sweeps.' Although the technologies discussed previously

are designed for use with complementary methods, gravity gradiometry offers distinct advantages over current security approaches. A GGI-based PIDS could improve overall perimeter security capability while reducing the number of sensors required to monitor a given area.

Gravity Gradients

Measuring and manipulating the gravity gradient, or the nine component spatial double derivative of the gravity vector, is what allows a GGI-based PIDS to be useful for an intrusion detection application. Newton's Law of Universal Gravitation states every mass is attracted to every other mass with a force given by Equation 5, where F is the magnitude of force between the two masses, m_1 and m_2 are the attracted and attracting masses, l is the separating distance between the masses, and G is the gravitational constant ($6.67428 \cdot 10^{-11} \ m^3 kg^{-1} s^{-2}$).

$$F = G \frac{m_1 m_2}{l^2}$$

Equation 5: Newton's Law of Universal Gravitation

Consider a system comprised of two point masses in a rectangular coordinate system where m_1 rests at position (x_1, y_1, z_1) and m_2 rests some distance l away at position (x_2, y_2, z_2), as described pictorially in Figure 9.
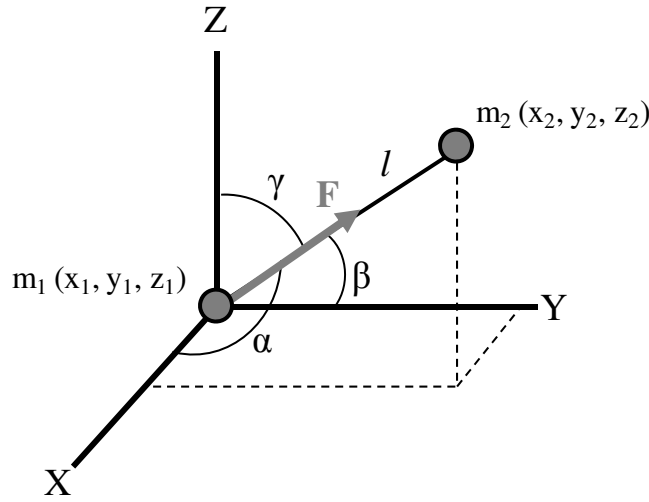
**Figure 9: The Force Vector Described in Rectangular Coordinates, based on
Hofmann-Wellenhof and Moritz [20]**

Hofmann-Wellenhof and Moritz [20] give the three-component force vector in Equation
6, where X, Y, and Z are the force components and α, β, and γ are the respective angles
from the coordinate axis to the force vector.

$$X = -F \cos \alpha = -\frac{Gm_1}{l^2} \frac{x_2 - x_1}{l} = -Gm_1 \frac{x_2 - x_1}{l^3}$$

$$Y = -F \cos \beta = -\frac{Gm_1}{l^2} \frac{y_2 - y_1}{l} = -Gm_1 \frac{y_2 - y_1}{l^3}$$

$$Z = -F \cos \gamma = -\frac{Gm_1}{l^2} \frac{z_2 - z_1}{l} = -Gm_1 \frac{z_2 - z_1}{l^3}$$

$$l = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

**Equation 6: Gravitational Force Vector Components between Two Point
Masses [20]**

A system of spatially variable components is often easiest described by a single potential
function, φ, mathematically defined by Equation 7.

$$X = \frac{\partial \varphi}{\partial x}, Y = \frac{\partial \varphi}{\partial y}, Z = \frac{\partial \varphi}{\partial z}$$

Equation 7: Definition of the Potential Function

In Equation 8, Hofmann-Wellenhof and Moritz [20] give the potential function describing Newton's Law of Universal Gravitation for a simple, two-mass system in the previous coordinate system. For simplification, the attracting mass is scaled to unity leaving only a single mass term, m [20].

$$\varphi = \frac{Gm}{l}$$

Equation 8: Gravitational Potential Function of a Simple System [20]

The previous equation holds for a point-mass in a closed system. Equation 9 shows Equation 8 expanded to calculate the gravitational potential of multiple point masses, where n is the total number of point masses considered.

$$\varphi = G \sum_{i=1}^{n} \frac{m_i}{l_i}$$

Equation 9: Potential Function for a Complex System [20]

The equation is simply the sum of each individual mass contribution. If the system is continuous, as it would be for any earthly application, the summation from Equation 9 can be mathematically restated as a volumetric integration covering the bounds of the system. The result of such a maneuver is called Newton's Integral and is given by Equation 10 where the mass density, ρ, and volume of integration, dV, replace the mass term m.

$$\varphi = G \iiint_V \frac{\rho}{l} dV = G \iiint_V \frac{\rho(x,y,z)}{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}} dxdydz$$

Equation 10: Newton's Integral [20]

Derivation of the potential function in accordance with Equation 7 yields the gravitational

force components for a system composed of an infinite distribution of mass. Equation 11

shows the gravitational force components.

$$X = -G \iiint_V \frac{x_2 - x_1}{l^3} \rho dV$$

$$Y = -G \iiint_V \frac{y_2 - y_1}{l^3} \rho dV$$

$$Z = -G \iiint_V \frac{z_2 - z_1}{l^3} \rho dV$$

Equation 11: Gravitational Force Components for a Complex System [20]

In accordance with the definition of a potential function, each gravitation force

component g_i of the gravitational force vector g^n is given by the first derivative of the

Equation 10, as shown in Equation 12.

$$g^n \equiv \nabla \varphi_g = \begin{pmatrix} g_x \\ g_y \\ g_z \end{pmatrix}$$

Equation 12: Relation of the Potential Equation to the Gravitational Vector [21]

The gravity vector is relatively easy to conceptualize, and is affected by a large mass like the mountain seen in Figure 10.
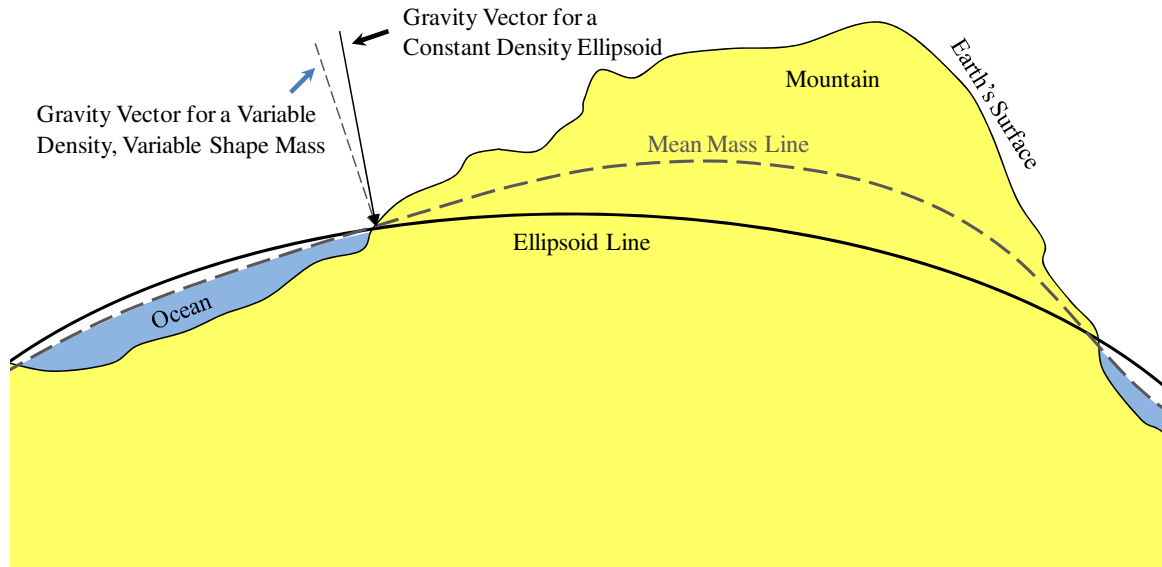
Figure 10: Effect of Variable Topography on the Gravitational Vector at the Earth's Surface, based on Burkard [22]

The gravitational vector on the surface of a constant density sphere can be conceptualized by treating the sphere as a point mass and applying Newton's Law of Universal Gravitation with a separation distance of the radius of the sphere. The vector will point toward the center of mass independent of the surface position, as long as a constant radius is maintained. The gravitational vector for a complex object with variable shape and density will not always point toward the center of mass, instead depending on the local mass distribution. The large mass of a mountain on the surface of the Earth in Figure 10 influences the gravity vector by pulling it slightly toward the area of greater mass. In this way, g^n is unique at every location on the Earth's surface since no mass distribution is exactly the same at any two positions.

However, realizing any change in the gravitational vector from one spot to another is challenging without an exceptionally precise instrument, a difficulty exacerbated in areas of relatively even density distributions and gradual topography changes. Because it is more responsive to small changes in the gravitational vector, the gravity gradient is usually measured instead of the actual gravitational vector. The gravity gradient is formed by differentiating the potential function again to form the nine-component tensor term described by Richeson in Equation 13 [21].

$$\Gamma^n \equiv \nabla\nabla^T \varphi_g = \begin{pmatrix} \Gamma_{xx} & \Gamma_{xy} & \Gamma_{xz} \\ \Gamma_{yx} & \Gamma_{yy} & \Gamma_{yz} \\ \Gamma_{zx} & \Gamma_{zy} & \Gamma_{zz} \end{pmatrix}$$

Equation 13: The Complete Gravitational Gradient [21]

The gravity vector is easy to conceptualize, but the gravitational gradient term is less so. For ease of understanding when applied to a system with an already well-known frame of reference, the rectangular x, y, z notation is switched to the rectangular north (N), east (E), down (D) frame of reference where N corresponds to x, E to y, and $-D$ to z. The gravitational gradient with revised notation is in Equation 14.

$$\Gamma^n = \begin{pmatrix} \Gamma_{NN} & \Gamma_{NE} & \Gamma_{ND} \\ \Gamma_{EN} & \Gamma_{EE} & \Gamma_{ED} \\ \Gamma_{DN} & \Gamma_{DE} & \Gamma_{DD} \end{pmatrix}$$

Equation 14: Complete Gravitation Gradient with NED Notation

To clarify what the components in Equation 14 mean physically, consider the Γ_{NE} component: the inner subscript represents the measured component while the outer subscript indicates across which direction the measurement is made, so Γ_{NE} is a

measurement of the change in the *N* gravity component in the *E* direction. Hofmann-Wellenhof and Moritz [20] point out that if the location of a gravitational gradient measurement is considered to be free space when compared to the density of surrounding attractive bodies, the mathematical properties of the potential function form the zero-value version of Poisson's equation in Equation 15, a particular form known as Laplace's equation. Since the intended GGI application is above ground and the density of the atmosphere at sea level is only about 0.05 percent of the mean density of the earth's crust, this is a reasonable assumption [21].

$$\nabla^2 \varphi = \frac{\partial^2 \varphi}{\partial N^2} + \frac{\partial^2 \varphi}{\partial E^2} + \frac{\partial^2 \varphi}{\partial D^2} = \Gamma_{NN} + \Gamma_{EE} + \Gamma_{DD} = 0$$

Equation 15: Poisson's Equation for Gravitational Gradiometry in Free Space [20]

Mickus and Hinojosa [23] note the gravitational gradient tensor forms a symmetric matrix, which causes the redundancy of terms described in Equation 16.

$$\frac{\partial g_D}{\partial E} = \frac{\partial g_E}{\partial D} = \Gamma_{DE} = \Gamma_{ED}$$

$$\frac{\partial g_N}{\partial D} = \frac{\partial g_D}{\partial N} = \Gamma_{ND} = \Gamma_{DN}$$

$$\frac{\partial g_E}{\partial N} = \frac{\partial g_N}{\partial E} = \Gamma_{EN} = \Gamma_{NE}$$

Equation 16: Redundancies in the Gravitational Gradient Tensor [23]

With Equation 15 and Equation 16 in mind, it becomes clear the gravitational gradient tensor has only five independent terms when the previous assumptions are included. A

fewer number of tensors to measure reduces the instrumentation burden, but provides less

unique information than a complete independent nine-component derivative would.
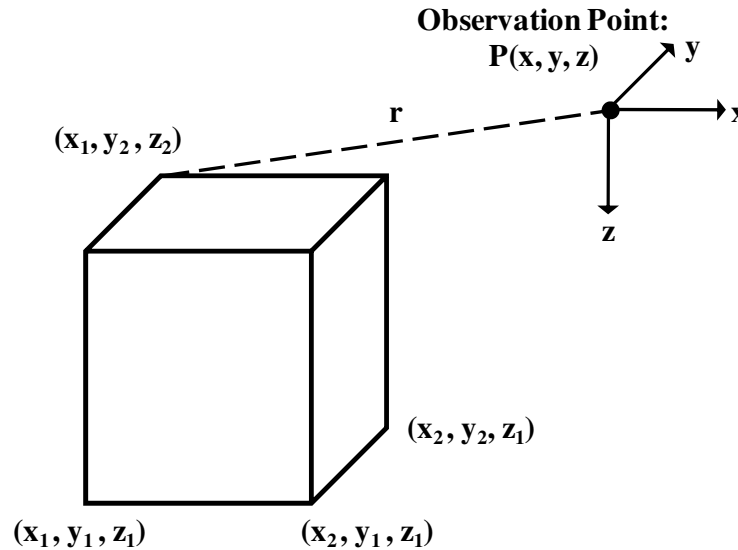
Figure 11: Rectangular Prism Annotated to Correspond with Equation 17, based on Nagy, Papp, and Benedek [24]

Those simplifications also allow derivation of closed form equations to calculate the

gravity gradient for simple objects. According to Nagy, Papp, and Benedek [24], the

gravity gradient disturbance tensor for any prism with a uniform density ρ can be

described by Equation 17. Figure 11 demonstrates the notation used in Equation 17.

Equation 17 implies each gravity disturbance gradient tensor calculation results

from adding eight calculated values—one for each of the prism corners. Note here

Equation 17 describes the gravitational disturbance gradient—rather than the total

gravitational gradient—at some point in free space outside of the prism interior. This

means implementing the equation will result in the gravity gradient contribution of the

prism only, and not, for example, the total gravity gradient reading if the prism were

30

sitting on the Earth's surface since the effect of the Earth's mass would not be included in the calculation.

$$T_{xx} = G\Delta\rho \sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}(-1)^{i+j+k}\tan^{-1}\left(\frac{(y-y_i)(z-z_k)}{(x-x_i)r}\right)$$

$$T_{yy} = G\Delta\rho \sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}(-1)^{i+j+k}\tan^{-1}\left(\frac{(x-x_i)(z-z_k)}{(y-y_i)r}\right)$$

$$T_{xy} = G\Delta\rho \sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}(-1)^{i+j+k}\ln\left((z-z_k)+r\right)$$

$$T_{xz} = G\Delta\rho \sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}(-1)^{i+j+k}\ln\left((y-y_k)+r\right)$$

$$T_{yz} = G\Delta\rho \sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2}(-1)^{i+j+k}\ln\left((x-x_k)+r\right)$$

$$r = \sqrt{(x-x_i)^2 + (y-y_j)^2 + (z-z_k)^2}$$

Equation 17: Closed Form Solution for the Independent Gravitational Disturbance Gradient Tensors for a Prism [24]

Because Γ represents the total gravity gradient, a gravity disturbance gradient is typically described by T. The gravity gradient caused by multiple prisms can be calculated by summing the calculated gravity gradient values for each prism. Thus, the gravity gradient for objects with a complex shape is approximated by dividing the object into many small prisms and summing the calculated gradients for each of the smaller prisms to yield the whole object gravitational disturbance gradient.

As an example of what a theoretical gravitational disturbance tensor might look like, Rogers [25] calculated a six-component map of the gravitational gradient caused by a theoretical $50m \times 10m \times 6m$ prism with a uniform density of $1.5g/cm^3$. Figure 12 shows the prism centered on a $250m \times 250m$ grid. If the prism were sitting on the Earth, the x- and y-axis would run along the surface while the z-axis would rise into the sky. Figure 13 shows the value of the independent gravitational gradient tensors at a constant height of $50m$ above the reference plane. Note the T_{zz} tensor is a function of the T_{xx} and T_{yy} tensors, meaning it is not independent and only shown in the figure for reference. The gradient map makes it clear that even with a complete, accurate gravitational gradient map in hand, determining the physical characteristics of the source object is not intuitive because the shape, density, and position of an object are all variable. A small, dense object far from a GGI causes similar gradient reading as a large, low-density object close to a GGI. Without some idea about at least one of the variables, using gravity gradiometry to identify a source object is essentially impossible.
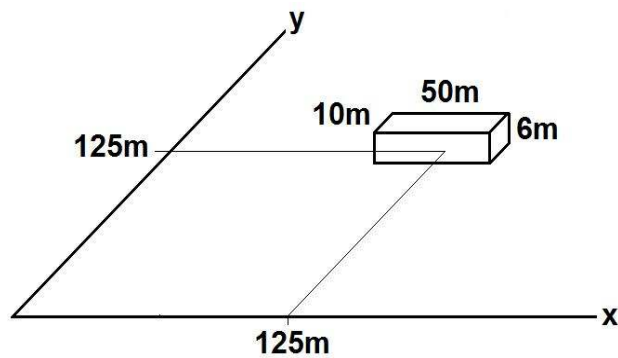
Figure 12: Hypothetical Prism, taken from [25]
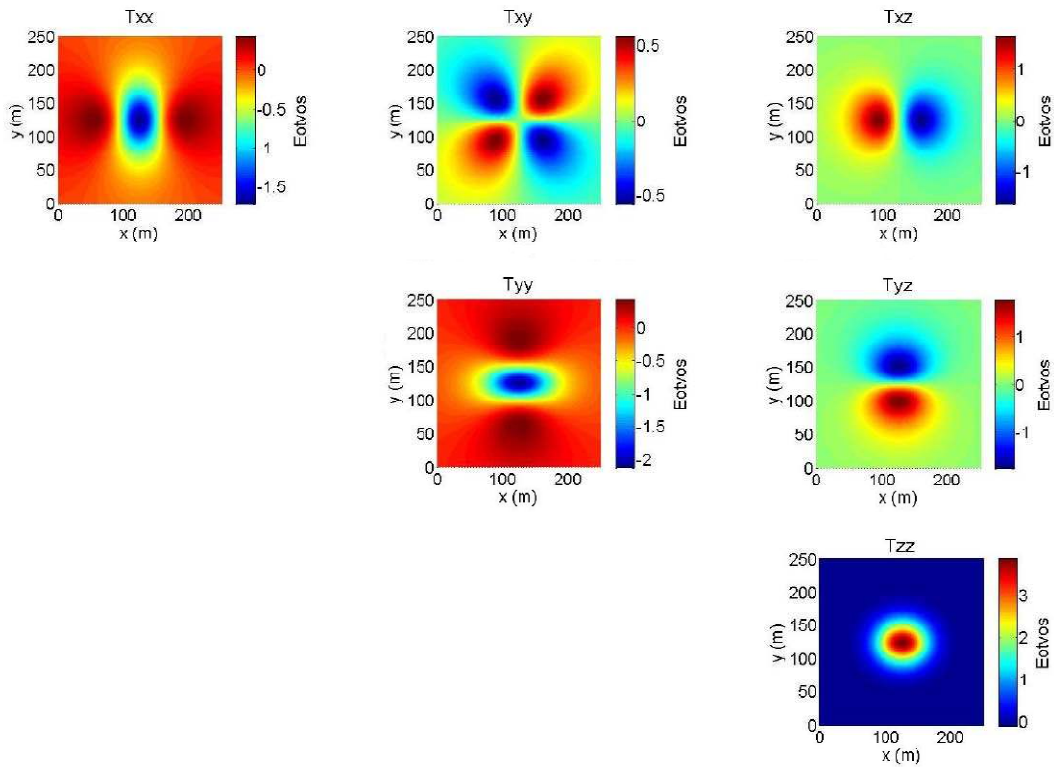
Figure 13: Gravitational Gradiometry Map of a Hypothetical Prism, taken from Rogers [25]

Figure 13 describes the strength of gravity gradients in units of Eötvös (*Eö*), which is equivalent to $10^{-9}s^{-2}$. Although not a recognized SI unit, the Eötvös is commonly used within the gravitational gradiometry community due to the diminutive magnitudes of measured gradients. In accordance with Newton's Law of Universal Gravitation, 1*Eö* is about equal to the gravitational gradient induced by 10 sand granules each with a mass of 1*mg* at a distance of 1*cm* [26]. While it might seem 1*Eö* is indicative of extreme sensitivity in gravitational gradient measurement, an object as massive as a 1000*kg* rock at a distance of 10*m* only creates a gradient of about 0.10*Eö*, a result owing to the derivative of Newton's Law of Universal Gravitation having an l^{-3} relationship.

It should be noted that 'gravitational' and 'gravity' do not technically refer to the same thing, with the former describing only the relationship defined by Newton's Law of Universal Gravitation and the latter describing the sum of all acceleration contributions in some system. The earth offers a good example of the difference, since the gravity vector is actually the sum of the acceleration induced by mass attraction (gravitation) and the centripetal acceleration caused by the earth's rotation. Richeson [21] explains the centripetal acceleration term is a constant value dependent on latitude and angular velocity, implying it can be treated as a bias and ignored in this investigation. Therefore, there will be no further distinction between 'gravity' and 'gravitational.'

Measurement and mapping of gravitational gradients is not a new concept and has been applied across a variety of industries. The first device for gravity gradient measurement was the product of Hungarian scientist Baron Roland von Eötvös. In 1890, he employed it to deduce the topography beneath a frozen lakebed, with accurate results confirmed by physical measurement in warmer months. However, the sensitive nature of the instrument made it bulky and created difficulty getting accurate readings in unfavorable weather conditions. Additionally, the multiple component gradient maps were almost undecipherable by inexperienced technicians causing gravity gradiometry to be altogether abandoned in favor of measuring the actual gravity vector even though it contained less information. Initial interest in gravity measurements stemmed from a desire to see beneath the earth and identify valuable density anomalies like salt domes, precious metal concentrations, and oil and gas fields. Today, gravity gradiometry

research continues in the areas of hypersonic aircraft navigation, airplane terrain avoidance, underground structure detection, and valuable resource detection [21, 27-30].

Existing Gravity Gradient Instruments

A complete investigation into the history and types of GGIs is not necessary in this study since the only parameter of interest is how well the best modern instrument measures gravity gradients (though the author recommends the thorough but readable treatment given to the development of GGIs by Rogers [25]). It is clear from the earlier discussion of the relationship between mass, separation distance, and gravity gradient magnitude that a PIDS using gravity gradiometry requires an extremely sensitive instrument capable of collecting data about every second. Development of modern GGIs has focused on implementation within navigation and geological surveying systems on an aircraft, driving a requirement for mobility. Such an environment is characterized by constant noise in the form of vibration and turbulence, so the goal of creating a highly sensitive GGI is not a trivial one.

Two approaches have emerged for creating a GGI with adequate sensitivity levels for a PIDS application. The first takes advantage of the Meissner effect and flux quantization and is classified as a superconducting gravity gradiometer (SGG). GGIs utilizing this technology and built for use onboard an aircraft have demonstrated sensitivities of $1 E\ddot{o}/Hz^{-1/2}$. However, designers of the most advanced SGG instrument have noted that if the mobility requirements were removed, designing a device with a sensitivity of $10^{-3} E\ddot{o}/Hz^{-1/2}$ is "quite possible" [31]. The second promising approach

utilizes atom interferometry to measure the path of an atom as it is subjected to external forces. Although the measured sensitivities of such an instrument are lower than the superconducting method at about $5E\ddot{o}/Hz^{-1/2}$, research remains robust [32]. The following development of a GGI-based PIDS assumes a stationary GGI exists capable of 10^{-3} $E\ddot{o}/Hz^{-1/2}$ sensitivity levels.

Perimeter Security System Metrics

The primary value of any PIDS lies in the system's ability to detect an intruder accurately across a wide range of operating conditions and entry methods while minimizing false alarms, all with little or no operator control. It seems the effort to determine the best PIDS among several competing systems is an opportunity to apply the scientific method by having some impartial authority set up multiple systems in the same area and compare how each system reacts to a variety of intruder scenarios. This approach is a sound one in cases where the area of intended use closely resembles the test area, but widespread variation between PIDS performance occurs based on topography, vegetation, wildlife, background noise, meteorological conditions, and type of soil or pavement. Williams [5] notes no such thing as a "typical site" exists since the particular combination of the factors previously listed make each site unique. He cautions extracting broad performance data from comparative, single-site testing, since a non-site specific test could result in unrealistic conclusions. Although cost is not a consideration during this early stage of a GGI-based PIDS technology evaluation, comparing multiple systems at every site requiring a PIDS is clearly cost prohibitive.

The security industry has developed a set of metrics used to characterize how well a PIDS works, though the type of metric used depends in large part on what role the PIDS is expected to fulfill. In their analysis of various volumetric intrusion detection technologies, Lester and Smith [33] describe the four objectives of the widespread "Defense in Depth" approach to perimeter security to be (1) Deter the intruder (2) Detect the intruder (3) Delay the intruder, and (4) Respond to the intruder. Although (3) lends itself to a physical barrier like a chain link fence or a barbwire impediment, an effective PIDS would contribute to the other three aspects of perimeter security.

The quantitative values used to describe how well a PIDS works is straightforward, though some understanding of statistics is necessary to frame the various values in the correct context. The most widespread measures of performance are the probability of detection, usually denoted by P_d and defined by Equation 18, and the false alarm rate, typically abbreviated FAR and defined by Equation 19.

$$P_d = \frac{Attacks\ Detected}{Attacks\ Performed}$$

Equation 18: Definition of Detection Probability [34]

$$FAR = \frac{Number\ of\ False\ Alarms}{Amount\ of\ Perimeter\ Observed \cdot Time\ Observed}$$

Equation 19: Definition of False Alarm Rate [34]

Horner and Leach [35] point out that requiring a specific percentage of attacks to be detected is not the same as specifying a P_d since no confidence intervals are specified. In other words, P_d could be estimated using the outcome of a series of tests and Equation

18, but the estimated P_d would only become the true P_d when the number of tests reaches infinity. To avoid confusion, the detection rate (*DR*) is often used in place of P_d when the value is estimated through experimentation as it more accurately reflects the significance of the value. For example, a PIDS detecting an intruder five out of five times gives a *DR* of 100%, but the limited test does not mean the system P_d is 100% as well.

Since the P_d is one of the main PIDS performance characteristics, a variety of statistical methods are used to calculate the true P_d at some confidence level given the *DR* and number of tests performed. Leach [36] considers the normal approximation to binomial, chi-squared, and Poisson approximation distributions in an effort to select the best statistical approach to take. Unless the true P_d is known beforehand, the number of tests is large, and there are more than five attempts not detected in the course of all the tests, none of the methods is suitable when P_d is expected to be very close to unity. Leach [36] notes a complicated, binomial-type expression can be applied if there is equal likelihood the true value of P_d is between zero and one. Equation 20 calculates the best guess for the detection probability, $P_{d\ Best\ Guess}$, using this method with the equal likelihood assumption for a 95% confidence interval, meaning an experimenter can expect 95% of the PIDS tested to have a true P_d value greater than $P_{d\ Best\ Guess}$ given some experimental data.

$$P_{d\ Best\ Guess} = \frac{Attacks\ Detected + 1}{Attacks\ Performed + 2}$$

Equation 20: Estimation of P_d Given Experimental Data

While the assumption of an even P_d distribution might seem unreasonable, Leach [36] explains commissioned PIDS are more likely to show P_d values near the extremes of zero and unity which works to balance the distribution and make the assumption more valid.

The *FAR* is also subject to confusion because manufacturers have varying definitions of a false alarm, and changing the magnitude and circumstances of the *time observed* quantity results in vastly different *FAR* values. In hopes of decreasing the apparent *FAR,* some PIDS distributors recategorize false alarm to only include instances where a system malfunction caused the false alarm and ignore cases where a known entity other than an intruder, like wildlife, caused the alarm. A *FAR* definition considering only internal system errors as sources for alarms is more appropriately called an unattributed alarm rate (*UAR*) since all attributed alarm sources are discounted [34]. This paper will use the traditional *FAR* definition given by Equation 19, an approach recognizing an alarm caused by anything other than an intruder to be a burden to the PIDS operator whether the source can be determined or not.

Reaching an overall conclusion about how one PIDS compares to another is difficult, even when several system metrics of interest are known exactly. Leach [34] has suggested a few comparative measures integrating P_d and *FAR* to form a single term, but notes a combining type of approach is only as useful as the formulation weighting. For example, the relative importance between P_d and *FAR* is dependent on the application, and there is ambiguity whether a logarithmic or linear scale should be used. Equation 21 gives two examples of comparative measures.

$$System\ Performance = \frac{P_d}{FAR} \quad or \quad = \frac{1}{(1 - P_d)FAR}$$

Equation 21: Examples of Overall PIDS Comparative Measures [34]

Leach [34] suggests the best method for determining the value of a system is to perform an investment analysis by calculating the net fiscal effect caused by the system characteristics, where factors like poor reliability, false alarms, installation cost, and operating cost are summed and subtracted by the amount of money saved by intruder prevention. The cost approach suffers from the formidable hurdle of supplying accurate quantitative data, especially in cases where the goal of intruder detection is to avoid symbolic damage rather than monetary loss, as might be the case with breaching a military base perimeter. Additionally, estimating initial and lifetime costs on a developing technology with no commercial equivalent is an impossible task. Evaluation of an overall system performance measure will thus have to be a work in progress since no good example already exists [34]. A few other PIDS evaluation characteristics are occasionally used including reliability, ease of operation, ease of training future staff, ease of expansion, vulnerability to defeat, and cost [34-36]. Those criteria are ignored in this study because they represent characteristics of a manufactured system with known operator interfaces, and this study aims to compare technologies rather than specific systems.

Unfortunately, typical performance metrics for various PIDS technologies are not available, probably because performance is so dependent on the specific application of any system. A 20-sensor seismic PIDS, for example, monitoring an oil refinery perimeter

in a flat, isolated, vegetation-free environment would perform differently than the same system installed on the vegetated, hilly perimeter of a nuclear power plant in a population dense area. Geographic differences, let alone variation in the number and quality of sensors installed in any given system, make direct performance comparisons imprudent. Despite dedicated efforts to establish an experimental perimeter designed to test commercial PIDS, most notably in the United Kingdom by the Home Office Scientific Development Branch, the effort remains in its infancy [37-39]. Hennin and Germana [40] specified several "common" and "typical" values of interest to a PIDS designer as part of an overview for an integrated perimeter security system, though the bases for those values are unknown. Table 3 shows some of the performance requirements.

Table 3: Common PIDS Performance Requirements, taken from Hennin, Germana, and Garcia [40]

Source	Minimum Cross Sectional Area	Minimum Speed	Maximum Speed	Operational Availability	Probability of Detection, P_d	False Alarm Rate, FAR
Hennin, Germana, and Garcia	$0.5 m^2$	$0.1 m/s$	$30 m/s$	99.9%	0.95	Facility Dependent

Meaningful evaluation of a PIDS depends on prior knowledge of what types of threats the system should face. Past evaluation of other PIDS has helped to lay the groundwork for formulating what a battery of defeat attempt tests might consist of, though no standard exists. Of five defeat tests performed in the evaluation of a combination of passive infrared and microwave sensors at an Australian university, only the slow movement and crawling test is applicable to testing a GGI-based PIDS. Other

41

tests, like moving behind a shield made of cardboard, aid GGI performance since increasing intruder mass increases the magnitude of the gravity gradient tensor at a given radius. Likewise, most attempts to provoke a false alarm in the same study—cycling a heat source, shining a white light at the sensors, and rigging an aluminum foil covered fan to spin in the test area—would not prove useful, though tests like small animal movement and random small object motion are applicable [33]. Although gravity gradiometry shares some common ground with other PIDS technologies, its uniqueness as a perimeter security technology warrants a specialized set of effectiveness tests to characterize its performance against different defeat attempts within a noisy environment.

III. Methodology

Since this investigation sought to examine the feasibility of using future technology to achieve an objective, all conclusions are the result of computer simulation rather than experimentation. Initial efforts resulted in a computer program capable of calculating the gravitational disturbance gradient caused by a human relative to a stationary GGI. A brief investigation into the nature of a human induced gravity disturbance gradient yielded an approach for an algorithm to detect intruders. Added options allowing the user to test system performance for various off-baseline conditions provided confidence an array of GGIs could detect and track a human in a realistic security situation. Devising metrics to identify the strengths and weaknesses of the algorithm rounds out the discussion in this chapter.

Simulation Overview

All simulations were performed using MATLAB R2010b (7.11.0.584) on a personal computer system running Microsoft Windows XP Professional with a Xeon X5482 processor and 3.25 gigabytes of random access memory. The general steps of the computer program are outlined next, with in depth explanation of each step following the general steps. The term 'computer program' refers to the complete MATLAB program that generates gravity gradient signals, runs the detection and tracking algorithm, and calculates performance metrics, while 'algorithm' refers to just the portion of the program that detects and tracks an intruder given gravity gradients from a GGI.

Figure 14 shows a flowchart of the computer program. The majority of the program involves setting up a grid of the detection area, placing the GGI array, and calculating gravity gradients at the grid locations. The computer program generates the gravity disturbance gradients used by the detection and tracking algorithm.
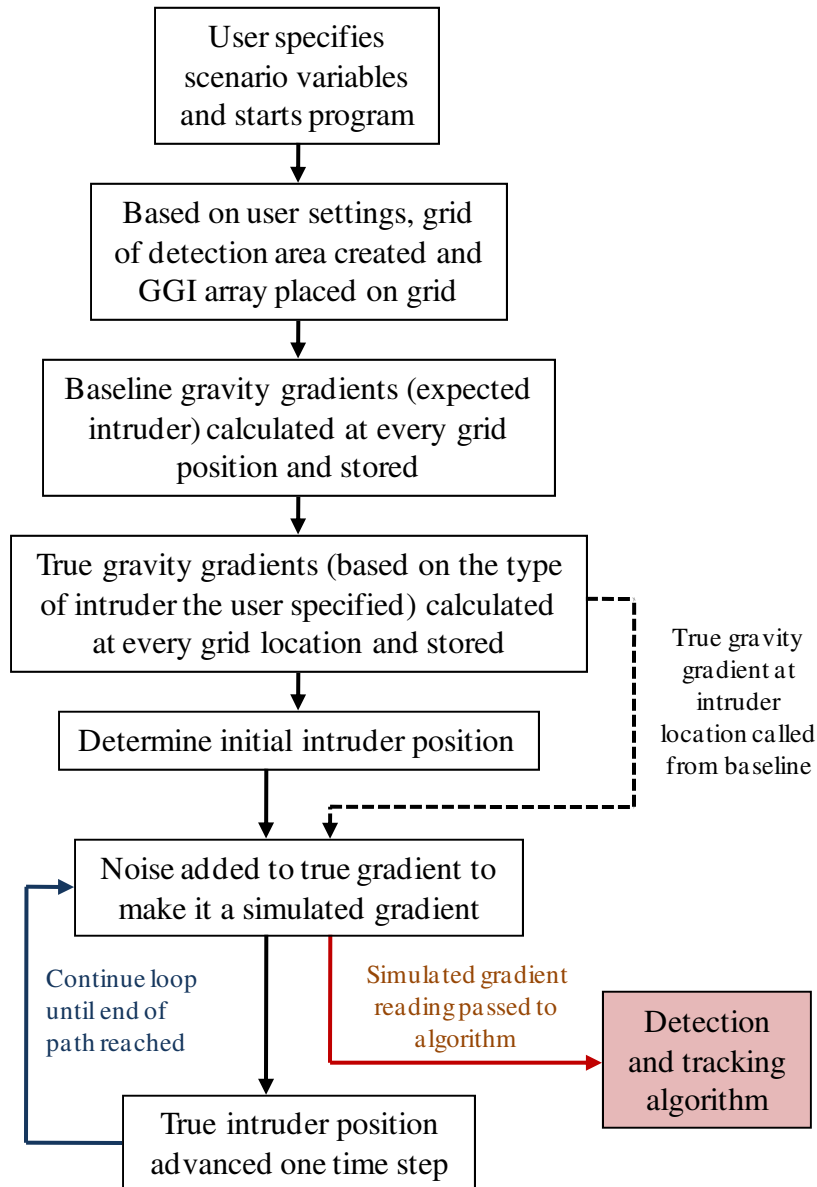
Figure 14: Computer Program Sequence

An installed GGI-based PIDS would use only the detection and tracking algorithm, since a real intruder would generate the gravity disturbance gradient a real array of GGIs would detect. Before running the simulation, the user must specify values for the controllable variables. These variables include aspects like GGI array geometry, intruder size and movement speed, and algorithm characteristics. The next section explains all variable options more thoroughly. When started, the program uses the user inputs to populate the array of GGIs on a grid of the area the PIDS is supposed to protect.

In general, the algorithm works by comparing the noisy gravity disturbance gradient reading with a map of truth, or baseline, gradient readings in the area near the security system. The algorithm detects and locates an intruder when all five tensors of the gradient reading closely match a location on the baseline gradient map. For this reason, the computer program has to generate two gravity gradient maps over the area of interest before simulating an intrusion. The true gradient map calculates the gradient caused by an intruder with user specified characteristics at every position in the area of interest. The baseline gradient map does the same thing, except gradients are calculated for an intruder with expected characteristics—the baseline map for a real GGI-based PIDS would be created during system calibration. If the expected intruder is a human but the true intruder is a large vehicle, the gradient maps will not be similar. Each GGI in the array gets its own true and baseline gradient maps, since the gravity disturbance gradient differs depending on GGI location.

With the true and baseline gradient maps created, the computer program determines the intruder starting position. The program then calls the five-component

45

gradient value for the intruder position from the true gradient map, and adds noise to simulate a real GGI reading for each of the GGIs in the array. Those readings are passed to the algorithm, which compares the reading to the expected intruder gradient map to decide whether to sound the intrusion alarm. Each time step advances the intruder position along a user-specified path, which generates a new simulated GGI reading. The program loops for a set number of iterations, and then ends by calculating algorithm performance metrics.

The following sub sections cover all aspects of the computer program in depth: available user inputs, the methodology for creating off-baseline conditions, GGI noise characteristics, the detection and tracking algorithm, and evaluation metrics.

User Inputs

Before starting the program, the user must set the intrusion situation by specifying a variety of program variables. The first choice is the scenario type. The initial aim of this project was to determine the feasibility of using gravity gradiometry as the primary technology in a perimeter security system, but perimeters come in many forms. A user begins by choosing between one of two types of scenarios. Intended to characterize how well the algorithm could detect an intruder when using a single cluster of GGIs, the open area scenario models intrusion detection for any small area in open space, like the footprint of an oil derrick, cell phone tower, or wind turbine. The second security scenario simulates large perimeter security with clusters of GGIs spaced at equal intervals along a line. The user must specify the number of GGIs in each cluster, as well as the

cluster geometry by specifying height off the ground, and GGI separation distance within the cluster. Figure 15 shows a bird's eye diagram of an example perimeter scenario with four GGIs in each cluster and dashed black lines showing a 20*m* range from each cluster center.
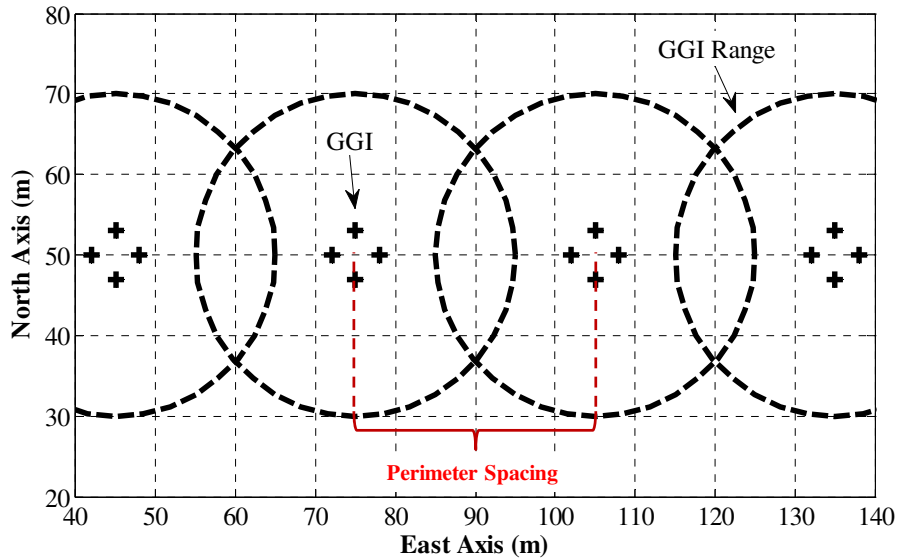
Figure 15: Bird's Eye View of Perimeter Scenario GGI Layout

While the figure only shows four clusters because of a magnified view, the perimeter actually extends further in both directions and there are more equally spaced GGI clusters along the perimeter. The same view of the open area scenario looks similar, but there is only a single GGI cluster in the area of interest instead of a line of GGI clusters. Figure 15 shows dashed black lines marking a radius of 20*m* from each cluster center. The 'Perimeter Spacing' measurement represents the distance between cluster centers along a perimeter.

The grid size, or area the PIDS is supposed to protect, depends on the scenario type. The open area scenario utilizes a 100*m* × 100*m* area with the GGI cluster in the

middle centered at position (50*m*, 50*m*), while the perimeter scenario grid has a variable

length depending on the GGI perimeter spacing selected by the user and a grid depth of

100*m*. Grid length for the perimeter scenario can never be more than 200*m*, however,

since GGIs placed further away contribute negligibly to intruder detection since the

gravity disturbance gradient for an intruder in the area of interest would be too small at

those large separation distances. In both scenarios, grid resolution is 1*m*, and the origin is

always at the point on the grid furthest South and West according to the NED reference

frame. The discussion on GGI noise later in the chapter will demonstrate the grid size

choice is reasonable.

There can be 1, 2, 4, or 6 GGIs in a cluster. The shape of each cluster of GGIs is

hard coded into the computer program, but the dimensions of the shape are a user input.

Figure 16 shows a single cluster with 6 GGIs. 'Cluster Spacing' indicates how far from

the center of a cluster each GGI is, and 'Cluster Height' indicates how far above ground

the cluster center is—note distance above the ground is negative in the NED reference

frame since the origin is at a height of zero and the positive direction is down. Therefore,

in the figure the cluster center is at a position of (50, 50, -10), or 10*m* above the ground.

A single GGI is positioned in the center of the cluster center; a 2-GGI cluster is arranged

with each GGI spaced equal distance from the cluster center along the East axis on the

horizontal plane; a 4-GGI cluster has GGIs spaced equal distance from the cluster center

along the East and North axis on the horizontal planes; and a 6-GGI cluster is arranged

like a six pointed tack so there is GGI variation on every plane, but all are equal distance

from the cluster center.
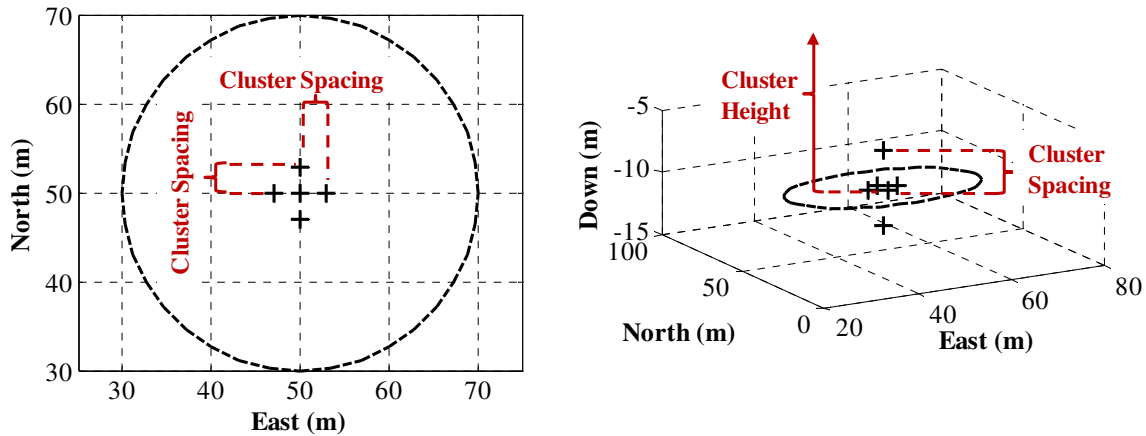
Figure 16: Description of Cluster Geometry

When setting up a scenario, a user can specify an intruder path by choosing the mean speed while choosing between two types of paths. Speed choices are slow, medium, or fast, and respectively designed to represent the speed of a human crawl, moderate walk, and fast jog. Table 4 shows the values used by the computer program based on speed choice.

Table 4: Intruder Movement Speed Values

Speed Choice	Representative Human Movement	Speed (m/s)	Speed (mph)
Slow	Crawl	0.3	0.7
Medium	Moderate Walk	1.5	3.4
Fast	Fast Jog	5.0	11.2

The user must choose between a linear or snaking path. The procedure for determining the intruder starting position is different for each scenario type, but common to the path type. When a user selects the perimeter scenario, either path type will move the intruder along a path roughly perpendicular to the perimeter. The initial North-axis

position is 45*m* south of the perimeter line, while the initial East-axis position is random within a defined range of allowable starting positions. The range of allowable East-axis starting locations is marked in green on Figure 17. The allowable starting region is centered at the middle of all modeled GGI clusters, and spans one perimeter spacing length. Therefore, for the example snaking path on the perimeter scenario shown in Figure 17, the allowable starting range extends 30*m* along the East-axis (since the cluster spacing is 30*m*), and is centered on the East-axis at 90*m* (since East = 90*m* is the center of the six GGI clusters).
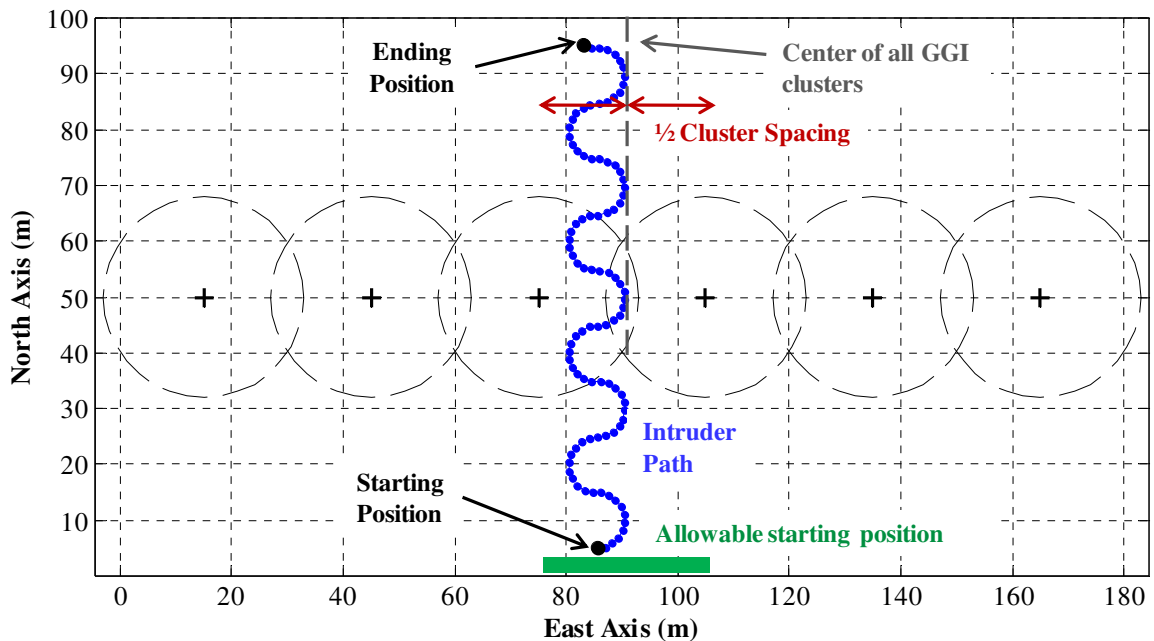
Figure 17: Example Snaking Path on Perimeter Scenario

Because the perimeter runs perpendicular to the East-axis 50*m* north of the East-axis, the North-axis intruder starting position is at 5*m* (since it is 45*m* south of the perimeter line). The starting position is random within the allowable starting region to ensure the intruder

path can fall everywhere within the range of one cluster spacing increment on successive runs. Since the allowable starting range encompasses the complete distance between any two GGI clusters, the algorithm's measured detection and tracking performance is the same as it would be for any position along an infinite perimeter.

For the open area scenario, starting position is determined by randomly selecting a start angle around the single GGI cluster. The starting position is a 45m distance from the GGI cluster center in the direction of the starting angle. The direction of intruder movement is directly at and through the cluster center. Figure 18 shows an example of the linear path on an open area scenario.
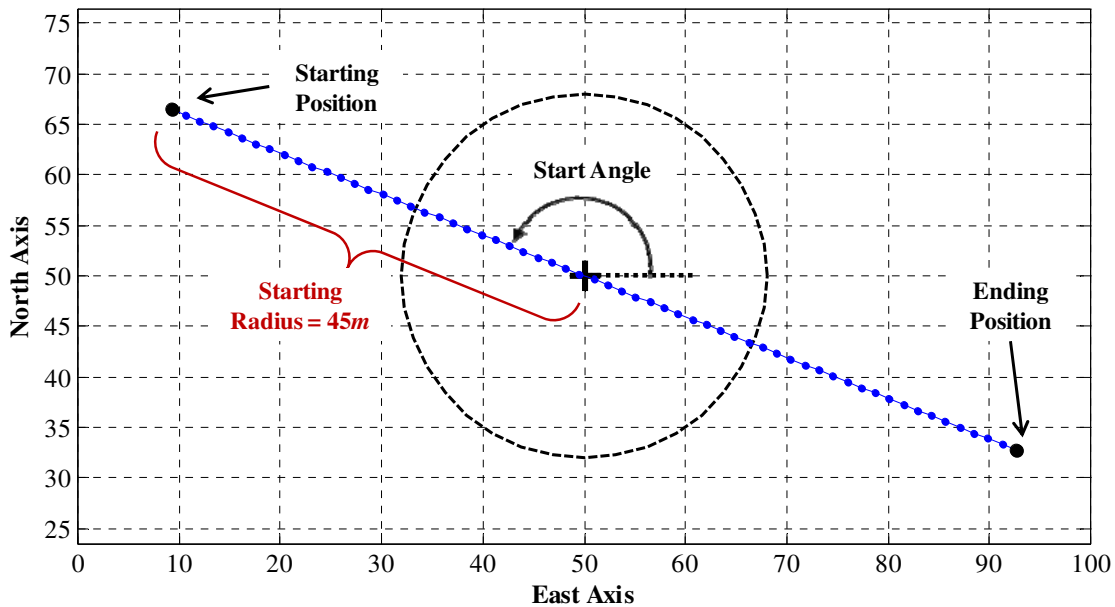
Figure 18: Example Linear Path on Open Area Scenario

The two different path type options represent intruder movement in two different ways to determine if the tracking ability of the algorithm is dependent on how an intruder moves. The linear path is the simplest case, and advances the intruder a single distance

increment according to the speed and time step. The snaking path is a series of 5m radius

half circles centered on a line in the path direction. Figure 17 and Figure 18 show

examples of both path types, with each position advance indicated by a small blue filled

circle. Both paths end as soon as the intruder is as far from the perimeter (in the case of

the perimeter scenario) or GGI cluster (in the case of the open area scenario) as when the

simulation started.

The user can test algorithm performance on off-baseline conditions using three

Since the closed form gravity gradient equations presented in the last chapter are

only valid for a prism aligned with the axis system, determining the gravity disturbance

gradient for a human or any other non-prism object requires approximating the object's

shape as a prism. The overall goal is to create an approximating prism with a constant

density but same total mass as the true object, a simple task given the dimensions of the

prism.

The user can test algorithm performance on off-baseline conditions using three

sizes of humans, as well as two objects other than humans. Table 5 shows all intruder

choices. For humans, height is the standing height of a human, width is the shoulder

breath of a standing human, depth is the linear distance from the small of the back to the

front of the waist, and mass is human mass. All human values in Table 5 are from a

compilation of human anthropometric data, where values for a 'small' human are the

reported values for a 5th percentile male, values for a 'medium human' are those for a

50th percentile male, and values for a 'large' human are those for a 95th percentile male.

While each dimension for a given human percentile does not necessarily correspond to

the same person, taken together they provide a good way for approximating the size of a

small, medium, and large human approximating prism in this application. Dimensional

values for the rabbit and large vehicle are the author's representative approximations and

not reported from any literature source.

Table 5: Dimensions and Mass of Various Objects, based on Woodson, Tillman, and Tillman [41]

Object	Height		Width		Depth		Mass / Weight	
	(ft)	(m)	(ft)	(m)	(ft)	(m)	(lb)	(kg)
Small Human	5.3	1.62	1.4	0.42	0.6	0.18	124	56.2
Medium Human	5.7	1.73	1.5	0.45	0.8	0.25	168	76.2
Large Human	6.1	1.85	1.6	0.50	1.0	0.31	224	101.6
Rabbit	0.4	0.13	0.5	0.15	0.8	0.25	5	2.3
Large Vehicle	6.2	1.90	6.6	2.00	18.0	5.50	8000	3629.0

The user can select the motion position of the human by choosing between an

upright or prone position. If the prone position is chosen, the program swaps the height,

width, and depth values to lay the prism on its face. Changing the motion position is not

an option for the rabbit or large vehicle. With the prism approximations in place, the

program uses the closed form equations for calculating the gravity gradient to generate

the true gradient map for the entire grid area.

Significant water mass in the form of precipitation falls onto the ground during

any extended period of rainfall and contributes to a gravity gradient reading. The final

user input allows the user to include the mass distribution effects of rain into the true

gravity disturbance gradient calculation. The contribution of rainfall to the gravity

gradient reading can be estimated by considering rainfall is absorbed some depth into the

ground over a given time and rainfall rate. If those values are known, the density

contribution to the ground can be calculated by dividing the total mass of the rain by the

volume of the soil it will be absorbed into, as shown in Equation 22 where $\rho_{+,soil}$ is the

soil density contribution due to rain, $\dot{r}_f$ is the rainfall accumulation rate, ρ_{H2O} is the

density of water, and d_p is the rain penetration depth.

$$\rho_{+,soil} = \frac{mass\ of\ rain}{volume\ of\ soil} = \frac{\dot{r}_f t \rho_{H2O}}{d_p}$$

Equation 22: Soil Density Contribution Due to Rain

Using the same definition for 'light' and 'heavy' rainfall the American Meteorology

Center uses along with Equation 22, Table 6 shows the density contribution over one

hour of rainfall and 0.5m rain penetration into the soil [42].

Table 6: Overview of Rainfall Characteristics

Precipitation Type	Intensity of Rainfall (cm/hr)	Density Contribution (kg/m^3)
Light Rain	0.25	5.0
Heavy Rain	0.76	15.2

Gravity gradient contribution from rain is then calculated by creating a prism with

density $\rho_{+,soil}$ and a depth dimension of 0.5m. The other two dimensions must be large

enough to approximate a surface. Since there is no computation penalty for calculating

the gravity disturbance gradient of a very large prism, the side dimensions of the prism

are 2000m each because it extends 1000m in each surface direction. Figure 19 shows how

the rain prism relates to the grid and GGI clusters. The contribution due to rain is only an

approximation and assumes the same amount of rain falls in every location on the grid,

the rain only soaks to a depth d_p, and the rain soaks into the ground in such a way so the water is evenly distributed in the prism.
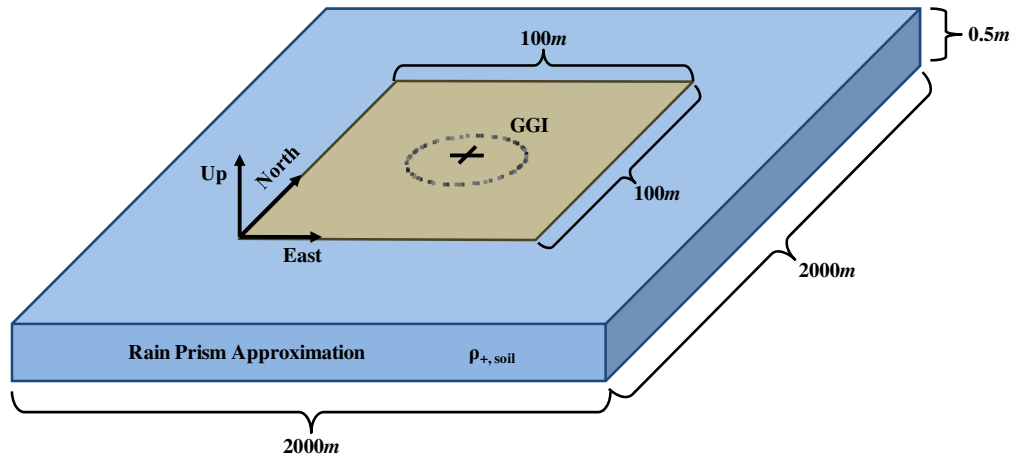
Figure 19: Diagram of Rain Prism (not to scale)

GGI Noise

Realistically simulating the reading of a GGI requires adding noise to the true gravity disturbance gradient. According to Rogers [25], gravity gradiometer manufacturers promote their instruments as having zero-mean Gaussian white noise over some bandwidth. The noise spectral density (NSD) parameter describes GGI noise, and it represents the power of GGI noise over a range of frequencies. Given a NSD and sampling frequency f_s, the root mean squared (RMS) noise can be calculated using Equation 23.

$$RMS\ noise = \sqrt{[NSD]^2 \cdot \frac{1}{2} f_s}$$

Equation 23: GGI RMS Noise Calculation

Since the noise is known to have a mean of zero, the RMS noise is equal to the noise standard deviation [25]. In his discussion of GGI noise, DeGregoria [27] notes the noisy signals produced by GGIs are often filtered using a low pass Butterworth filter to reduce noise. While any technique used to reduce noise is looked favorably upon for this investigation, a Butterworth filter is recursive because it uses previous data readings to filter the most current reading. A GGI signal is very small when a human is far away and relatively large when the human is close to the GGI, filtering causes too much signal lag and prevents accurate detection and tracking when the intruder is moving quickly. At the expense of a noisier signal, this investigation will forgo low pass filtering. After calculating an RMS value, MATLAB's "normrnd" function introduces noise into the calculated gravity disturbance gradient. Table 7 shows the expected RMS noise for several NSD values with a sampling frequency of $1Hz$. Recall a source in the previous chapter suggested a NSD of $0.001 E\ddot{o}/Hz^{1/2}$ would be likely in the future. Modern high performance GGIs perform internal filtering before outputting a signal, so a $1Hz$ is about the maximum signal production rate (sampling frequency).

Table 7: Future Unfiltered GGI Noise Characteristics

GGI Type	NSD ($E\ddot{o}/Hz^{1/2}$)	Sampling Frequency (Hz)	RMS Noise ($E\ddot{o}$)
Near Future	0.01	1	0.00707
Projected Future	0.001	1	0.00071
Long Term Future	0.0005	1	0.00035

In combination with the RMS noise in Table 7, the single tensor derivative of Newton's Law of Universal Gravitation provides insight into the effective range of a

56

GGI-based PIDS, and can help to determine what grid size is needed. DeGregoria's [27] reported derivative approximation for the magnitude of a gravity gradient is given in Equation 24 where the notation is the same as it was for Newton's Law of Universal Gravitation.

$$|\Gamma| \approx \frac{2GM}{l^3}$$

Equation 24: Gravity Gradient Approximation

That equation and the RMS noise calculations for the three GGIs shown in Table 7 yields Figure 20. It shows the distance from a GGI in which a large human intruder (mass = $101kg$) has a GGI signal to noise standard deviation ratio of unity. The least noisy GGI, with a NSD of $0.0005E\ddot{o}/Hz^{1/2}$ and noise RMS of $0.00035E\ddot{o}$, has a signal to noise standard deviation ratio of one at about $34m$.
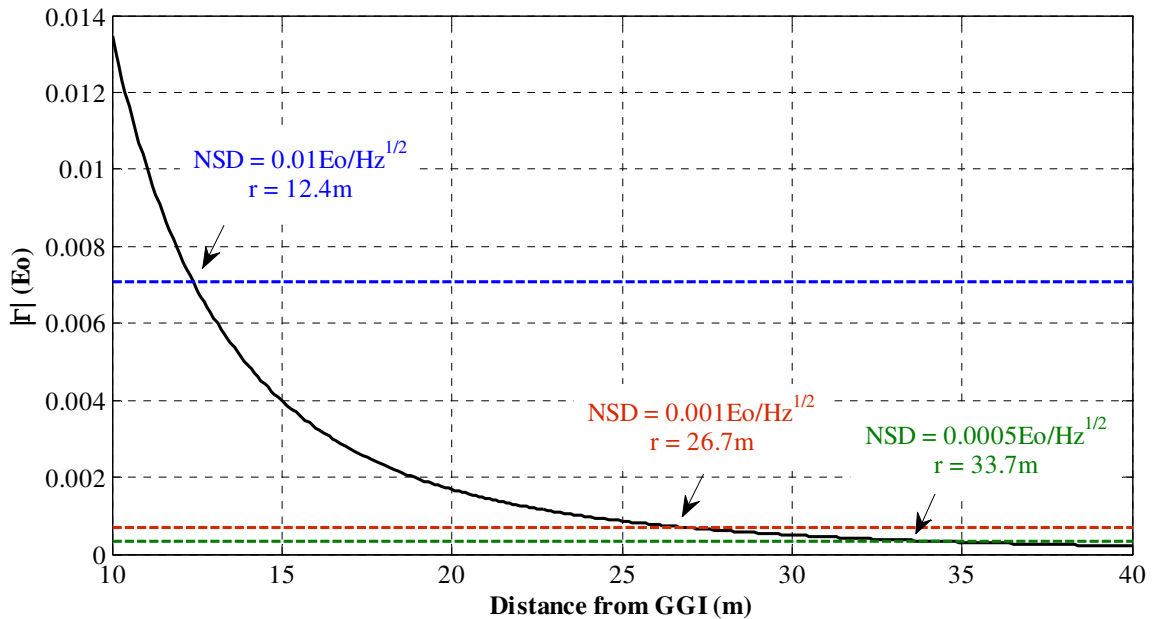
Figure 20: Estimated Maximum Effective Range for GGI Based PIDS

The graph itself is an optimistic prediction, since only about 68% of a normal distribution

falls within one standard deviation of the mean. Therefore, the noise levels can be more

than twice as high as the graph shows, further cutting anticipated effective range. Figure

20 confirms the grid size used for each intrusion scenario is appropriate, since the

minimum distance from GGI to grid edge for any intrusion scenario is 50m—well outside

the expected detection range for the noise characteristics evaluated.

Choosing to model the effect of rainfall soaking into the ground on a GGI reading

is a user option, but there is no model consideration for the raindrops themselves as they

fall through the atmosphere. Rainfall contributes little to the mean atmospheric density,

but ignoring the gradiometry effect of falling raindrops requires validation due to the

extreme sensitivity of the GGIs used for this project. Rain's density contribution to

atmospheric air is similar to rainfall's density contribution to soil discussed earlier.

Equation 25 calculates the mean atmospheric density increase caused by rainfall where t

is time, A is some surface area, V_r is the raindrop falling velocity, ρ_{H2O} is water density,

and $\dot{r}_f$ is the rainfall accumulation rate.

$$\rho_{+,air} = \frac{mass\ of\ rain}{volume\ of\ air} = \frac{\dot{r}_f t A \rho_{H2O}}{AV_r t} = \frac{\dot{r}_f \rho_{H2O}}{V_r}$$

Equation 25: Atmospheric Density Contribution Due to Rain

The atmospheric density contribution equation reduces to include only the rainfall

accumulation rate, water density, and raindrop falling velocity terms. Greater

accumulation rates result in higher atmospheric density while faster drop falling

velocities lower the atmospheric density contribution. Beard [43] shows rainfall terminal

velocity depends on a variety of factors, but ranges from 9*m/s* to 13*m/s* for drops with an equivalent spherical diameter larger than 3*mm*. Approximating an atmospheric prism is more complex than it was for approximating a soil prism since a GGI is above ground and within the atmosphere, while prism gradient equations are only valid for a reading location outside of the measured prism. Deployed GGIs are likely to be under a small cover, blocking rainfall and in effect carving a small area out of a larger, uniform-density rainfall prism. Figure 21 shows the rainfall prism with a small area kept dry by a cover over the GGI. Like the soil density contribution prism, the raindrop prism extends 1000m away from the GGI in every direction to simulate the gradient contribution from a whole atmosphere of rain. The GGI is inside of a small prism marked with dashed lines, representing the dry area. The size of the dry area was varied to determine if an installed GGI needs a minimum shelter roof size to increase distance between the falling drops and the GGI.

Figure 22 shows the calculated value of the strongest gravity gradient tensor (T_{xx}) for the raindrop prism shown in Figure 21. The plot shows the gradient value for multiple shelter radius values, along with notable rainfall rates and the noise standard deviation for a GGI with a NSD of 0.001*Eö/Hz$^{1/2}$*. T_{xx} has the largest magnitude of all tensors except for T_{yy}; T_{xx} is the same of T_{yy}, while the other three tensors are many orders of magnitude smaller.
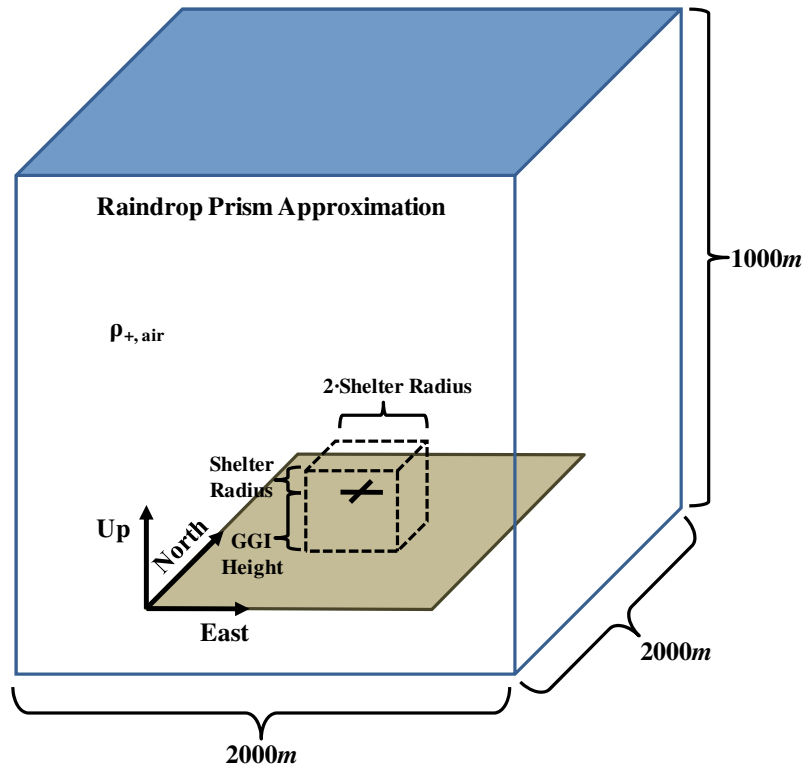
Figure 21: Diagram of Raindrop Prism (not to scale)

The plot shows the induced gravity gradient of a heavy (0.76*cm/hr*) rainfall is not

significant relative to GGI noise, even with a small shelter roof dimension of $0.2m \times$

$0.2m$. Rainfall rate does not contribute significantly to the gravity gradient reading until it

reaches about $0.03m/hr$, or just over an inch an hour. That level of rainfall is indicative of

rare, extreme weather, so ignoring the effect of rainfall in the atmosphere is a reasonable

assumption. Figure 22 shows an installed GGI should be sheltered in a structure with as

large a roof as possible to create maximum separation distance between falling rain and

the GGI. All rainfall approximations assume raindrops fall at the same rate in the area

surrounding the GGI, fall straight down, and together have a uniform mean density in the

atmosphere.
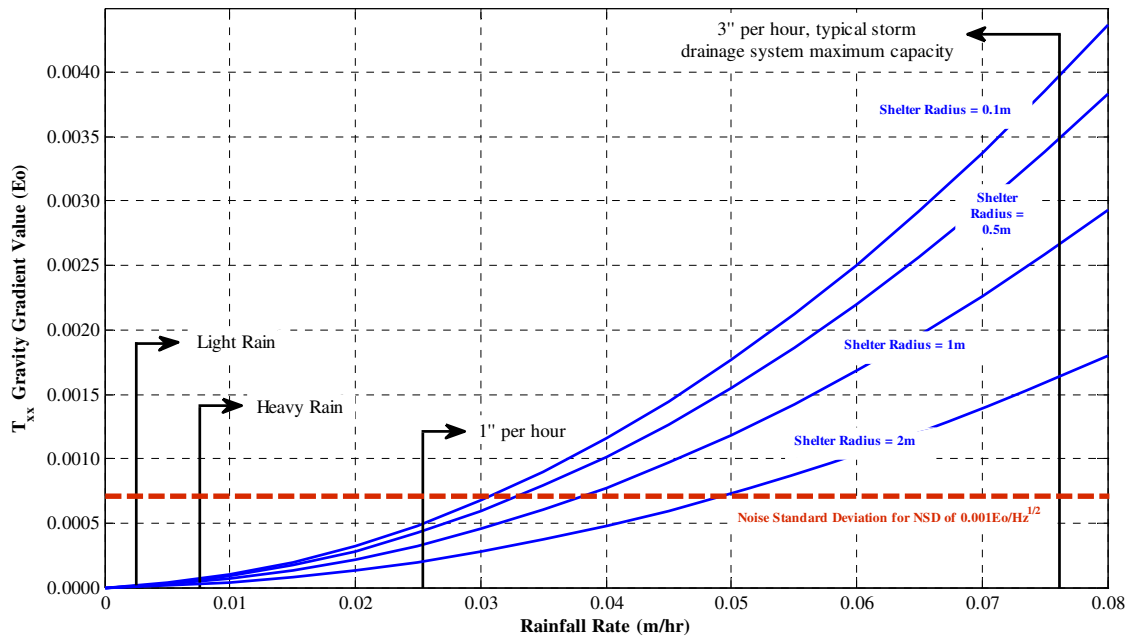
Figure 22: Effect of Rainfall on T_{xx} Gravity Gradient Tensor

In summary, the computer program utilizes user inputs to create an intrusion situation. The program generates the noisy, five-tensor gravity disturbance gradient the detection and tracking algorithm uses to detect and locate an intruder. While available user inputs do not cover every conceivable intrusion scenario, inputs do test algorithm effectiveness against varied intruder size and approach type, multiple GGI array geometries, and natural noise sources like rain. The rest of this chapter covers the detection and tracking algorithm, algorithm performance metrics, and the simulation test plan.

Detection and Tracking Algorithm

This section describes how the detection and tracking algorithm uses the noisy, five-component gravity disturbance gradient tensor to detect and localize and intruder.

61

The computer program flowchart showed the only information passed to the algorithm is the gravity disturbance gradient. Previous analysis of the gravity disturbance gradient equations for a prism indicated a gravity disturbance gradient reading is the result of three source object characteristics: size, density, and position. It might seem five independent gradient tensors could provide enough information to close a problem with only three unknowns. However, each tensor contains so little information that the five independent tensors together are only capable of determining one of the three source object characteristics given the other two characteristics.

An example gravity disturbance gradient for a human can help to illustrate the problem. Figure 23 shows contour plots of each of the five no-noise, independent gravity disturbance gradient tensors for a medium human, as determined by a single GGI $3m$ above the ground and positioned on the surface grid at ($50m$, $50m$). The plots show the baseline gravity gradient map for each tensor, with the contour values representing the gravity disturbance gradient caused by a medium human relative to the GGI at position ($50m$, $50m$). Once again, the black '+' symbol represents the GGI location, while positive gravity gradients have green contours and negative gradients have red contours. As expected, the gravity gradient for each tensor gets larger as the intruder gets closer to the GGI. The contour distribution for each of the plotted tensors shows why determining the precise location of a human requires more than one tensor.

Note: All contour values are in Eötvös
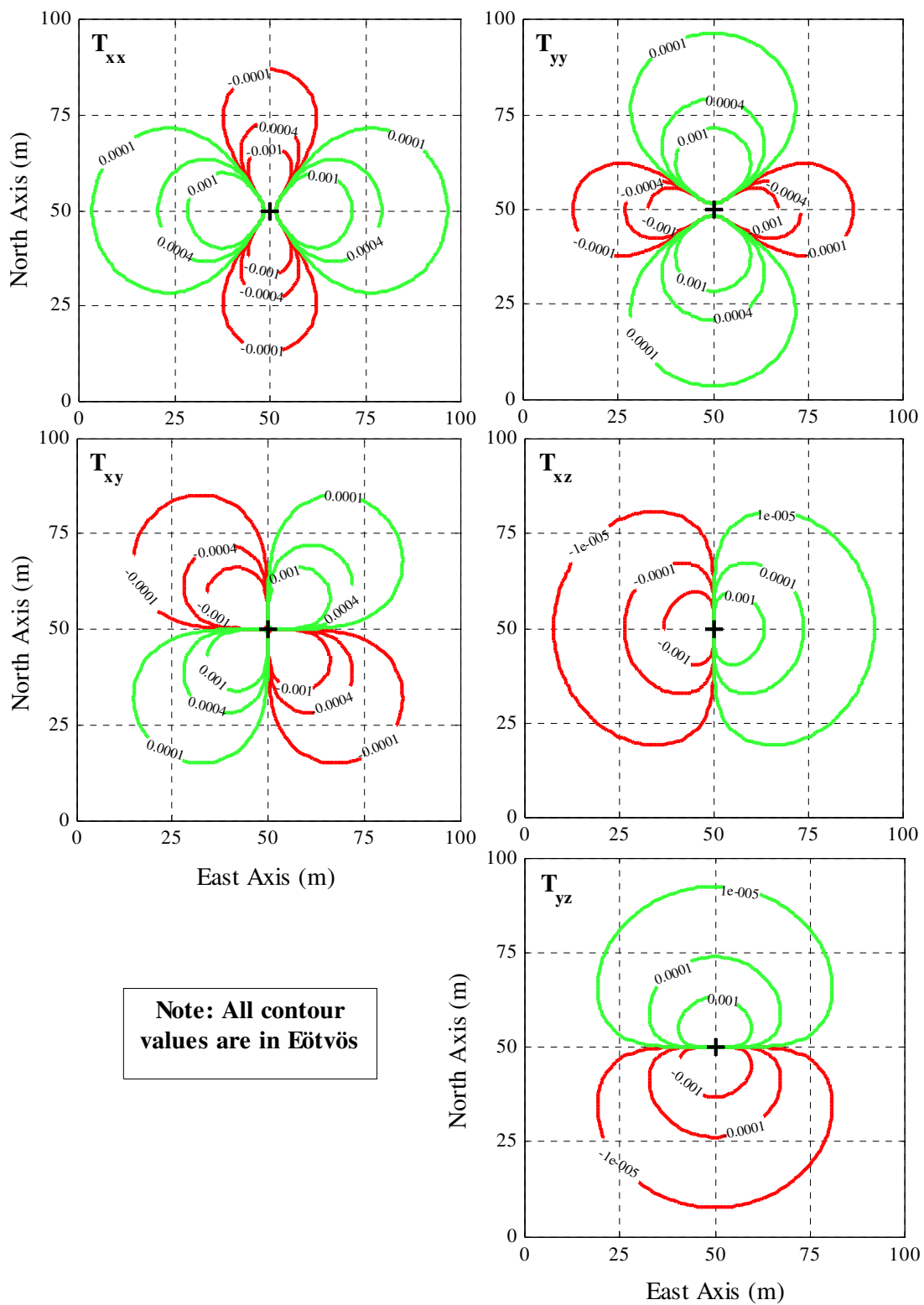
Figure 23: Five Component Gravity Gradient Contour for a Medium Human

63

For example, a T_{xx} reading of 0.0001$E\ddot{o}$ means the individual could be anywhere along the 0.0001$E\ddot{o}$ contour in the figure. A single gradient tensor reading is not good enough to determine direction or radius from the GGI. Observing both the magnitude of the contours for each tensor component and distance from the GGI suggests the T_{xz} and T_{yz} tensors are weak compared to the other three tensors.

Based on Figure 23, Table 8 shows the maximum distance from the GGI where a medium sized intruder would first cause a tensor gravity gradient of 0.001$E\ddot{o}$ is largest for the T_{xx}, T_{yy}, and T_{xy} components. Practically, this means those components will be the first to detect an intruder for any level of noise.
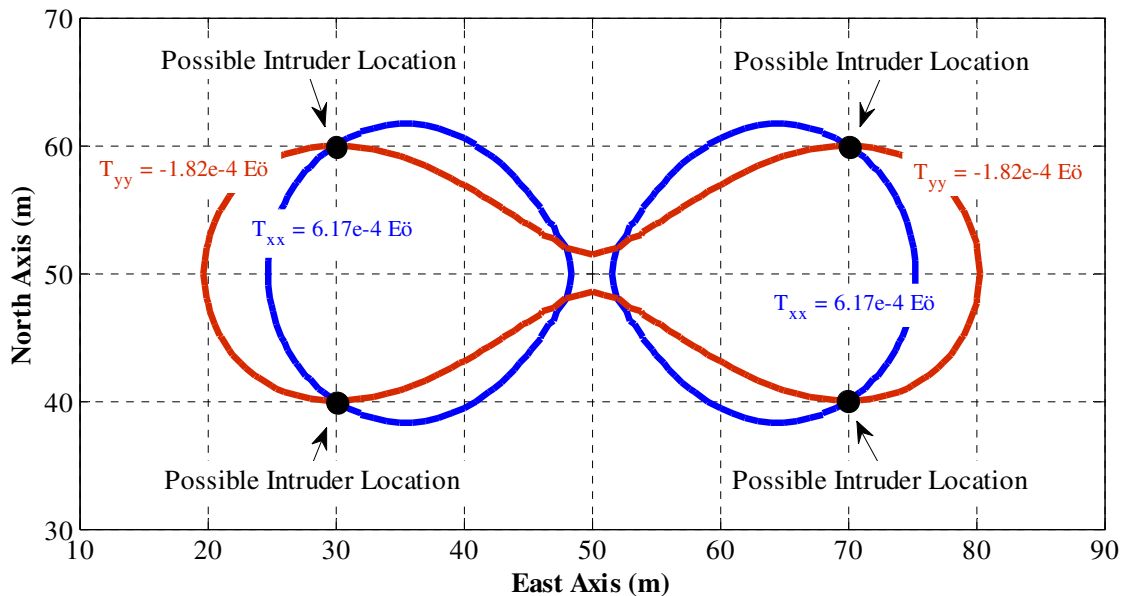
Figure 24: Determining Intruder Position Using T$_{xx}$ and T$_{yy}$ Components

Figure 24 shows how using only two components begins to narrow the intruder position. After plotting the T$_{xx}$ and T$_{yy}$ tensors readings on the same grid, it becomes obvious the intruder must be in one of the four positions where the reading contours

overlap. Figure 25 shows how considering four components makes deducing the true intruder position possible, because there is only one position on the grid where all the stored gravity disturbance gradient readings correspond to the actual GGI readings. While contours for only four components appear in Figure 25 for clarity, the contour of the non-plotted T_{yz} component also passes through the true intruder position.

Table 8: Maximum Radius for Medium Sized Human and Gravity Gradient of 0.001$E\ddot{o}$

Tensor	Approximate Distance (m)	Relative Strength
T_{xx}	22	Strong
T_{yy}	22	Strong
T_{xy}	20	Strong
T_{xz}	13	Weak
T_{yz}	13	Weak

A small random shift in all the contours plotted in Figure 25 helps to illustrate the negative effect the addition of noise has on determining an intruder position. After such a shift—which represents noise, or a small difference between the true gradient and the detected gradient—the tensor contours might not all overlap at a single point, or there might be multiple points where several components overlap suggesting more than one intruder position. Nygren [44] suggests a method for determining the likelihood of position on a grid given a set of noisy measurements if the measurement error is Gaussian. Equation 26 determines the most likely intruder position if each gravity gradient component is viewed as a unique measurement.
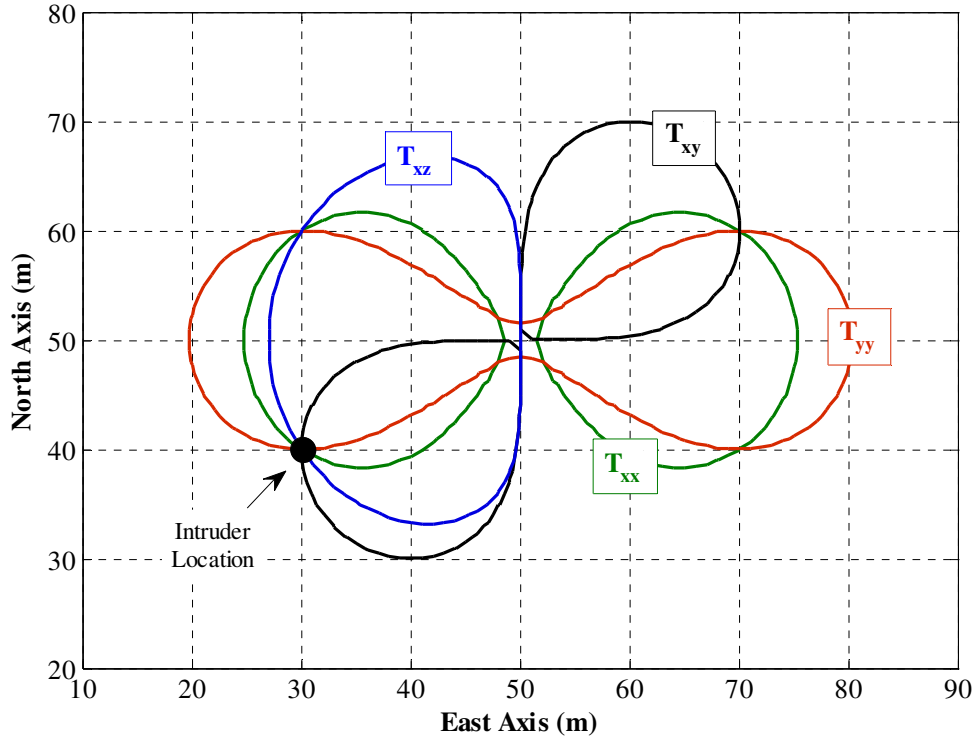
Figure 25: Determining Intruder Location Using Overlaid T_{xx}, T_{yy}, T_{xy}, and T_{xz} Component Contours

$$L(x_t, y_t) = \frac{1}{\sqrt{(2\pi\sigma_e^2)^N}} \exp\left(-\frac{1}{2\sigma_e^2}\sum_{k=1}^{N}(m - h_k(x,y))^2\right)$$

Equation 26: Likelihood Function

In the likelihood function, σ_e is the standard deviation of the measurement error, N is the number of measurements, x and y make up the Cartesian coordinate positions on a grid, the subscript t indicates a single location on the grid, m is the measured signal value at x_t and y_t, and h is true signal value at all grid locations. The true signal value h is the baseline gradient map containing the gravity gradient for the expected intruder type. The baseline gradient map is a matrix of gradient values at each grid position, so the result of

the likelihood function is a matrix of relative likelihood values for the entire grid. The

portion of the function before the exponential term does not vary with grid position, so it

acts like a bias. Dropping it from the calculation reduces computing power required

without changing the results since the function determines relative likelihood of a grid

positions. Figure 26 shows implementation of the likelihood function (Equation 26) for a

no-noise GGI and intruder position at ($30m$, $40m$), the same situation presented in the

previous two contour plots for both the two-component and four-component readings.

Unlike a contour plot, the likelihood function calculates relative likelihood at every grid

location. The likelihood maps show the most likely intruder locations are in the same spot

on the grid where contour plot lines cross. There are large grid areas where position

likelihood is almost as likely as the maximum when using only the two strongest

components to generate the likelihood map. This observation suggests accurate position

estimation requires all independent gradient tensors while demonstrating the low amount

of information contained in each tensor by itself.
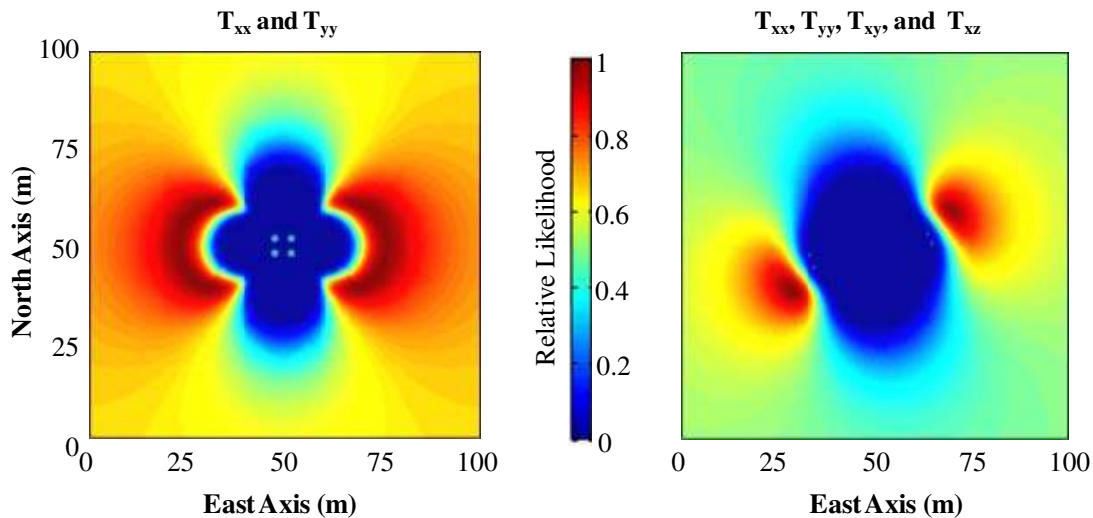
Figure 26: Example Intruder Location Determination Using Likelihood Function

Element-by-element multiplication rapidly combines likelihood maps generated for each GGI reading to form a single likelihood map for a grid with an array of GGIs. A user input averages each tensor of a GGI reading within a cluster before applying the likelihood function. Reading averaging is expected to reduce the number of incorrect position guesses since the mean of multiple noisy values should have an error more consistently closer to zero than any measurement on its own. The option to either use each GGI reading on its own or average the reading is called 'logic type.'

The algorithm also considers the movement capability of a human intruder when calculating position likelihood. If a human is detected at some position on the grid, it is unlikely the human could be in a position 50*m* away from the previous estimate in the next one-second cycle since a 50*m/s* position change is well outside the human range of mobility. Increasing likelihood in the areas where a human could move in one time step and reducing likelihood of position in the areas where movement is improbable aids the tracking ability of the algorithm. A multivariate normal distribution centered on the previous GGI reading cycle's most likely position estimate is a simple method of modifying the raw likelihood map. The distribution assigns a movement likelihood value to each grid position based on an expected movement speed to create a three dimensional likelihood curve. With no information known about the path, the multivariate normal distribution places the most likely position of current reading at the same location as the previous reading, and the likelihood decreases radially from that spot. For example, Figure 27 shows the likelihood curve for intruder position location given a previous position of (30*m*, 30*m*), movement speed standard deviation of 10*m/s* in the North and

East axes, and a time step of one second. The highest movement likelihood is in the area surrounding the previous position estimate, and the lowest likelihoods are at the grid positions furthest from the previous position estimate.
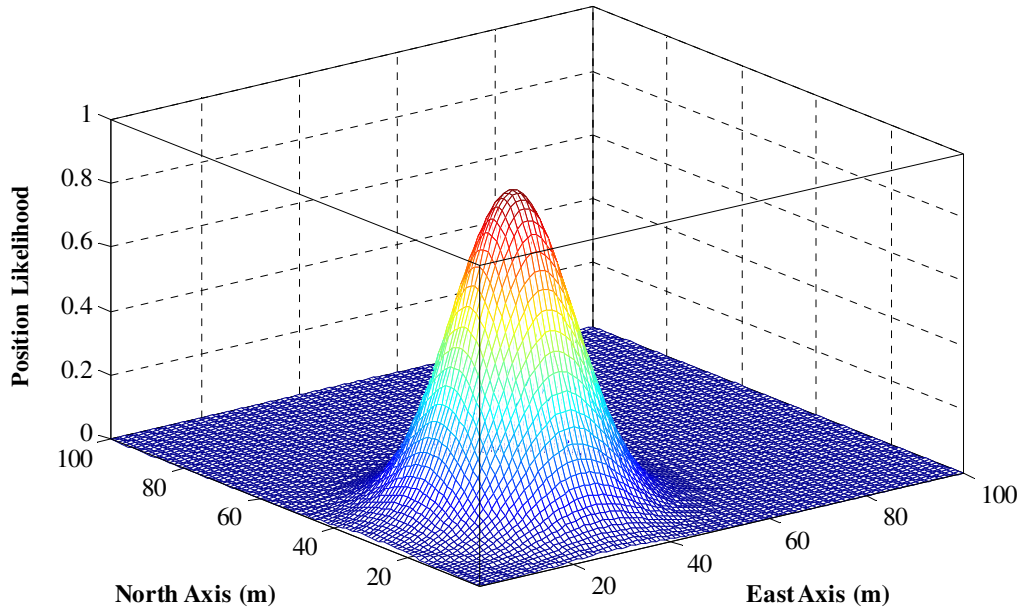
Figure 27: Movement Likelihood Radius Given a Previous Location at (30*m***, 30***m***)**

Element-by-element multiplication of the movement likelihood radius with the results of the likelihood function results in a revised likelihood map taking into account both a set of GGI readings and the limitations of human movement. The movement speed standard deviation was set at 10*m/s* in both axes for this investigation because that value creates a large movement likelihood footprint while simultaneously eliminating areas of the grid very far away from the initial position estimate. Applying this modification of the likelihood values created from the likelihood function alone is only possible after an intruder has been initially detected, meaning it can help track an intruder but will have no

positive effect until an initial intruder position is determined. If no intruder is detected the previous time step, the movement likelihood map is set to unity at all grid positions.

The earlier contour plots of a human gravity gradient demonstrated an object will always have a gravity disturbance gradient value, but the value approaches zero as the separation distance between the object and a GGI increases. At some separation distance, the magnitude of the gradient reading becomes much smaller than the GGI noise level resulting in grid likelihood values more reflective of the random nature of instrument noise rather than an actual disturbance gradient. The algorithm avoids the problem by accepting a position estimate only when the estimate is close enough to the GGI so the gradient magnitude at the estimated position is at least as large as the two-standard-deviation GGI noise magnitude at the intruder position.

Figure 28 shows the area on a grid where at least one tensor of the gravity disturbance gradient for a medium human is greater than the two-RMS noise level for a GGI with a NSD of 0.001 $E\ddot{o}/Hz^{1/2}$. A dashed black circle shows a 19m radius from the GGI in all directions, and estimates the radius where some of the gradient is larger than a two-RMS noise level. The 19m radius is an approximate GGI range within which a likelihood maximum could be often trusted to be accurate because the intruder signal is large relative to instrument noise. Though the likelihood maximum could occur at any position on the grid, the algorithm considers it indicative of an actual intruder when within the expected GGI range. A larger expected GGI range could potentially detect an intruder at a larger separation distance from the GGI, but such an advantage comes at the expense of more false alarms since the noise level is more likely to be closer to gradient

magnitude. Expected GGI range is an input left to the operator when setting up a scenario.
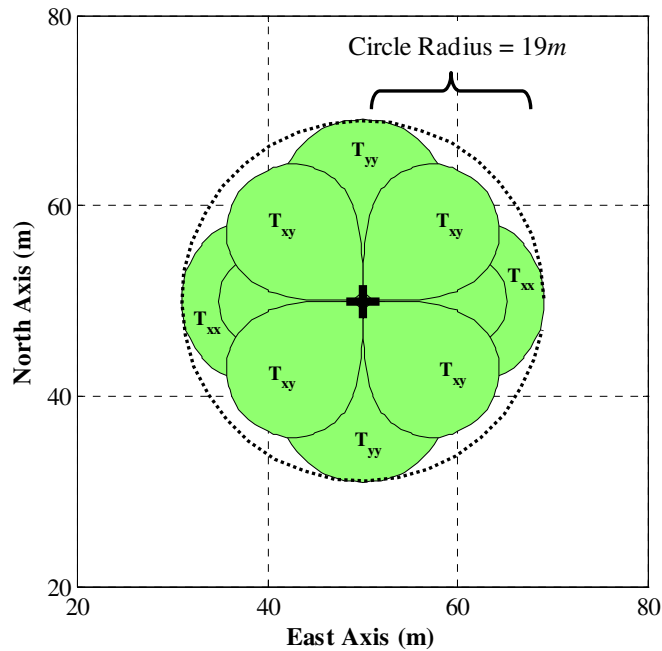
Figure 28: Area Where $|T_{ii}| \cdot (2 \cdot RMS_{Noise})^{-1}$ is Greater Than Unity

The likelihood function, movement likelihood radius, and expected GGI range concepts form the core of the detection and tracking algorithm. Consistent with the research objectives, the goal of the algorithm is to utilize GGI readings to recognize when an intrusion event has occurred. The aim of every GGI reading cycle is to decide whether to sound an alarm or not, and if so, be able to provide an accurate intruder position estimate.

The algorithm works by first applying the likelihood function to a noisy GGI reading using the baseline GGI gradient map. It then determines the most likely intruder grid position, and determines if the position estimate is within the expected GGI range. If

it is not, the algorithm discards the reading and waits for the next GGI reading. If the estimate is within the expected range, the reading is considered a possible intrusion detection and triggers creation of a movement probability radius for the possible detection location. The next GGI reading calculates a likelihood map using the likelihood function and the movement probability radius from the previous step. The algorithm repeats the previous steps indefinitely. If three consecutive readings are possible intrusion detections, the algorithm sounds a detection alarm and outputs the most likely intruder position from the most recent reading.
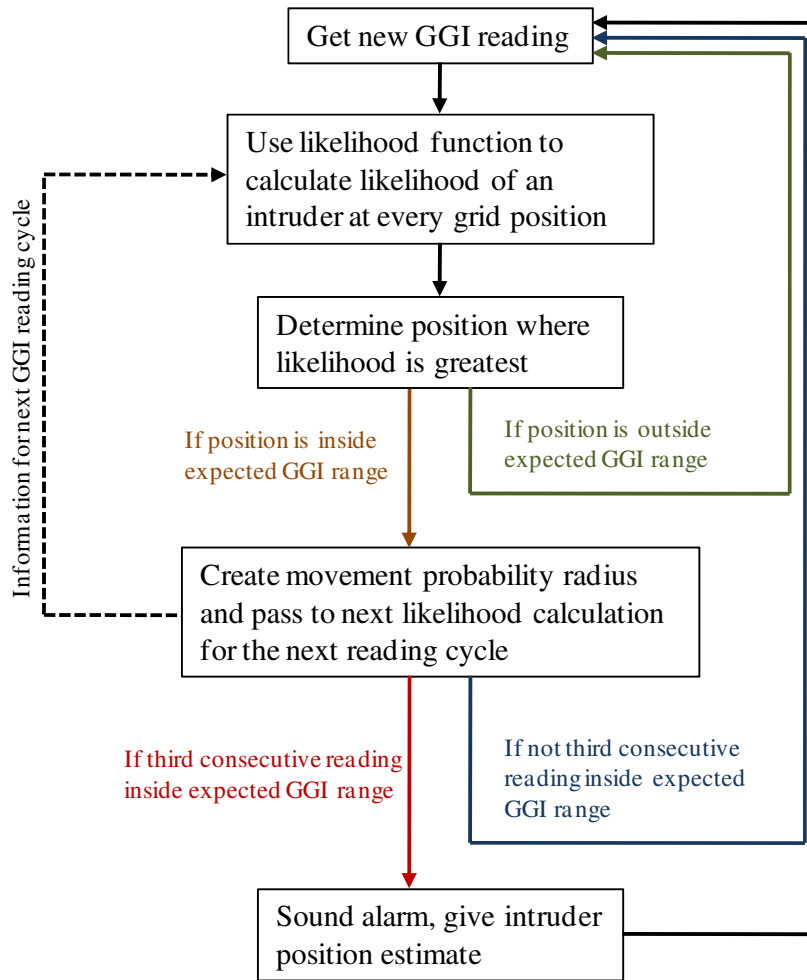
Figure 29: Detection and Tracking Algorithm Sequence

Figure 29 shows a flow chart of the detection and tracking algorithm sequence. The algorithm operates continuously and considers every GGI reading. A primary design consideration was simplicity so near real-time operation is feasible using a standard desktop computer and a gravity gradient collection rate of 1*Hz*.

The literature review in the last chapter demonstrated the most common metrics for evaluating existing PIDS is the probability of detection, P_d, and False Alarm Rate, *FAR*. While P_d is an easy value to ascertain from simulation, *FAR* is less so since it is a measure of how well a security system reacts within a real environment and is better suited to determination through field testing. P_d does not tell the full story for a PIDS designed to both detect and locate an intruder since it does not assess the tracking or pinpointing aspects of a system. Creating a new way to measure performance has the advantage of more accurately describing a system, but has the downside of preventing comparison to existing systems that do not use the new metric. Though constructed at the expense of easy comparison with existing systems, the next section explains the Detection and Tracking Parameter.

Detection and Tracking Parameter

The new Detecting and Tracking Parameter (*DTP*) metric evaluates overall system performance. It comprises three important aspects of an intruder detection system: the range at which an intruder is first detected, the average error of the position estimation, and the ability to provide a continuous tracking estimation. Equation 27 shows the relationship between the three performance values used in *DTP* calculation. In

73

the equation, range of first detection (*RFD*) is the true separation distance between the

intruder and GGI cluster (or intruder and perimeter) when the intruder alarm was first

sounded. The mean position estimate error (*MPE*) is the average difference between an

intruder position estimate and the true intruder position, while the continuous tracking

proportion (*CTP*) is the number of alarms divided by the number of measurement cycles

from the first alarm to the last alarm.

$$DTP = \left(\frac{1}{2}RFD(m) - MPE(m)\right) \cdot CTP$$

Equation 27: Detection and Tracking Parameter Calculation
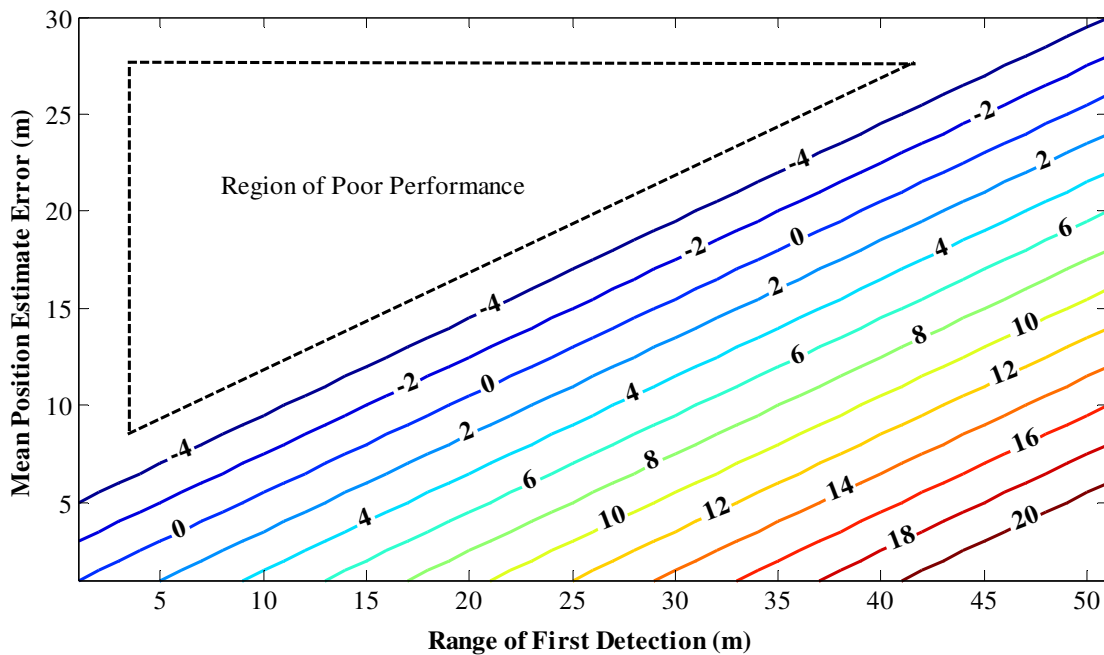
Figure 30: *DTP* for Various *RFD* and *MPE* Values, *CTP* = 1.0

Figure 30 shows how *DTP* changes for various *RFD* and *MPE* values. The one-half

weighting on the *RFD* term reflects higher value placed on position estimation fidelity

than large detection range. The *DTP* can be positive or negative, but positive values

represent a better system and result from first detecting an intruder further away, having

low position error estimates, and providing a position estimate every measurement cycle

after the alarm first sounds. Simulating multiple trials without changing the scenario

setup will yield a normal distribution of *DTP* values about some mean *DTP*, and is thus

considered using normal distribution statistics.

Figure 31 shows an example of an open area scenario and the resulting parameters

used in the *DTP* calculation. The left plot in the figure shows the intruder path, along

with red dots showing the various position estimates when an alarm was sounded. The

right plot in the figure shows a timeline of the algorithm and includes the true intruder

radius, whether the GGI reading was an alarm or not, and the position estimate error for

each alarm. The initial alarms do not accurately predict the intruder position, but as the

intruder nears the GGI the position estimate error decreases significantly. The trial shown

in Figure 31 has a *DTP* of 6.23.
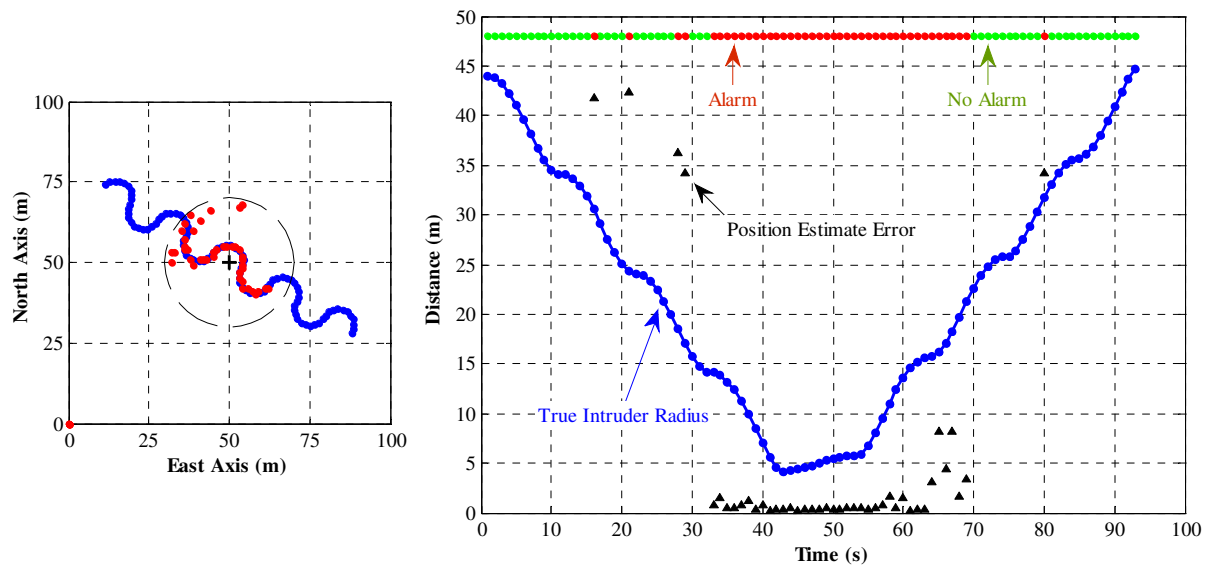
Figure 31: Example of *DTP* for an Open Area Trial

Since the true mean *DTP* for given scenario setup cannot be known without taking an infinite number of samples, which in this investigation would mean running an intrusion scenario simulation infinite times, the reported *DTP* is always be expressed within some confidence level. Triola [45] explains the confidence interval for a sample mean variable can be determined using a simple random sample and student t-test when the number of trials *n* is greater than 30 and the parameter standard deviation is not known. Put simply, a confidence interval expresses some level of confidence a true value falls within the calculated confidence interval about the sample value. As the number of trials approaches infinity, the confidence interval will shrink to zero since the sample mean would become the true mean when the number of samples is so large. Triola [45] points out a confidence level of 95% is typically used since it provides a good balance between precision (reflected in the width of the confidence interval) and reliability (expressed by the high confidence level). The true *DTP* can then be expressed to fall within the interval DTP_{mean} - E < DTP_{true} < DTP_{mean} +E where the margin of error E is given by Equation 28.

$$E = t_{\alpha/2} \frac{s}{\sqrt{n}}$$

Equation 28: Confidence Interval Estimate Using Student t-test

In the equation, $t_{\alpha/2}$ is a critical value and a function of the desired confidence interval and number of trials, *n* is the number of trials, and *s* is the sample standard deviation. Increasing the number of trials, or simulations, reduces the margin of error. In this investigation, the simulation number will be large enough so the reported *DTP* has a

margin of error of 0.3 or smaller; put another way, enough samples will be taken so the true *DTP* falls within the range $DTP_{reported} \pm 0.3$.

The *DTP* metric does not indicate when a PIDS is too sensitive and prone to generating alarms when no intruder is present. While the traditional *FAR* metric is not well suited to this project, the false alarm probability P_{FA} takes its place. Generating the false alarm probability requires running the algorithm without modeling an intruder gravity disturbance gradient. The program considers an alarm to be false if an alarm is sounded when the no-noise gravity gradient is zero. A P_{FA} value near unity is a characteristic of an overly sensitive algorithm, since GGI noise alone is enough to trigger an alarm without any intruder at all.

A third method of analyzing algorithm performance involves calculating the P_d at each grid location as a means of determining the range where the algorithm first detects an intruder. Calculating the P_d distribution map is a means of visualizing the intruder location where the alarm portion of the algorithm is first set into motion, and is useful when determining appropriate values for expected GGI range in future GGI-based PIDS designs. P_d is calculated by locating an intruder on a grid location and running the first portion of the algorithm. If the algorithm determines a possible intrusion detection occurs, the trial is considered a detection.

To condense a whole grid worth of data into a few comparable values, the mean radius to a P_d of 0.50 is calculated by taking the average distance between the GGI cluster center (or perimeter) and the grid location where $P_d = 0.50$ is first obtained on a straight line radiating away from the GGI. The mean grid distance required for the

detection probability to transition from $P_d = 0.10$ to $P_d = 0.90$ is also be determined as a means of suggesting how precise the security system is in the configuration tested. Better security systems will have a large $P_d = 0.50$ radius and a small radius difference from $P_d = 0.10$ to $P_d = 0.90$ since those characteristics suggest little uncertainty about how a system will perform given an intruder, and the true effective range is unambiguous.

Figure 32 shows an example of the detection probability grid generated for a 4-GGI cluster in an open area scenario with a medium human intruder and 100 simulations for each grid point. In the example, the $P_d = 0.50$ radius is a little less than $19m$ and the spread from $P_d = 0.10$ to $P_d = 0.90$ is about $5m$. The radial distance required to go from a near zero probability of detection to certainty of detection is thus relatively small, and the configuration's detection ability can be well characterized and repeated.
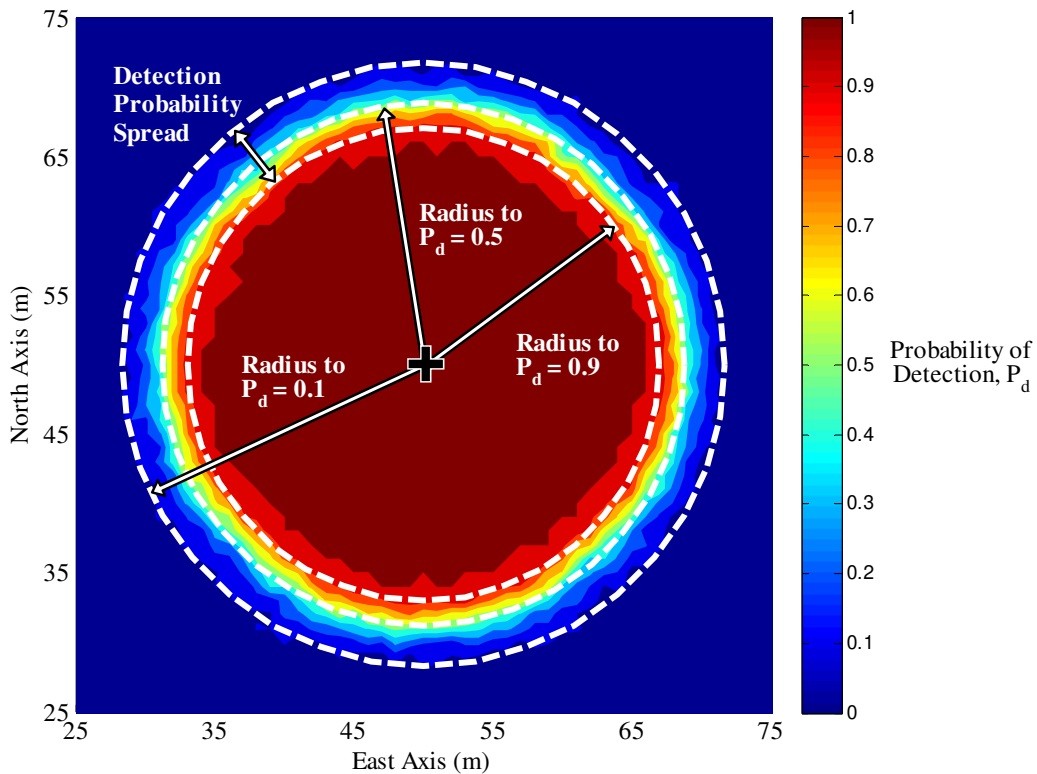
Figure 32: P_d Map for a 4-GGI Cluster

Like the *DTP*, confidence in the accuracy of at each grid location is of interest. The procedure used to calculate a level of confidence for P_d and P_{FA} is similar as it was for *DTP*, but the calculation changes slightly since those values are proportions rather than a population mean like *DTP*. According to Triola [45], Equation 29 expresses the margin of error for some proportion P where $z_{\alpha/2}$ is the critical z-value for a given two-tailed level of confidence and n is the number of samples.

$$E = z_{\alpha/2} \sqrt{\frac{P \cdot (1 - P)}{n}}$$

Equation 29: Margin of Error for a Proportion

For Equation 29 to be valid, the proportion must be a simple random sample, there must be a binomial distribution with a fixed amount of trials, and $n \cdot P \geq 5$ and $n \cdot (1 - P) \geq 5$. Based on the example detection probability grid shown in Figure 32, it is clear the last requirement is difficult to satisfy at all grid points since at a large distance from the GGI the P_d is zero and at a close distance the P_d is unity no matter how many simulations are run. Since no type of data distribution captures the essence of an unchanging value, there is little choice but to assume a binomial distribution for all grid locations. When a certain margin of error is desired but the expected proportion is not known before setting the number of trials, Trioli [45] suggests setting the $P \cdot (1 - P)$ term to 0.25 since it represents the largest possible value for the term.

The confidence level for P_{FA} is 95%. False alarm metrics are most useful when presented with some context of time, since a false $P_{FA} = 0.01$ means a very different thing if it results in one false alarm every 24 hours versus one false alarm every minute. All P_{FA}

values reported in this investigation are the result of 1 simulated hour of operation, or 3600 algorithm cycles at $1Hz$. Therefore, a reported P_{FA} value of 0.0014 means the tested configuration had five false alarms during the simulation and has a false alarm rate of five per hour. Since $n = 3600$, the PFA margin of error is $P_{FA} = P_{FA, reported} \pm 0.016$.

As with DTP and P_{FA}, the P_d maps use a 95% confidence level. Due to the large number of grid points, increasing n rapidly increases the required number of computations. For this reason $n = 100$ for all P_d simulations. That value of n yields a maximum margin of error of about 0.1 when assuming 0.25 for the $P_d \cdot (1 - P_d)$ term. While a 10% error margin is somewhat large for a simulation where the number of trials is controllable, a primary goal of this simulation is to gain some understanding of how the detection probability spread changes based on user inputs—an objective is still achievable even with a large margin of error. Furthermore, a P_d map is the not a means of evaluating the algorithm, and is only used in some instances to help explain the results.

In summary, every set of simulations collects DTP and P_{FA}, while P_d map generation is reserved for helping to understand unexpected results. With a method in place for evaluating the algorithm using the results of computer simulations, the final step before collecting data is to determine which user variables to test.

Test Plan

With 15 variables, a full factorial analysis requires simulations of several million variable combinations and is well outside the range of feasibility for this project. The goal of all simulations is to demonstrate the effects of changing user inputs. Table 9 overviews all user inputs, along with the baseline values for each variable. The baseline

values are those that produced good detection and tracking results during computer program development, but were not optimized. Part of the test plan includes simple sensitivity analysis for each variable using the baseline condition. The overview of user inputs shows the available inputs partitioned into three different parts of the system. Choosing only a few variable combinations for full factorial analysis limits the number of simulations while still focusing on the three distinct parts of the PIDS system outlined in Table 9. Table 10 shows the variables used for the GGI cluster geometry, intruder size, motion and speed, and GGI noise full factorial analysis. The effect of rain and the type of intruder were investigated separately.

Table 9: Overview of User Inputs

Part of System	Variable	Options	Baseline
Intruder Characteristics	Intruder Speed	Slow, Medium, Fast	Walking
	Path Type	Linear, Snaking	Linear
	Motion Position	Upright, Prone	Upright
	Human Size	Large, Medium, Small	Medium
	Type of Intruder	Human, Rabbit, Large Vehicle	Human
	Precipitation State	None, Light Rain, Heavy Rain	None
GGI Characteristics	Number of GGIs in Cluster	1, 2, 4, 6	4
	Perimeter Spacing	Any Positive Number	$30m$
	GGI Cluster Spacing	Any Positive Number	$2m$
	GGI Cluster Height	Any Positive Number	$1m$
	GGI Noise Spectral Density	Any Positive Number	$0.001 Eö/Hz^{1/2}$
Algorithm	Expected GGI Range	Any Positive Number	$18m$
	GGI Logic Type Used	Equal Weighting, Averaging	Equal Weighting

Table 10: Areas of Full Factorial Analysis

Area	Variables	Values
GGI Cluster Geometry	Number of GGIs in a Cluster	1, 2, 4, 6
	GGI Cluster Spacing (*m*)	1, 2 ,4 ,6
	GGI Cluster Height (*m*)	1, 3, 5
Intruder Size, Motion, and Speed	Intruder Speed	Slow, Medium, Fast
	Path Type	Linear, Snaking
	Motion Position	Upright, Prone
	Intruder Size	Large, Medium, Small
GGI Noise and Algorithm	GGI Noise Spectral Density ($E\ddot{o}/Hz^{1/2}$)	0.01, 0.001, 0.0005
	Expected GGI Range (*m*)	10, 15, 18, 20, 25, 30
	GGI Logic Type Used	Equal Weighting, Averaging

Summary of Assumptions

The usefulness of the results hinge on the handful of assumptions made to simplify the simulation. Perhaps most crucial is the idea a gravity gradient for a natural, non-intruded grid area can be accurately mapped and does not change significantly between GGI reading time steps. Much like taring a scale before use, every gravity gradient reading must first subtract the total natural gravity gradient at the GGI location before determining the disturbance gradient caused by an intruder. If the value of the natural gradient is not steady it would be impossible to know what portion of a gradient reading came from the changing background gradient and which portion came from an approaching intruder, making accurate detection and tracking impossible. There has been little previous work characterizing background gravity gradient shifts over time, but the

sorts of events with the potential to cause a noticeable shift are not difficult to imagine. Changes in underground aquifer levels, snow, soil erosion, and movement of very large objects could all influence the natural total gravity gradient.

In a similar vein, it is important to create an accurate disturbance map for the expected intruder type. This means the initial installation of a GGI PIDS requires significant calibration, and is probably easiest completed by recording the disturbance gradient as an individual stands at every grid location. This is not a trivial task, but the usefulness of the likelihood function decreases significantly if the baseline gradient map is inaccurate. In reality, a baseline gradient map would have some inaccuracies, but for this initial technology investigation the baseline gradient disturbance map is assumed perfect for the expected intruder type. While it is important the prism model of a human offer a good representation of the true gradient for a human, its use in this paper is limited as a method for determining approximate gradient magnitudes and spatial patterns. A functioning GGI PIDS would use a real human, rather than a prism model, to create the baseline disturbance gradient map, and thus the baseline would match the true expected intruder disturbance signal.

In conclusion, this chapter demonstrated how a MATLAB computer program utilizes user inputs to produce a simulated gravity gradient. It discusses how an algorithm uses the noisy, five-tensor simulated gradient reading to predict an intruder location with a size and density assumption. The methodology discussion ends with a discussion of performance metrics and the associated statistical confidence measures needed to evaluate simulation results. The next chapter describes the results of the simulations.

IV. Analysis and Results

The simulations tested security system performance in five general areas: GGI cluster geometry, GGI noise and algorithm characteristics, human physiology and movement, ground absorption of precipitation, and other-than-human intruders. It is impossible to analyze and present every nuance in data covering many thousands of simulations and multiple variable combinations. What follows is an observation of general trends, rules of thumb, and useful information for the future design and of a GGI-based PIDS. This results discussion most often uses the detection and tracking parameter (*DTP*) as a measure of overall system performance, but other important aspects like false alarm probability (P_{FA}) are included when relevant. Keep in mind all *DTP* values are reported at a 95% confidence level to $DTP \pm 0.3$, and P_{FA} is reported at a 95% confidence to $P_{FA} \pm 0.02$.

Table 11 shows the baseline performance of the modeled system for the open air and perimeter intrusion scenarios. There was little overall difference between the two scenarios, and the small differences in mean range at first detection (*RFD*) and mean position estimate error (*MPE*) were likely the result of where the simulation positions the intruder. In the open area scenario, the intruder moved directly toward the GGI cluster, so the *RFD* should have been close to the maximum detection range for a GGI cluster. In the perimeter scenario though, the *RFD* was the distance from the intruder to the perimeter— not from the intruder to a GGI cluster—so the intruder could get closer to the perimeter before entering the maximum detection range of one of the GGI clusters on the perimeter. In both baseline scenarios the mean continuous tracking proportion (*CTP*) was unity,

84

meaning the algorithm provided an alarm and position estimate every gradient collection cycle from first detection through the time the intruder left detection range. There were no false alarms for either scenario during the simulated hour-long false alarm test.

Table 11: Baseline Performance Values

Scenario Type	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Mean Continuous Tracking Proportion	Detection and Tracking Parameter	False Alarm Probability
Open Area	15.90	1.66	1.00	6.26	0.000
Perimeter	13.87	1.15	1.00	5.76	0.000

Intruder Speed: Walk, Path Type: Linear, Motion Position: Upright, GGIs Per Cluster: 4, Cluster Height: 1m, Cluster Spacing: 2m, NSD: 0.001Eö/Hz$^{1/2}$, Perimeter Spacing: 30m, Expected GGI Range: 18m, Logic Type: Equal Weighting

GGI Geometry

A full factorial analysis considered changes to the number of GGIs in a cluster, cluster height, and cluster spacing in an open area scenario while holding all other variables at the baseline value. Figure 33 shows a bar plot with calculated *DTP* for every combination of the available user input variables for GGI geometry. The simulations tested three cluster heights—1*m*, 3*m*, and 5*m*—and four spacing values—1*m*, 2*m*, 4*m*, and 6*m*—for clusters with one, two, four and six GGIs.

For all number of GGIs in a cluster, system performance increased as cluster height increased. The result makes intuitive sense based on the nature of the independent tensors. The strongest gravity gradient tensors are those in the same plane of movement as the source object because the gravitational vector points from the detection location to

the source object. The earlier table in the methodology discussion showing relative component strengths for this application makes sense with these results in mind—if measured near the plane of human movement (the ground), the T_{xx}, T_{yy}, and T_{xy} tensors are the strongest. Since there is little variation in the gravitational vector in the vertical direction because an intruder and GGI are bound to ground level, the T_{xz} and T_{yz} tensors are weakest. Raising the GGI off the ground increases the vertical component of the gravitational vector for an approaching intruder, which means the T_{xz} and T_{yz} tensors increase magnitude as well. The algorithm detects and locates an intruder best when all five tensors have information, e.g. it performs best when all five tensors have a relative strength of 0.2, rather than when three tensors each have a relative strength of 0.33 and two tensors have a relative strength of zero.
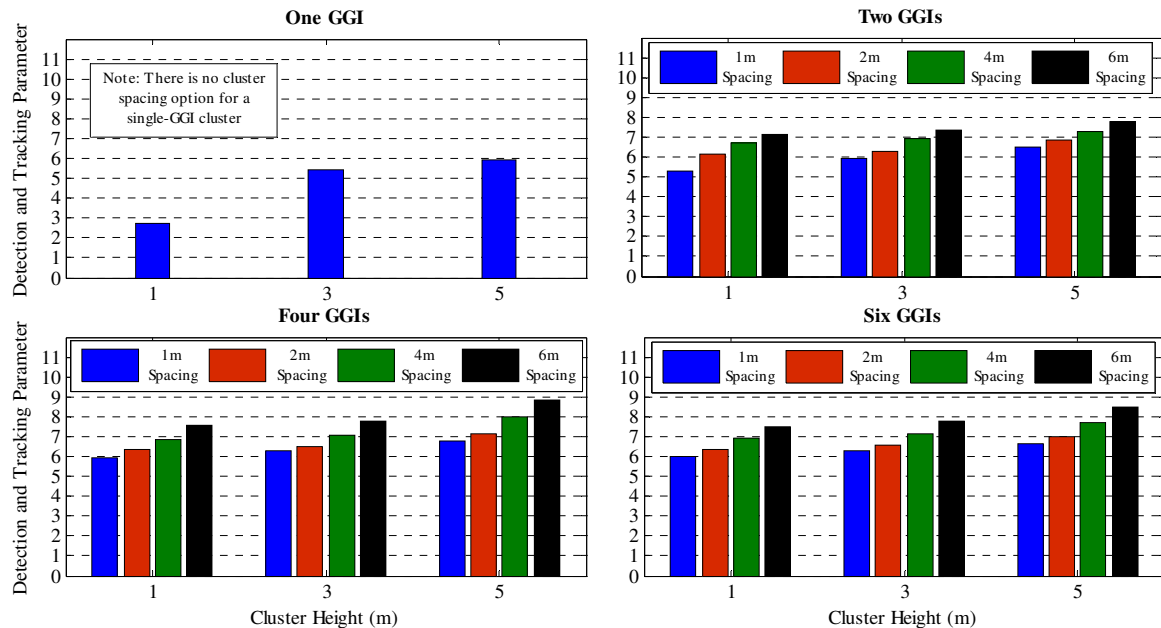
Figure 33: Results of Cluster Geometry Full Factorial Analysis

There are two disadvantages to raising a GGI off the ground to balance gravity gradient tensor magnitudes. First, it increases the separation distance between an approaching intruder and the GGI, reducing the magnitude of all tensors, whether balanced or not. Second, it provides a logistical challenge to mount and service a GGI many meters off the ground. Figure 33 shows changing the cluster height from $1m$ to $5m$ increased DTP, but by a maximum of about 1.0. An extra two meters of detection range or one meter lower position estimate error is probably not enough reason to raise a GGI high off the ground for the disadvantages just discussed. Nonetheless, cluster height does affect how a GGI detects an intruder and is an option for future investigation.

At first glance, increasing cluster spacing also seems to increase algorithm performance by raising DTP. Figure 33 shows DTP increased by about 0.25 for every meter of cluster spacing for each clusters with two, four and six GGIs. Since mean range of first detection is defined as the distance from the cluster center to the intruder, increasing cluster spacing places a GGI nearer to an approaching intruder. The increased DTP is a result of positioning individual GGIs further away from the cluster center rather than the detection footprint of any single GGI increasing. Still, concluding a larger array of GGIs increases the detection characteristics of the PIDS is important to designers, even if it is the obvious and expected result.

Increasing the number of GGIs in a cluster increased DTP for all geometries tested, though the increase was negligible beyond a 4-GGI cluster. The most marked improvement occurred between the 1- and 2-GGI clusters. When the cluster was at ground level, the DTP nearly doubled with the addition of a single GGI to the cluster.

87

Adding more GGIs increased overall performance by reducing the mean position estimate error to about 1*m*, but the *DTP* increase was small. Though a 4-GGI cluster was the baseline for the rest of the simulations, Figure 34 shows a 2-GGI cluster offered the best balance between a low number of instruments and good system performance.

Scenario Type: Open Area, Intruder Speed: Walk, Path Type: Linear, Motion Position: Upright, Cluster Height: 1m, Cluster Spacing: 2m, NSD: 0.001Eö/Hz$^{1/2}$, Expected GGI Range: 18m, Logic Type: Equal Weighting
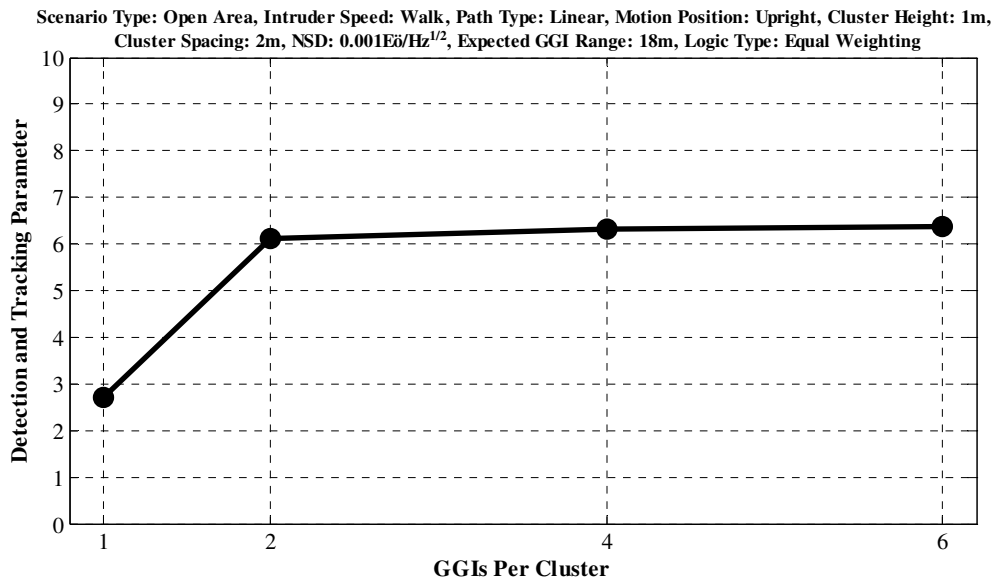
Figure 34: Effect of GGIs Per Cluster on Baseline Performance

DTP is not the only relevant performance measure. Even an exceptional system able to first detect an intruder at long ranges with little position estimation error is useless if plagued by many false alarms. Figure 35 shows how cluster geometry affected the false alarm rate. There were no false alarms observed for any configuration of the 4- or 6-GGI clusters. The 1- and 2-GGI clusters, however, were prone to false alarms at many cluster height and spacing combinations. The single GGI cluster was most unreliable, with a maximum false alarm rate of nine every hour when the cluster height was 3*m*. Other

cluster heights resulted in fewer false alarms, but even the 2-GGI clusters had at least one

false alarm per hour at every cluster spacing value simulated. Large cluster spacing

values most consistently provoked false alarms, though smaller spacing values were not

entirely immune.

Scenario Type: Open Area, Intruder Speed: Walk, Path Type: Linear, Motion Position: Upright, NSD: 0.001Eö/Hz$^{1/2}$, Expected GGI Range: 18m, Logic Type: Equal Weighting

Note: No 4- or 6-GGI clusters registered a false alarm for the configurations tested

Legend:
- GGIs in Cluster: 1, Cluster Spacing: -
- GGIs in Cluster: 2, Cluster Spacing: 1
- GGIs in Cluster: 2, Cluster Spacing: 2
- GGIs in Cluster: 2, Cluster Spacing: 4
- GGIs in Cluster: 2, Cluster Spacing: 6

Figure 35: Effect of Cluster Geometry on False Alarms

Depending on the particular security system application, a false alarm rate of one

per hour could be reasonable. It is difficult to imagine a situation where nine alarms every

hour is acceptable, but configurations with just a few alarms every hour is not reason to

discount those configurations automatically. Overall, the simulations showed that all

aspects of cluster geometry affected the detection and tracking parameter. The best

tracking and locating performance occurred with a 6-GGI cluster spaced at 6*m* and a

height of 5*m* above ground level. A 2-GGI cluster detected and tracked an intruder nearly

as well as the 4- or 6-cluster GGI, but had an average of more than one false alarm per hour. All geometry configurations were remarkably stable across changes to the other variables, differing *DTP* by less than one unit. Geometric stability has both positive and negative aspects. It suggests GGI installation in a future system to be a simple undertaking, since there is little performance advantage to place GGIs in a precise, specific pattern. A placement tolerance of several meters in any direction promises to allow quick GGI deployment and setup by avoiding complex tasks like a physical site survey. On the other hand, low performance sensitivity to cluster geometry means GGI array geometry cannot be fine-tuned to greatly increase detection and tracking performance. The burden to know GGI position accurately, however, is not relaxed by the low position precision requirement since intruder position estimates are made relative to GGI location.

Human Physiology and Movement

Cluster geometry was an important consideration for a GGI-based PIDS because it offered insight into some factors important for eventual system optimization. This simulation section showed how the algorithm reacted to an intruder slightly different from the one used to make the baseline gravity gradient map, and the results detail how much tolerance the algorithm had for an off-baseline human intruder.

Figure 36 shows *DTP* for different intruder sizes, path types, and movement speed. Movement speeds have the same colored line; a solid line indicates a linear path, while a dashed line indicates a snaking path. All combinations are plotted against the

three simulated human intruder sizes. *DTP* for all intruder speeds and path types was largest for large intruders, the result of a larger intruder having more mass and creating a stronger gravity gradient at a given range than the other intruder sizes.
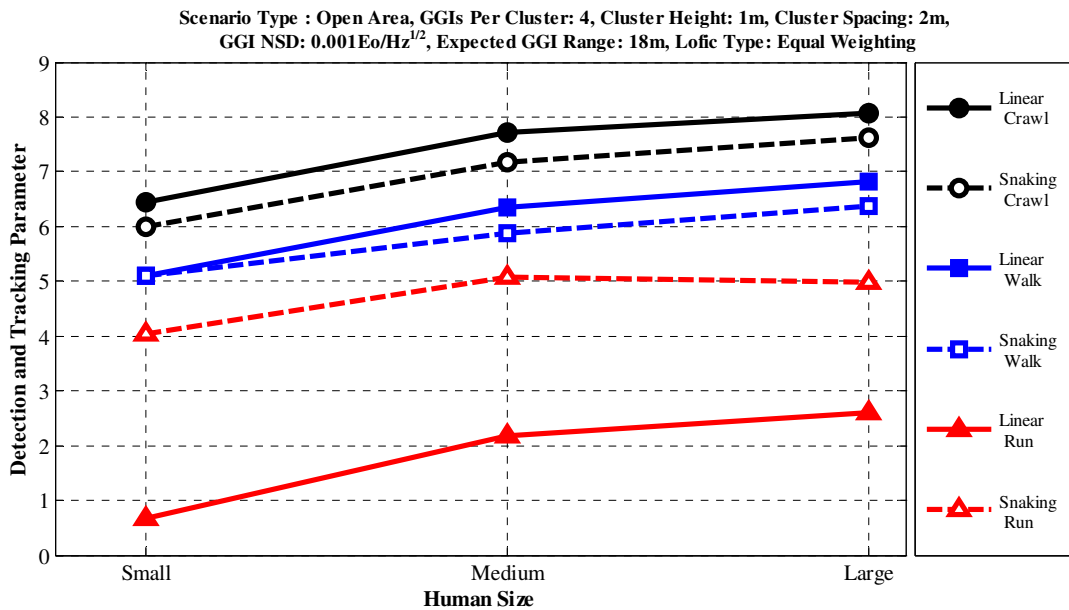
Figure 36: Results of Intruder Size, Motion, and Speed Full Factorial Analysis

Further investigation into Table 17 in Appendix B shows *DTP* increase for intruder size is the result of a greater *RFD* alone, with *MPE* steady between 1*m* and 3*m* for all of the human physiology and movement full factorial simulations. The mean range of first detection was as large of 21*m* for the large, slow moving human, while a small, slow moving intruder reduced the *RFD* to about 17*m*. The decrease in detection range was similar for all small-large human combinations, showing an approximate detection range increase of 1*m* for every 11*kg* increase in intruder mass. The larger build of a more massive intruder did contribute to increasing the gravity disturbance gradient, but the

dimensional differences between the large and small intruder were slight enough to conclude human mass—not size—had the greater effect on gravity gradient strength and overall algorithm performance.

Path type contributed little to *DTP* variation at crawling and walking speeds, though for all human sizes at those movement speeds the snaking path *DTP* was slightly lower than the linear path *DTP*. Again, Table 17 in Appendix B reveals the approximate 0.5 unit *DTP* decrease for path type was due to a larger position estimate error, rather than a decrease in first detection range. In theory, centering the movement likelihood radius on the last known intruder position eliminated the effect of path choice on algorithm performance because the likelihood of movement was the same in every direction. Though the effect of path type was small at slow and medium speeds, further investigation should study whether more complex path types exacerbate the *DTP* decrease seen for the snaking path. Running speed significantly decreased *DTP* relative to the two slower speeds, but the result makes sense in light of the algorithm's requirement for three consecutive possible intruder detections before sounding an alarm. When travelling 5*m/s* directly at the GGI cluster, the intruder could travel 15*m* within the GGI detection range during three one-second detection cycles before the first alarm. Since the snaking path did not move the intruder straight at the GGI cluster, the *DTP* decrease from slower movement speeds was more reasonable. Smaller *RFD* values for the fast movement speed did not indicate algorithm misperformance, but rather reflected the smaller amount of gradient information available when the intruder quickly crossed through the detection range.

Table 12 shows motion position had almost no effect on the algorithm's detection and tracking ability. The table presents the mean *DTP* difference between the upright and prone position for both movement speed and intruder size. The prone *DTP* was lower than the upright *DTP* for every variable combination, but the difference was nearly always about one *DTP* confidence interval, or close to statistical insignificance. A running intruder had the largest upright-prone *DTP* difference, but at under 0.6 units the mean difference is nearly within the *DTP* uncertainty range. Additionally, prone human movement at 5*m/s* is unlikely.

Table 12: Effect of Motion Position on *DTP* for Various Intruder Sizes and Speeds

		Crawl	Walk	Run	Mean DTP Difference Between Upright and Prone
Small Intruder	Upright	6.45	5.11	0.68	
	Prone	6.21	5.05	0.02	**0.32**
Medium Intruder	Upright	7.72	6.36	2.17	
	Prone	7.41	6.25	1.50	**0.36**
Large Intruder	Upright	8.08	6.81	2.60	
	Prone	7.71	6.71	2.21	**0.29**
Mean DTP Difference Between Upright and Prone		**0.31**	**0.09**	**0.57**	

Scenario Type: Perimeter, Path Type: Linear, GGIs Per Cluster: 4; Cluster Height: 1m, Cluster Spacing: 2m, NSD: 0.001Eö/Hz$^{1/2}$, Perimeter Spacing: 30m, Expected GGI Range: 18m, Logic Type: Equal Weighting

Of all intruder movement and physiology variables, intruder mass and movement speed had the greatest effect on the GGI-based PIDS performance. The algorithm performed strongly against all tested variations of a human intruder, though fast movement speeds decreased the range of first detection to about 5*m*. Mean position

estimation error ranged from 1*m* to 3*m*, changing little for any of the full factorial combinations. Fundamentally, these results confirm the uniqueness of the technology in a security application. The intruder characteristics having little effect on detection and tracking ability, like motion position and physical size, are of primary interest to a radar or infrared detector. Likewise, the most important characteristic for a GGI—intruder mass—has no theoretical effect on the performance of a seismic sensor or video camera.

GGI Noise and Algorithm Settings

The previous two full factorial analyses tested GGI geometry and an off-baseline human intruder. The final factorial analysis used the remaining algorithm variables to characterize the effect of GGI noise, logic type, and the expected GGI range on the algorithm.
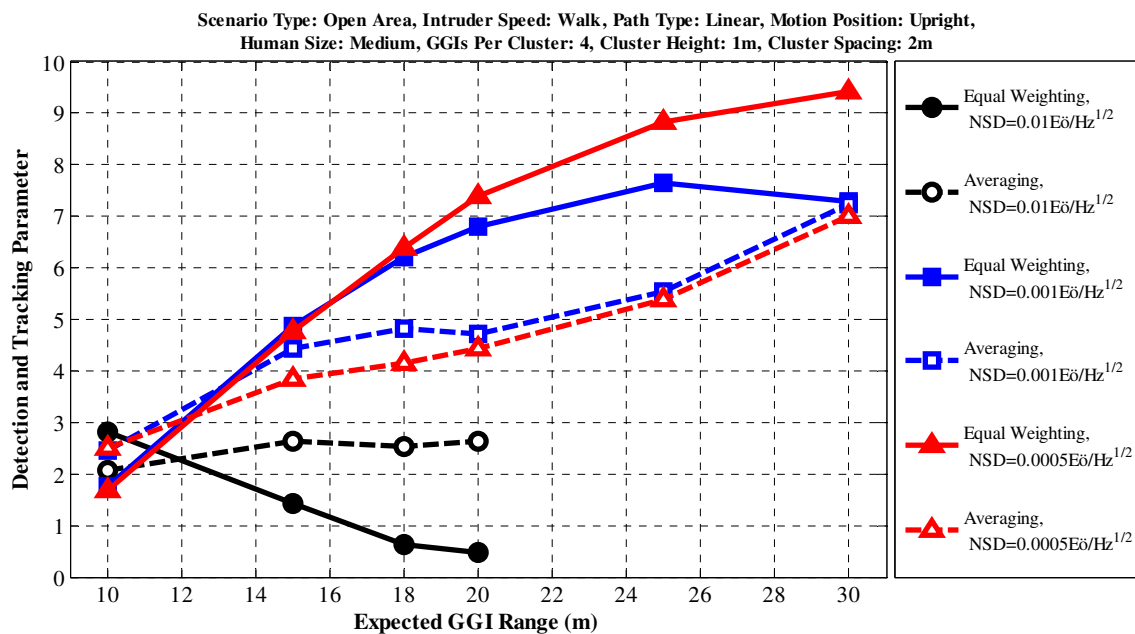
Figure 37: Results of GGI Noise and Logic Type Full Factorial Analysis

Figure 37 shows the simulated *DTP* across six expected GGI ranges for the three GGI noise values and two algorithm logic types. The same line color is common to the noise spectral density (NSD) value, while a solid line represents the equal weighting logic type and dashed line represents the averaging logic type. At an expected GGI range of 10*m*, all noise levels had about the same *DTP*. The *DTP* value was low relative to the baseline value of about 6.0 because the small expected GGI range reduced *RFD* to below 10*m*. As expected GGI range increased, *DTP* for the noisiest simulation decreased to below 1.0 and stayed low through an expected GGI range of 20*m*. The other two noise levels tested had nearly the same *DTP* through an expected GGI range of 18*m* before the more noisy of the two—the 0.001*Eö/Hz$^{1/2}$* GGI—trailed off, eventually maintaining a *DTP* of about 7.0 up to an expected GGI range of 30*m*. The GGI cluster with a NSD value a half magnitude better than the baseline case increased *DTP* across the entire simulated expected GGI range, though the amount of increase slowed at a 30*m* expected range. It had a maximum *DTP* of about 9.5. As expected, the general trend was for a less noisy GGI to have a better detection range and higher *DTP*.

Averaging each gravity gradient tensor within a cluster prior to applying the likelihood function was detrimental to the *DTP* at all expected GGI ranges for the two least noisy conditions. In most cases, the *RFD* was nearly the same as the equal weighting logic. However, averaging typically caused an *MPE* several meters larger than it was for equal weighting, which acted to drive *DTP* down. The noisiest condition had a higher DTP using averaging logic, but both were low and had *MPE* values near 20*m*.

95

Since the algorithm works by ignoring the positions of maximum likelihood when they fall outside the expected GGI range, every GGI noise level has an optimal expected range value. When optimized, the expected GGI range is the largest possible before it extends so far that position estimates become unreliable because the disturbance gradient signal is not stronger than GGI noise. The detection and tracking parameter alone does not contain enough information to determine the best expected GGI range value for a given noise level, because the effect of large position estimate errors on *DTP* is outweighed by large first detection ranges. Figure 38 shows how the probability for false alarm changed for each NSD-expected GGI range combination.

Scenario Type: Open Area, Intruder Speed: Walk, Path Type: Linear, Motion Position: Upright, Human Size: Medium, GGIs Per Cluster: 4, Cluster Height: 1m, Cluster Spacing: 2m
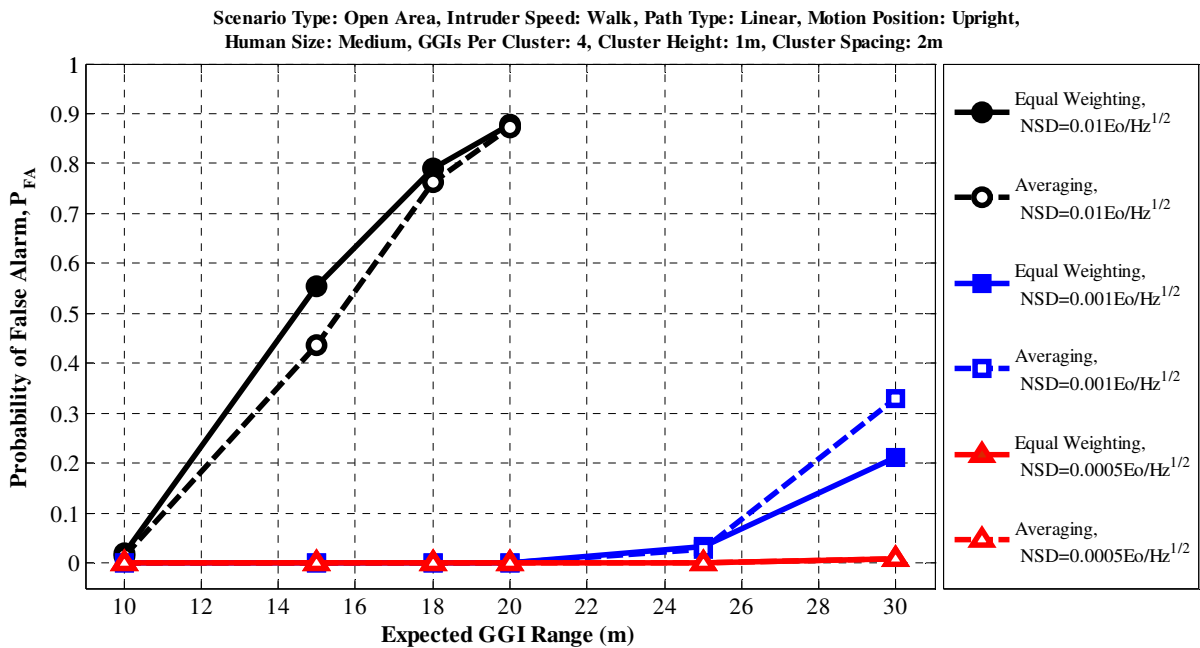
Figure 38: Effect of NSD and Expected GGI Range on P_{FA}

It shows P_{FA} is a better measure than *DTP* of where the optimal GGI expected range is for a given NSD. Since a P_{FA} value as small as 0.001 is equivalent to nearly four false alarms per hour, the optimal expected GGI range for a given NSD level is the maximum before $P_{FA} > 0$. The resolution for the simulated data is low since only six expected GGI values were tested, but the plot shows optimal expected GGI range is less than 10*m* for NSD = 0.01*Eö/Hz*$^{1/2}$, between 20*m* and 25*m* for NSD = 0.001*Eö/Hz*$^{1/2}$, and between 25*m* and 30*m* for NSD = 0.0005*Eö/Hz*$^{1/2}$. Comparing Figure 37 and Figure 38 reveals *DTP* does not begin to decrease until several meters after the optimal expected GGI range, so it is not a good metric to pinpoint the most favorable expected GGI range value for a given GGI noise level. Since averaging logic did not consistently reduce the false alarm probability and provided no better *DTP* than equal weighting logic, there was no apparent advantage to using averaging over equal weighting.

Comparing the P_d maps for a given GGI noise level helped to solidify why selecting the best expected GGI range is important. Figure 39 shows how increasing the expected GGI range resulted in an expansion of the detection area, but when increased too large the spread between high and low probability contours became very large. The methodology discussion explained a larger spread between the radius of high detection probability and low detection probability was undesirable because it indicated inconsistent detection ability at certain ranges. Noting the probability spread is very large for the for the expected GGI radius of 25*m* in Figure 39, while the false alarm probability for the same NSD and expected detection range is well above zero, further confirms the thought. Large spreads drove up the false alarm probability. On the other hand, choosing

too small an expected GGI range for given GGI noise level constrained the algorithm so the GGI underperformed. The 10m expected GGI range plot in Figure 39 shows the underperformance: while the probability spread is very narrow (< 1m), the effective detection range was about 10m. Figure 38 shows the best expected GGI range was at least 20m for a NSD of $0.001 E\ddot{o}/Hz^{1/2}$ since there was a zero probability of false alarms up to that expected GGI range value. The results show a GGI array was tunable using the expected GGI range value. Lower values reduced the probability of false alarms and rigorously defined the effective detection footprint, while higher values increased the effective GGI cluster range at the expense of a higher probability of false alarms.
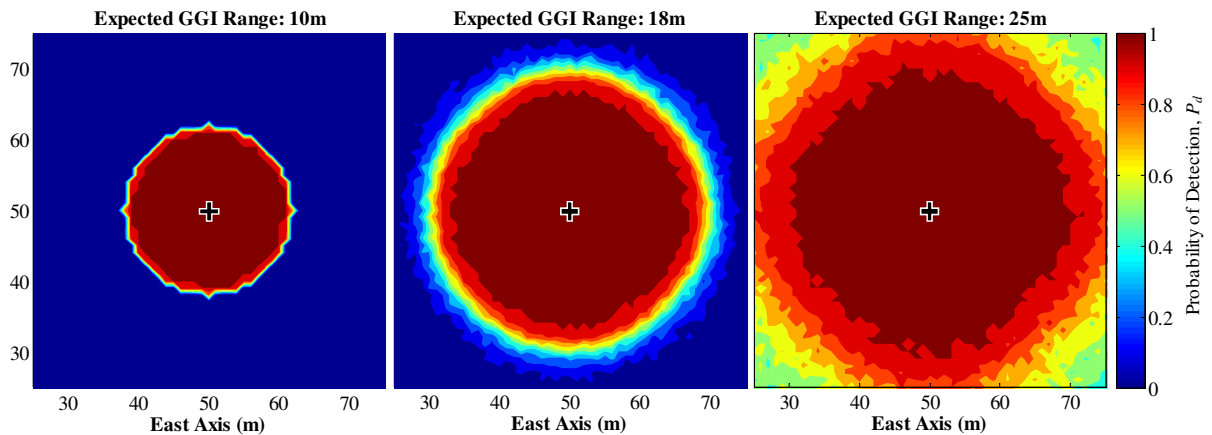
Figure 39: P_d Map Comparison for NSD = $0.001 E\ddot{o}/Hz^{1/2}$

Precipitation Infiltration

The final two simulation areas evaluated the algorithm's response to far off-baseline conditions. Whereas the three full-factorial analyses evaluated the effect of small changes to variable combinations known to successfully detect and track an intruder,

98

precipitation infiltration into the ground and other than human intruders represent far off-baseline conditions.
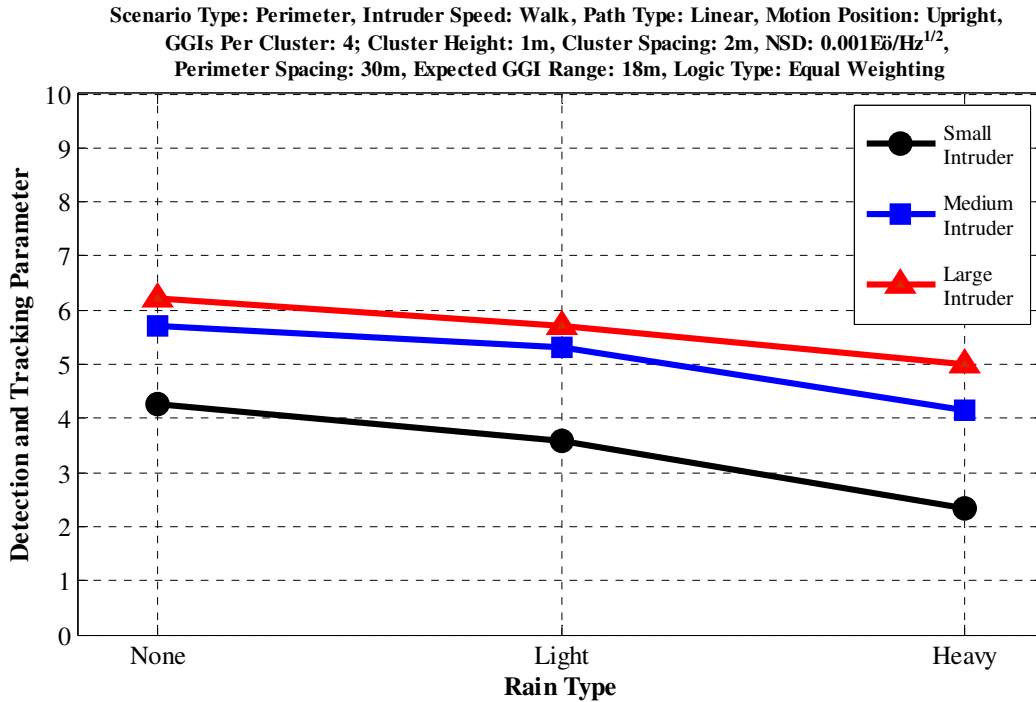
Figure 40: Effect of Precipitation on *DTP* for Various Intruder Sizes

Figure 40 shows how *DTP* changed with varied rates of precipitation and intruder size. *DTP* decreased as the amount of rain increased for all intruder sizes. The algorithm's ability to track a small human suffered the most, as *DTP* dropped about 2.0 units. *DTP* for the medium and large intruder dropped about 1.5. Table 19 in Appendix B shows full precipitation results, and identifies the culprit of rain-induced *DTP* decrease to be the mean range at first detection. While the position estimate error remained constant at about 1.5*m* for all rain rate-human size combinations, *RFD* dropped by about 4*m* from the no rain to the heavy rain scenario. Rain infiltration's effect on the GGI reading is particularly interesting because the addition of mass over a large area beneath the GGI

cluster did not increase the position estimate error. Though rain seeping into the ground

reduced the effective detection range of the system, it was not enough to prevent effective

algorithm performance. Additionally, Table 19 reports the false alarm probability for all

rain types was zero, a positive indication of a GGI-based PIDS ability to be useful in

adverse weather.

The simulation only tested rain accumulation over a single hour, so the results say

nothing about algorithm detection capability at the end of a long, steady rainstorm where

several hours worth of precipitation soaks into the ground. Figure 40 is still relevant,

however, because there are a few ways to roughly estimate and account for the effect of

long periods of precipitation on the gravity disturbance gradient.

The simplest method uses a common rain gauge. A gravity disturbance gradient

could be calculated using specific soil infiltration information for the installation site

along with the amount of rain in the rain gauge. Subtracting the estimated rain

disturbance gradient from every gradient reading would yield a gravity gradient reading

free from the influence of rain infiltration. The true rain disturbance gradient would be

only an estimate using such an imprecise method, but Figure 40 shows the algorithm had

some tolerance for unaccounted rainfall. In-ground sensors could further improve the rain

gauge estimate by calculating the specific amount of rain infiltration at different soil

depths to increase the accuracy of the rain prism approximation.

A complete investigation into rainfall approximation requires detailed soil models

able to predict accurately how rain moves through the ground, and a method for

predicting the influence of a complex underground rain density structure on a surface

GGI reading. Both tasks are outside the scope of this project, and probably easiest left to experimentation on an installed GGI cluster. The rain results show how any amount of rainfall decreased the effectiveness of gravity gradiometry as a security system technology by decreasing the first detection range. Nevertheless, rudimentary techniques likely prevent rendering useless a GGI cluster with a NSD as low as $0.001 E\ddot{o}/Hz^{1/2}$ in a prolonged rainstorm.

Non-Human Intruders

The final area of evaluation was algorithm response to a small animal and large vehicle. This test was important because it offered indicators of how well the likelihood function technique at the algorithm's core operated when the baseline gradient map for the expected human intruder was very different from the true gradients caused by a non-human. *DTP* is a measure of how well the system performed against the expected intruder type, not against other types of intruders. Though the *DTP* value meant little for this application, some of the variables contained within were important, like the range at first detection and the position estimate error. The proportion of simulations triggering an alarm was calculated and reported in Table 20 in the Appendix B along with rest of the results for this series of simulations. The proportion of simulations triggering an alarm is the likelihood the intruder type caused the algorithm to sound at least one alarm from the start to the end of the path. Less massive intruders should have a low proportion of sounded alarms since their gravity disturbance gradient is small relative to the baseline

values, while intruders larger than the baseline should cause an alarm every simulation

because their gravity disturbance gradient is larger than the expected value.

Scenario Type: Perimeter, Intruder Speed: Walk, Path Type: Linear, Motion Position: Upright, GGIs Per Cluster: 4,
Cluster Height: 1m, Cluster Spacing: 2m, NSD: 0.001Eö/Hz$^{1/2}$, Perimeter Spacing: 30m, Logic Type: Equal Weighting
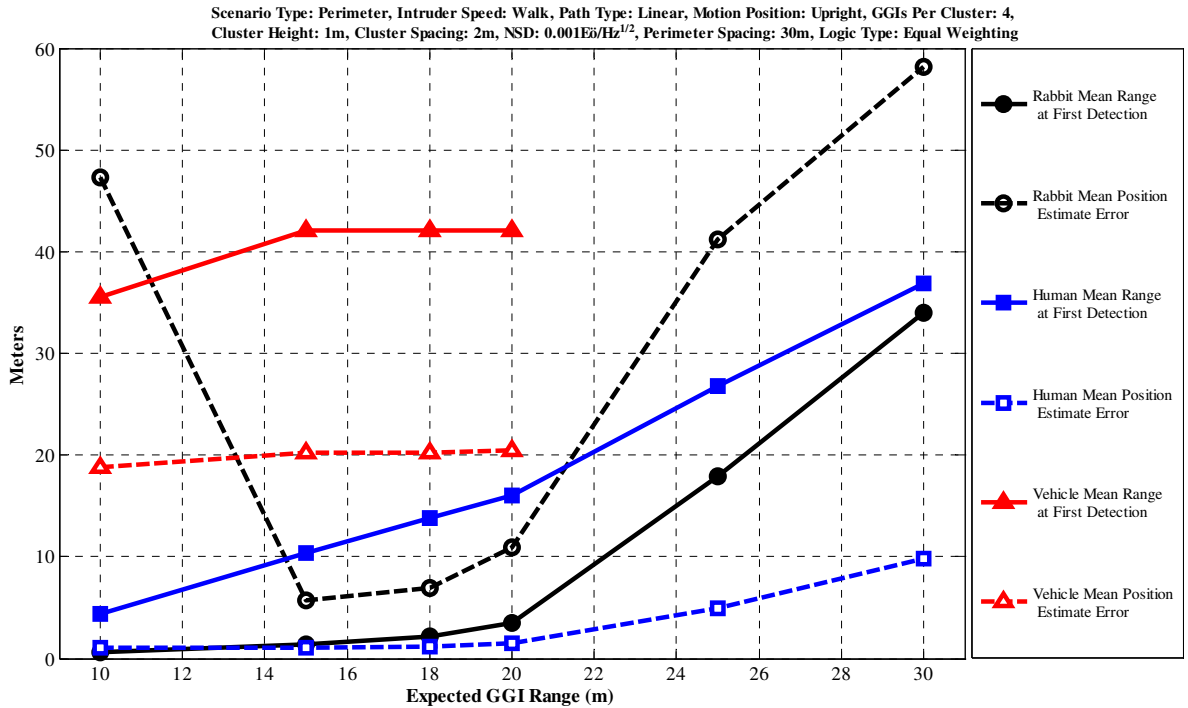
Figure 41: Effect of Intruder Type on Detection Characteristics

Figure 41 shows the *RFD* and *MPE* for each of the three intruder types across a

changing expected GGI range. Discussed previously, the first detection range for a

human increased as the expected GGI range increased, though the position error estimate

also rapidly increased after the expected GGI range exceeded about 20*m*. A high first

detection range and low position estimate error implies good algorithm performance in

general. The figure shows the algorithm performed poorly for the two non-human

intruder types at all expected GGI ranges tested. The algorithm first detected the rabbit at

a range of less than 5*m* when the expected GGI range was less than 20*m*, though the

position estimate error was well over 10*m* in some cases. The large vehicle caused poor algorithm performance, with high first detection ranges and equally high position estimate errors for all expected GGI ranges evaluated.

The results showed large disturbance gradient values for a vehicle the size of a pickup truck allowed detection at large ranges, but the human baseline map was too dissimilar from the true pickup gravity gradients to predict position accurately. A small object the size and mass of a rabbit did not cause an alarm until very close to the GGI cluster, but the baseline map for a human was still too dissimilar from the true rabbit gravity gradient to result in good tracking capability.

Table 13 shows the proportion of simulations that triggered an alarm for each intruder type. As expected GGI range increased, the intruder's ability to trigger an alarm increased as well. The large gravity gradients of the vehicle triggered an alarm every simulation for all the expected GGI range values tested.

Table 13: Percentage of Simulations that Triggered at Least One Alarm

Expected GGI Range (m)	Intruder Type		
	Rabbit	Medium Human	Large Vehicle
10	27	77	100
15	36	100	100
18	40	100	100
20	32	100	100
25	91	100	100
30	100	100	100

Scenario Type: Perimeter, Intruder Speed: Walk, Path Type: Linear, Motion Position: Upright, GGIs Per Cluster: 4; Cluster Height: 1m, Cluster Spacing: 2m, NSD: $0.001E\ddot{o}/Hz^{1/2}$, Perimeter Spacing: 30m, Logic Type: Equal Weighting, Number of Simulations: 150

Combined with Figure 41, the table suggests a few possible ways to improve algorithm performance against off-baseline intruders. Most potential intruders smaller than a human—rabbits, gophers, dogs, etc—are not true security threats, meaning an ideal algorithm would ignore them without sounding an alarm. The rabbit's small *RFD* of less than 3*m* suggests a physical barrier could prevent the gravity gradient for small animals from ever becoming large enough to trigger an alarm. When the expected GGI range was 10*m*, the rabbit was detected in only 27% of the simulations, likely evading detection when it moved between two clusters on the perimeter. A simple barrier like a fence surrounding each GGI cluster with a radius of a few meters would have prevented the rabbit from ever entering the cluster detection range for so small an animal. Raising the GGI off the ground could provide a similar separation effect.

Larger, more massive objects are a more challenging problem. Gravity gradiometry is not useful for distinguishing between objects about the same size and mass, so an algorithm could probably never distinguish a small deer from a human using gravity gradiometry alone. While more mass increases possible detection range, accurate position estimates require the baseline gradient map to match the true intruder characteristics. In the case of a pickup, the many-fold mass increase allows detection at ranges in excess of 40*m* but without correct position estimation. Simulating other than human intruders demonstrated the detection portion of the algorithm was robust against varied intruder shapes and sizes, but the tracking ability suffered when the intruder did not resemble the intruder used to create the baseline gradient map. Appendix B shows the full results of all simulations in Table 14 through Table 20.

V. Conclusions and Recommendations

Gravity Gradiometry as a PIDS Technology

With no previous published work linking gravity gradiometry to outdoor perimeter security, this project was the first step toward merging a longstanding scientific concept with an industry need. Though current technology levels cannot yet produce a GGI capable of the high sensitivity required to detect a human mass change at ranges of tens of meters, new gravity gradient measurement approaches promise future GGIs with noise characteristics several orders of magnitude better than they are today. This project showed how combining that level of progression with the unique nature of gravity gradiometry could allow a cluster containing multiple GGIs to detect and accurately track a human intruder at ranges of $20m$ or more with a position estimation error of less than $2m$, all with a false alarm likelihood of less than one per hour.

Initial gravity gradient calculations demonstrated the gradient contribution of falling rain was negligible relative to future GGI noise levels, while simple estimation techniques allowed successful algorithm performance in spite of large net mass increases near a GGI due to ground absorption of precipitation. Spatially manipulating the GGI array geometry by up to $5m$ in every direction had little effect on overall detection capability, though placing at least two GGIs near each other significantly decreased position estimation error. Varying human intruders from the 5th to 95th size and mass percentiles had only a small effect on detection capability. Detection capability was most dependent on intruder mass, with the more massive intruders first detected at a greater range than smaller intruders. There was almost no difference between detection and

tracking performance when the intruder was in a prone position versus a standing position, further highlighting the importance of mass over shape as the primary indicator of detection range. Faster intruders eluded detection for longer than did slow moving intruders, but intruders were eventually detected at all extremes of the human range of motion.

The created algorithm determined intruder presence and position by applying the likelihood function, comparing every noisy gradient reading to a map of true gradient readings for the expected intruder type—in this case a medium sized human. The algorithm approach resulted in good performance when the intruder was similar to the baseline, but position estimation suffered when there was little similarity. Replacing the human intruder with a large vehicle caused the algorithm to detect the vehicle mass at a range in excess of $40m$, but the gravity gradient tensor value combination was too dissimilar from the baseline gradient map to result in accurate position estimation. While the non-human intruder simulations revealed the algorithm gave poor position estimates when the true intruder was much different from the expected intruder, all types of intruders were detected. Those characteristics mean a GGI-based PIDS has particularly high potential in circumstances where the security system need is limited to detection capability alone.

If predictions of future sensitivity levels are eventually realized, gravity gradiometry appears at least as useful in a security application as existing volumetric PIDS technologies. Like radar, infrared, seismic, and optical intruder detection methods, effective intruder detection using gravity gradiometry is susceptible to natural sources of

noise and decreased performance when the intrusion scenario is different from the design point. Unlike the present technologies, gravity gradiometry is not fooled by deliberate movement methods, heat-masking techniques, or the dark of night. All PIDS work better by combining multiple technologies to overlap the weaknesses of one method with the strengths of other. Gravity gradiometry could also gain from such a combination, but this early investigation suggests it might be a strong enough technology in the future to stand on its own with little performance penalty.

Recommendations for Future Research

As this was an initial feasibility study, many aspects of using gravity gradiometry for perimeter security require more investigation. The most severe need for deeper exploration lies in characterizing natural gravity gradient shifts at the milli-Eötvös level. Results from this project are valid only when the background gravity gradient signal at some location is constant. Natural processes like plate tectonics and the water cycle redistribute large amounts of mass, often without visual indicators. If changes in the natural gravity gradient are common at the sensitivity levels of future instruments, a GGI-based PIDS for human intruder detection would have to take a different approach.

The detection and tracking algorithm needs modification to track and detect multiple intruders. Successful results for a simple, one-person intrusion scenario demonstrated broad concept feasibility, but a perimeter security system requires a stronger algorithm able to handle more intruder types and combinations of intruders. Using multiple baseline gravity gradient maps and better intruder tracking logic are just a

few ways the algorithm might be improved. The algorithm is simple enough to run on a basic desktop computer much faster than real time in its current state, so margin exists for the computational penalty incurred by added complexity.

In the security system realm, a large gap exists in the area of standardized system testing. With only a few common performance metrics and no universally accepted governing body, there are few ways to consistently compare a new security system concept with existing systems. The most reliable and useful method, side-by-side comparison, is an expensive option not even available to systems at the conceptual design level. The need here is two pronged. First, there should be simple, easily replicated tests covering the most common security system applications. Ideally, those tests should lend themselves to computational implementation to estimate the performance of systems still on the drawing board, like the GGI-based PIDS algorithm resulting from this project. Second, the metrics for overall system performance must improve beyond P_d and *FAR*. Those are important parameters to specify and test, but aspects like intruder tracking ability are equally significant. Creating the *DTP* for this project represented a straightforward attempt to combine the most important aspects of a security system, but a concentrated effort to make a more comprehensive, universal PIDS performance parameter could result in an even better measure.

Future studies should aim to optimize GGI arrays and design novel ways to eliminate the effect of undesired sources of gravity gradients like snow and rain. The cluster geometry full factorial analysis in the previous chapter suggested moderate changes to GGI cluster shape had little effect on overall performance, but cluster height,

spacing, and number of GGIs per cluster were all somewhat limited variables because they were all based on a non-varying cluster GGI pattern. Combining more cluster geometry variables with dedicated optimization tools could result in a much better GGI-based PIDS. Similarly, modeling site design options like advanced underground water drainage systems and variable topography could eliminate precipitation and erosion concerns.

Finally, the results of this study indicate gravity gradiometry is useful for an interior security system application. With projected detection and tracking ranges near $20m$ and a high tolerance against spoofing techniques, the characteristics make gravity gradiometry unique from other security technologies make it ideal for something like bank vault security or detecting movement through a hallway. The interior environment is more controlled and has a lower range requirement than an outdoor perimeter does. Gravity gradiometry can 'see' through walls, further expanding its potential application options and separating it from current security technology. Future work should identify shortfalls in current interior security systems and evaluate gravity gradiometry as an interior security system technology.

Appendix A: MATLAB Computer Program

The following program calculates *DTP* and *P_{FA}* for intrusion scenarios based on user inputs. All input options span the first 21 lines of code following the header. All simulations used this code in MATLAB version R2010b to obtain the reported results.

```matlab
%***************************************************************************%
% Jared D. Tuinstra, Air Force Institute of Technology, March 2011
%
% This program simulates a physical intrusion detection system and is
%    intended to evaluate the effectiveness of using gravity gradiometry to
%    detect a human intruder in a variety of user-controlled circumstances.
%
% NOTES:
% - The characteristics of the gravity gradiometer instrument (GGI)
%    array geometry and the simulated intruder, as well as GGI instrument
%    noise and algorithm options is specified in the "User Inputs" block
%    immediately following this header
% - The expected (baseline) intruder is a medium, upright human
% - The default simulated perimeter length is 150m
% - No outside files are required: adjust user inputs and begin!
%
%***************************************************************************%
clear all; close all; clc;

%% User Inputs
num_samples=150; %User inputs number of desired simulation runs
test_for_false_alarm=0; %Available types: 1: Calculates Probability of
                        % False Alarm, Any other value: Calculates DTP
scenario_type=2; %Available types: 1:Straight perimeter, 2:Open area
approach_type=2; %Available types: 1:Constant crawling speed, 2:Constant
                 % moderate-paced walking speed, 3:Constant running speed
path_type=1; %Available types: 1:Linear, 2:Snaking
cluster_type=4; %Available types: 1:Single GGI, 2:Two GGIs, 4:Four pt
                % cross, 6:Six pt 3-D star
perimeter_spacing=30; %Cluster to cluster spacing along perimeter, [m]
                      % {Only used when scenario_type=1}
int_size=1; %Available types: 1:Medium (baseline), 2:Large, 3:Small
NSD=0.001; %Noise spectral density, [Hz/Eo^1/2]
motion_type=1; %Available types: 1:Upright, 2:Prone (crawling)
cluster_spacing=2; %Distance from center position, [m]
cluster_height=1; %Distance cluster is above ground, [m]
GGI_range=18; %Expected effective radius of each GGI cluster, [m]

int_object=1; %Available types: 1:Human, 2:Truck, 3:Rabbit
rain_type=1; %Available types: 1:No rain, 2:Light rain, 3:Heavy rain
logic_type=1; %Available types: 1: Each GGI reading given equal weight,
              % 2: GGI readings averaged in cluster before probabilities
              % calculated

%***************************************************************************
% END OF USER INPUTS
%***************************************************************************

%% Other coded option not investigated for the thesis
if (int_object==2 || int_object==3) && motion_type==2;
 error('Only a human can crawl. Modify int_object or motion_type.')
end

%GGI Cluster Characteristics
 perimeter_position=50; %Depth location of perimeter on grid, [m]
 width_center=50; %Width location of cluster center on grid, [m]
%GGI characteristics
 prod_rate=1.0; %Rate of signal production, [Hz]
%Algorithm characteristics
 expected_sigma_move=10; %Expected moving velocity standard deviation [m/s]
                         % (used to find probability radius),
 rain_length=1; %Duration of rain prior to scenario starting, [hrs]
                % {Use only if rain_type is other than 1}
 %Baseline intruder characteristics (same as medium human)
baseline_mass=76.2; %[kg]
 human_x=0.45; %Body width, [m]
 human_y=0.25; %Body thickness, [m]
 human_z=1.73; %Body height, [m]

%% Assign intruder mass and dimensions
```

```matlab
switch int_object
 case 1
   switch int_size %Indicates the user choice
     case 1 %Medium person
       int_dim(1)=76.2; %Body mass, [kg]
       int_dim(2)=0.45; %Body width, [m]
       int_dim(3)=0.25; %Body thickness, [m]
       int_dim(4)=1.73; %Body height, [m]
     case 2 %Large person
       int_dim(1)=101.6;
       int_dim(2)=0.50;
       int_dim(3)=0.31;
       int_dim(4)=1.85;
     case 3 %Small Person
       int_dim(1)=56.2;
       int_dim(2)=0.42;
       int_dim(3)=0.18;
       int_dim(4)=1.62;
   end
 case 2 %Truck dimensions and weight, all values approximate for 2010
        % Ford F-150
   int_dim(1)=3629; %Vehicle weight, [kg] {8,000 lbs}
   int_dim(2)=2.00; %Vehicle width, [m]
   int_dim(3)=5.50; %Vehicle length, [m]
   int_dim(4)=1.90; %Vehicle height, [m]
 case 3 %Rabbit dimensions and weight, notional values and estimations
        % for Eastern Cottontail Rabbit
   int_dim(1)=2.27; %Animal weight, [kg] {5 lbs}
   int_dim(2)=0.152; %Animal width, [m]
   int_dim(3)=0.254; %Animal length, [m]
   int_dim(4)=0.127; %Vehicle height, [m]
end

if motion_type==2 %Reposition dimensions for crawling human
 int_dim(2,1)=int_dim(1,1);
 int_dim(2,2)=int_dim(1,2);
 int_dim(2,3)=int_dim(1,4);
 int_dim(2,4)=int_dim(1,3);
end

%% GGI Placement and Grid Size Determination
switch scenario_type %Places GGIs on grid according to user inputs
 case 1
   switch cluster_type
     case 1
      for clust=1:ceil(150/perimeter_spacing+1);
       P(clust,:)=[15+perimeter_spacing*(clust-1), perimeter_position,...
           -cluster_height];
      end
     case 2
      for clust=1:ceil(150/perimeter_spacing+1);
       P(2*clust-1,:)=[15+perimeter_spacing*(clust-1)-cluster_spacing, perimeter_position, -cluster_height];
       P(2*clust,:)=[15+perimeter_spacing*(clust-1)+cluster_spacing, perimeter_position, -cluster_height];
      end
     case 4
      for clust=1:ceil(150/perimeter_spacing+1);
       P(4*clust-3,:)=[15+perimeter_spacing*(clust-1)-cluster_spacing, perimeter_position, -cluster_height];
       P(4*clust-2,:)=[15+perimeter_spacing*(clust-1)+cluster_spacing, perimeter_position, -cluster_height];
       P(4*clust-1,:)=[15+perimeter_spacing*(clust-1), perimeter_position-cluster_spacing, -cluster_height];
       P(4*clust,:)=[15+perimeter_spacing*(clust-1), perimeter_position+cluster_spacing, -cluster_height];
      end
     case 6
      for clust=1:ceil(150/perimeter_spacing+1);
       P(6*clust-5,:)=[15+perimeter_spacing*(clust-1)-cluster_spacing, perimeter_position, -cluster_height];
       P(6*clust-4,:)=[15+perimeter_spacing*(clust-1)+cluster_spacing, perimeter_position, -cluster_height];
       P(6*clust-3,:)=[15+perimeter_spacing*(clust-1), perimeter_position-cluster_spacing, -cluster_height];
       P(6*clust-2,:)=[15+perimeter_spacing*(clust-1), perimeter_position+cluster_spacing, -cluster_height];
       P(6*clust-1,:)=[15+perimeter_spacing*(clust-1), perimeter_position, -cluster_height+cluster_spacing];
       P(6*clust,:)=[15+perimeter_spacing*(clust-1), perimeter_position, -cluster_height-cluster_spacing];
      end
   end
 case 2
   clust=1;
   switch cluster_type
     case 1
      P(1,1)=width_center; P(1,2)=perimeter_position; P(1,3)=-cluster_height;
     case 2
      P(1,1)=width_center-cluster_spacing; P(1,2)=perimeter_position; P(1,3)=-cluster_height;
      P(2,1)=width_center+cluster_spacing; P(2,2)=perimeter_position; P(2,3)=-cluster_height;
     case 4
      P(1,1)=width_center-cluster_spacing; P(1,2)=perimeter_position; P(1,3)=-cluster_height;
      P(2,1)=width_center+cluster_spacing; P(2,2)=perimeter_position; P(2,3)=-cluster_height;
      P(3,2)=width_center; P(3,2)=perimeter_position-cluster_spacing; P(3,3)=-cluster_height;
      P(4,1)=width_center; P(4,2)=perimeter_position+cluster_spacing; P(4,3)=-cluster_height;
     case 6
      P(1,1)=width_center-cluster_spacing; P(1,2)=perimeter_position; P(1,3)=-cluster_height;
      P(2,1)=width_center+cluster_spacing; P(2,2)=perimeter_position; P(2,3)=-cluster_height;
      P(3,2)=width_center; P(3,2)=perimeter_position-cluster_spacing; P(3,3)=-cluster_height;
      P(4,1)=width_center; P(4,2)=perimeter_position+cluster_spacing; P(4,3)=-cluster_height;
      P(5,1)=width_center; P(5,2)=perimeter_position; P(5,3)=-cluster_height-cluster_spacing;
      P(6,1)=width_center; P(6,2)=perimeter_position; P(6,3)=-cluster_height+cluster_spacing;
```

```matlab
            end
        end

    GGI_count=size(P,1);

    switch scenario_type
     case 1
        grid_length=perimeter_spacing*GGI_count/cluster_type+15;
        grid_depth=100;
        x_entry_range=[(mean(P(cluster_type*(clust-1)+1:GGI_count,1))-mean(P(1:cluster_type,1)))/2+15- ...
            perimeter_spacing/2 (mean(P(cluster_type*(clust-1)+1:GGI_count,1))-mean(P(1:cluster_type,1)))/2+15+ ...
            perimeter_spacing/2];
        rad_length=grid_depth-5;
     case 2
        grid_length=100;
        grid_depth=100;
        rad_length=45;
    end

    %% Form Expected GGI Range Area
    GGI_bound=ones(grid_depth,grid_length);
    for GGI_rad_count=1:GGI_count;
     [PX,PY] = meshgrid((-P(GGI_rad_count,1)+1):(grid_length-P(GGI_rad_count,1)),(-P(GGI_rad_count,2)+1):(grid_depth- ...
        P(GGI_rad_count,2)));
     L_GGI_rad=GGI_range<(PX.^2+PY.^2).^0.5;
     GGI_bound=GGI_bound.*L_GGI_rad;
    end

    %%  Calculate true gravity gradient values and allocate arrays
    G=6.67E-11;
    sigma_noise=sqrt(NSD^2*0.5*prod_rate); %Calculate noise standard deviation

    Txx_ref=zeros(grid_depth,grid_length,GGI_count);
    Tyy_ref=zeros(grid_depth,grid_length,GGI_count);
    Txy_ref=zeros(grid_depth,grid_length,GGI_count);
    Txz_ref=zeros(grid_depth,grid_length,GGI_count);
    Tyz_ref=zeros(grid_depth,grid_length,GGI_count);

    Txx_ref_base=zeros(grid_depth,grid_length,GGI_count);
    Tyy_ref_base=zeros(grid_depth,grid_length,GGI_count);
    Txy_ref_base=zeros(grid_depth,grid_length,GGI_count);
    Txz_ref_base=zeros(grid_depth,grid_length,GGI_count);
    Tyz_ref_base=zeros(grid_depth,grid_length,GGI_count);

    r=zeros(GGI_count,8); Txx=zeros(GGI_count,8); Tyy=zeros(GGI_count,8);
    Txy=zeros(GGI_count,8); Txz=zeros(GGI_count,8); Tyz=zeros(GGI_count,8);

    count=0;

    switch rain_type %Create rain penetration prism
     case 1
      soil_depth=0.5;
      delta_rho_soil=0;
     case 2
      soil_depth=0.5; %rain penetration depth, [m]
      delta_rho_soil=rain_length*0.0025*1000/soil_depth; %kg/m^3
     case 3
      soil_depth=0.5; %rain penetration depth, [m]
      delta_rho_soil=rain_length*0.0076*1000/soil_depth; %kg/m^3
    end

    for prism_num=1:2;
     for grid_pos_x=1:grid_length;
      for grid_pos_y=1:grid_depth;
       prisx=[grid_pos_x-int_dim(motion_type,2)/2 grid_pos_x+int_dim(motion_type,2)/2 -1000 1000]; %100-150
       prisy=[grid_pos_y-int_dim(motion_type,3)/2 grid_pos_y+int_dim(motion_type,3)/2 -1000 1000]; %120-130
       prisz=[-int_dim(motion_type,4) 0 0 soil_depth]; %-3-3
       prisrho=[int_dim(motion_type,1)/(int_dim(motion_type,2)*int_dim(motion_type,3)*int_dim(motion_type,4)) ...
           delta_rho_soil];
       x=[prisx(2*prism_num-1) prisx(2*prism_num)];
       y=[prisy(2*prism_num-1) prisy(2*prism_num)];
       z=[prisz(2*prism_num-1) prisz(2*prism_num)];
       rho=prisrho(prism_num);
       for i=1:2;
        for j=1:2;
         for k=1:2;
          count=count+1;
          for GGI_num=1:GGI_count; %Closed form gravity gradient equations
           x_diff=P(GGI_num,1)-x(i); y_diff=P(GGI_num,2)-y(j); z_diff=P(GGI_num,3)-z(k);
           r=sqrt(x_diff^2+y_diff^2+z_diff^2);
           Txx(GGI_num,count)=G*rho*10^9*(-1)^(i+j+k)*atan(y_diff*z_diff/(x_diff*r));
           Tyy(GGI_num,count)=G*rho*10^9*(-1)^(i+j+k)*atan(x_diff*z_diff/(y_diff*r));
           Txy(GGI_num,count)=-G*rho*10^9*(-1)^(i+j+k)*log(z_diff+r);
           Txz(GGI_num,count)=-G*rho*10^9*(-1)^(i+j+k)*log(y_diff+r);
           Tyz(GGI_num,count)=-G*rho*10^9*(-1)^(i+j+k)*log(x_diff+r);
          end
         end
        end
       end
       for GGI_num1=1:GGI_count;
        Txx_ref(grid_pos_y,grid_pos_x,GGI_num1)=Txx_ref(grid_pos_y,grid_pos_x,GGI_num1)+sum(Txx(GGI_num1,:));
```

```matlab
        Tyy_ref(grid_pos_y,grid_pos_x,GGI_num1)=Tyy_ref(grid_pos_y,grid_pos_x,GGI_num1)+sum(Tyy(GGI_num1,:));
        Txy_ref(grid_pos_y,grid_pos_x,GGI_num1)=Txy_ref(grid_pos_y,grid_pos_x,GGI_num1)+sum(Txy(GGI_num1,:));
        Txz_ref(grid_pos_y,grid_pos_x,GGI_num1)=Txz_ref(grid_pos_y,grid_pos_x,GGI_num1)+sum(Txz(GGI_num1,:));
        Tyz_ref(grid_pos_y,grid_pos_x,GGI_num1)=Tyz_ref(grid_pos_y,grid_pos_x,GGI_num1)+sum(Tyz(GGI_num1,:));
      end
      count=0;
    end
    grid_pos_y=0;
 end
end

%% Calculate baseline gravity gradients
if logic_type==2
 for assign_P_avg=1:clust
  P_avg(assign_P_avg,1)=mean(P((assign_P_avg-1)*cluster_type+1:(assign_P_avg-1)*cluster_type+cluster_type,1));
  P_avg(assign_P_avg,2)=mean(P((assign_P_avg-1)*cluster_type+1:(assign_P_avg-1)*cluster_type+cluster_type,2));
  P_avg(assign_P_avg,3)=mean(P((assign_P_avg-1)*cluster_type+1:(assign_P_avg-1)*cluster_type+cluster_type,3));
 end
 P=P_avg;
 tot_GGI_num=GGI_count;
 GGI_count=clust;
end

for grid_pos_x=1:grid_length;
  for grid_pos_y=1:grid_depth;
   x=[grid_pos_x-human_x/2 grid_pos_x+human_x/2];
   y=[grid_pos_y-human_y/2 grid_pos_y+human_y/2];
   z=[-human_z 0];
   rho=baseline_mass/(human_x*human_y*human_z);
   for i=1:2;
    for j=1:2;
     for k=1:2;
      count=count+1;
      for GGI_num=1:GGI_count; %Closed form gravity gradient equations
       x_diff=P(GGI_num,1)-x(i); y_diff=P(GGI_num,2)-y(j); z_diff=P(GGI_num,3)-z(k);
       r=sqrt(x_diff^2+y_diff^2+z_diff^2);
       Txx(GGI_num,count)=G*rho*10^9*(-1)^(i+j+k)*atan(y_diff*z_diff/(x_diff*r));
       Tyy(GGI_num,count)=G*rho*10^9*(-1)^(i+j+k)*atan(x_diff*z_diff/(y_diff*r));
       Txy(GGI_num,count)=-G*rho*10^9*(-1)^(i+j+k)*log(z_diff+r);
       Txz(GGI_num,count)=-G*rho*10^9*(-1)^(i+j+k)*log(y_diff+r);
       Tyz(GGI_num,count)=-G*rho*10^9*(-1)^(i+j+k)*log(x_diff+r);
      end
     end
    end
   end
   for GGI_num1=1:GGI_count;
    Txx_ref_base(grid_pos_y,grid_pos_x,GGI_num1)=sum(Txx(GGI_num1,:));
    Tyy_ref_base(grid_pos_y,grid_pos_x,GGI_num1)=sum(Tyy(GGI_num1,:));
    Txy_ref_base(grid_pos_y,grid_pos_x,GGI_num1)=sum(Txy(GGI_num1,:));
    Txz_ref_base(grid_pos_y,grid_pos_x,GGI_num1)=sum(Txz(GGI_num1,:));
    Tyz_ref_base(grid_pos_y,grid_pos_x,GGI_num1)=sum(Tyz(GGI_num1,:));
   end
   count=0;
  end
  grid_pos_y=0;
end

if logic_type==2
 GGI_count=tot_GGI_num;
end

%% Set initial detection conditions and run time loop
 ctct=1;

for cyc=1:num_samples %start simulation loop
 t=1;

 Prob_rad=ones(grid_depth,grid_length);
 if cyc==1
  tic %Estimate total simulation time
 end

 switch scenario_type %Determine intruder starting position
  case 1
    intrud1_x(1)=x_entry_range(1)+(x_entry_range(2)-x_entry_range(1))*rand(1);
    intrud1_y(1)=5;
    x_dumm_fin(1)=0;
    approach_angle=-90;
  case 2
    start_angle=randi(360,1);
    x_dumm_fin(1)=0;
    intrud1_x(1)=width_center+44*cosd(start_angle);
    intrud1_y(1)=perimeter_position+44*sind(start_angle);
    approach_angle=start_angle;
 end

 switch approach_type
  case 1
    mean_speed=0.3; %slow walk/crawl speed, [m/s]
  case 2
    mean_speed=1.5; %medium walk speed, [m/s]
```

113

```matlab
  case 3
    mean_speed=5; %fast jog speed, [m/s]
  end

  if test_for_false_alarm==1
    mean_speed=90/3600; %Ensure loop simulates one hour of data collection
  end

  while (sqrt((intrud1_x(t)-width_center)^2+(intrud1_y(t)-perimeter_position)^2)<rad_length && ...
         intrud1_y(t)<grid_depth-5)

%% Determine Intruder Path
  switch path_type %Calculate new intruder position based on path type
   case 1
     intrud1_x(t+1)=intrud1_x(t)+mean_speed*cosd(approach_angle+180);
     intrud1_y(t+1)=intrud1_y(t)+mean_speed*sind(approach_angle+180);
   case 2
     y_dumm(1)=0;
     y_dumm_fin(1)=0;
     path_rad=5;
     snake_angle=mean_speed/path_rad*t-pi/2;
     x_dumm_fin(t+1)=x_dumm_fin(1)+path_rad*cos(snake_angle);
     y_dumm(t+1)=path_rad+path_rad*sin(snake_angle);
     y_dumm_fin(t+1)=y_dumm_fin(t)+abs(y_dumm(t+1)-y_dumm(t));
     intrud1_x(t+1)=x_dumm_fin(t+1)*cosd(approach_angle+90)-y_dumm_fin(t+1)*sind(approach_angle+90)+intrud1_x(1);
     intrud1_y(t+1)=x_dumm_fin(t+1)*sind(approach_angle+90)+y_dumm_fin(t+1)*cosd(approach_angle+90)+intrud1_y(1);
  end

%% Calculate Measured Intruder GG Signal
   for GGI_num2=1:GGI_count; %Interpolate using intruder position and gravity gradient grid
    T_true(1,GGI_num2)=interp2(1:grid_length,1:grid_depth,Txx_ref(:,:,GGI_num2),intrud1_x(t),intrud1_y(t));
    T_true(2,GGI_num2)=interp2(1:grid_length,1:grid_depth,Tyy_ref(:,:,GGI_num2),intrud1_x(t),intrud1_y(t));
    T_true(3,GGI_num2)=interp2(1:grid_length,1:grid_depth,Txy_ref(:,:,GGI_num2),intrud1_x(t),intrud1_y(t));
    T_true(4,GGI_num2)=interp2(1:grid_length,1:grid_depth,Txz_ref(:,:,GGI_num2),intrud1_x(t),intrud1_y(t));
    T_true(5,GGI_num2)=interp2(1:grid_length,1:grid_depth,Tyz_ref(:,:,GGI_num2),intrud1_x(t),intrud1_y(t));
   end
   if test_for_false_alarm==1
    T_true=T_true.*0; % Set mean GGI reading to zero for false alarm
   end                % calculation(no intruder)
   T_reading=normrnd(T_true,sigma_noise); %Add noise to measured signal

%% Apply Likelihood Function to grid area
  switch logic_type
   case 1
     Error_reading=ones(grid_depth,grid_length);
     Error_true=ones(grid_depth,grid_length);
     for GGI_num3=1:GGI_count;
      Er1=(T_reading(1,GGI_num3)-Txx_ref_base(:,:,GGI_num3)).^2;
      Er2=(T_reading(2,GGI_num3)-Tyy_ref_base(:,:,GGI_num3)).^2;
      Er3=(T_reading(3,GGI_num3)-Txy_ref_base(:,:,GGI_num3)).^2;
      Er4=(T_reading(4,GGI_num3)-Txz_ref_base(:,:,GGI_num3)).^2;
      Er5=(T_reading(5,GGI_num3)-Tyz_ref_base(:,:,GGI_num3)).^2;
      Error_reading=Error_reading.*exp(-1/(2*sigma_noise^2)*(Er1+Er2+Er3+Er4+Er5));
     end
     Prob_best_guess=Error_reading.*Prob_rad;
   case 2
     for avg_read=1:clust %Average cluster readings
       T_reading_avg(1,avg_read)=mean(T_reading(1,(avg_read-1)*cluster_type+1:(avg_read-1)*cluster_type+ ...
          cluster_type));
       T_reading_avg(2,avg_read)=mean(T_reading(2,(avg_read-1)*cluster_type+1:(avg_read-1)*cluster_type+ ...
          cluster_type));
       T_reading_avg(3,avg_read)=mean(T_reading(3,(avg_read-1)*cluster_type+1:(avg_read-1)*cluster_type+ ...
          cluster_type));
       T_reading_avg(4,avg_read)=mean(T_reading(4,(avg_read-1)*cluster_type+1:(avg_read-1)*cluster_type+ ...
          cluster_type));
       T_reading_avg(5,avg_read)=mean(T_reading(5,(avg_read-1)*cluster_type+1:(avg_read-1)*cluster_type+ ...
          cluster_type));
     end
     Error_reading_clust_avg=ones(grid_depth,grid_length);
     for clust_ct=1:clust
      ER1=(T_reading_avg(1,clust_ct)-Txx_ref_base(:,:,clust_ct)).^2;
      ER2=(T_reading_avg(2,clust_ct)-Tyy_ref_base(:,:,clust_ct)).^2;
      ER3=(T_reading_avg(3,clust_ct)-Txy_ref_base(:,:,clust_ct)).^2;
      ER4=(T_reading_avg(4,clust_ct)-Txz_ref_base(:,:,clust_ct)).^2;
      ER5=(T_reading_avg(5,clust_ct)-Tyz_ref_base(:,:,clust_ct)).^2;
      Error_reading_clust_avg=Error_reading_clust_avg.*exp(-1/(2*sigma_noise^2)*(ER1+ER2+ER3+ER4+ER5));
     end
     Prob_best_guess=Error_reading_clust_avg.*Prob_rad;
  end

%% Calculate Most Likely Intruder Positions
  max_liklihood=max(Prob_best_guess(:));
  [r1,c1]=find(Prob_best_guess==max_liklihood);
  row(t)=r1(1);
  col(t)=c1(1);

  if size(r1,1)>10; %If no maximum probability, assume no detection
   if t>1
    switch scenario_type
     case 1
       row(t)=P(find(((row(t-1)-P(:,2)).^2+(col(t-1)-P(:,1)).^2).^0.5==min(((row(t-1)-P(:,2)).^2+ ...
```

```matlab
          (col(t-1)-P(:,1)).^2).^0.5),1),2);
        col(t)=P(find(((row(t-1)-P(:,2)).^2+(col(t-1)-P(:,1)).^2).^0.5==min(((row(t-1)-P(:,2)).^2+ ...
          (col(t-1)-P(:,1)).^2).^0.5),1),1);
      case 2
        row(t)=perimeter_position;
        col(t)=width_center;
      end
    else
      row(t)=1;
      col(t)=1;
    end
  end

  if GGI_bound(row(t),col(t))>0; %Indicate no possible detection
    row(t)=1;
    col(t)=1;
    Prob_rad=ones(grid_depth,grid_length);
    detection_logic(t)=0;
  else
    [X1,Y1] = meshgrid((-col(t)+1):(grid_length-col(t)),(-row(t)+1):(grid_depth-row(t)));
    radius=(X1.^2+Y1.^2).^0.5;
    Prob_rad=exp(-1/(2*expected_sigma_move^2).*radius.^2);
    detection_logic(t)=1;
  end

  % Calculate true inruder distance and position estimate error
  r_error(t)=sqrt((intrud1_x(t)-col(t))^2+(intrud1_y(t)-row(t))^2);
  if scenario_type==1
    r_true(t)=abs(50.1-intrud1_y(t));
  else
    r_true(t)=sqrt((intrud1_x(t)-width_center)^2+(intrud1_y(t)-perimeter_position)^2);
  end

  %% Determine if alarm should be sounded
  if t>2
    if sum(detection_logic(t-2:t))==3
      alarm(t)=1;
    else
      alarm(t)=0;
    end
  else
    alarm(t)=0;
  end

  t=t+1;
end

%% Calculate DTP Metrics
t_first_detect=find(alarm,1); %time when intruder was first detected
if sum(t_first_detect)==0;
  DTP(cyc)=0; %Indicates intruder not detected at all
  fprintf('No detect!!!\n')
  num_alarms=0;
else
  cyc_count(ctct)=cyc;
  r_true_first_detect(cyc)=r_true(t_first_detect); %true radius from center position/perimeter when intruder
                                                   % is first detected
  r_error_alarm_mean(cyc)=mean(r_error(find(r_error.*alarm))); %average position error when intruder is detected
  num_alarms=length(find(alarm==1)); %total number of alarms
  num_cycles=max(find(alarm))-t_first_detect+1; %cycles from first alarm to last alarm
  continuous_trck_param(cyc)=num_alarms/num_cycles;
  DTP(cyc)=(0.5*r_true_first_detect(cyc)-r_error_alarm_mean(cyc))*continuous_trck_param(cyc);
  ctct=ctct+1;
end

if test_for_false_alarm==1 %Output P_FA
  fprintf('-The false detection probability is %2.5f \n',num_alarms/length(alarm));
  error('False alarm calculation complete.')
end

if cyc==1 %Allow option to end program if estimated run time is too long
  cont=input(['Expected run time is ',num2str(toc*(num_samples-1)/60),' more minutes. Continue [1=yes,2=no]? ']);
    if cont==2
      error('Program terminated by user')
    end
    minutes=clock;
    fprintf('-Start time is %2.0f:%2.0f \n',minutes(4),minutes(5));
  end
end

%% Print Performance Parameters to Command Window
DTP_mean=mean(DTP);
n_req=(1.984*std(DTP)/0.3)^2; %According to two tail student t-test for 95% conifdence level that the true TP_mean
                              % is within +/-0.5 of the simulate DTP_mean value
SE_true=1.984*std(DTP)/sqrt(num_samples);

fprintf('-The mean range at first detection is %4.2f\n',mean(r_true_first_detect))
fprintf('-The mean position estimate error is %4.2f\n',mean(r_error_alarm_mean))
fprintf('-The continuous tracking proportion is %4.2f\n',mean(continuous_trck_param))
fprintf('-The mean DTP (+/-%4.2f) is %4.2f\n',SE_true,DTP_mean)
fprintf('-The the approximate number of samples needed for DTP_mean standard error to be 0.3\r')
```

```matlab
fprintf('(95 percent confidence) is %4.1f\n', n_req)
fprintf('-The actual number of samples was %4.0f\n', num_samples)

if int_object==2 || int_object==3
 fprintf('-The mean range at first detection was %4.3f\n', mean(r_true_first_detect(cyc_count)))
 fprintf('-The mean position estimate error was %4.3f\n',  mean(r_error_alarm_mean(cyc_count)))
 fprintf('-The detection likelihood is %4.3f\n',length(cyc_count)/num_samples)
end

%% Plot Results of Last Simulation to Double Check Configuration
subplot(2,2,1)
 hold on; hist(DTP); plot([DTP_mean DTP_mean],[0 num_samples], '--r','LineWidth',2); ...
     text(min(DTP),90,['Mean DTP is ',num2str(DTP_mean,'%4.2f\n')])  ;axis([min(DTP)-2 max(DTP)+2 0 100])
 title('Histogram of DTP values for all Simulations')
 xlabel('DTP'),ylabel('Number of Instances')
subplot(2,2,[3 4])
 hold on; plot(r_error.*alarm,'^b','MarkerSize',5,'MarkerEdgeColor','b','MarkerFaceColor','b'); ...
     plot(find(alarm==0), alarm(find(alarm==0)).*20,'o','MarkerEdgeColor','g','MarkerFaceColor','g','MarkerSize', ...
     6);plot(find(alarm),alarm(find(alarm)).*20,'o','MarkerEdgeColor','r','MarkerFaceColor','r','MarkerSize',6); ...
     plot(1:t-1,r_true,'k')
 xlabel('Time (s)'), ylabel('Distance (m)')
 title('Time History of Alarms, Intruder Range, and Position Estimation Error')
subplot(2,2,2)
 hold on; plot(intrud1_x,intrud1_y,'--bo','LineWidth',1,'MarkerEdgeColor','b','MarkerFaceColor','b','MarkerSize',3);
 plot(col.*alarm,row.*alarm,'o','MarkerEdgeColor','r','MarkerFaceColor','r','MarkerSize',2); ...
     axis([0 grid_length 0 grid_depth])
 text(5,90,['DTP for this run is ',num2str(DTP(cyc),'%4.2f\n')])
 switch scenario_type
  case 1
    for plot_param=1:clust
     plot(15+perimeter_spacing*(plot_param-1)+GGI_range*cosd(0:10:360), ...
         perimeter_position+GGI_range*sind(0:10:360),'--','LineWidth',2,'Color',[0.6 0.6 0.6]);
    end
    plot3(P(:,1),P(:,2),P(:,3),'+k')
  case 2
    plot(width_center+GGI_range*cosd(0:10:360),perimeter_position+GGI_range*sind(0:10:360),'--','LineWidth', ...
        2,'Color',[0.6 0.6 0.6]);
    plot3(P(:,1),P(:,2),P(:,3),'+k')
 end
 xlabel('East Axis'), ylabel('North Axis')
 title('Birds Eye View of Intruder Path and Location Prediction')
```

Appendix B: Tables of Results

Table 14 through Table 20 shows simulation results for the variable combinations discussed in the project methodology discussion. To recap, Table 14 and Table 15 are the results of a simple sensitivity analysis showing the effect of change from the baseline variable set for the perimeter and open area scenarios. Table 16, Table 17, and Table 18 show the results of full factorial analyses in the respective areas of cluster geometry, intruder size, motion and speed, and GGI noise and algorithm settings. Finally, Table 19 and Table 20 present GGI-based PIDS simulated effectiveness against the outside influences of precipitation absorption into the ground and non-human intruders.

All tables in this appendix show the user option settings for every simulation: table rows are simulations while table columns show the user settings and algorithm performance characteristics. The majority of the variables are spelled out entirely, but the 'Scenario' and 'Logic Type' columns use abbreviations to save space. A scenario of 'P' indicates perimeter scenario, a scenario of 'OA' indicates open area scenario, a logic type of 'EW' indicates equal GGI weighting, and a logic type of 'AVG' indicates cluster averaging prior to applying the likelihood function. The six columns on the far right of every table shows the three parameters used to calculate *DTP* and the *DTP* value, as well as the number of simulations and false alarm probability. The number of simulations is based on the number of samples needed to ensure a maximum *DTP* confidence interval of ±0.3—a greater number of simulations indicates a greater *DTP* spread for that combination of user inputs. Since false alarm probability is calculated without an intruder

117

(no-noise gravity gradient of zero), false alarm values are only reported for changes in

cluster geometry and GGI characteristic user inputs.

Table 14: Sensitivity Results—Perimeter Scenario

Scenario	Intruder Speed	Path Type	Motion Position	Human Size	GGIs in Cluster	Perimeter Spacing (m)	GGI Cluster Spacing (m)	GGI Cluster Height (m)	GGI Noise Spectral Density (Eö/Hz$^{1/2}$)	Expected GGI Range (m)	Logic Type	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Mean Continuous Tracking Proportion	Number of Simulations	Detection and Tracking Parameter (±0.3)	False Alarm Probability (±0.02)
BASELINE																	
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
INTRUDER SPEED																	
P	Crawl	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	16.82	1.10	0.96	110	7.03	-
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	-
P	Run	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	5.84	1.15	1.00	170	1.77	-
PATH TYPE																	
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	-
P	Walk	Snaking	Upright	Medium	4	30	2	1	0.001	18	EW	15.17	1.15	0.98	130	6.31	-
MOTION POSITION																	
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	-
P	Walk	Linear	Prone	Medium	4	30	2	1	0.001	18	EW	13.87	1.27	1.00	110	5.65	-
HUMAN SIZE																	
P	Walk	Linear	Upright	Small	4	30	2	1	0.001	18	EW	11.84	1.72	1.00	110	4.19	-
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	-
P	Walk	Linear	Upright	Large	4	30	2	1	0.001	18	EW	15.92	1.78	1.00	120	6.18	-
GGIs IN CLUSTER																	
P	Walk	Linear	Upright	Medium	1	30	2	1	0.001	18	EW	13.36	3.80	0.93	400	2.59	0.004
P	Walk	Linear	Upright	Medium	2	30	2	1	0.001	18	EW	12.66	1.33	0.99	120	4.96	0.001
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
P	Walk	Linear	Upright	Medium	6	30	2	1	0.001	18	EW	13.82	1.13	1.00	110	5.77	0.000
PERIMETER SPACING																	
P	Walk	Linear	Upright	Medium	4	20	2	1	0.001	18	EW	15.30	1.16	1.00	120	6.49	0.000
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
P	Walk	Linear	Upright	Medium	4	40	2	1	0.001	18	EW	11.41	1.45	0.97	300	4.25	0.000
GGI CLUSTER SPACING																	
P	Walk	Linear	Upright	Medium	4	30	1	1	0.001	18	EW	12.58	1.18	1.00	130	5.08	0.000
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
P	Walk	Linear	Upright	Medium	4	30	5	1	0.001	18	EW	16.23	1.38	0.99	130	6.70	0.000
GGI CLUSTER HEIGHT																	
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
P	Walk	Linear	Upright	Medium	4	30	2	3	0.001	18	EW	14.02	1.10	1.00	130	5.91	0.000
P	Walk	Linear	Upright	Medium	4	30	2	5	0.001	18	EW	13.89	0.92	1.00	130	6.01	0.000
GGI NOISE SPECTRAL DENSITY																	
P	Walk	Linear	Upright	Medium	4	30	2	1	0.01	18	EW	41.90	39.76	0.92	130	-17.37	0.876
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
P	Walk	Linear	Upright	Medium	4	30	2	1	0.0005	18	EW	13.59	1.09	1.00	130	5.79	0.000
EXPECTED GGI RANGE																	
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	10	EW	4.67	0.97	0.79	100	1.36	0.000
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	15	EW	10.03	0.94	1.00	100	4.08	0.000
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	20	EW	15.79	1.39	0.99	120	6.46	0.001
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	25	EW	26.27	4.46	0.85	300	7.29	0.036
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	30	EW	37.11	10.70	0.82	300	6.45	0.221
LOGIC TYPE																	
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	EW	13.87	1.15	1.00	100	5.76	0.000
P	Walk	Linear	Upright	Medium	4	30	2	1	0.001	18	AVG	14.02	2.12	1.00	350	4.88	0.000

Table 15: Sensitivity Analysis—Open Area

Scenario	Intruder Speed	Path Type	Motion Position	Human Size	GGIs in Cluster	GGI Cluster Spacing (m)	GGI Cluster Height (m)	GGI Noise Spectral Density ($E\ddot{o}/Hz^{1/2}$)	Expected GGI Range (m)	Logic Type	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Mean Continuous Tracking Proportion	Number of Simulations	Detection and Tracking Parameter (±0.3)	False Alarm Probability (±0.02)
BASELINE																
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	0.000
INTRUDER SPEED																-
OA	Crawl	Linear	Upright	Medium	4	2	1	0.001	18	EW	18.79	1.56	0.97	150	7.59	-
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	-
OA	Run	Linear	Upright	Medium	4	2	1	0.001	18	EW	7.87	1.89	1.00	150	2.04	-
PATH TYPE																
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	-
OA	Walk	Snaking	Upright	Medium	4	2	1	0.001	18	EW	16.27	2.17	1.00	170	5.96	-
MOTION POSITION																
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	-
OA	Walk	Linear	Prone	Medium	4	2	1	0.001	18	EW	15.98	1.78	1.00	170	6.21	-
HUMAN SIZE																
OA	Walk	Linear	Upright	Small	4	2	1	0.001	18	EW	14.23	1.98	1.00	170	5.14	-
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	-
OA	Walk	Linear	Upright	Large	4	2	1	0.001	18	EW	17.79	2.10	1.00	170	6.79	-
GGIs IN CLUSTER																
OA	Walk	Linear	Upright	Medium	1	2	1	0.001	18	EW	14.65	4.23	0.98	500	3.03	0.001
OA	Walk	Linear	Upright	Medium	2	2	1	0.001	18	EW	15.46	1.69	0.99	300	6.00	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	0.000
OA	Walk	Linear	Upright	Medium	6	2	1	0.001	18	EW	15.87	1.57	1.00	150	6.35	0.000
GGI CLUSTER SPACING																
OA	Walk	Linear	Upright	Medium	4	1	1	0.001	18	EW	15.23	1.59	1.00	170	6.01	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	0.000
OA	Walk	Linear	Upright	Medium	4	5	1	0.001	18	EW	18.65	2.16	1.00	170	7.14	0.000
GGI CLUSTER HEIGHT																
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	0.000
OA	Walk	Linear	Upright	Medium	4	2	3	0.001	18	EW	15.92	1.41	1.00	170	6.54	0.000
OA	Walk	Linear	Upright	Medium	4	2	5	0.001	18	EW	15.87	0.86	1.00	170	7.07	0.000
GGI NOISE SPECTRAL DENSITY																
OA	Walk	Linear	Upright	Medium	4	2	1	0.01	18	EW	4.48	19.34	0.89	170	0.73	0.784
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.0005	18	EW	15.95	1.61	1.00	170	6.37	0.000
EXPECTED GGI RANGE																
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	10	EW	8.05	1.97	1.00	150	2.06	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	15	EW	13.06	1.54	1.00	150	4.99	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	20	EW	18.24	2.12	0.99	150	6.94	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	25	EW	27.75	5.47	0.88	150	7.24	0.031
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	30	EW	38.08	10.00	0.88	300	6.97	0.197
LOGIC TYPE																
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	EW	15.90	1.66	1.00	150	6.26	0.000
OA	Walk	Linear	Upright	Medium	4	2	1	0.001	18	AVG	16.05	4.23	1	500	3.79	0.000

Table 16: Full Factorial Analysis—Cluster Geometry

Scenario	Intruder Speed	Path Type	Motion Position	Human Size	GGIs in Cluster	GGI Cluster Height (m)	GGI Cluster Spacing (m)	GGI Noise Spectral Density (Eö/Hz$^{1/2}$)	Expected GGI Range (m)	Logic Type	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Mean Continuous Tracking Proportion	Number of Simulations	Detection and Tracking Parameter (±0.2)	False Alarm Probability (±0.02)
GGIs IN CLUSTER and GGI CLUSTER SPACING and GGI CLUSTER HEIGHT																
OA	Walk	Linear	Upright	Medium	1	1	-	0.001	18	EW	14.55	4.53	0.99	300	2.71	0.001
OA	Walk	Linear	Upright	Medium	1	3	-	0.001	18	EW	14.81	1.79	0.98	350	5.44	0.001
OA	Walk	Linear	Upright	Medium	1	5	-	0.001	18	EW	15.15	1.32	0.96	350	5.90	0.003
OA	Walk	Linear	Upright	Medium	2	1	1	0.001	18	EW	14.73	2.06	1.00	350	5.29	0.000
OA	Walk	Linear	Upright	Medium	2	1	2	0.001	18	EW	15.64	1.65	0.99	350	6.11	0.000
OA	Walk	Linear	Upright	Medium	2	1	4	0.001	18	EW	17.02	1.70	0.99	200	6.72	0.001
OA	Walk	Linear	Upright	Medium	2	1	6	0.001	18	EW	17.59	1.57	0.99	200	7.12	0.001
OA	Walk	Linear	Upright	Medium	2	3	1	0.001	18	EW	14.59	1.33	1.00	200	5.94	0.000
OA	Walk	Linear	Upright	Medium	2	3	2	0.001	18	EW	15.49	1.45	1.00	200	6.28	0.000
OA	Walk	Linear	Upright	Medium	2	3	4	0.001	18	EW	16.78	1.36	0.99	200	6.94	0.001
OA	Walk	Linear	Upright	Medium	2	3	6	0.001	18	EW	17.98	1.48	0.99	200	7.36	0.000
OA	Walk	Linear	Upright	Medium	2	5	1	0.001	18	EW	14.89	0.89	0.99	200	6.51	0.000
OA	Walk	Linear	Upright	Medium	2	5	2	0.001	18	EW	15.80	0.93	0.99	200	6.85	0.001
OA	Walk	Linear	Upright	Medium	2	5	4	0.001	18	EW	16.76	0.95	0.98	200	7.29	0.001
OA	Walk	Linear	Upright	Medium	2	5	6	0.001	18	EW	17.91	1.00	0.98	200	7.75	0.001
OA	Walk	Linear	Upright	Medium	4	1	1	0.001	18	EW	15.18	1.67	1.00	200	5.91	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	18	EW	15.91	1.61	1.00	200	6.33	0.000
OA	Walk	Linear	Upright	Medium	4	1	4	0.001	18	EW	17.62	1.93	1.00	200	6.86	0.000
OA	Walk	Linear	Upright	Medium	4	1	6	0.001	18	EW	19.58	2.18	1.00	200	7.57	0.000
OA	Walk	Linear	Upright	Medium	4	3	1	0.001	18	EW	15.16	1.28	1.00	200	6.29	0.000
OA	Walk	Linear	Upright	Medium	4	3	2	0.001	18	EW	15.79	1.41	1.00	200	6.47	0.000
OA	Walk	Linear	Upright	Medium	4	3	4	0.001	18	EW	17.52	1.70	1.00	200	7.04	0.000
OA	Walk	Linear	Upright	Medium	4	3	6	0.001	18	EW	19.48	1.93	1.00	200	7.77	0.000
OA	Walk	Linear	Upright	Medium	4	5	1	0.001	18	EW	15.13	0.80	1.00	200	6.76	0.000
OA	Walk	Linear	Upright	Medium	4	5	2	0.001	18	EW	15.94	0.83	1.00	200	7.13	0.000
OA	Walk	Linear	Upright	Medium	4	5	4	0.001	18	EW	17.65	0.83	1.00	200	7.97	0.000
OA	Walk	Linear	Upright	Medium	4	5	6	0.001	18	EW	19.45	0.83	1.00	200	8.86	0.000
OA	Walk	Linear	Upright	Medium	6	1	1	0.001	18	EW	15.17	1.57	1.00	200	6.01	0.000
OA	Walk	Linear	Upright	Medium	6	1	2	0.001	18	EW	15.94	1.59	1.00	200	6.38	0.000
OA	Walk	Linear	Upright	Medium	6	1	4	0.001	18	EW	17.65	1.89	1.00	200	6.93	0.000
OA	Walk	Linear	Upright	Medium	6	1	6	0.001	18	EW	19.35	2.18	1.00	200	7.47	0.000
OA	Walk	Linear	Upright	Medium	6	3	1	0.001	18	EW	15.05	1.25	1.00	200	6.27	0.000
OA	Walk	Linear	Upright	Medium	6	3	2	0.001	18	EW	15.95	1.40	1.00	200	6.56	0.000
OA	Walk	Linear	Upright	Medium	6	3	4	0.001	18	EW	17.63	1.67	1.00	200	7.14	0.000
OA	Walk	Linear	Upright	Medium	6	3	6	0.001	18	EW	19.35	1.89	1.00	200	7.77	0.000
OA	Walk	Linear	Upright	Medium	6	5	1	0.001	18	EW	14.98	0.86	1.00	200	6.63	0.000
OA	Walk	Linear	Upright	Medium	6	5	2	0.001	18	EW	15.96	1.00	1.00	200	6.97	0.000
OA	Walk	Linear	Upright	Medium	6	5	4	0.001	18	EW	17.77	1.15	1.00	200	7.71	0.00
OA	Walk	Linear	Upright	Medium	6	5	6	0.001	18	EW	19.28	1.10	1.00	200	8.51	0.000

Table 17: Full Factorial Analysis—Intruder Size, Motion, and Speed

Scenario	Intruder Speed	Path Type	Motion Position	Human Size	GGIs in Cluster	GGI Cluster Height (m)	GGI Cluster Spacing (m)	GGI Noise Spectral Density ($E\ddot{o}/Hz^{f/2}$)	Expected GGI Range (m)	Logic Type	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Mean Continuous Tracking Proportion	Number of Simulations	Detection and Tracking Parameter (±0.3)
INTRUDER SPEED and PATH TYPE and MOTION POSITION and HUMAN SIZE															
OA	Crawl	Linear	Upright	Small	4	1	2	0.001	18	EW	16.99	1.86	0.97	80	6.45
OA	Crawl	Linear	Upright	Medium	4	1	2	0.001	18	EW	18.84	1.57	0.97	80	7.72
OA	Crawl	Linear	Upright	Large	4	1	2	0.001	18	EW	20.84	2.06	0.97	80	8.08
OA	Crawl	Linear	Prone	Small	4	1	2	0.001	18	EW	16.97	2.09	0.97	80	6.21
OA	Crawl	Linear	Prone	Medium	4	1	2	0.001	18	EW	18.86	1.80	0.97	80	7.41
OA	Crawl	Linear	Prone	Large	4	1	2	0.001	18	EW	20.55	2.21	0.97	80	7.81
OA	Crawl	Snaking	Upright	Small	4	1	2	0.001	18	EW	17.07	2.35	0.97	80	6.00
OA	Crawl	Snaking	Upright	Medium	4	1	2	0.001	18	EW	18.89	2.09	0.98	80	7.18
OA	Crawl	Snaking	Upright	Large	4	1	2	0.001	18	EW	21.12	2.66	0.97	80	7.62
OA	Crawl	Snaking	Prone	Small	4	1	2	0.001	18	EW	16.96	2.91	0.97	80	5.37
OA	Crawl	Snaking	Prone	Medium	4	1	2	0.001	18	EW	18.79	2.49	0.97	80	6.72
OA	Crawl	Snaking	Prone	Large	4	1	2	0.001	18	EW	21.20	2.90	0.97	80	7.45
OA	Walk	Linear	Upright	Small	4	1	2	0.001	18	EW	14.06	1.92	1.00	80	5.11
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	18	EW	15.91	1.59	1.00	80	6.36
OA	Walk	Linear	Upright	Large	4	1	2	0.001	18	EW	17.79	2.08	1.00	80	6.81
OA	Walk	Linear	Prone	Small	4	1	2	0.001	18	EW	14.10	2.00	1.00	60	5.05
OA	Walk	Linear	Prone	Medium	4	1	2	0.001	18	EW	16.10	1.80	1.00	60	6.25
OA	Walk	Linear	Prone	Large	4	1	2	0.001	18	EW	17.98	2.26	1.00	60	6.71
OA	Walk	Snaking	Upright	Small	4	1	2	0.001	18	EW	14.97	2.39	1.00	60	5.10
OA	Walk	Snaking	Upright	Medium	4	1	2	0.001	18	EW	16.09	2.15	1.00	60	5.88
OA	Walk	Snaking	Upright	Large	4	1	2	0.001	18	EW	18.39	2.77	0.99	60	6.38
OA	Walk	Snaking	Prone	Small	4	1	2	0.001	18	EW	14.86	2.96	1.00	60	4.46
OA	Walk	Snaking	Prone	Medium	4	1	2	0.001	18	EW	16.02	2.50	1.00	60	5.51
OA	Walk	Snaking	Prone	Large	4	1	2	0.001	18	EW	17.87	2.99	1.00	60	5.92
OA	Run	Linear	Upright	Small	4	1	2	0.001	18	EW	5.65	2.14	1.00	100	0.68
OA	Run	Linear	Upright	Medium	4	1	2	0.001	18	EW	7.96	1.81	1.00	100	2.17
OA	Run	Linear	Upright	Large	4	1	2	0.001	18	EW	9.89	2.35	1.00	100	2.60
OA	Run	Linear	Prone	Small	4	1	2	0.001	18	EW	5.65	2.80	1.00	100	0.02
OA	Run	Linear	Prone	Medium	4	1	2	0.001	18	EW	7.70	2.35	1.00	100	1.50
OA	Run	Linear	Prone	Large	4	1	2	0.001	18	EW	9.60	2.59	1.00	100	2.21
OA	Run	Snaking	Upright	Small	4	1	2	0.001	18	EW	12.34	2.14	1.00	100	4.03
OA	Run	Snaking	Upright	Medium	4	1	2	0.001	18	EW	13.88	1.86	1.00	100	5.08
OA	Run	Snaking	Upright	Large	4	1	2	0.001	18	EW	15.07	2.54	1.00	100	4.99
OA	Run	Snaking	Prone	Small	4	1	2	0.001	18	EW	12.56	2.63	1.00	100	3.65
OA	Run	Snaking	Prone	Medium	4	1	2	0.001	18	EW	13.55	2.65	1.00	100	4.13
OA	Run	Snaking	Prone	Large	4	1	2	0.001	18	EW	14.84	3.05	1.00	100	4.37

Table 18: Full Factorial Analysis—GGI Noise and Algorithm

Scenario	Intruder Speed	Path Type	Motion Position	Human Size	GGIs in Cluster	GGI Cluster Height (m)	GGI Cluster Spacing (m)	GGI Noise Spectral Density (Eö/Hz^{G2})	Expected GGI Range (m)	Logic Type	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Mean Continuous Tracking Proportion	Number of Simulations	Detection and Tracking Parameter (±0.3)	False Alarm Probability (±0.02)
NOISE SPECTRAL DENSITY and EXPECTED GGI RANGE and LOGIC TYPE																
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	10	EW	15.48	3.52	0.78	170	2.81	0.021
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	10	AVG	12.84	3.59	0.85	170	2.08	0.014
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	15	EW	38.34	17.08	0.74	170	1.43	0.555
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	15	AVG	38.45	15.38	0.71	170	2.64	0.437
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	18	EW	40.17	19.32	0.64	170	0.64	0.789
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	18	AVG	40.39	17.31	0.88	170	2.54	0.762
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	20	EW	40.74	19.85	0.94	170	0.47	0.877
OA	Walk	Linear	Upright	Medium	4	1	2	0.01	20	AVG	40.80	17.61	0.95	170	2.64	0.873
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	10	EW	8.04	2.25	1.00	170	1.77	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	10	AVG	8.17	1.63	1.00	170	2.46	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	15	EW	13.28	1.77	1.00	170	4.87	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	15	AVG	13.32	2.23	1.00	170	4.43	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	18	EW	15.92	1.76	1.00	170	6.20	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	18	AVG	16.04	3.21	1.00	170	4.81	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	20	EW	18.18	2.23	0.99	150	6.80	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	20	AVG	18.32	4.35	0.99	150	4.73	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	25	EW	28.24	5.32	0.88	150	7.66	0.033
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	25	AVG	28.76	8.08	0.90	150	5.55	0.028
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	30	EW	35.97	9.71	0.88	180	7.28	0.213
OA	Walk	Linear	Upright	Medium	4	1	2	0.001	30	AVG	37.86	10.91	0.91	180	7.24	0.331
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	10	EW	8.04	2.34	1.00	180	1.68	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	10	AVG	8.09	1.54	1.00	180	2.50	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	15	EW	13.04	1.76	1.00	180	4.76	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	15	AVG	13.25	2.78	1.00	180	3.85	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	18	EW	16.02	1.61	1.00	180	6.40	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	18	AVG	16.08	3.88	1.00	180	4.16	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	20	EW	18.05	1.64	1.00	180	7.39	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	20	AVG	18.25	4.69	1.00	180	4.43	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	25	EW	23.31	2.72	0.99	180	8.83	0.001
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	25	AVG	23.57	6.33	0.99	180	5.38	0.000
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	30	EW	30.68	5.20	0.94	180	9.41	0.010
OA	Walk	Linear	Upright	Medium	4	1	2	0.0005	30	AVG	31.65	8.36	0.94	180	7.01	0.008

Table 19: Precipitation Effects

Scenario	Intruder Speed	Path Type	Motion Position	Human Size	Rain Type	GGIs in Cluster	GGI Cluster Height (m)	GGI Cluster Spacing (m)	GGI Noise Spectral Density (Eö/Hz^{ö2})	Perimeter Spacing (m)	Expected GGI Range (m)	Logic Type	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Mean Continuous Tracking Proportion	Number of Simulations	Detection and Tracking Parameter (±0.3)	False Alarm Probability (±0.02)
PRECIPITATION TYPE and HUMAN SIZE																		
P	Walk	Linear	Upright	Small	None	4	1	2	0.001	30	18	EW	11.81	1.64	1.00	180	4.26	0.000
P	Walk	Linear	Upright	Small	Light	4	1	2	0.001	30	18	EW	10.87	1.85	1.00	180	3.58	0.000
P	Walk	Linear	Upright	Small	Heavy	4	1	2	0.001	30	18	EW	8.99	2.15	1.00	180	2.34	0.000
P	Walk	Linear	Upright	Medium	None	4	1	2	0.001	30	18	EW	13.77	1.17	1.00	180	5.71	-
P	Walk	Linear	Upright	Medium	Light	4	1	2	0.001	30	18	EW	12.82	1.11	1.00	180	5.30	-
P	Walk	Linear	Upright	Medium	Heavy	4	1	2	0.001	30	18	EW	10.79	1.26	1.00	180	4.14	-
P	Walk	Linear	Upright	Large	None	4	1	2	0.001	30	18	EW	16.07	1.77	1.00	180	6.23	-
P	Walk	Linear	Upright	Large	Light	4	1	2	0.001	30	18	EW	14.47	1.53	1.00	180	5.70	-
P	Walk	Linear	Upright	Large	Heavy	4	1	2	0.001	30	18	EW	12.58	1.28	1.00	180	5.01	-

Table 20: Intruder Type Effects

Scenario	Intruder Speed	Path Type	Motion Position	Intruder Type	GGIs in Cluster	GGI Cluster Height (m)	GGI Cluster Spacing (m)	GGI Noise Spectral Density (Eö/Hz^{ö2})	Perimeter Spacing (m)	Expected GGI Range (m)	Logic Type	Number of Simulations	Mean Range at First Detection (m)	Mean Position Estimate Error (m)	Proportion of Simulations that Triggered at Least One Alarm
PRECIPITATION TYPE and HUMAN SIZE															
P	Walk	Linear	-	Rabbit	4	1	2	0.001	30	10	EW	150	0.59	47.29	0.267
P	Walk	Linear	-	Rabbit	4	1	2	0.001	30	15	EW	150	1.36	5.70	0.360
P	Walk	Linear	-	Rabbit	4	1	2	0.001	30	18	EW	150	2.18	6.94	0.400
P	Walk	Linear	-	Rabbit	4	1	2	0.001	30	20	EW	150	3.52	10.93	0.320
P	Walk	Linear	-	Rabbit	4	1	2	0.001	30	25	EW	150	17.87	41.24	0.907
P	Walk	Linear	-	Rabbit	4	1	2	0.001	30	30	EW	150	33.96	58.17	1.000
P	Walk	Linear	Upright	Medium Human	4	1	2	0.001	30	10	EW	150	4.36	1.02	0.767
P	Walk	Linear	Upright	Medium Human	4	1	2	0.001	30	15	EW	150	10.37	1.07	1.000
P	Walk	Linear	Upright	Medium Human	4	1	2	0.001	30	18	EW	150	13.76	1.15	1.000
P	Walk	Linear	Upright	Medium Human	4	1	2	0.001	30	20	EW	150	16.05	1.45	1.000
P	Walk	Linear	Upright	Medium Human	4	1	2	0.001	30	25	EW	150	26.83	4.89	1.000
P	Walk	Linear	Upright	Medium Human	4	1	2	0.001	30	30	EW	150	36.87	9.84	1.000
P	Walk	Linear	-	Large Vehicle	4	1	2	0.001	30	10	EW	150	35.52	18.83	1.000
P	Walk	Linear	-	Large Vehicle	4	1	2	0.001	30	15	EW	150	42.10	20.23	1.000
P	Walk	Linear	-	Large Vehicle	4	1	2	0.001	30	18	EW	150	42.10	20.29	1.000
P	Walk	Linear	-	Large Vehicle	4	1	2	0.001	30	20	EW	150	42.10	20.42	1.000

Bibliography

[1] U.S. Department of Homeland Security, "U.S. Department of Homeland Security Strategic Plan Fiscal Years 2008-2013," Washington: GPO, 2008.

[2] Moteff, J., and Parfomak, P., "Critical Infrastructure and Key Assets: Definition and Identification," Congressional Research Service, RL32631, Washington: GPO, 2004.

[3] Committee on Armed Services, "Security Against Terrorism On U.S. Military Bases," Hearing, 107th Congress, 1st Session, 2001, pp. 1-88.

[4] United States Air Force, "Khobar Towers Bombing," URL: http://www.af.mil/shared/media/photodb/photos/960626-N-0000G-102.jpg [cited 7 Feb 2011].

[5] Williams, J. D., "Advanced technologies for perimeter intrusion detection sensors," *European Convention on Security and Detection (Conf. Publ. No.408)*, London, UK: IEEE, 1995.

[6] Electronic Security Systems Engineering Division, *Perimeter Security Sensor Technologies Handbook*, North Charleston, SC: NISE East, 1997.

[7] Butler, W., Poitevin, P., and Bjomholt, J., "Benefits of wide area intrusion detection systems using FMCW radar,"*, 41st Annual IEEE International Carnahan Conference on Security Technology*, pp. 176-182. Ottawa, ON, Canada: IEEE, 2007.

[8] Cory, P., Everett, H. R., and Pastore, T. H., "Radar-based intruder detection for a robotic security system," *Proceedings of the 1998 Mobile-Robots XIII and Intelligent Transportation Systems*, Vol. 3525, pp. 62-72. Boston: SPIE, 1998.

[9] Smith, G. D., Arlowe, H. D., and Williams, J. D. *A Survey of Early Warning Technologies*. SAND-95-1329C. Palm Desert, CA: Sandia National Laboratories, 1995.

[10] Succi, G., Prado, G., Gampert, R. "Problems in seismic detection and tracking," *Unattended Ground Sensor Technologies and Applications II*, Vol. 4040, pp. 165-173. Orlando, FL: SPIE, 2000.

[11] Sleefe, G. E., Ladd, M. D., McDonald, T. S., "Acoustic and seismic modalities for unattended ground sensors," *Proceedings of the 1999 Unattended Ground Sensor Technologies and Applications Conference*, Vol. 3713, pp. 2-9. Orlando, FL: SPIE, 1999.

[12] U.S. Geological Survey, "Earthquake Glossery-Body Wave," URL: http://earthquake.usgs.gov/learn/glossary/?term=body%20wave [cited 7 Feb. 2011].

[13] Pakhomov, A., and Goldburt, T., "Zero false alarm seismic detection and identification systems," *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense VII*, Vol. 6943, pp. 1-12. Orlando, FL: SPIE, 2008.

[14] Leet, L. D., *Earth waves*. Harvard monographs in applied science, Vol. 2. Cambridge, MA: Harvard University Press, 1950.

[15] Pakhomov, A., "Single and three axis geophone: Footstep detection with bearing estimation, localization and tracking," *Unattended Ground Sensor Technologies and Applications V*, Vol. 5090, pp. 155-161. Orlando, FL: SPIE, 2003.

[16] Lindgren, D., Habberstad, H., and Holmberg, M., "Robust fusion of multiple microphone and geophone arrays in a ground sensor network," *Symposium on Battlefield Acoustic Sensing for ISR Applications*, SET-107, pp. 1-16. Amsterdam, Netherlands: NATO Research and Technology Orgainisation, 2006.

[17] Rosenberg, P. A., *Facility Security: New Threats, New Strategies*. Quincy, MA: National Fire Protection Association, 2002.

[18] Doctor, A. P., "Passive infrared motion sensing technology," *Proceedings of the Aerial Surveillance Sensing Including Obscured and Underground Object Detection Conference,* Vol. 2217, pp. 403-422. Orlando, FL: SPIE, 1994.

[19] Baran, D., O'Brien, B., and Fung, N., *Multi-Camera Persistent Surveillance Test Bed*. ARL-TR-4031, Adelphi, MD: Army Research Lab, 2007.

[20] Hofmann-Wellenhof, B., and Moritz, H., *Physical Geodesy*. New York: SpringerWienNewYork, 2006.

[21] Richeson, J. A., *Gravity Gradiometer Aided Inertial Navigation Within Non-GNSS Environments*. PhD dissertation. University of Maryland, College Park, MD, 2008.

[22] Burkard, R. K., *Geodesy for the Layman* (4th Edition). St. Louis: United States Air Force Geophysical and Space Science Branch, Chart Research Division, 1964.

[23] Mickus, K. L., and Hinojosa, J.H., "The complete gravity gradient tensor derived from the vertical component of gravity: a Fourier transform technique," *Journal of Applied Geophysics,* 46: 159-174 (March 2001).

[24] Nagy, D., Papp, G., and Benedek, J., "The gravitational potential and its derivatives for the prism," *Journal of Geodesy,* 74: 552-560 (2000).

[25] Rogers, M. M., *An Investigation Into the Feasibility of Using a Modern Gravity Gradiometer Instrument for Passive Aircraft Navigation and Terrain Avoidance*. MS Thesis, AFIT/GAE/ENY/09-M16. School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, 2009.

[26] Gray, S. D., Parmentola, J. A., and LeSchack, R., "Estimating the weight of very heavy objects with a gravity gradiometer," *Journal of Physics D: Applied Physics,* 28: 2378-88 (Nov. 1995).

[27] DeGregoria, A., *Gravity Gradiometry and Map Matching: An Aid to Aircraft Inertial Navigation Systems*. MS Thesis, AFIT/GAE/ENY/10-M06. School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, 2010.

[28] Kirkendall, B., Li, Y., and Oldenburg, D., "Imaging cargo containers using gravity gradiometry," *IEEE Transactions on Geoscience & Remote Sensing,* 45: 1786-97 (June 2007).

[29] Streland, A. H., "Going Deep: A System Concept for Detecting Deeply Buried Facilities From Space," Air War College, Maxwell AFB, AL: 2003.

[30] Bell, R. E., "Gravity gradiometry," *Scientific American,* 278: 75-79 (Nov. 1998).

[31] Lumley, J. M. and others, "A Superconducting Gravity Gradiometer Tool for Exploration," ARKex, 2004.

[32] Kasevich, M., "Light-pulse atom-interference-based precision inertial navigation system development program (LPA-PINS)," Stanford University, CA, 2008.

[33] Lester, A. J., and Smith, C. L., "Analyses of performance of volumetric intrusion detection technologies," *Proceedings of the 33rd Annual International Carnahan Conference on Security Technology*. Madrid, Spain: IEEE, 1999.

[34] Leach, G., "A single performance measure for perimeter intruder detection systems," *European Conference on Security and Detection*, London, UK: IEEE, 1997.

[35] Horner, M., Leach, G., and Peaty, S., "Development and use of a method to commission a perimeter intrusion detection system to a performance standard," *Proceedings of the 28th Annual International Carnahan Conference on Security Technology*, Albuquerque, NM: IEEE, 1994.

[36] Leach, G., and Sanson, L.D., "Testing of detection systems," *Proceedings of the 29th Annual International Carnahan Conference on Security Technology*, Sanderstead, UK: IEEE, 1995.

[37] Feraday, S.T., "Standardised procedures for evaluating PIDS," *Proceedings of the 42nd Annual International Carnahan Conference on Security Technology*, Prague, Czech Republic: IEEE, 2008.

[38] Thompson, J.D., "Standardised procedures for evaluating PIDS — An update," *Proceedings of the 43rd Annual International Carnahan Conference on Security Technology*, Zurich, Switzerland: IEEE, 2009.

[39] Maki, M., Nieh, R., and Dickie, M., "Field testing of outdoor intrusion detection sensors," *Proceedings of the 36th Annual International Carnahan Conference on Security Technology*, Atlantic City, NJ: IEEE, 2002.

[40] Hennin, S., Germana, G., and Garcia, L., "Integrated perimeter security system," *Technologies for Homeland Security, 2007 IEEE Conference on,* Raytheon Company, Woburn, MA, 2007, pp. 70-75.

[41] Woodson, W. E., Tillman, B., and Tillman, P., *Human Factors Design Handbook : Information and Guidelines for the Design of Systems, Facilities, Equipment, and Products for Human Use*. New York: McGraw-Hill, 1992.

[42] Glickman, T. S., *Glossary of Meteorology*. Boston, MA: American Meteorological Society, 2000.

[43] Beard, K. V., "Terminal velocity and shape of cloud and precipitation drops aloft," *Journal of the Atmospheric Sciences,* 33: 851-864 (1976).

[44] Nygren, I., "Robust and efficient terrain navigation of underwater vehicles," *Proceedings of the 2008 IEEE/ION Position, Location and Navigation Symposium*, Monterey, CA: IEEE, 2008.

[45] Triola, M. F., *Elementary Statistics*. Boston, MA: Pearson/Addison-Wesley, 2005.

Vita

Second Lieutenant Jared Tuinstra graduated from Pasco High School in Pasco, Washington, and entered undergraduate studies at the United States Air Force Academy in the summer of 2005. He bid farewell to Colorado Springs and Squadron 29 in May 2009 after receiving a Bachelor of Science in Aeronautical Engineering with Distinguished Graduate honors and officer commission into the United States Air Force. He currently lives in Dayton, Ohio, and will continue his work with the Air Force at the Air Force Research Laboratory's aircraft propulsion directorate following graduation.

REPORT DOCUMENTATION PAGE

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of the collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to an penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) 24-03-2011	2. REPORT TYPE Master's Thesis	3. DATES COVERED (From – To) March 2010 – March 2011

4. TITLE AND SUBTITLE Perimeter Security And Intruder Detection Using Gravity Gradiometry: A Feasibility Study	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) Tuinstra, Jared D., 2Lt, USAF	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/ENY) 2950 Hobson Way, Building 640 WPAFB OH 45433-8865	8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GAE/ENY/11-M30

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Stephen Malys (Stephen.Malys@nga.mil) National Geospatial-Intelligence Agency InnoVision Directorate 12310 Sunrise Valley Dr Reston, VA 20191	10. SPONSOR/MONITOR'S ACRONYM(S) NGA
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES
This material is declared a work of the U.S. Government and is not subjected to copyright protection in the United States.

14. ABSTRACT

Changes in the mass distribution around some point on the Earth's surface induce corresponding changes to the magnitude and direction of the gravity vector at that location. The nine-tensor derivative of the gravity vector, or gravity gradient, is sensitive to very small changes in the gravity vector. With some assumptions, continuous measurement of the gravity gradient using a gravity gradiometer (GGI) can be used to determine the location of a mass change in the local area near the instrument. This investigation sought to determine the effectiveness, operating characteristics, and limitations of a physical perimeter security system that uses an array of GGIs to detect and locate a human intruder. Results were obtained via computer simulations that utilized the closed form solution for calculating a gravity gradient given an object's size and mass, as well as industry-predicted future GGI performance characteristics.

15. SUBJECT TERMS
Terrorism Prevention, PIDS, Perimeter Security, Gravity, Intruder Detection, Volumetric

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Richard E. Huffman, Lt. Col., USAF
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (Include area code) (937) 255-6565x7490 (richard.huffman@us.af.mil)
U	U	U	UU	142	