

5-2009

## Human identification using pyroelectric infrared sensors

Aron E. Suppes  
University of Nevada, Las Vegas

Follow this and additional works at: <https://digitalscholarship.unlv.edu/thesesdissertations>



Part of the [Computer Engineering Commons](#), and the [Signal Processing Commons](#)

---

### Repository Citation

Suppes, Aron E., "Human identification using pyroelectric infrared sensors" (2009). *UNLV Theses, Dissertations, Professional Papers, and Capstones*. 975.  
<https://digitalscholarship.unlv.edu/thesesdissertations/975>

This Thesis is protected by copyright and/or related rights. It has been brought to you by Digital Scholarship@UNLV with permission from the rights-holder(s). You are free to use this Thesis in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you need to obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself.

This Thesis has been accepted for inclusion in UNLV Theses, Dissertations, Professional Papers, and Capstones by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact [digitalscholarship@unlv.edu](mailto:digitalscholarship@unlv.edu).

HUMAN IDENTIFICATION USING PYROELECTRIC  
INFRARED SENSORS

by

Aron Edward Suppes

Bachelor of Science Computer Science  
Hawaii Pacific University  
2005

A thesis submitted in partial fulfillment  
of the requirements for the

**Master of Science Degree in Computer Science  
School of Computer Science  
Howard R. Hughes College of Engineering**

**Graduate College  
University of Nevada, Las Vegas  
May 2009**

UMI Number: 1484373

All rights reserved

INFORMATION TO ALL USERS

The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if material had to be removed, a note will indicate the deletion.



UMI 1484373

Copyright 2009 by ProQuest LLC.

All rights reserved. This edition of the work is protected against unauthorized copying under Title 17, United States Code.



ProQuest LLC  
789 East Eisenhower Parkway  
P.O. Box 1346  
Ann Arbor, MI 48106-1346

Copyright by Aron Edward Suppes 2009  
All Rights Reserved



**Thesis Approval**  
The Graduate College  
University of Nevada, Las Vegas

APRIL 3rd, 2009

The Thesis prepared by

ARON EDWARD SUPPES

Entitled

HUMAN IDENTIFICATION USING PYROELECTRIC SENSORS

is approved in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

[Redacted Signature]

*Examination Committee Chair*

[Redacted Signature]

*Dean of the Graduate College*

[Redacted Signature]

*Examination Committee Member*

[Redacted Signature]

*Examination Committee Member*

[Redacted Signature]

*Graduate College Faculty Representative*

## ABSTRACT

### **Human Identification Using Pyroelectric Infrared Sensors**

by

Aron E. Suppes

Dr. Evangelos Yfantis, Examination Committee Chair  
Professor of Computer Science  
University of Nevada, Las Vegas

The objective of this thesis is to discuss the viability of using pyroelectric infrared (PIR) sensors as a biometric system for human identification.

The human body emits infrared radiation, the distribution of which varies throughout the body, and depends upon the shape and composition of the particular body part. A PIR sensor utilizing a Fresnel lens will respond to this infrared radiation. When a human walks, the motion of the body's individual components form a characteristic gait that is likely to affect a PIR sensor field in a unique way.

A statistical model, such as a Hidden Markov Model, could be used for the identification process. The model would consist of two phases; learning and testing. The learning phase would train the model on a particular feature or signature. The testing phase would take a signature as input and determine which of the trained models it matches with the highest probability.

## TABLE OF CONTENTS

APPROVAL PAGE.....	ii
ABSTRACT .....	iii
LIST OF FIGURES.....	vi
LIST OF TABLES.....	vii
ACKNOWLEDGMENTS.....	viii
CHAPTER 1 INTRODUCTION.....	1
CHAPTER 2 BACKGROUND .....	3
Gait for Human Identification .....	3
Infrared Radiation and the Human Body.....	4
Measuring Gait using Infrared Radiation.....	4
CHAPTER 3 METHODOLOGY.....	6
Sensor Module.....	6
Pyroelectric Infrared (PIR) Sensor.....	6
Fresnel Lens Array .....	8
Motion Detection Circuit.....	8
Dwell Time.....	9
Retriggering .....	9
Circuit Design .....	10
Signal Generation.....	12
Decay .....	13
Digital Signal Processing .....	14
Analog to Digital Conversion .....	14
Feature Generation.....	15
Normal Distribution.....	16
Thresholding.....	17

Low-pass Filtering .....	19
Temporal Pattern Recognition .....	20
Classification .....	20
Hidden Markov Models (HMM) .....	21
HMM Parameters .....	22
Forward Algorithm .....	24
Backward Algorithm.....	25
Forward-Backward Algorithm .....	26
Model Training .....	29
Model Testing .....	29
Maximum Likelihood.....	30
CHAPTER 4 DATA ANALYSIS .....	31
Configuration .....	31
Microcontroller .....	32
PIR Identification Application.....	34
Initial Testing.....	35
Experiment.....	41
CHAPTER 5 CONCLUSIONS.....	44
BIBLIOGRAPHY .....	47
VITA.....	48



## LIST OF FIGURES

Figure 1. Schematic for motion detection circuit.....	11
Figure 2. The component side of the motion detection circuit. ....	12
Figure 3. The sensor side of the motion detection circuit. ....	12
Figure 4. Signal as it decays with respect to time. ....	14
Figure 5. Waveform showing decay.....	18
Figure 6. Sensor module.....	31
Figure 7. PIR stand with four sensor modules. ....	32
Figure 8. Schematic of Analog-to-Digital converter.....	33
Figure 9. Demo board containing Javelin Stamp and ADC.....	34
Figure 10. PIR Human Identification Application.....	35
Figure 11. Experiment walking path.....	36
Figure 12. Signal generated from sensor 1.....	37
Figure 13. Signal generated from sensor 2.....	37
Figure 14. Signal generated from sensor 3.....	38
Figure 15. Signal generated from sensor 4.....	38
Figure 16. Sample means for three subjects.....	40
Figure 17. Sample standard deviations for three subjects. ....	40

## LIST OF TABLES

Table 1	Experiment Results (Training Set = 500), (Testing Set = 500) .....	41
Table 2	Experiment Results (Training Set = 500), (Testing Set = 200) .....	41
Table 3	Experiment Results (Training Set = 500), (Testing Set = 100) .....	42
Table 4	Experiment Results (Training Set = 500), (Testing Set = 50) .....	42
Table 5	Experiment Results (Training Set = 1000), (Testing Set = 500) .....	42
Table 6	Experiment Results (Training Set = 1000), (Testing Set = 200) .....	42
Table 7	Experiment Results (Training Set = 1000), (Testing Set = 100) .....	42
Table 8	Experiment Results (Training Set = 1000), (Testing Set = 50) .....	43

## ACKNOWLEDGEMENTS

Special thanks to all the people who helped me complete this thesis. In particular I would like to thank Ghada, Josh, and Shirin for their patience and understanding. I would also like to thank my examination committee, in particular Dr. Evangelos Yfantis for all his help.

## CHAPTER 1

### INTRODUCTION

A biometric system relies on pattern recognition for the measurement and analysis of unique physical or behavioral characteristics as a means of verifying personal identity. Many aspects of human form and behavior are characteristic and therefore, suitable for identification purposes. Typical biometrics may include the fingerprint, face, hand, iris, DNA, signature, gait, and voice. Biometric systems work by measuring a particular biometric using a sensor and then transforming the sensed data into a digital feature.

Human gait has been shown to be a characteristic trait of human behavior that can be used for identification. For analysis, a person's gait must first be recorded, typically using a video camera. Image processing techniques are then utilized to measure the principal components of the human gait as they change over time. The relative expense of video cameras and the supporting data systems make their use somewhat limited and impractical for many common tasks.

Pyroelectric (PIR) sensors have been used for many years as a primary component of motion detection devices. These sensors and the associated electronics are widely available and very inexpensive. These sensors work by detecting the infrared radiation released from the human body. When a human

walks, the motion of the body's individual components are likely to affect a PIR sensor field in a unique way. Measuring gait with sensors such as these is desirable because of the low system costs and data loads. The objective of this paper is to discuss the viability of measuring the human gait as a biometric for human identification, using pyroelectric infrared (PIR) sensors.

## CHAPTER 2

### BACKGROUND

#### Gait for Human Identification

Gait refers to the manner of walking or running. It has been shown that humans have the capability of recognizing people from a distance solely based on their gait. This work seems to suggest that there must be some sort of characteristic nature to a person's gait, which if properly extracted, could be used by a biometric system for identification purposes.

For normal walk, gait sequences are repetitive and exhibit nearly periodic behavior, particularly in the movement of the legs and in the swing of the arms. While a person's gait may seem random and unpredictable, studies have shown that there are 24 different components to human gait and that if all movements are considered, gait is unique. There are two aspects to human gait that need to be measured in order for it to be distinctive; structural content and dynamic content. Structural content includes data obtained from the shape of the subject's profile taken at successive frames to show how it changes over time. Dynamic content includes data obtained from measurements that focus on the velocity of the subject's motion. While distinct, both types of information are required for the gait to be characteristic and therefore, cannot be decoupled.

## Infrared Radiation and the Human Body

All objects radiate energy continuously in the form of electromagnetic waves due to the thermal vibrations of their molecules. As such, any object that has a temperature above absolute zero is known to emit electromagnetic radiation at some wavelength, with the wavelength being directly proportional to the temperature of the object. We can calculate the peak wavelength of the radiation emitted by an object, such as the human body, by taking advantage of this known relationship between wavelength and temperature called Wein's displacement law:

$$\lambda_{max} = \left\{ \frac{0.2898 \times 10^{-2} \text{m} \times \text{°K}}{T} \mid T = \text{temperature of object in } \text{°K} \right\}$$

The temperature of the skin of a typical human being is approximately 34 °C. By converting this temperature to Kelvin, we can determine the peak wavelength as:

$$\lambda_{max} = \frac{0.2898 \times 10^{-2} \text{m} \times \text{°K}}{307 \text{°K}}$$

$$\lambda_{max} \approx 9.5 \times 10^{-6} \text{m or } 9.5 \text{ } \mu\text{m}$$

Thus, it can be shown that the black body radiation emitted by a person peaks between 9 to 10  $\mu\text{m}$ , which lies in the infrared (IR) portion of the electromagnetic spectrum.

## Measuring Gait using Infrared Radiation

In order for a person's gait to be analyzed, it must first be recorded or sensed in some way. Typically, video cameras are set up to record the movements while image processing techniques are utilized to measure the principal components of

the human gait. Another method, which is the topic of this paper, is to utilize a person's emitted infrared radiation as a way to measure their gait.

There are many factors to consider when designing a biometric system, the primary one in many cases being cost. Visible light and infrared cameras are relatively expensive and also have drawbacks including high data loads to transfer and store video and large computational costs to perform the necessary image processing. Pyroelectric sensors, on the other hand, potentially offer a much cheaper method of human detection and/or identification. They also would involve much lighter data loads, computational costs, and cheaper system costs.

As we have already discussed, the human body radiates energy in the near-infrared, approximately 9 to 10  $\mu\text{m}$ . This radiation, however, is not constant as the human body is irregularly shaped and due to movement, its overall surface area will change relative to a common point. From the point-of-view of a PIR sensor, the magnitude and duration of the IR energy detected can be used to track this movement in the spatial domain.

Due to its complex geometry and the way heat is radiated to the environment, the human body can be viewed as a distributed IR source that varies both in time and in the spatial domain. Therefore, it seems plausible to assume that recognizing how a person's infrared signature varies over space and time as they walk, may correlate with their gait.



## CHAPTER 3

### METHODOLOGY

#### Sensor Module

Many PIR sensor modules are commercially available, but these are typically built for use as devices that simply detect motion and close a relay for an external lighting or alarm circuit. These devices are unsuitable for human identification purposes. What is needed for our experiment is a package that contains a dual-element sensor, motion detection circuit, and a Fresnel lens array. We will refer to this package as a sensor module.

For this experiment, we chose commercially available motion detectors that contained a PIR sensor and circuit board located in a molded plastic package with an attached Fresnel lens array. The PIR sensor and circuit board were discarded and replaced with a purpose-built motion detection circuit that is more suited for human identification applications. The following sections will describe each of these components more thoroughly.

#### Pyroelectric Infrared (PIR) Sensor

A pyroelectric infrared (PIR) sensor is a device that generates a surface electric charge in response to being exposed to infrared radiation. The sensor used in this experiment is the PIR325, available from the Glolab Corporation.

The sensor package consists of a sensing element, a field-effect transistor (FET), and an infrared filter.

The sensing element of a PIR sensor is composed of a crystalline material that possesses the pyroelectric property. Pyroelectricity is defined as the ability to generate an electric potential when exposed to changes in temperature. The amount of electric potential generated by the charge from the sensing element is usually very small, on the order of 1 mV. This flow of electric charge drives a field-effect transistor (FET).

The crystalline material used in the sensing element is sensitive to a wide range of radiation. PIR sensors used for motion detection generally are only interested in the band with which the human body radiates; approximately 8 to 14  $\mu\text{m}$ , with a peak wavelength of approximately 9.5  $\mu\text{m}$ . The sensor package includes an infrared filter to limit the incoming radiation to this range for this very purpose.

For the purposes of detecting motion, the sensor package includes two sensing elements, referred to as a dual-element sensor. This is necessary to offset signals caused by direct sunlight, changing room temperature, and vibration from generating a voltage. This is accomplished by connecting the two sensing elements in series opposition. With this configuration, the voltage produced by the sensor is the difference of the voltage produced by both of the sensing elements. The sensing elements are oriented along the horizontal plane within the sensor package. The energy being radiated from an object passing in front of the sensor will invariably reach one of the sensing elements before the

other, thereby causing a positive or negative voltage to be induced. Any radiation that strikes both sensing elements at the same exact time will generate the same voltage in both sensing elements and therefore will be cancelled.

### Fresnel Lens Array

Fresnel lenses are typically used in motion detectors to focus infrared radiation onto the sensing elements of a PIR sensor. A Fresnel lens can capture more infrared radiation and focus it onto a smaller point on the PIR sensing element. This focal point moves across the sensor as the IR source moves and exposes one element at a time. The use of a Fresnel lens can increase the maximum distance of detection from a few meters to approximately 30 meters and provide more stable detection.

Fresnel lenses share all the optical characteristics of a conventional plano-convex lens but are physically quite different. A Fresnel lens array is a single molded piece of plastic that is transparent to infrared radiation but only translucent to visible radiation. This piece of plastic has multiple individual Fresnel lenses molded into it. They are used to produce a broad, tailored field of view through many repeated narrow fields of view.

### Motion Detection Circuit

In order to generate a characteristic digital signature using a motion detector, the circuit that implements motion detection must exhibit certain characteristics. Namely, a low dwell time and a fast trigger time. Each of these topics will be covered in the following sections.

### Dwell Time

Dwell is the amount of time that a circuit remains on after motion is detected. For example, in a conventional motion detection circuit it may be desired to use a relay to close an external circuit once motion is detected, such as an outdoor lighting circuit. In this circumstance, you would want a dwell time equal to the amount of time that the lighting circuit should be closed and the lights to remain on. For the purposes of digital signature generation, the dwell time should be less than the polling interval of the device that's checking the state of the motion detector. If this is not the case, a single motion detection event will appear in the digital feature as multiple samples. This would reduce the resolution of the system and increase network traffic and storage requirements with redundant information. The details of these processes will be discussed in the next section.

### Retriggering

A circuit's ability to retrigger is a timing issue that describes whether, and how fast it, can generate a motion detection event after a preceding motion detection event. Using the outdoor lighting circuit example from above, a fast trigger may be unnecessary if the idea is to close an external circuit for a relatively long time once a motion detection event has occurred. In this circumstance, a motion detection event will initially close the circuit for a length of time determined by the dwell time of the circuit. Retriggering the circuit while it is in dwell is unnecessary because the external lighting circuit is already closed and on. In this case, the circuit can effectively ignore multiple repeated motion detection events. Stated

more clearly, the time to retrigger the circuit should be greater than or equal to the circuit's dwell time. For the purposes of digital signature generation, a fast retriggering of the circuit is necessary, in part because of the circuit's requirement of a fast dwell time. It is also desired to encapsulate a single motion detection event in a single sample when polling. A fast trigger prevents the device that is polling the motion the state of the motion detector from sampling a single motion detection event multiple times, thereby increasing resolution and decreasing network bandwidth and storage requirements. The need for fastest trigger possible eliminates the need for a timing circuitry to be added to the motion detection circuit, usually implemented as an integrated circuit. In this case, the circuit's time to retrigger is determined by the amount of time needed by the sensing elements to build up the necessary charge to drive the on-board field-effect transistor in the motion sensor package. This is the fastest possible retrigger time for the circuit and is determined by the individual sensor's physical limitations of its sensing elements and the resistive and capacitive elements of the circuit. This fundamentally provides an upper-bound on how fast the circuit can effectively be polled.

### Circuit Design

Commercially available motion detection circuits are designed for a much different purpose than human identification and as such, are not robust enough to perform this task. For this experiment, four motion detection circuits were built for this specific purpose. The motion detection circuit is powered by a +5V power supply, connected to the drain pin of the PIR325 sensor (see Figure 1). As

stated above, the voltage generated by the PIR325 sensor in response to a motion detection event is low, approximately 1 mV, thus amplification is necessary. This is accomplished using a LM324 operational amplifier in two stages. A capacitor is placed across the power supply and ground to prevent oscillation. In addition, a 100K pull-down resistor connected across the PIR325s source pin to ground. This keeps the output of the circuit at approximately 0 V when no motion is detected. Below is a schematic of the motion detection circuit:

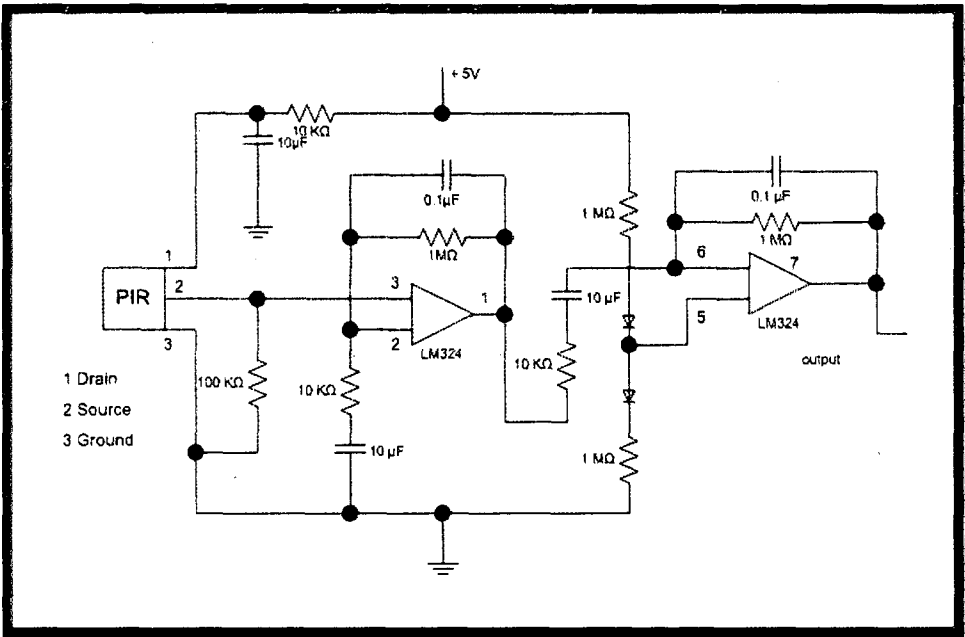


Figure 1. Schematic for motion detection circuit.

Two photographs, one of the sensor side and another of the component side of the motion detection circuit are shown in the figures below:

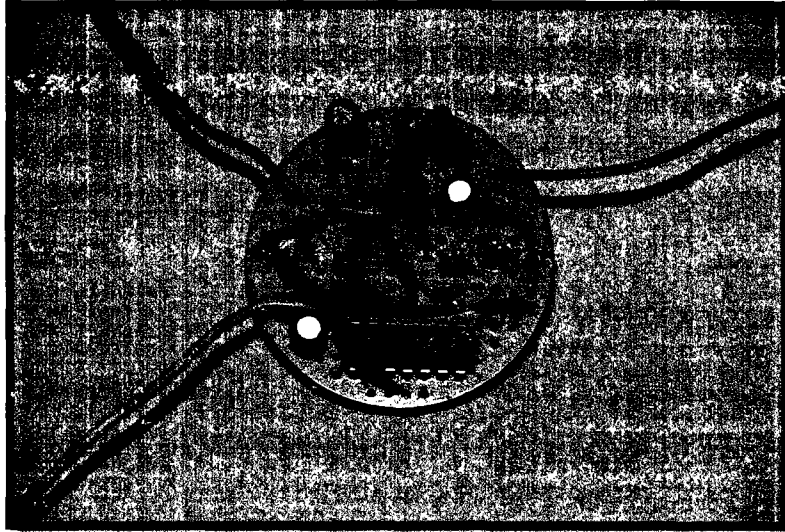


Figure 2. The component side of the motion detection circuit.

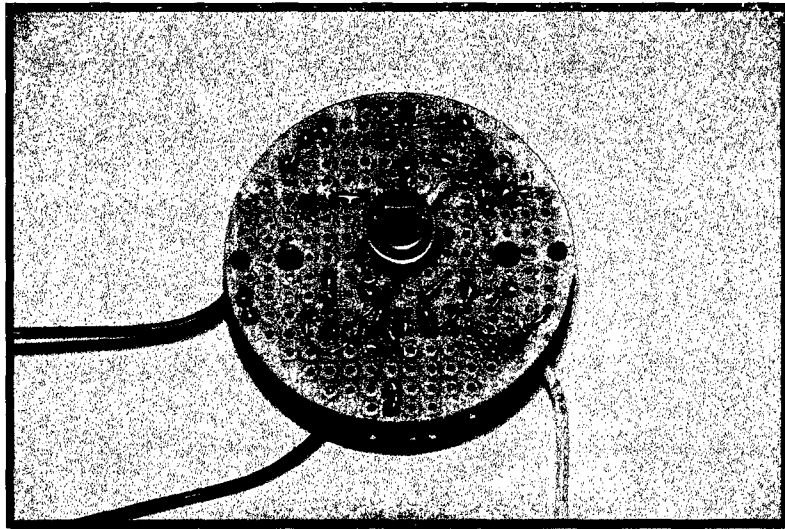


Figure 3. The sensor side of the motion detection circuit.

### Signal Generation

The output of the motion detection circuit is an analog signal whose strength is relative to the amount of charge generated by the elements of the motion sensor, which in turn represents the amount of infrared radiation striking them.

When power is initially applied to the circuit, the capacitive elements must reach a stable state, which will cause the circuit's output to be unstable. After approximately 30 seconds the circuit will stabilize, generating approximately 0 V when no motion is detected. During a motion detection event, the charge generated by the PIR sensing elements will drive a FET which will produce output. The resulting signal will be amplified 72 dB using a LM324 op-amp. This will produce a peak voltage of 3.9V. Due to the motion of the object passing in front of the sensor, a voltage will be induced in one sensing element before the other, producing a positive or negative voltage. As the object moves, the voltage will be induced in the other sensing element and the voltage will alternate.

#### Decay

The amplitude of the resulting positive and negative spikes will vary based on the magnitude of the infrared radiation detected by the sensing elements, with the peak being  $\pm 3.9$  V. If motion is suddenly stopped, the signal will begin to decay to 0 V. This decay is a function of the resistive and capacitive elements in the circuit and therefore is impossible to completely eliminate. This unwanted decay can have the detrimental effect of potentially causing the device polling the motion detector to register a motion detection event on a decaying signal. An example of this can be shown in the following example:



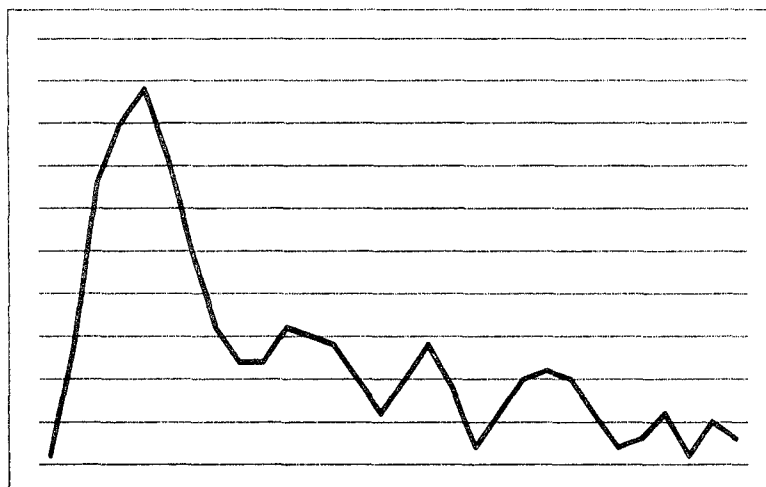


Figure 4. Signal as it decays with respect to time.

The effects of decay can be reduced to acceptable levels through the use of signal processing, which will be discussed later in this chapter.

### Digital Signal Processing

The output of the motion detection circuit will be an analog signal. In order to input this signal to a computer for the purposes of extracting a characteristic digital feature, it must be processed. The signal processing techniques used in this experiment include analog-to-digital conversion, noise detection, thresholding, and low-pass filtering. The following sections will describe each of these techniques more thoroughly.

### Analog to Digital Conversion

The signal that is generated by the motion detection circuit is analog and represents how the voltage generated by a PIR sensor varies with time. For this signal to be analyzed by a digital computer, it must first be converted to digital form. This process is known as analog-to-digital conversion (ADC).

When converting an analog signal to a digital form, there are two important processes to consider: sampling and quantization. Sampling refers to the process of converting the independent variable (time in this case) from continuous to discrete.

Quantization, on the other hand, converts the dependant variable (voltage in this case) from continuous to discrete. With respect to the motion detection circuit, the continuous range of voltage values contained in the analog signal must be approximated by an 8-bit integer, with values ranging from 0 to 255. This approximation results in an inevitable error that can be as high as  $\pm 0.5$  LSB (Least Significant Bit). Quantization results in the addition of a specific amount of random noise and is uniformly distributed and possesses a mean of zero. Since the uncorrelated noise in a system is additive, the noise generated by quantization will simply add to whatever noise is already present in the analog signal.

#### Feature Generation

Once the motion vector has been digitized and corrected for noise, it will consist of integers that can range from 0 to 255 that represent a voltage between -5V and +5V, with 127 being the zero reference or 0V. It is important to note that the polarity of the voltage at any point is irrelevant; it simply corresponds to a particular sensing element in a dual element sensor receiving infrared radiation a moment before the other sensing element. What is important is the magnitude or intensity of the radiation. For example, there is no important difference between -2V and +2V when it comes to generating a feature, as they both correspond to

the same intensity of infrared radiation emitted from the source. Therefore, at this point it would be advantageous to adjust the signal so that the number 0 will provide the reference point of 0V, while using the absolute values to eliminate any notion of polarity. Subtracting 127 from the digital number and then taking the absolute value can accomplish this as shown in the following formula:

$$y = \{abs(x - 127) | x = old\ DN, y = new\ DN\}$$

The resulting motion vector will consist of integers in the range 0 – 127.

### Normal Distribution

Unlike a square wave or a triangle wave, for example, the waveform created by a motion event has a random appearance. This is due to minor differences in intensity of the infrared source, as well as a random temporal element due to the unpredictable and often unrepeatable exact speed of motion of the source with respect to the position of the sensor. This randomness means that the signal will have a bell-shaped probability density function (pdf) that approximates a Gaussian curve, or said to be normally distributed.

The fact that the underlying process exhibits a normal distribution provides two parameters that allow us to characterize the signal, the mean and the standard deviation. The mean describes the peak of the Gaussian curve and is of the form:

$$\mu = \frac{1}{N} \sum_{i=1}^N x_i$$

The mean represents the average value of the signal. It is also commonly referred to as the signal's DC component.

The standard deviation of a signal describes the breadth of the Gaussian curve and represents the average amount of variance between the individual values and the mean, and is of the form:

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu)^2}$$

Utilizing the mean and the standard deviation will allow us to perform the thresholding step described in the next section.

### Thresholding

Thresholding describes the process of converting a digital value with an arbitrary range (i.e. 8 bit) into a binary value represented by either a 0 or 1. Thresholding is critical step in the generation of a digital feature as it involves the separation of the carrier signal from the information contained within it.

In order to fully understand why the signal behaves the way it does, it is necessary to review how it is generated. When a PIR sensor detects a motion event, a voltage is generated which is measured by the onboard circuit. The instant this peak voltage is generated, it begins to decay. This decay will cause the signal to ramp down until either another motion event is detected, or the resistive and capacitive elements of the circuit reach equilibrium and zero voltage is present in the circuit. This decaying signal represents a voltage present in the circuit and therefore will generate a digital number in the analog to digital conversion process. The following figure shows an example waveform that

represents a motion vector, in which it is easy to see this decaying process between peaks:

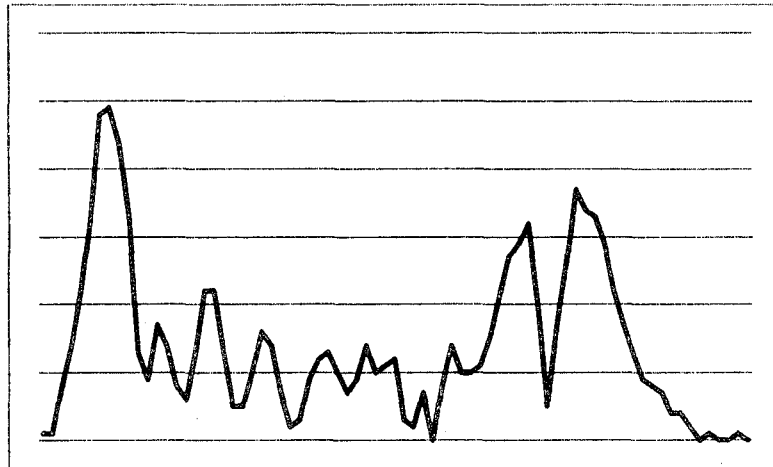


Figure 5. Waveform showing decay.

This decay means that during a repetition of motion events, the signal will very rarely show a value of 0V. This means that when generating a binary-valued signal using thresholding, it is not practical to simply assign a value of 1 when a voltage is present and a 0 otherwise. This would result in a binary signal that would contain all 1s for a motion event and 0s when no motion is detected (although due to the white noise generated by the electronics as discussed in a previous section, there exists a high probability that a voltage, regardless of how small, will be present in the circuit.) In order to separate the information from the signal, a more sophisticated algorithm must be employed.

The key to thresholding lies in understanding the underlying process that generates the signal. The parts of the signal with the utmost interest are the peaks, as these are representative of a motion event. The decaying signal is

representative of a type of noise that is not characteristic of the underlying process. This noise will blend in with the white noise already present in the system. Ideally we would like to design a thresholding function that captures the higher amplitude peaks and ignores the lower amplitude areas between the peaks. This function should generate a binary-valued sequence of alternating 1s and 0s that should be representative of a human motion feature.

By sampling the white noise present on the system, we can design a thresholding function that can exclude this noise and capture the information contained in the signal. This thresholding function must be proportional to the noise level of the system. By first calculating the mean, and then the standard deviation of the white noise, it is possible to form a threshold wherein only values that lie above a particular amplitude are allowed to generate a binary 1. This method provides a way to reliably extract the information generated by the underlying process from the noisy signal.

#### Low-pass Filtering

The output of the thresholding function creates a binary signal, but this signal still contains many high-frequency components that were contained in the original signal. These high-frequency components are indicative of noise present in the system. What we are interested in is the overall motion event contained in the signal. For this reason, we pass the signal through a low-pass filter.

The filter we used is simply an averaging filter that is designed to reduce the high-frequency components of the signal by smoothing. It is implemented by using a spatial mask that encompasses a value and its two neighbors to either

side. A new value corresponding to the original value is arrived at by adding the original's binary value to the values of its four neighbors, and then dividing by five.

### Temporal Pattern Recognition

Once we have extracted the features or digital signatures from the dataset derived from the motion sensor array, we must employ a classification mechanism to aid in pattern recognition.

Hidden Markov Models were chosen as the statistical model in this experiment, primarily because they are well suited for applications in temporal pattern recognition.

### Classification

The chief purpose of any pattern recognition system is to classify the data based on either a priori or statistical information. The classifiers used in this experiment will be constructed using hidden Markov models (HMMs), which will be the topic of the next section.

A classifier partitions the space of digital features into  $K$  disjoint subsets,  $a_1, \dots, a_k$ , such that for sample with feature  $X = (x_1, \dots, x_g) \in a_k$ , the predicted class is  $K$ . A learning set can be defined as:

$$L = \{(x_1, y_1), \dots, (x_n, y_n)\}$$

Next, a classifier  $C$  can be built from the learning set as follows:

$$C(., L): X \rightarrow \{1, 2, \dots, K\}$$

When sequence  $X$  is observed, classifier  $C$  will decide:

$$C(X, L) = k \text{ if } X \in a_k$$

The system built for this experiment will focus on the construction of two separate and distinct classifiers. One classifier will simply determine if human motion is present by examining the observation sequence. This classifier can be defined as above with  $k = 2$ , motion detected or motion not detected.

The second classifier will attempt to classify individual subjects for identification purposes. This classifier can be defined as above with:

*K = number of registered subjects*

#### Hidden Markov Models (HMM)

A Hidden Markov Model (HMM) is a statistical model in which the system being modeled is considered to be a Markov chain with unknown parameters. A Markov chain is a stochastic process having the Markov property, meaning that given the present state, future states are independent of the past states. A Markov chain is considered a memory-less system and can be represented as a finite state machine. Markov chains are stochastic in that future states will be reached through a probabilistic process rather than a deterministic one.

Markov chains consist of various states and transitional probabilities between those states. At each step the system may change from one state to another state, or remain in the same state based a particular probability distribution. When a system changes states, it is referred to as a transition. The probability that the system will change to a particular state is called the transition probability. Transition probabilities exist between every state and itself.



## HMM Parameters

A Hidden Markov Model can be characterized by various known and unknown parameters. The states that make up a regular Markov model are directly visible to the observer, but with respect to a hidden Markov model, the states are hidden and not directly visible. Mathematically, the set of hidden states is denoted using set notation below:

$$S = \{s_1, \dots, s_N, N = \text{number of states}\}$$

The observation sequence is the observable data that is output from the model at time  $t$ . Formally, the set of observations can be represented as :

$$O = \{o_1, \dots, o_T, T = \text{number of observations}\}$$

The observation sequence described above can only contain a finite number of discrete symbols. Formally, the set of discrete observation symbols can be represented as:

$$V = \{v_1, \dots, v_M, M = \text{number of discrete symbols}\}$$

One assumption of a Markov model is that any state can follow another. Since all states are possible, probabilities are used to determine the likelihood of one state following another. The set of transition probabilities can be described mathematically using set notation as shown below:

$$A = \left\{ \begin{array}{l} a_{1,1}, a_{2,2}, \dots, a_{N,N}, N = \text{number of states,} \\ a_{i,j} = \Pr(s_{t+1} = j | s_t = i) \\ 1 \leq i, j \leq N \\ 1 \leq t \leq T \\ a_{i,j} \geq 0 \\ \sum a_{i,j} = 1 \end{array} \right\}$$

Each  $a_{i,j}$  pair in the set of transition probabilities represents the probability of transitioning from state  $s_i$  to state  $s_j$ . Emission probabilities describe the probability of a particular discrete observation symbol being emitted by a particular state. The set of emission probabilities can be described mathematically by using set notation as shown below:

$$B = \left\{ \begin{array}{l} b_i(v_k) | b_i(v_k) = \Pr(v_k \text{ at } t | s_t = i) \\ 1 \leq i \leq N \\ 1 \leq k \leq M \\ b_i(v_k) \geq 0 \\ \sum b_j(v_k) = 1 \end{array} \right\}$$

Each  $b_i(v_k)$  represents a function which takes a particular observation symbol and returns the probability of that symbol being emitted at state  $s_i$ . The last parameter of a hidden Markov model is the start state probabilities. These probabilities describe the state of the model at its initialization when  $t = 1$ . The set of start state probabilities can be described mathematically using set notation as shown below:

$$\Pi = \left\{ \begin{array}{l} \pi_i | \pi_i = \Pr(s_1 = i) \\ 1 \leq i \leq N \\ \pi_i \geq 0 \\ \sum \pi_i = 1 \end{array} \right\}$$

Each  $\pi_i$  represents the probability of state  $s_i$  being the start state.

The transition, emission, and start state probabilities form a tuple  $(A, B, \Pi)$  and when used together, fully describe a particular hidden Markov model and is represented by  $\lambda = (A, B, \Pi)$ .

## Forward Algorithm

One of the basic problems associated with hidden Markov models is evaluation: given a particular observation sequence  $O$  and an HMM  $\lambda$ , what is the method for computing  $\Pr(O|\lambda)$ , or the probability of  $O$  given the model? Using a brute-force approach, this computation can be expressed as the sum of the probabilities of all possible state sequences in the HMM:

$$\Pr(O|\lambda) = \left\{ \sum_X \Pr(O|X) \Pr(X), X = \text{Possible hidden state sequence} \right\}$$

This is very computationally expensive, however, as there are on the order of  $N^T$  possible state sequences. This clearly doesn't scale very well and is considered impractical for most real-life problems. The solution to this problem can be solved using a variant of the Viterbi algorithm called the Forward algorithm.

The Forward algorithm uses a dynamic programming approach that utilizes recursion and the idea of partial probabilities to reduce the overall computational complexity to approximately  $N^2T$  computations. Partial probabilities can be described mathematically as:

$$\alpha_t(i) = \Pr(o_1, \dots, o_t, q_t = s_i | \lambda)$$

This can be explained as the probability that the current state is  $s_i$  at time  $t$  and the partial observation  $o_1, \dots, o_t$  has been observed, given the model  $\lambda$ . The sum of these partial probabilities can simply be added to obtain the overall probability  $\Pr(O|\lambda)$ .

The Forward algorithm works in three steps: initialization, induction, and termination. The initialization step describes the algorithm at time  $t = 1$ .

$$\alpha_1(i) = 1, 1 \leq i \leq N$$

The induction step describes the approach of the algorithm between the initialization and termination steps.

$$\alpha_t(j) = \left\{ \sum_{i=1}^N \alpha_{t-1}(i) a_{i,j} \right\} b_j(o_t), 2 \leq t \leq T, 1 \leq j \leq N$$

This algorithm computes the probability that the current state is  $s_j$  and the partial observation  $o_1, \dots, o_t$  has been observed at time  $t$  by computing the sum of the partial probabilities of all the states at time  $t - 1$ . The termination step simply sums all of the computed partial probabilities at time  $T$ .

$$\Pr(O|\lambda) = \sum_{i=1}^N \alpha_T(i)$$

The Forward algorithm is able to provide an enormous improvement over the naïve approach, running in the order of  $N^2T$  computations versus  $N^T$  computations. Another benefit of this algorithm is how it lends itself to recursion. An efficient way to compute the forward probabilities is to start with the termination step and work backward recursively through the inductive step until time  $t = 1$ . The initialization step is then used to end the recursion.

### Backward Algorithm

The Backward algorithm is similar to the Forward algorithm in that it used to solve the evaluation problem. The Backward algorithm, however, works in the opposite direction. This algorithm also uses a dynamic programming approach

and utilizes recursion and partial probabilities. This algorithm calculates the probability that given an HMM, and given a state  $s_i$  at time  $t$ , the partial observation  $o_{t+1}, \dots, o_T$  has been generated, or stated mathematically:

$$\beta_t(i) = \Pr(o_{t+1}, \dots, o_T | q_t = s_i, \lambda)$$

Like the Forward algorithm, the Backward algorithm works in three steps: initialization, induction, and termination. The initialization step describes the algorithm at time  $T$ .

$$\beta_T(i) = 1, 1 \leq i \leq N$$

The induction step describes the approach of the algorithm between the initialization and termination steps.

$$\beta_t(i) = \left\{ \left[ \sum_{j=1}^N a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j) \right], t = T - 1 \dots 1, 1 \leq i \leq N \right\}$$

The termination step simply sums all of the computed partial probabilities at time  $t = 1$ .

$$\Pr(O | \lambda) = \sum_{i=1}^N \pi_i \beta_1(i)$$

The Backward algorithm is asymptotically equal to the Forward Algorithm.

### Forward-Backward Algorithm

One of the most important problems associated with hidden Markov models is the ability to adjust a model's parameters  $\lambda = (A, B, \Pi)$  to maximize  $\Pr(O|\lambda)$ .

This problem is known as learning.

It is often the case where the parameters of a hidden Markov model are known a priori or estimated from training data. In such situations it is

unnecessary to perform this step. However, in many cases the parameters describing the underlying model may not be known but we would like to fit the model parameters to a dataset. In such a case we are looking for a new model that best fits the data or said mathematically:

$$\lambda' = \operatorname{argmax}_{\lambda} \Pr(O|\lambda)$$

Given an initial model  $\lambda$ , it is possible to find another model  $\lambda'$  such that  $\Pr(O|\lambda') \geq \Pr(O|\lambda)$ . The process of learning is accomplished using an Expectation-Maximization algorithm called the Forward-Backward algorithm (also known as the Baum-Welch algorithm).

The Forward-Backward algorithm requires an initial guess to be made for the parameters. The accuracy of these initial parameters is not important, however, more accurate estimations may lower the time this algorithm takes to converge. Once the initial parameters have been instantiated the algorithm proceeds in two steps: an expectation step and a maximization step. The expectation step estimates the expected value of the unknown variables, given the current parameter estimate. The maximization step re-estimates the distribution parameters to maximize the likelihood of the data, given the estimates of the expectations of the unknown variables. The expectation and maximization steps are alternated until there is no appreciable improvement in maximum likelihood.

In order to generate a new model  $\lambda' = (A', B', \Pi')$  from the initial model  $\lambda = (A, B, \Pi)$ , all three parameters must be re-estimated. When re-estimating the transition probabilities:  $A$ , we need to know the probability of being in state  $s_i$  at

time  $t$ , and going to state  $s_j$  at time  $t - 1$ , given the current observation sequence and model parameters. This can be stated in the following equation:

$$\xi_t(i, j) = \{\Pr(q_t = s_i, q_{t+1} = s_j \mid O, \lambda)\}$$

This probability, called the transition probability, can be calculated as:

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{i,j} b_j(o_{t+1}) \beta_{t+1}(j)}$$

Intuitively, the transition probability re-estimation can be described as the expected number of transitions from state  $s_i$  to state  $s_j$ , divided by the total number of transitions from state  $s_i$ , as shown in the following equation:

$$a'_{i,j} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \sum_{j'=1}^N \xi_t(i, j')}$$

To help clarify the above equation, we can define:

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j)$$

This simplifies the above equation to the following:

$$a'_{i,j} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

To re-estimate the initial state probabilities, we simply need to calculate the number of times that state  $s_i$  is a start state. This is just a special case of the original equation, where time  $t = 1$ . This can be shown in the following equation:

$$\pi'_i = \gamma_1(i)$$

The last parameter that we need to re-estimate is the emission probabilities. Intuitively, this can be described as the expected number of times in state  $s_i$  where

we observe symbol  $v_k$ , divided by the total expected number of times in state  $s_i$ , as shown in the following equation:

$$b'_i(k) = \left\{ \frac{\sum_{t=1}^T \delta(o_t, v_k) \gamma_t(i)}{\sum_{t=1}^T \gamma_t(i)}, \delta(o_t, v_k) = 1 \text{ if } o_t = v_k, \text{ and } 0 \text{ otherwise} \right\}$$

Using these parameter re-estimation formulas, we can now calculate a improved model  $\lambda' = (A', B', \Pi')$  based on the original model  $\lambda = (A, B, \Pi)$ . This re-estimation process is known as the maximization step. After this step is completed, a expectation step is performed to calculate the forward and backward probabilities of the given model, with the expectation that  $\Pr(O|\lambda') \geq \Pr(O|\lambda)$ . This process will continue in an iterative manner until there is no appreciable improvement or  $\Pr(O|\lambda') - \Pr(O|\lambda) \approx 0$ .

### Model Training

The Forward-Backward algorithm mentioned above will adjust a model's parameters to fit a particular dataset. This dataset will be a digital signature or feature that corresponds to a human motion vector. This is an example of the supervised learning method of pattern recognition. As with any biometric system, a user must be registered with the system before identification can take place. For this experiment, a user will be registered by having them walk along a predefined path for a particular amount of time. Each user will use identical paths and training times for each test.

### Model Testing

Once a series of features have been registered using the training process, we can test the system by introducing an unknown observation sequence and then



determining which model best fits by finding the one that returns the maximum likelihood. The method which we will use to calculate the probability of a testing sequence corresponding to a particular training sequence is the Forward algorithm. As discussed in an earlier section, the Forward algorithm computes the probability of a particular observation (training) sequence  $O$ , occurring given an HMM  $\lambda_i$ , or  $\Pr(O | \lambda_i)$ .

### Maximum Likelihood

Maximum likelihood criterion will be used in order to determine which training sequence a particular testing sequence most likely matches. This can be stated mathematically as follows:

$$X \in \lambda_i, i = \operatorname{argmax}_i \{\Pr(X | \lambda_i)\}, 1 \leq i \leq K$$

Simply put, after the system has seen a complete observation sequence, it will calculate the Forward probability that a particular HMM  $\lambda_i$  could generate that sequence. It will calculate the Forward probability for each HMM  $\lambda_i$ , for  $K$  HMMs. The value of  $\lambda_i$  for which the Forward Probability is the largest,  $\Pr(O | \lambda_i)$ , will be chosen.

## CHAPTER 4

### DATA ANALYSIS

#### Configuration

For this experiment, four sensor modules were custom-built and placed on a stand. The height of the stand is fully adjustable and sensor modules can be placed anywhere along the vertical axis to allow for maximum variability. The figures below show a sensor module and a stand populated with four sensor modules.

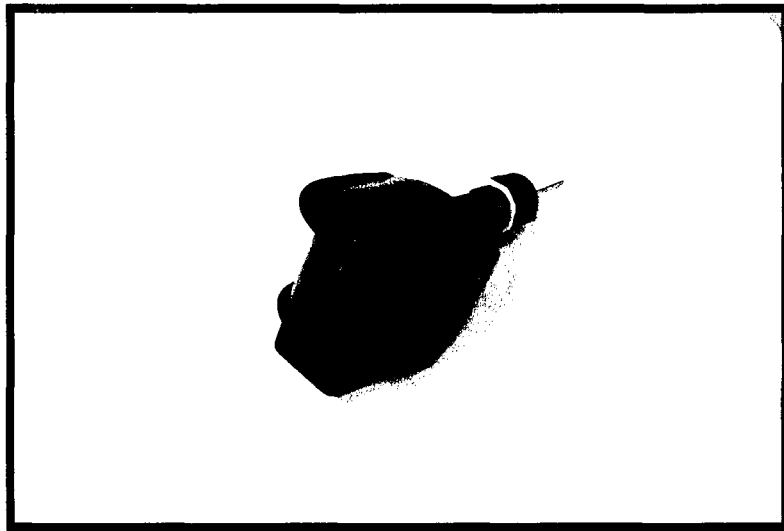


Figure 6. Sensor module.



Figure 7. PIR stand with four sensor modules.

Each sensor module is cabled to a microcontroller. This device receives the analog signal, performs the analog-to-digital conversion, and the required signal processing. The microcontroller is connected with a serial cable to a RS-232 port on a PC. On the PC is an application built for this experiment that reads the digital signatures and performs the training and testing phases. This application also allows the user to interact with the system.

#### Microcontroller

The microcontroller used for this project is the Javelin Stamp available from Parallax. The microcontroller is attached to the Javelin Stamp Demo board. The Demo board also contains an onboard Sigma-Delta analog-to-digital converter,

which is used to convert the analog signals generated by the sensor modules to a digital signal that can be used by the microcontroller. A Uart located on the Demo board provides a way for the microcontroller to communicate with a PC over a serial cable. This enables the microcontroller to transmit the binary digital feature to a connected PC for additional processing.

The Sigma-Delta A/D converter is implemented by a combination a RC network that is shown in the schematic below:

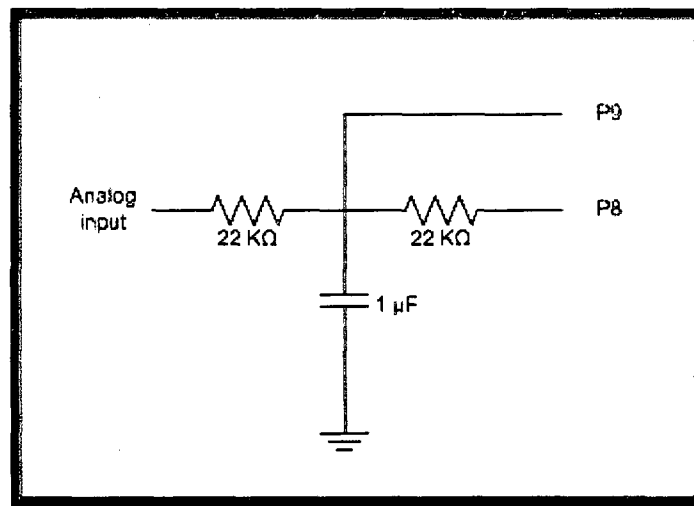


Figure 8. Schematic of Analog-to-Digital converter.

A picture of the Demo board, Javelin Stamp, and A/D converter with the associated wiring can be seen below:

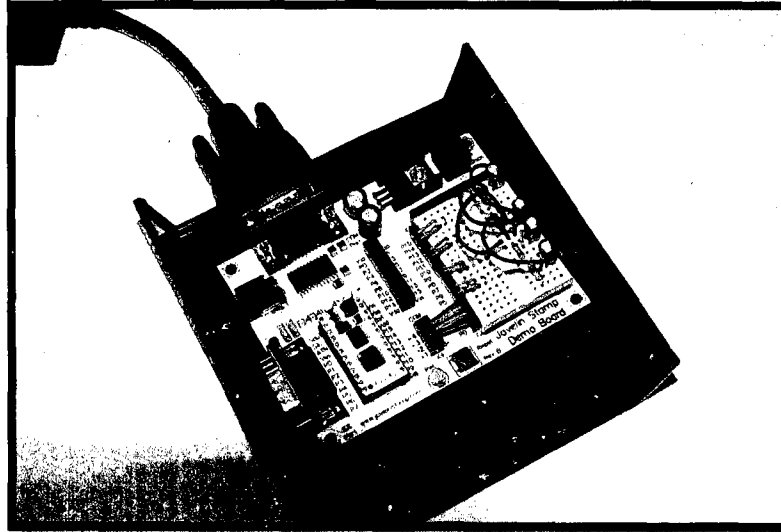


Figure 9. Demo board containing Javelin Stamp and ADC.

### PIR Identification Application

In order to implement a pattern recognition system for identification, an application was custom built. This application was created using C++ and utilized the Microsoft Foundation Class (MFC) library.

The application used three threads, a GUI thread that allow the user to interact with the interface, a worker thread that computed the HMMs, and another worker thread that polls the Windows COM port for the signal sent by the microcontroller. The user can choose whether a given observation sequence is a learning event or a testing event and enter the associated parameters for each. The application then performs the necessary calculations for either learning or testing and outputs the received binary digital feature to the screen. The following figure shows the user interface of the PIR identification application:

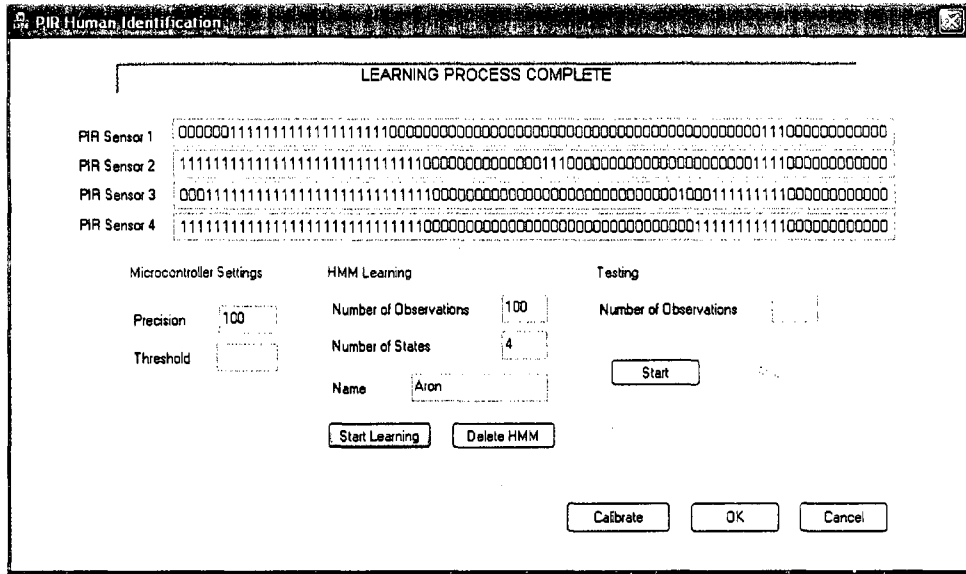


Figure 10. PIR Human Identification Application.

### Initial Testing

Using the configuration described in the previous sections, the focus then turned to testing individuals in order to view the characteristics of their generated waveforms.

The initial tests used four sensor modules located at a 35 in., 42 in., 48 in., and 54 in. from the floor, respectively, and aligned on the vertical axis. The output from each sensor was fed through an analog-to-digital converter and then sent to the microcontroller. The microcontroller processed each signal by subtracting 127 from each sample and then taking the absolute value, which results in a signal whose amplitude that is directly proportional to the generated voltage on its respective sensor. The microcontroller sampled the signal every 100 ms or 10 times a second. The sensor modules were placed in the middle of a room, with a path marked with tape located perpendicularly to the sensors.

This path is two feet away from the sensors at its closest point. Below is a diagram showing the position of the sensors and the walking path:

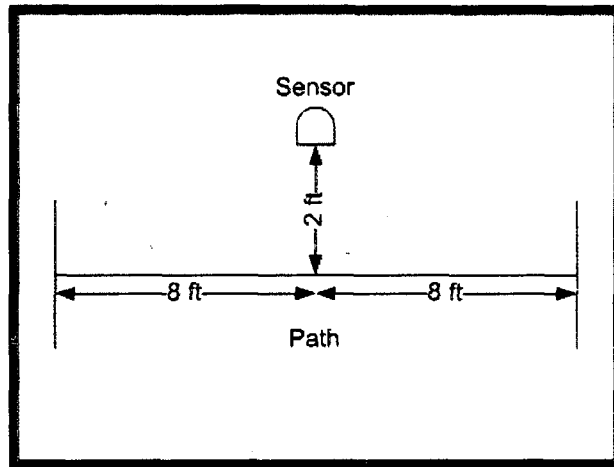


Figure 11. Experiment walking path.

The subject was instructed to walk in a straight line along the path, and then turn around once the end of the path was reached. The subject walked for 20 seconds, which produced 200 samples. The following four figures show the signals generated for each of the four sensors:

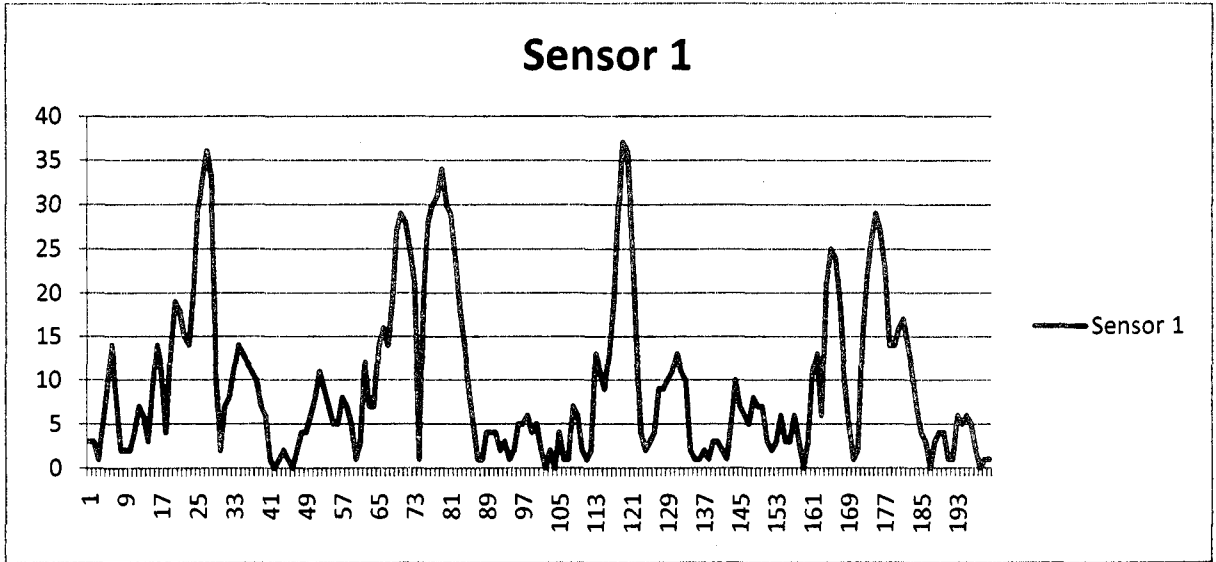


Figure 12. Signal generated from sensor 1.

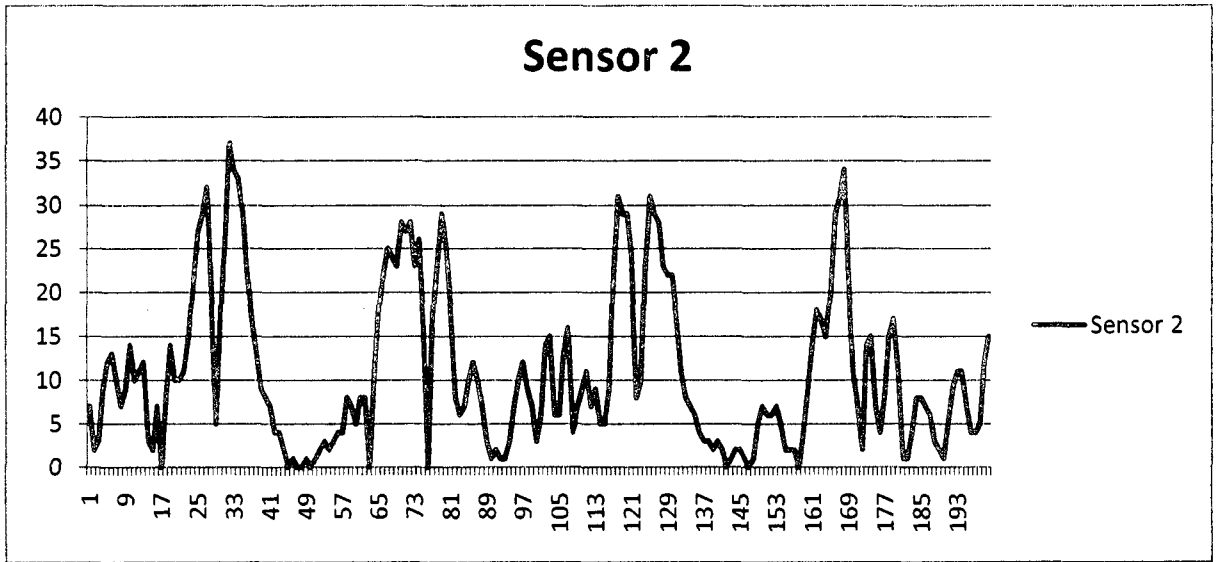


Figure 13. Signal generated from sensor 2.



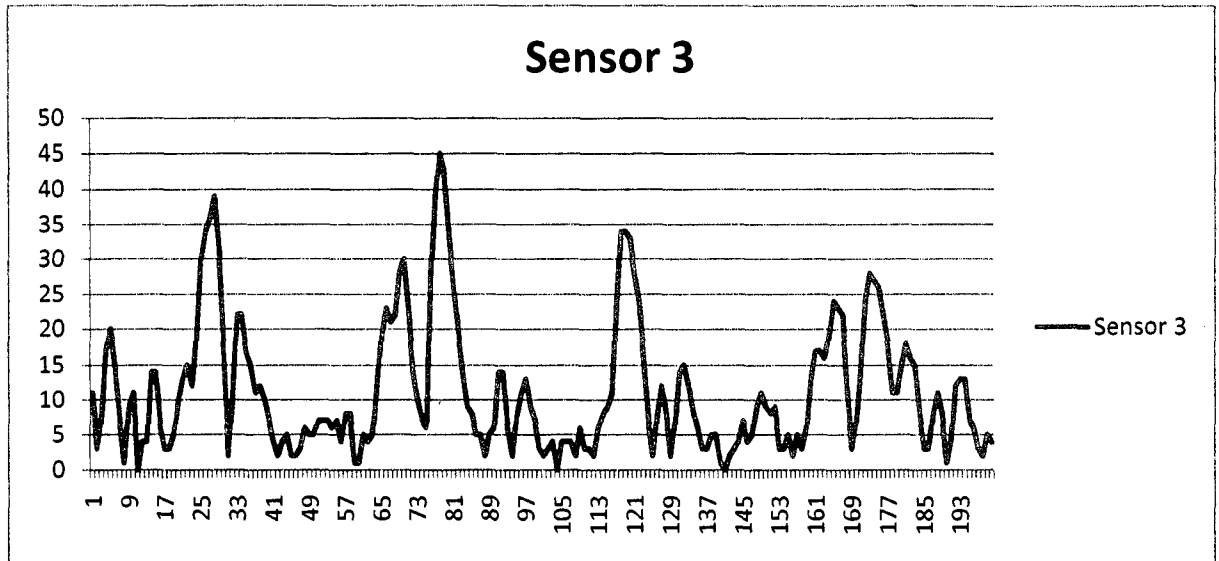


Figure 14. Signal generated from sensor 3.

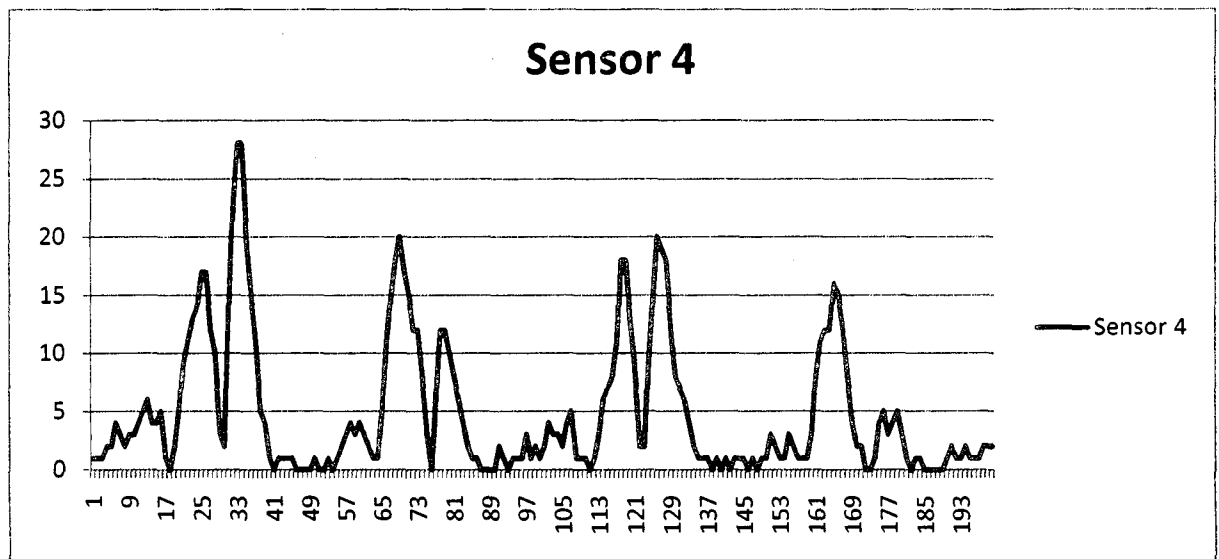


Figure 15. Signal generated from sensor 4.

In the figures above, the amplitude is directly proportional to the subject's proximity to the sensor. High frequency noise dominates the areas between the peaks where little or no motion has been detected.

Intuitively, it seems that due to the characteristic nature of a person's gait, each person should affect a sensor field in a unique way. As a person moves through a sensor field, the magnitude and spread of the voltage generated on a particular sensor is proportional to the infrared energy detected over time. The amount of energy depends on the movement and composition of the body parts that the sensor is focused on. This can affect the signal in two distinct ways. The amplitude of the peaks will be more if the sensor is pointed towards the warmer areas of the body, such as the torso. A person's height is one factor that will affect whether a sensor is pointed near the warmer areas of the body or colder areas such as the head and legs. We can compare the amplitudes of two or more signals by calculating the mean of each signal. The spread of a signal will be determined by a number of factors, such as the speed of the walker, the swing of the arms, and the shape of the body. By calculating the standard deviation of a signal, we can measure its spread. Therefore, if it is assumed that the noise is constant in every signal, we may be able to determine if the signal is unique by comparing the mean and standard deviation of the signals generated by multiple subjects. The next two figures show bar charts that represent the mean and standard deviation, respectively, of three different subjects. Each subject is represented by four bars, corresponding to each of the four sensors.

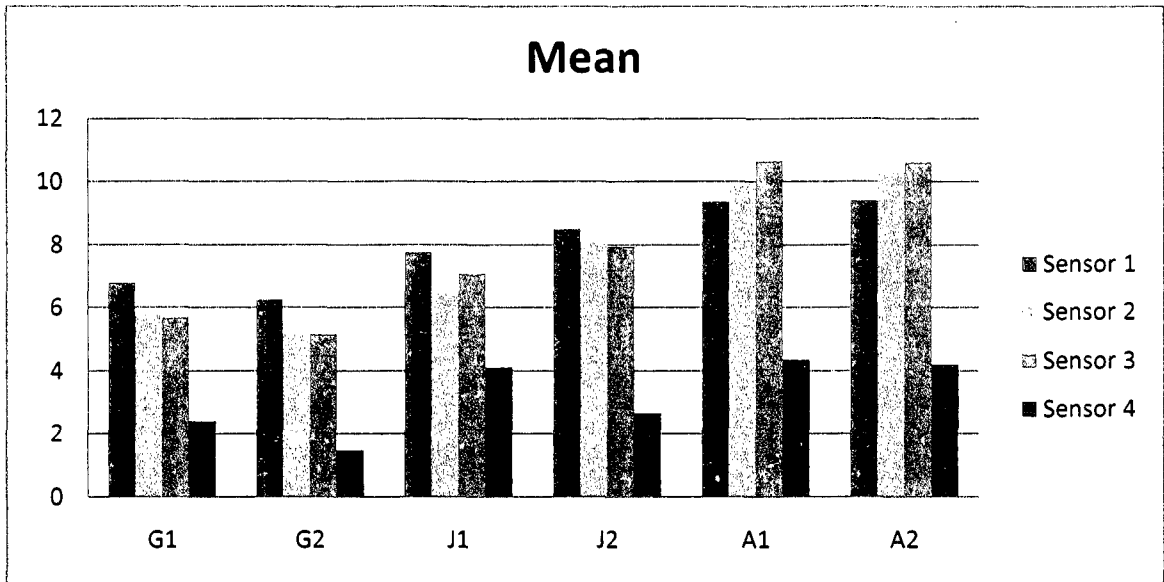


Figure 16. Sample means for three subjects.

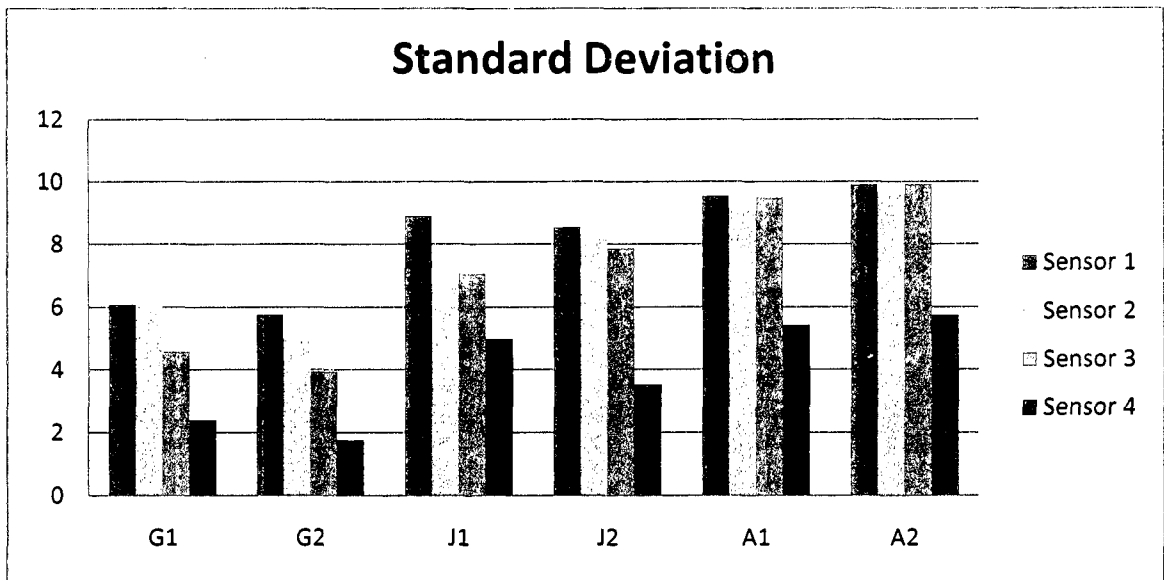


Figure 17. Sample standard deviations for three subjects.

It can be seen in the charts that the mean and standard deviations of the signals representing a particular subject are consistent between trials. This demonstrates that these signals are repeatable. It can also be seen that the

signals representing a subject are dissimilar from that of the other subjects. This demonstrates that these signals may be characteristic of the subject that generated them.

### Experiment

The experiment was carried out using four subjects. Each subject conducted a walk of particular duration to train the system and then performed five walks each of varying durations to test the system.

The variables of the experiment are the time taken to train the system, and the time taken to test the system. There were two different training times used, 50 sec (500 samples) and 100 sec (1000 samples). For each training time, there were four different testing times, 500 sec, 200 sec, 100 sec, and 50 sec. Each subject tested at each level five times. The other HMM parameters, such as states, were kept constant throughout the experiment. The results of the experiment are shown in the tables below:

Table 1 Experiment Results (Training Set = 500), (Testing Set = 500)

Name	Aron	Ghada	Josh	Shirin
Aron	80%	20%		
Ghada		80%	20%	
Josh			100%	
Shirin	20%			80%

Table 2 Experiment Results (Training Set = 500), (Testing Set = 200)

Name	Aron	Ghada	Josh	Shirin
Aron	60%	40%		
Ghada		80%	20%	
Josh	20%		40%	40%
Shirin	40%			60%

Table 3 Experiment Results (Training Set = 500), (Testing Set = 100)

Name	Aron	Ghada	Josh	Shirin
Aron	60%		40%	
Ghada		60%	40%	
Josh	20%		60%	20%
Shirin	60%			40%

Table 4 Experiment Results (Training Set = 500), (Testing Set = 50)

Name	Aron	Ghada	Josh	Shirin
Aron	60%		40%	
Ghada		100%		
Josh		40%	60%	
Shirin	60%			40%

Table 5 Experiment Results (Training Set = 1000), (Testing Set = 500)

Name	Aron	Ghada	Josh	Shirin
Aron	100%			
Ghada		100%		
Josh	40%		60%	
Shirin	60%		20%	20%

Table 6 Experiment Results (Training Set = 1000), (Testing Set = 200)

Name	Aron	Ghada	Josh	Shirin
Aron	60%		40%	
Ghada		100%		
Josh	20%		20%	60%
Shirin	80%			20%

Table 7 Experiment Results (Training Set = 1000), (Testing Set = 100)

Name	Aron	Ghada	Josh	Shirin
Aron	40%		60%	
Ghada		100%		
Josh			80%	20%
Shirin	20%		60%	20%

**Table 8 Experiment Results (Training Set = 1000), (Testing Set = 50)**

Name	Aron	Ghada	Josh	Shirin
Aron	60%		40%	
Ghada		100%		
Josh	40%		60%	
Shirin	20%		20%	60%

From each of the tables above, it can be seen that the longer testing times generally result in a higher probability of positive identification. The two training times, however, didn't seem to have much of an effect one way or the other on the results. There is a trade-off between short testing times and a high probability of identification. It is desirable to have as short of a testing time as possible so that a person could be identified quickly. However, with a short testing time there is a relatively small amount of samples compared to its corresponding training sequence, which could limit the system's precision. One interesting note about the results show that while the results are much better at when a test with 500 samples are used, there is no noticeable drop-off in identification rate from tests that used 200 samples down to 50 samples. In fact, the test with 50 samples slightly out performed some of the longer tests.

## CHAPTER 5

### CONCLUSIONS

The goals of this paper were threefold; two determine if the digital features generated by a motion sensor are characteristic of human motion, to build a classifier that can detect human presence, and build another classifier that can perform human identification. It is not completely certain that the digital features generated by the motion sensors used in this experiment and using the signal processing techniques described in this paper, are truly a characteristic measure of the human gait.

The second goal was to design a classifier that can detect the presence of a human in a room. It was demonstrated that the hardware and software developed for this experiment certainly was successful in implementing a system that could accurately detect human motion. The amount of motion necessary for detection is minimal enough that even a human attempting to stand still would generate a motion event. This system could be used for numerous applications, much like conventional motion detection systems.

The third goal of this paper was to design a classifier that could be used a biometric device; able to perform the complex task of human identification. Ultimately, successfully achieving this goal is tied to the accomplishment of the first goal, which is to fulfill the task of generating a characteristic digital feature.

It's impossible to say with any level of confidence that this goal was achieved, however, some of the results seemed promising. The random chance of positive identification in any given trial with four subjects would be one in four, or 25%. While the results in most of the trials were far from perfect, none had an overall identification rate less than 50%. Additionally, out of the 32 individual trials run in this experiment, only three had positive identification rates that were lower than 25%. The most successful trials were the two that used 500-sample testing sets. Notably, the trial characterized by a 500-sample training set and a 500-sample testing set was able to achieve an overall positive identification rate of 85%. Although the number of trials run is way too small to be free of statistical noise, it is still hard to imagine an outcome of 85% being attributed to random chance.

It is uncertain that this application, if designed properly, would be suitable as a standalone biometric system. This question would have to be the topic of a future paper. One major limitation of the implementation described in this paper is that there is no easy way to determine if the subject being tested was ever trained to begin with. Using hidden Markov models as a probability model, the system simply chooses the trained model that has the highest probability of matching the current observation sequence. In order to provide a way to numerically determine that the current observation sequence does not correspond to any trained model, some sort of adaptive threshold would have to be used. It seems apparent that the real use and power of a system such as the one presented would be as an alternate identification mechanism that could pair



in tandem with other biometric systems as a way to increase the overall likelihood of correct authentication.

Ultimately, future work must be done in this area before any judgment could be made on the subject matter. The quality of the electronics used in the sensor modules, analog-to-digital conversion, and signal processing could vastly be improved in production than what was used for this experiment. The method of generating the digital feature used in this experiment is also rather simple and rudimentary and could most likely be improved. Lastly, other statistical models, such as principle components analysis (PCA), neural networks, etc. could possibly lead to better results and therefore, should be explored.

## BIBLIOGRAPHY

- Infrared Parts Manual. Glolab Corporation. 2003.
- Gonzalez, Rafael C. Digital Image Processing. 3<sup>rd</sup> edition. Upper Saddle River, NJ: Pearson Prentice Hall, 2008.
- Bertsekas, Dimitri P., Tsitsiklis, John N. Introduction to Probability. Belmont, MA: Athena Scientific, 2002.
- Petruzella, Frank D. Essentials of Electronics: A Survey. New York, NY: Glencoe/McGraw-Hill, 1993.
- Agresti, Alan, Franklin, Chris. Statistics: The Art and Science of Learning from Data. 2<sup>nd</sup> edition. Upper Saddle River, NJ: Pearson Prentice Hall, 2007.
- Serway, Raymond A., Faughn, Jerry S. College Physics. 6<sup>th</sup> edition. Volume 1. Pacific Grove, CA: Brooks/Cole-Thomson Learning, 2003.
- Serway, Raymond A., Faughn, Jerry S. College Physics. 6<sup>th</sup> edition. Volume 2. Pacific Grove, CA: Brooks/Cole-Thomson Learning, 2003.
- Al Williams Consulting. Javelin Stamp User's Manual. Version 1.1.
- J.S. Fang, Q. Hao, D. Brady, B. Guenther, K. Hsu. Real-time human identification using a pyroelectric detector array and hidden Markov models.
- N. Kakuta, S. Yokoyama, M. Nakamura. Estimation of radiative heat transfer using a geometric human model. IEEE Trans. Biomed. Eng. 48, 324-331, 2001.
- A. Kale, N. Cuntoor, B. Yegnanarayana, A.N Rajagopalan, R.Chellappa. Gait analysis for human identification.
- J.Cutting, L.Kozlowski, Recognizing friends by their walk: gait perception without familiarity cues. Bulletin of the Psychonomic Society, vol.9, pp.353-356, 1977.

VITA

Graduate College  
University of Nevada, Las Vegas

Aron Edward Suppes

Home Address:

10091 Cranbrook Falls Ct  
Las Vegas, NV 89148

Degrees:

Bachelor of Science, Computer Science, 2005  
Hawaii Pacific University

Thesis Title:

Human Identification Using Pyroelectric Sensors

Thesis Examination Committee:

Chairperson, Dr. Evangelos A. Yfantis, Ph. D.  
Committee Member, Dr. Jan Pedersen, Ph. D.  
Committee Member, Dr. Laxmi P. Gewali, Ph. D.  
Graduate Faculty Representative, Dr. Georg Mauer, Ph. D.